STAG: Smart Tools and Applications in Graphics (2022)
D. Cabiddu, T. Schneider, G. Cherchi, and R. Scateni (Editors)

Accurate molecular atom selection in VR

E. Molina and P. Vázquez

ViRVIG Group
Universitat Politècnica de Catalunya

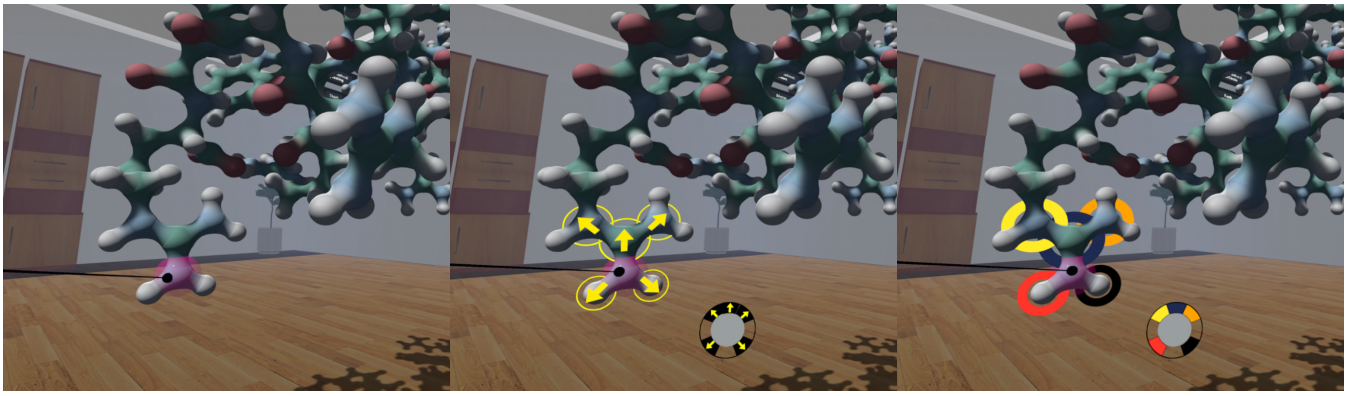

Figure 1: *Our new selection methods (center and right) as compared to classical ray selection (left). To facilitate atom picking in crowded scenes, we have created two new world-space visual feedback elements. From left to right: Raycasting, Ray + arrows and Ray + colors. Visual feedback is coupled with a technique to select the desired neighbours through the use of the touchpad, which requires no precision.*

Abstract

Target acquisition is a basic task that is part of almost any high-level interaction in 3D environments. Therefore, providing accurate selection is a necessity for most applications and games. When targets are small and scenes are cluttered, selection becomes inaccurate. This may lead to selecting the wrong elements which, apart from the time consumed, may become a frustrating experience. Besides the unintentional tremor, the button/trigger press for effectively selecting an element further reduces our stability, increasing the probability of an incorrect target acquisition. In this paper, we focus on molecular visualization and address the problem of selecting atoms, which are rendered as small spheres. We build upon previous progressive selection algorithms and present two alternatives that accelerate the selection of neighbors after an initial selection. We have implemented and analyzed such techniques through a formal user study and found that they were highly appreciated by the users. These selection methods may be suitable for other crowded environments beyond molecular visualization.

(see https://www.acm.org/publications/class-2012)

CCS Concepts

• ***Human-centered computing*** → ***Interaction design process and methods;*** *Activity centered design;*

1. Introduction

Immersive analytics is the field that investigates how Virtual Reality (VR) and 3D interaction can be used to support visual analysis of complex data [MSD*18]. It is becoming rapidly popular in many areas. One of such areas is biology, for example for the analysis of molecular models. Some advantages of Virtual Environments (VE) over desktop-based systems include the spatial understanding of 3D structures or the possibility of natural interaction gestures. However, molecular visualization virtual reality systems (e.g., [PGH*21; DCP*14; KBL*19; NGEB15; LBO*20; GBS*18]) are still not on par with desktop and present some limitations in interaction, collaboration, or data visualization [GBS*18]. In this paper, we focus on interaction. More concretely, we address the problem of accurate selection of small objects. Tiny elements

DOI: 10.2312/stag.20221257

are present in molecular models, for example in the form of atoms, but they are also common in almost any visualization technique (e.g., scatterplots, dot plots, etc.). And those elements are difficult to select when we have highly cluttered scenes, something that is highly common in molecular datasets.

Raycasting is the most popular selection technique in Virtual Environments. The fact that it only requires two degrees of freedom and works at any distance makes it suitable for scenes of all sizes. Unfortunately, it is not free of limitations. When the visual size of the target is small, or the scene is cluttered, raycasting is slow and error-prone [SP04; AA13]. Scenarios like molecular visualization pose serious difficulties for accurate selection of a certain element, and have their particularities: *i)* occlusion (the element may be partially or completely hidden), *ii)* small size: atoms are small and thus difficult to point at, *iii)* ambiguity: the imprecision in the ray definition as well as the complexity of the scene results in several candidates being feasible, for a ray direction, and *iv)* neighbor navigation: some tasks require selecting more than one atom, such as for torsion angle query. However, currently, raycasting is the common selection system of the molecular visualization packages we have analyzed. Maybe due to its simplicity.

To address these problems, we propose and evaluate two different progressive techniques. In both cases, we decouple the selection procedure in two steps: an initial ray-based selection (likely on an atom close to the final goal), and a posterior navigation through the neighbors thanks to visual cues that does not require pointing precision. The differences in both proposals lie in the visual feedbacks used to indicate the neighboring atoms that can be selected through the touchpad (arrows on the spheres, or colored circles around the atoms, as shown in Figure 1). The advantages of our technique are threefold:

- Uses as a basis a familiar method.
- Preserves the 3D structure of the scene.
- Enables neighbor navigation, which is useful for other high-level tasks (that may imply the selection of several atoms, such as for measure queries).

We implemented these techniques over UnityMol, a molecular viewer and prototyping platform [DCP*14] coded in C# with Unity3D game engine, which is actively developed by Marc Baaden's team at the LBT laboratory (IBPC institute of CNRS in Paris) and with an HTC Vive.

The rest of the paper is organized as follows. Section 2 deals with the related work. In Section 3 we propose the new techniques, that are initially evaluated in Section 4 in a pilot study. In Section 5 we present the improvements and the final study. Finally, Section 7 discusses the results and Section 8 concludes our work.

2. Related Work

Despite the great popularity of ray-casting techniques for target acquisition, problems remain, especially when scenes are densely populated of tiny objects. As a result, several techniques have been proposed to facilitate pointing. These use different strategies, such as considering a group of candidates and working on a disambiguation strategy, or by making the initial selection easier or faster. However, as we will see, not all these techniques are suitable for molecular models.

2.1. Pointing facilitation

These strategies have the goal of making the selection easy or fast. Argelaguet and Andujar [AA09] use sticky targets. This strategy, inspired from a 2D technique, consists in making targets *attract* the pointer when the ray/pointer is close. It facilitates selecting small objects, and these can also be upscaled when the ray is close to ease its identification. But when many, close objects such as touching atoms, like in molecular models, it is still difficult to select the proper one. Lu et al. [LYS20] propose an extension of the bubble technique to select small objects without requiring precision. Elmqvist and Fekete [EF08] propose the use of semantic information to accelerate target acquisition. It is based on the idea that selectable targets are known by the application, and it is rarely used. However, these techniques are less useful when there are many candidates, such as in highly cluttered scenes.

2.2. Target disambiguation

Another group of strategies performs a progressive, multiple-step selection method [AA13]. An initial selection step, that can be realized in different ways, generates a group of candidates, and the techniques work on mechanisms to facilitate the disambiguation. The initial selection, thus, considers a volume of selection, which can be either a sphere, or a cylinder around the pointing ray.

For example, Grosmann and Balakrishnan use the *Lock Ray* strategy [GB06]: the initial selection locks a ray, and all the objects intersecting the ray are selected as candidates. Then, a depth marker is used to disambiguate the interesting element. Similarly, Baloup et al. [BPC19] implement a ray cursor in 3D. After an initial selection, several candidates are identified along the ray direction, and the user can use the touchpad to move among those.

However, these techniques are mostly useful when lower levels of occlusion are present, such as when scenes are composed of small, but sparse objects. When many candidate points may be in the same line, some disambiguation techniques may be required. For example, Monclús et al. [MVÁ13] also use a ray-cursor for medical models in the context of volume rendering. In this case, to facilitate the disambiguation in largely occluded scenes, two helper views are shown, with a custom transfer function, and projected onto two different planes, that provide contextual information on the surfaces where the candidate points are.

Instead of adding extra helper views, other systems focus on the disambiguation by a progressive refinement that uses secondary views, different from the original. After an initial selection, candidate objects may be presented in the form of pies or menus [GB06; RO13]. Kopper et al. [KBB11] address this problem by another progressive approach: an initial selection gathers objects in a region, and then iteratively reduces the set of selectable objects by placing groups of objects in quadrants of a 2D plane the user can further click on. At the end, a single object will appear in each region. This facilitates the selection, since each subset of objects is separated from the other. Thus, the user only needs to select one quadrant.

However, for scenarios such as molecular models, where the positions of the atoms may be relevant for the selection, displacing the atoms would remove important structural information. Moreover, this technique also shows a degradation in time for scenes with large densities of objects, due to the increased number of steps to select the target.

In contrast to these approaches, our goal is to leave the 3D structure of the original scene untouched, and adding visual cues to facilitate the quick selection of the object of interest through the navigation across neighbors. And make this navigation easy through the touchpad, thus reducing the precision required because the user does not need to point at the scene.

3. Two-step ray-based selection

We propose two new progressive target acquisition techniques intended to facilitate the precise selection of atoms. The core idea is to facilitate the quick access among neighboring atoms in a region of interest, without requiring precision. We call these techniques: *Ray + arrows* and *Ray + colors*. Both techniques work with the same two-step process, illustrated in Figure 2. It works as follows:

1. An initial selection is made using the Raycasting method (step 1 at the top).
2. The user can hop to any of the (up to) eight neighbors of the selected atom using the touchpad.
   - The candidate is then highlighted in white (step 2).
   - When the user has defined the desired atom, it can be selected with the trigger (step 3).

The novelty of our approach is the way we select the candidate neighbors, the visual feedback provided, and the technique to select the neighboring atom, which relies on the use of the touchpad and the trigger, without further need of pointing at the screen. In both cases, upon the initial atom selection, the atom is marked, and then, the system automatically highlights a set of up to 8 neighbors the user can travel to.

The difference of both techniques consists in providing a different visual feedback on the neighbors. In one case, the candidate directions of the touchpad, that represent the neighboring atoms where the user can move, are indicated by arrows over the atoms. In the other, these are indicated by colors that are painted as disks around the atoms. Both have their particularities: arrows might seem more intuitive, and circles may be less prone to occlusion by other atoms. In both cases, we further reinforce the set of valid directions with an informative widget that displays the valid directions in front of the user (on top of their nose). Each candidate destinations are in one of the main directions: up, down, left, right and the corresponding diagonals. And these directions map to eight regions of the controller's touchpad. Initially, we designed the version with feedback consisting in colors. However, we had the impression that associating the color of the atoms with the position of the touchpad would require a steep learning curve. Therefore, we designed the second alternative, with the arrows, so that we could compare the performance.

The candidate neighbors are defined in object space. Both techniques generate the candidate set using the same algorithm. The region in which neighbors are searched for is defined by a radius that could be easily configured by the user. For the experiments, though, we do not let the user change this parameter so that they do not change it during the tasks, therefore we fix it. Since UnityMol is built over Unity, the elements of the scene (atoms in the examples shown) are represented as spheres, that are stored in a spatial structure, to accelerate operations such as collision detection. This is calculated with the Unity function *OverlapSphere*.

The candidates' selection follows this scheme:

- Upon atom selection, the algorithm searches for the eight atoms closer to the selected one inside the defined radius.
- To create the visual feedback, those candidate atoms are projected onto a virtual plane perpendicular to the axis that goes from the viewer to the center of the selected atom.
- The virtual plane is subdivided in sectors (see Figure 3), and the atoms are assigned to the sector they fall into. If more than one atom projects into the same sector, we keep the one that is closer to the camera.

The resulting valid directions are encoded in the accompanying widget as colors or arrows. Directions without candidates are left empty. We now proceed to describe both techniques in detail.

Method 1: Ray + Arrows. The available neighbors are shown by a yellow arrow and a yellow ring around the corresponding atom. Both elements are oriented so that their planes are perpendicular to the user's vision. A guide appears over the user's nose, showing the user which directions are available. The user can select the desired direction, resulting in the arrow and the ring turning white as feedback. In the same way, the direction indicated in the guide turns white. The rationale behind that is that we believe that showing arrows naturally leads to the user to decide which touchpad direction is the desired one.

Method 2: Ray + Colors. Following the same model as in the previous method, we show the possible neighbors employing a colored ring that surrounds the corresponding atom, oriented perpendicular to the user's vision. These rings are of a different color, identifying the different directions they represent. In the same way, a guide is displayed over the users' nose that will show the available directions utilizing the corresponding colors. Likewise, the ring and the color in the guide of the selected direction turn white when their direction is selected. In contrast to the previous method, although here the directions need to be learned, the visual cues (rings around atoms) are seemingly more visible in cluttered scenes.

Our initial implementation, the one tested in our pilot study, placed the guide widget top left, in the peripheral view of the user. As described later, the change of position was a result of the analysis of the pilot study and the interviews with participants.

Neighbors calculation. In both cases, the available neighbors are obtained with the selection sphere centered on the currently selected atom and the directions are calculated by a transformation to a new coordinate system. The new system is made up of:

A = current atom position, C = Camera position
Origin = A, $\vec{Z}$ = A - C, $\vec{Y}$ = Camera up, $\vec{X} = \vec{Z} \times \vec{Y}$

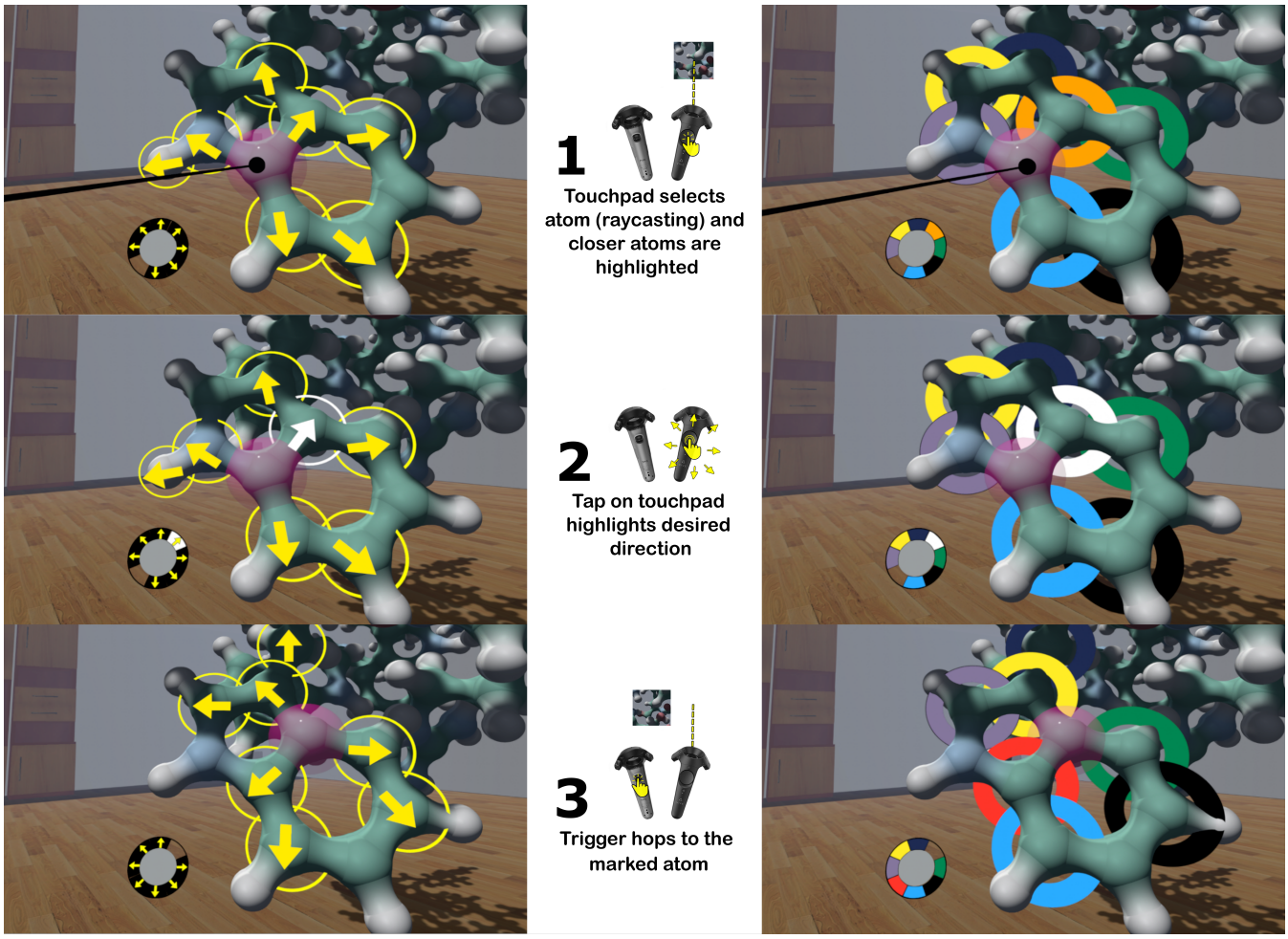

Figure 2: *Process to select an atom and choose and navigate to the available neighbors.* ***Left:*** *Ray + arrows technique.* ***Right:*** *Ray + colors technique. Initially, the user selects an atom clicking the touchpad (top). Then, candidates are highlighted, and the user can navigate to them using the touchpad (center). When the user is satisfied with the desired candidate, the trigger can be used to select the atom (bottom). Upon this selection, neighbors are recalculated and highlighted, and the user can repeat the neighbor navigation as many times as needed.*

With these new coordinates we will know the corresponding direction obtaining the angle defined with $\arctan(\frac{y}{x})$. See Figure 3.

The initial implementation was analyzed by the authors and tested informally with a naïve user with more than 20 years of experience in VR environments. She gave us numerous suggestions that we included in the implementation. With those modifications, we performed a pilot study to evaluate how the users interacted with the newly developed techniques and to get more insights on the possible problems (see Section 4). After analyzing in depth the initial results, as well as the comments of the users during the discussion sessions, we decided to introduce several changes to the interaction metaphors and perform the final study, described in Section 5.

4. Pilot study

This initial implementation was evaluated semi-informally.

4.1. Experiment design

The pilot study consisted in the selection of 6 atoms with a protein of less than 200 atoms, 2M7D from the PDB database [BKW*13], and 6 selections using a protein of about 17K atoms, 6EZN [WKE*18]. Initially, the participant sees the protein and the menu. The user can always move freely around the room or interact with the molecule to move or rotate it. To accomplish this, while aiming at the molecule and holding down the trigger, moving the arm will move the model, and wrist rotation will rotate it. When the user is ready, the selection task starts. The task is defined as:

1. The participant clicks a button on the virtual menu to activate the target.
2. The target is shown in cyan as a sphere larger than the atom it indicates. Then, it reduces its size after 2 seconds, and adjusts to a size similar to that of the target atom.
3. The user selects the atom with the current method.

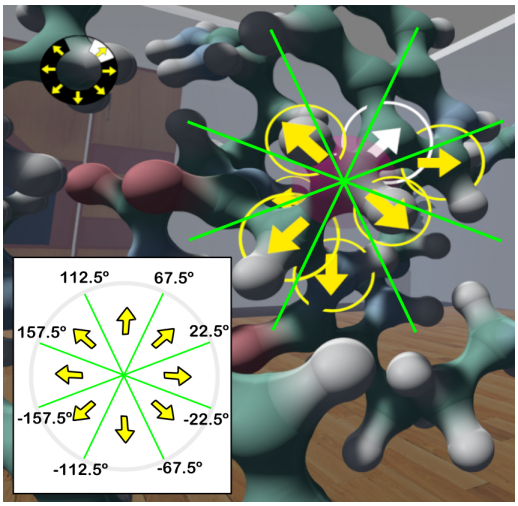

Figure 3: *Available neighbors and their corresponding directions based on the angle in Ray + arrows.*

4. Upon objective selection, the task ends. If there are still pending tasks, a new task can be started by going to step 1.

The user always knows how many selections are still unfinished, since the interface updates a message upon each selection. After the user has finished with one protein, the other one is loaded, and the remaining tasks can be performed.

The target marker is a sphere of size equal to the atom's size, multiplied by 0.1. To indicate the participant where it is, we scale its size and then reduce it. The upscaled size is calculated by multiplying it with a = $\frac{0.1}{s}$, where s means the global scale of the molecule (*lossyScale* in Unity).

Before starting the tasks, the users are introduced to the experiment with a briefing session and a video that shows the different interactions and visual feedbacks. After the video is reproduced, participants practice with a test molecule to become familiar with the techniques. They can perform as many search tasks as necessary until they feel comfortable with the method.

4.2. Participants

9 participants with ages between 18 and 36 carried out the experiment, one of them female. Only 3 of them had medium or high experience in VR. All had studies related to computer science or bioinformatics. The experiments were performed using Latin squares to sort participants, to avoid learning and fatigue effects. However, the order of selection of atoms in each molecule and method was always the same. Between each method and molecule, users could take a break if needed. After the tests, participants were asked to fill a questionnaire with 19 questions to be answered in a 1-7 Likert scale. Finally, they also give a global score from 1 to 10 to each technique. The questions asked whether:

- The selection is made more easily with one of the techniques than with the others
- The technique is comfortable to use
- The technique is easy to learn
- Most people would quickly understand the technique
- The way of showing the colors/arrows and the atom that is selected with the touchpad is understandable
- Overall score

4.3. Results

We can see in the Figure 4 the results of the questionnaire with a Likert scale of 1-7 in terms of perceived comfort with the technique, usefulness (as the ease of achieving the objective), and ease of learning. In the first two cases, comfort and learning, it appears that users prefer the Raycasting technique. However, when evaluating its performance, users assign the same value to Raycasting (5.25) than to the *Ray + arrows* (5.25) technique, and *Ray + colors* is barely behind (5.125).

When the users gave a grade between 1 and 10 to the techniques (see Figure 5), all techniques achieved a high value, which indicates that users recognize the potential of the new techniques proposed. More concretely, the grades were, in average, 8.625 for Raycasting, 8 for *Ray + arrows*, and 7.375 for *Ray + colors* (7.375). Thus, raycasting was ranked slightly higher. We believe, though, that the wording of the questions might have been slightly misleading (implying that the raycasting was not part of the other two selection techniques). In addition, users themselves commented on this fact, indicating that the ray alone was more complicated when they worked with the larger molecule. Since we believe our techniques might be suitable for this scenario, we tested larger molecules and explored scenes with greater occlusion in our final study.

After analyzing the results of the pilot study, we implemented several changes. The most relevant ones are: First, the informative widget, which in the pilot study was located in the upper left region of the vision, was moved to its final position on the nose. Second, the colors of the *Ray + colors* technique were modified. Initially, we were using a rainbow palette (see Figure 6), and we changed it to a categorical palette that is color-blind safe. The new colors are the ones in the Figure 2. Third, the color of the target was also changed into a more salient one.

5. Final Study

5.1. Experiment design

The experiment was designed similarly to the pilot study: First, users click to start a task. Second, a marker highlights the element to select (see Figure 8). Third, the user uses the current metaphor to select the item, and finally, confirms the selection. Tasks are solved using the same interactions. However, in this case, the scenes were closer to a real-world scenario. First, proteins are larger. They still have a significant difference in the number of atoms, but now they consist of 1QM5 from the PDB database [WMG*99], with around 14K atoms, and 1BR1 [DFTC98], with around 30K atoms. This size is in the range of the largest proteins that exist, which have between 30K and 35K atoms. Second, the tasks now also include the selection of internal atoms, which are more challenging. We also give the users feedback upon the completed selection in the form of a soft chime, preventing them from advancing until they achieve

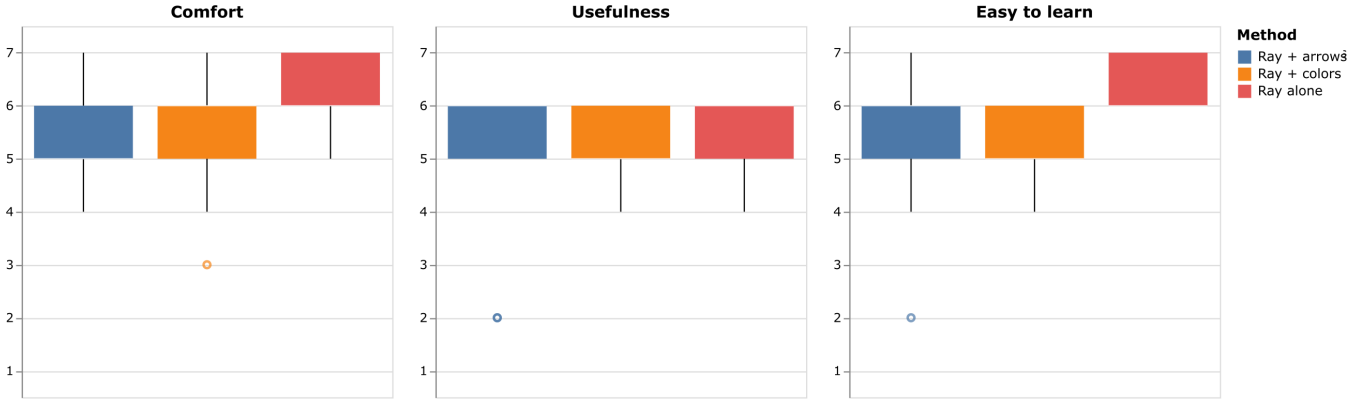

Figure 4: *Perceived comfort (left), usefulness (center), and ease of use (right) of our first implementation of the two-step selection techniques against simple ray selection (red). In this case, ray-based selection is the technique preferred.*

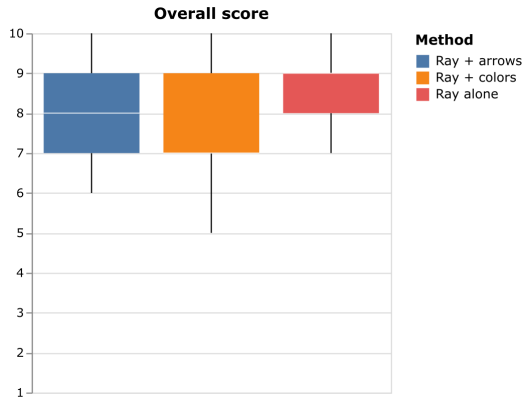

Figure 5: *Overall score in our first implementation of the two-step selection techniques, Ray + arrows (blue) and Ray + colors (yellow), against simple ray selection (red). In this case, Raycasting (8.625) was the preferred one over Ray + arrows (8). Ray + colors was the least valued technique (7.375).*

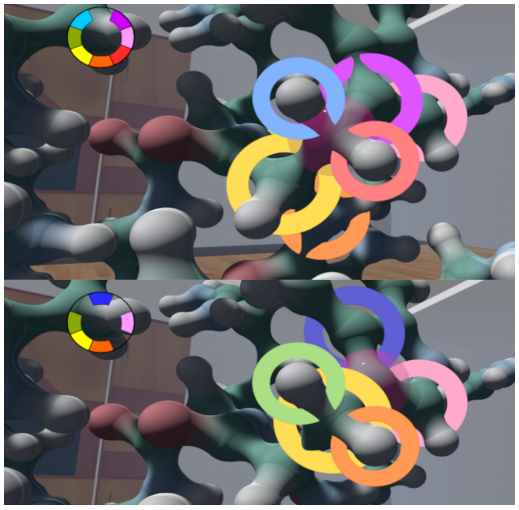

Figure 6: *Two examples of the colors used for the Ray + colors technique in the pilot study and how they are seen on screen. The initial palette was a rainbow one, and was changed to a colorblind safe one in the final implementation.*

this. As previously indicated, the sphere marker was changed to a saturated green, since it seemed more salient than the cyan in the used scenes.

The tasks have the same structure as the ones defined in the pilot study. The user has feedback on the current selection method, as well as the number of tests remaining. In Figure 7 we show how these visual feedbacks look for the initial training.

For this experiment, a pattern was designed for the searches for each molecule. For the small molecule the order was: atoms 1 and 2 were close but distant from the others, atom 3 was distant from the others, then, the other pairs 4-5, 6-7 and 8-9 behaved like the first pair. This last pair was very occluded in an internal part of the molecule. For the large molecule, we used the same scheme, but without occlusion. We consider close atoms as those where the user can travel from one to the other by simply employing the complementary techniques in one or two hops.

5.2. Participants

We recruited 13 participants (5 female), with ages between 18 and 30. None of them had participated in the pilot study. 5 of them had medium or high experience in VR. Only 4 of them with studies related to computer science. The tutorial part was the same as in the pilot study.

For this experiment, users used each technique to perform 9 searches with a protein of nearly 14K atoms, 1QM5 [WMG*99], and another of about 30K atoms, 1BR1 [DFTC98]. The configurations were sorted using Latin squares, to avoid learning and fatigue effects, but the pattern of selection of atoms was kept.

The procedure was the same as in the previous study: consent, tutorial, experiment, and questionnaire. The questions in the ques-

Figure 7: *The menu options the user sees and needs to click to perform the tasks. The top view shows the menu before starting the task, the center image shows the status in the middle of the training (we offered many selections, but users could stop as they were ready), and the bottom one shows the task finished. The upper part indicates the method being tested, and the lower part allows you to start and end searches.*

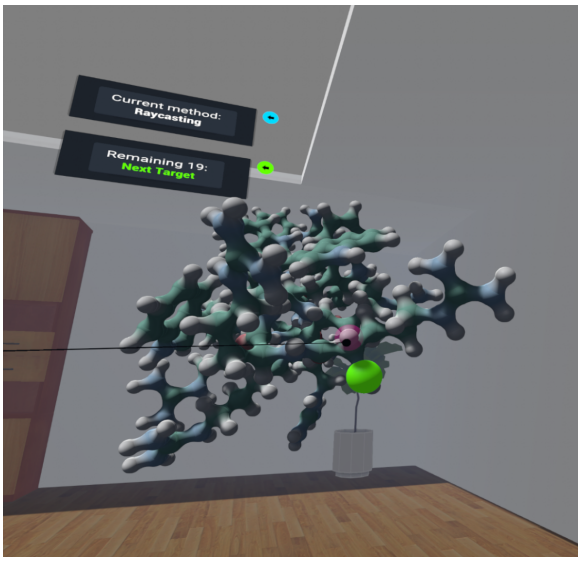

Figure 8: *The aspect of the screen when the user receives the hint on the atom to select (bottom right sphere, encoded in green, with its normal size).*

tionnaire were reformulated to be more clarifying. In addition, we collected information on the selection actions they performed and on the time it took users to carry out each selection. From the moment they started a new search by pressing a button on the virtual menu, until they selected the correct atom, receiving a success sound as feedback, and pressed again the same button. Each session was also recorded.

6. Results

Besides gathering the opinions of the users regarding the comfort and learnability of the techniques, we additionally tracked the time required to solve each of the tasks, as well as the distances the users moved throughout the experiment. This enables both a qualitative and a quantitative analysis of the results, that are presented next. When comparable data was available, we also tested the results using ANOVA and Tukey's post-hoc analysis when necessary.

6.1. Qualitative analysis

We discarded the data of a participant who did not follow the instructions. We also had a problem when gathering data for one participant: the file where data was saved did not update properly, and we lost the last 5 searches in the large molecule scenario with the *Ray + colors* technique. We decided to keep the correctly recorded data since the participant had not experimented any interaction issue during the experiment.

The new versions of the progressive techniques were widely approved by the participants. When asked about the comfort and preference (comparing the usefulness of the technique regarding using the ray alone), users graded our new techniques above raycasting, as it can be seen in Figure 9 (left and center). Comfort of the techniques showed significant differences (p-value = 0.015, F = 4.70) and Tukey determined Ray + colors was significantly more comfortable than Ray alone. Preference also yielded significant differences (p-value = 0.004, F = 6.522) and post-hoc analysis showed both Ray+Colors and Ray+Arrows were significantly preferred higher. Regarding learnability, raycasting still stays as the best technique (right), with significant differences (p-value =0.038, F = 3.587), but the Tukey posthoc test only confirms those for the pair Ray and Ray+Arrows. We somewhat expected ray alone to be ranked higher because the other two techniques also use the ray for selection, and then require an extra step that must be learned. In general, users preferred colors to arrows, partly because of the design. The arrows are located in front of the atom they refer to, and can sometimes be hidden from other atoms by a bond or by transformations applied to the molecule. In addition, we were surprised to see that some users rapidly got used to the color order, and as a consequence, stopped consulting the informative widget. This eliminates the initial advantage of the arrows of having an associated direction.

Users commented that the ray alone fell short on certain occasions, and that they appreciated having the other techniques to refine the search. However, despite the expressed user's preference, our quantitative analysis (see Section 6.2) shows that raycasting usually requires shorter times to perform the tasks.

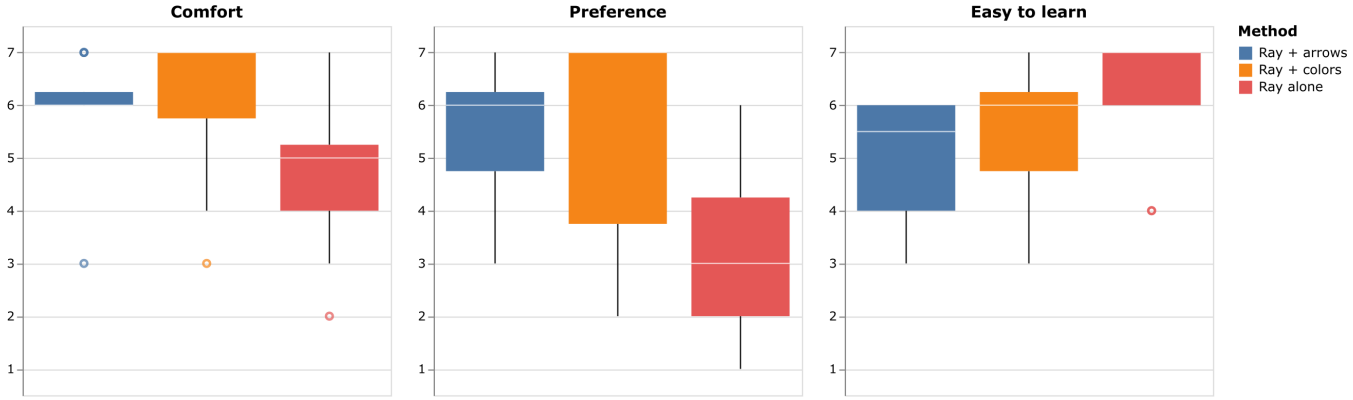

Figure 9: *Our improved implementation of the techniques exhibit a better perceived comfort (left), preference (center), and ease of use (right) than the first implementations. Moreover, both helper techniques (blue and orange) are felt more comfortable and useful than simply using the ray. In terms of time, ray alone (red) is usually faster.*

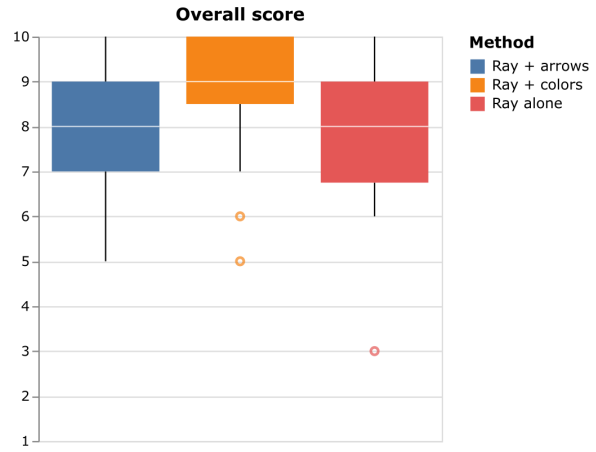

Figure 10: *Overall score in our improved implementation. In this case, Ray + colors (8.58) was the preferred one over Ray + arrows (8.08). Raycasting was the least valued technique (7.67).*

In line with the previous responses, the scoring questionnaire also shows that the new implementations of our progressive selection methods are appreciated by the users, but no significant differences were found. In the case of the *Ray + colors*, the method received higher ranking than raycasting. Note, however, that there are two outliers in this overall score (see Figure 10). Raycasting received the lower average grade (7.67), while the *Ray + colors* got 8.58, and was the preferred technique. *Ray + arrows* was still over raycasting, and obtained an 8.08.

6.2. Quantitative analysis

In addition to the qualitative analysis, we were also interested in the actual performance of the participants when using the different methods. So, we measured the time required to select each atom for all the task. Selection time was measured from the moment the menu button was clicked to start a search, until it was clicked again, after hearing the feedback sound when selecting the right target.

Calculating rough averages among all tasks would somewhat lose detail because not all atoms are equally accessible. As explained, we purposely selected some close and some distant atoms, even one which was especially occluded. Thus, to better understand the user performance, we plot the times per atom in a line chart, that plots every atom selection time in the same order they were performed. In Figure 11, on the left, we can see the times to select the atoms in the small molecule, while the right chart shows the times required to select the atoms in the larger protein. Fully opaque lines indicate the median of the observations (since it is more robust to outliers), and the background semi-transparent area charts represent the confidence intervals. The first impression, when we see the charts next to each other, is that the selection in the large protein takes slightly higher, which is to be expected. But the time differences are minimal, which is a bit surprising.

We now proceed to dig into the details of the small molecule (Figure 11-left). The second insight we obtain is that, as expected, when atoms are closer to the previous selection (atoms 2, 5, 7, and 9, in the small molecule, as described earlier), the subsequent selection is faster, as can be seen through the peaks followed by a descending slope. This is especially notable that atom 8, which was purposely selected as an internal atom, quite occluded. In this case, users took quite a long to achieve the selection. We can see more detail on the required times to select each of the atoms for the smaller protein in Figure 12. Atoms that are closer to the previous selection require less time to acquire with any of the techniques, with no technique clearly superior to the others. Distant object acquisition is slower, and also typically presents a larger number of outliers. As an example, atom 8, besides taking more time, also has larger variance in its results. The same behavior repeats for the larger protein, as shown in Figure 11-right and 13: closer atoms (2, 5, 7, and 9) require less time to be selected than their predecessors. And distant atoms (4, 6, and 8) exhibit larger times, variances, and outliers.

To show more insights on the time required to select non-occluded versus occluded atoms, we generated a boxplot of the

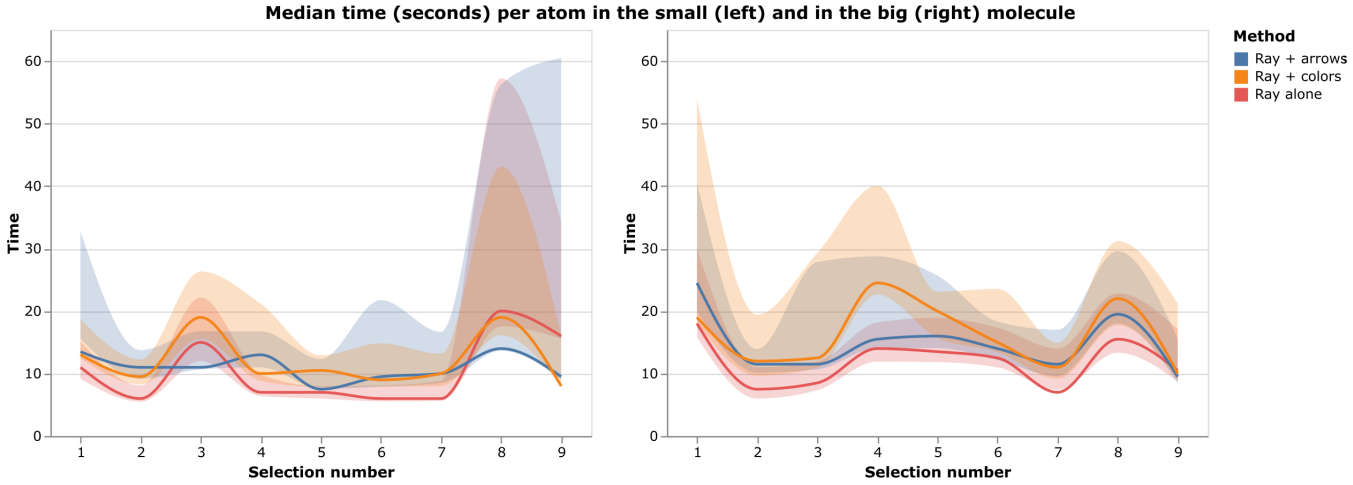

Figure 11: *The lines show the median time (seconds) per atom selection, with monotonic interpolation, and the areas show the variance in the data. On the left we have data regarding the small molecule and on the right about the big molecule. The peaks and descending slopes clearly show that when atoms are far and followed by a closer one, the first selection takes more time, and the second, less. We also see the effect in time of an atom being occluded, as in the eighth atom of the small molecule, that requires significantly larger to be picked.*

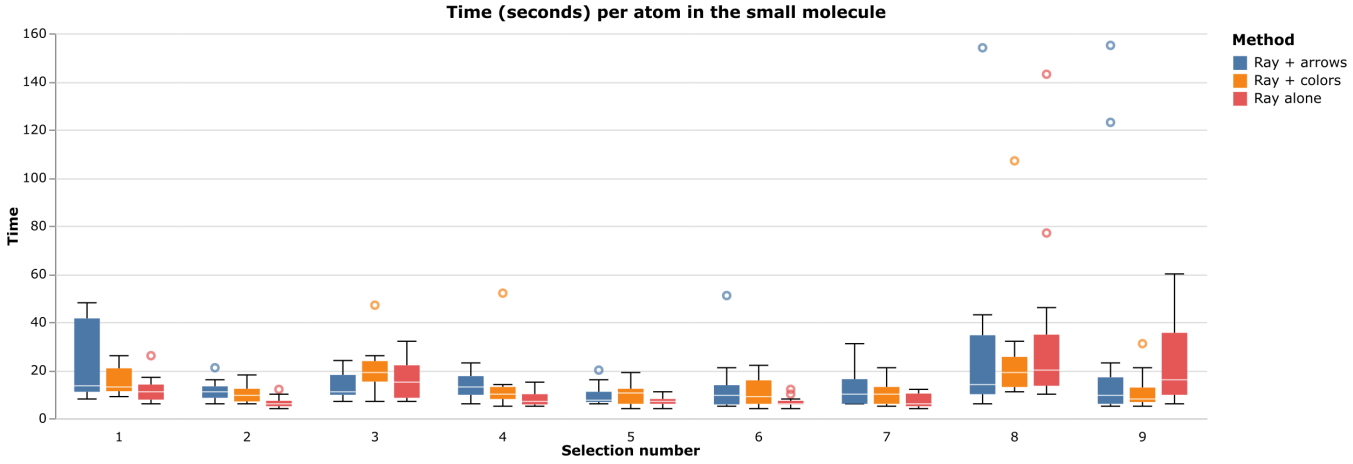

Figure 12: *Time (seconds) required to make the 9 selections for each technique in the* ***small protein****: Raycasting (red), Ray + arrows (blue) and Ray + colors (yellow). Note how atoms 2, 5, and 7 require less time to be selected, since they were close to the previous one. Atom 8 was highly occluded, which causes higher acquisition times and larger variances, and 9 close to the previous one, with reduced times again.*

times, partitioned based on the occlusion condition, as shown in Figure 14. We can see that the three techniques require larger times, and exhibit larger variances than the same techniques when used for the selection of non occluded atoms, even if they are distant. We also analyzed the time distributions for significance and found that, indeed, those present some differences. For the small molecule, ANOVA yielded a p-value of 0.025, F = 3.703, and Tukey confirmed that the ray was significantly faster than Ray+Arrows. Ray and Ray+Colors show no differences. The same happens for the large molecule (p-value = 0.001, F = 6.815), and in this case, the Ray alone is significantly faster than the other two techniques.

7. Discussion

The enhanced interaction techniques achieved higher appreciation (comfort, preference, and overall grading), although Raycasting remained the most intuitive. The technique that ended up with a higher overall score was *Ray + colors* (8.58 out of 10), while Raycasting achieved the lowest (7.67 out of 10). The analysis of quantitative data also shows a difference between atoms with greater occlusion and atoms clearly exposed. Occluded ones seem more difficult to achieve with the ray alone method.

From these two studies, we can extract that the Raycasting technique benefits from some method of refinement for the cases in which the occlusion prevents a correct direct selection, or for cluttered scenes where the initial selection may be wrong, but close to

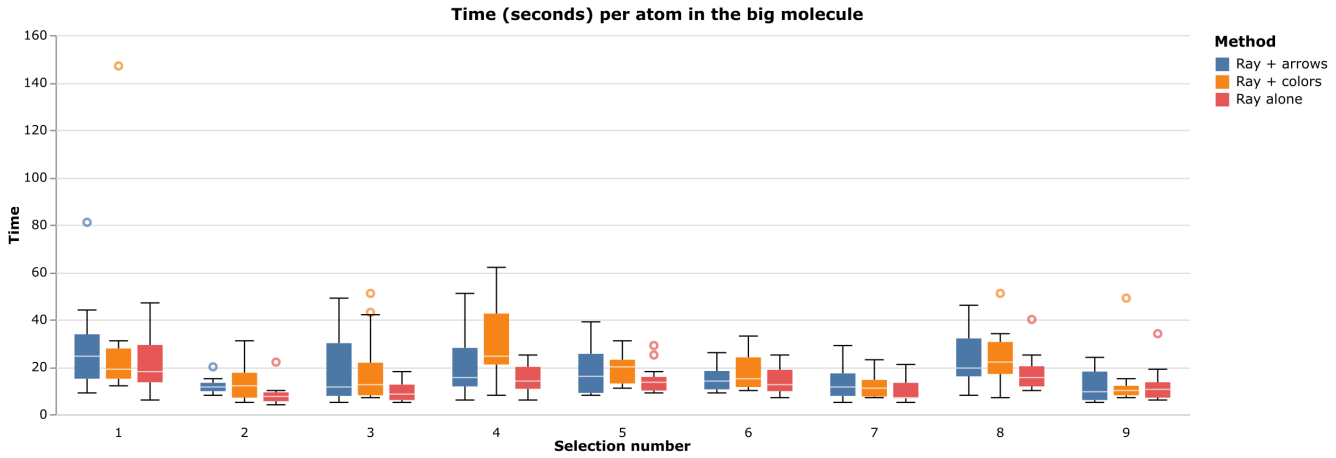

Figure 13: *Time (seconds) required to make each of the 9 selections for each technique in the case of the* ***large protein****: Raycasting (red), Ray + arrows (blue) and Ray + colors (yellow). Like in the previous case, distant atoms (e.g., 4, 6, and 8) require larger times than their closer counterparts (e.g., 2, 7, and 9), which are generally acquired in shorter times.*

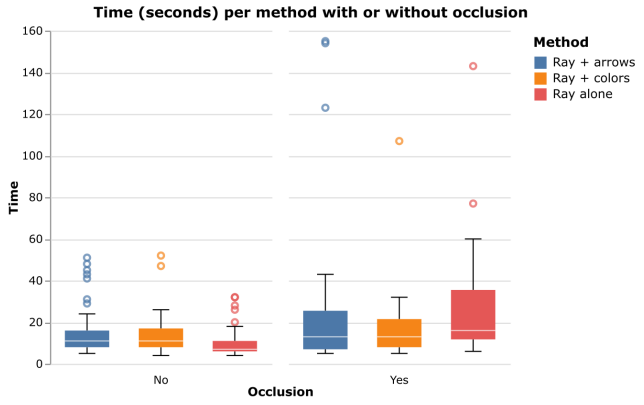

Figure 14: *Selecting visible versus occluded atoms in the small molecule. We can see how non occluded targets (atoms 1-7) require significantly less time to be selected than occluded ones (atoms 8-9). The technique that uses only the ray (red) would be faster than the ones that may use auxiliary methods (blue and yellow) for the exposed atoms. But they seem to help for occluded ones. Note that the sample size of the occluded is smaller.*

the final desired target. Although these additions lead to a longer training time, the results show that it is compensated by the selection time and user appreciation. In addition, users end up getting used to the techniques, and they did not need to check the direction widget to match colors and directions. Furthermore, in the case of prolonged sessions, the arm could rest since the neighbor navigation does not require pointing at the virtual space. Statistical analysis has also shown that using only the ray tends to achieve better times. However, the users feel significantly more comfortable with the other techniques. As a result, our advice would be to use the *Ray + colors* technique, although we want to continue with new studies.

8. Conclusions and Future Work

In this paper, we presented two new target acquisition techniques for accurate selection of small elements in cluttered scenes. A pilot study was initially carried out, which showed both techniques were understandable and easy to use. Users still seemed to prefer the ray interaction instead of the newly developed techniques, albeit recognizing that it was difficult to use when molecules are huge. With the lessons learned from the pilot study, we modified the visual feedback of both techniques in sensitive ways. And then, a formal study was prepared to deeply explore the relevant cases of interest. These include scenes with a more significant number of atoms and, thus, smaller spheres. The results show that these new techniques, although slower than ray alone, were preferred and found more comfortable by the participants. In the future, we want to further explore the amount of movement, number of movements and hops, errors, and fatigue of all the presented methods. Furthermore, we would like to have color-blind users to test our application. It would also be interesting to explore the effect of limiting the movement that the user can make in the room and compare the results to a free motion scenario.

Acknowledgments

This paper has been supported by TIN2017-88515-C2-1-R(GEN3DLIVE), from the Spanish Ministerio de Economía y Competitividad and PID2021-122136OB-C21 from the Ministerio de Ciencia e Innovación, by 839 FEDER (EU) funds. Elena Molina has been supported by FI-SDUR doctoral grant from Generalitat de Catalunya, and FPU grant from the Ministerio de Ciencia e Innovación.

References

[AA09] ARGELAGUET, FERRAN and ANDUJAR, CARLOS. "Efficient 3D pointing selection in cluttered virtual environments". *IEEE Computer Graphics and Applications* 29.6 (2009), 34–43 2.

[AA13] ARGELAGUET, FERRAN and ANDUJAR, CARLOS. "A survey of 3D object selection techniques for virtual environments". *Computers Graphics* 37.3 (2013), 121–136. ISSN: 0097-8493. DOI: https://doi.org/10.1016/j.cag.2012.12.003. URL: https://www.sciencedirect.com/science/article/pii/S0097849312001793 2.

[BKW*13] BYRNE, AIMEE, KIER, BRANDON L., WILLIAMS, D. V., et al. "Circular permutation of the Trp-cage: fold rescue upon addition of a hydrophobic staple". *RSC Adv.* 3 (43 2013), 19824–19829. DOI: 10.1039/C3RA43674H. URL: http://dx.doi.org/10.1039/C3RA43674H 4.

[BPC19] BALOUP, MARC, PIETRZAK, THOMAS, and CASIEZ, GÉRY. "Raycursor: A 3d pointing facilitation technique based on raycasting". *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 2019, 1–12 2.

[DCP*14] DOUTRELIGNE, SÉBASTIEN, CRAGNOLINI, TRISTAN, PASQUALI, SAMUELA, et al. "UnityMol: Interactive scientific visualization for integrative biology". *2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV)*. 2014, 109–110. DOI: 10.1109/LDAV.2014.7013213 1, 2.

[DFTC98] DOMINGUEZ, R., FREYZON, Y., TRYBUS, KM., and COHEN, C. "Crystal structure of a vertebrate smooth muscle myosin motor domain and its complex with the essential light chain: visualization of the pre-power stroke state." *Cell* 94.5 (1998), 559–571. DOI: https://doi.org/10.1016/S0092-8674(00)81598-6 5, 6.

[EF08] ELMQVIST, NIKLAS and FEKETE, JEAN-DANIEL. "Semantic pointing for object picking in complex 3d environments". *Proceedings of graphics interface 2008*. ACM. 2008, 243–250 2.

[GB06] GROSSMAN, TOVI and BALAKRISHNAN, RAVIN. "The design and evaluation of selection techniques for 3D volumetric displays". *Proceedings of the 19th annual ACM symposium on User interface software and technology*. 2006, 3–12 2.

[GBS*18] GODDARD, THOMAS D., BRILLIANT, ALAN A., SKILLMAN, THOMAS L., et al. "Molecular Visualization on the Holodeck". *Journal of Molecular Biology* 430.21 (2018), 3982–3996. ISSN: 0022-2836. DOI: https://doi.org/10.1016/j.jmb.2018.06.040. URL: https://www.sciencedirect.com/science/article/pii/S002228361830696X 1.

[KBB11] KOPPER, REGIS, BACIM, FELIPE, and BOWMAN, DOUG A. "Rapid and accurate 3D selection by progressive refinement". *2011 IEEE symposium on 3D user interfaces (3DUI)*. IEEE. 2011, 67–74 2.

[KBL*19] KINGSLEY, LAURA J., BRUNET, VINCENT, LELAIS, GERALD, et al. "Development of a virtual reality platform for effective communication of structural data in drug discovery". *Journal of Molecular Graphics and Modelling* 89 (2019), 234–241. ISSN: 1093-3263. DOI: https://doi.org/10.1016/j.jmgm.2019.03.010. URL: https://www.sciencedirect.com/science/article/pii/S1093326318303929 1.

[LBO*20] LAUREANTI, JOSEPH, BRANDI, JUAN, OFFOR, ELVIS, et al. "Visualizing biomolecular electrostatics in virtual reality with UnityMol-APBS". *Protein Science* 29.1 (2020), 237–246. DOI: https://doi.org/10.1002/pro.3773. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pro.3773. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/pro.3773 1.

[LYS20] LU, YIQIN, YU, CHUN, and SHI, YUANCHUN. "Investigating bubble mechanism for ray-casting to improve 3D target acquisition in virtual reality". *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE. 2020, 35–43 2.

[MSD*18] MARRIOTT, KIM, SCHREIBER, FALK, DWYER, TIM, et al. *Immersive analytics*. Vol. 11190. Springer, 2018 1.

[MVÁ13] MONCLÚS, EVA, VÁZQUEZ, PERE-PAU, and ÁLVARO, ISABEL NAVAZO. "DAAPMed: A data-aware anchor point selection tool for medical models in vr environments". *International Conference on Computer Graphics Theory and Applications*. Vol. 2. SCITEPRESS. 2013, 308–317 2.

[NGEB15] NORRBY, MAGNUS, GREBNER, CHRISTOPH, ERIKSSON, JOAKIM, and BOSTRÖM, JONAS. "Molecular Rift: Virtual Reality for Drug Designers". *Journal of Chemical Information and Modeling* 55.11 (2015). PMID: 26558887, 2475–2484. DOI: 10.1021/acs.jcim.5b00544. eprint: https://doi.org/10.1021/acs.jcim.5b00544. URL: https://doi.org/10.1021/acs.jcim.5b00544 1.

[PGH*21] PETTERSEN, ERIC F., GODDARD, THOMAS D., HUANG, CONRAD C., et al. "UCSF ChimeraX: Structure visualization for researchers, educators, and developers". *Protein Science* 30.1 (2021), 70–82. DOI: https://doi.org/10.1002/pro.3943. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pro.3943. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/pro.3943 1.

[RO13] REN, GANG and O'NEILL, EAMONN. "3D selection with freehand gesture". *Computers & Graphics* 37.3 (2013), 101–120 2.

[SP04] STEED, ANTHONY and PARKER, CHRIS. "3D selection strategies for head tracked and non-head tracked operation of spatially immersive displays". *8th international immersive projection technology workshop*. Vol. 2. 2004 2.

[WKE*18] WILD, REBEKKA, KOWAL, JULIA, EYRING, JILLIANNE, et al. "Structure of the yeast oligosaccharyltransferase complex gives insight into eukaryotic N-glycosylation". *Science* 359.6375 (2018), 545–550. DOI: 10.1126/science.aar5140. eprint: https://www.science.org/doi/pdf/10.1126/science.aar5140. URL: https://www.science.org/doi/abs/10.1126/science.aar5140 4.

[WMG*99] WATSON, K.A., MCCLEVERTY, C., GEREMIA, S., et al. "Phosphorylase recognition and phosphorolysis of its oligosaccharide substrate: answers to a long outstanding question". *The EMBO Journal* 18.17 (1999), 4619–4632. DOI: https://doi.org/10.1093/emboj/18.17.4619. eprint: https://www.embopress.org/doi/pdf/10.1093/emboj/18.17.4619. URL: https://www.embopress.org/doi/abs/10.1093/emboj/18.17.4619 5, 6.