HAL
open science

3D Object Position using Automatic Viewpoint Transitions

Michael Ortega

To cite this version:

Michael Ortega. 3D Object Position using Automatic Viewpoint Transitions. CHI 2013 - International Conference on Human-computer Interaction, Apr 2013, Paris, France. pp.193-196, 10.1145/2470654.2470681 . hal-00871566

HAL Id: hal-00871566

https://hal.science/hal-00871566

Submitted on 15 Oct 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

3D Object Position using Automatic Viewpoint Transitions

Michaël Ortega
Laboratoire d'Informatique de Grenoble
Université Grenoble I / CNRS
385, rue de la Bibliotheque, B.P. 53 F-38041 Grenoble cedex 9, France
michael.ortega@imag.fr

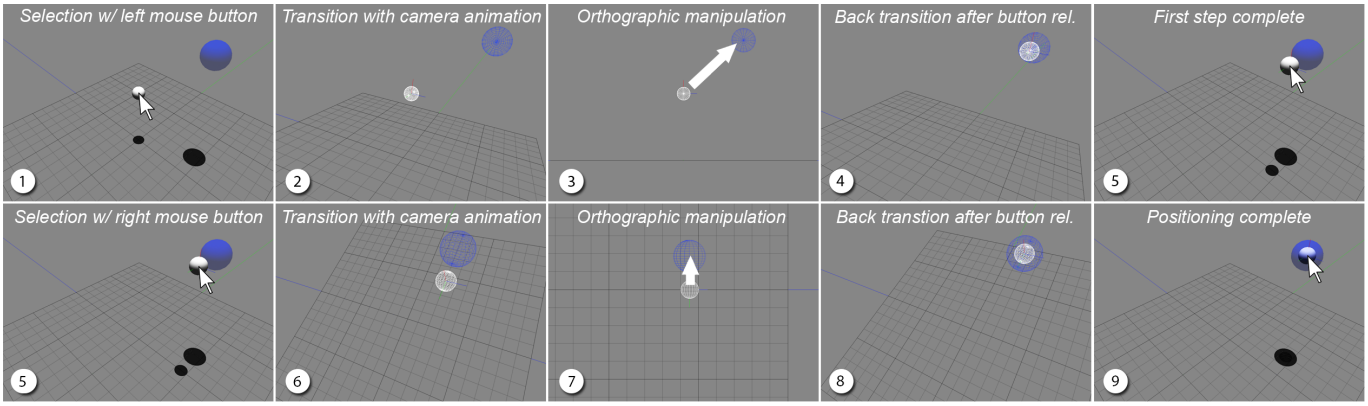

Figure 1. Steps with IUCA. 1: Selection of the object with the left mouse button down;1 to 3: Animation of the camera parameters (extrinsic and intrinsic); 3: Manipulation of the object to reach the target, then button release; 3 to 5: Reverse animation; 5: Second selection of the same object with right mouse button down; 5 to 7: Second animated transition; 7: Manipulation of the sphere then mouse release; 7 to 9: Reverse animation.

ABSTRACT
This paper presents IUCA (Interaction Using Camera Animations), a new interaction technique for 3D objects manipulation. IUCA allows efficient interaction in a full-resolution perspective view by integrating transients animated transitions to orthographic views *into* the manipulation task. This provides an interaction in context, with precise object positioning and alignment. An evaluation of the technique shows that, compared to the classical configurations, IUCA allows to reduce pointing time by 14% on average. Testing with professional 3D designers and novice users indicate that IUCA is easy to use and to learn; and that users feel comfortable with it.

Author Keywords
3D interaction; Four-views configuration; 3D modeling; 3D pointing; Fitts' law

ACM Classification Keywords
H5.2 Information interfaces and presentation: User Interfaces - Graphical user interfaces

Copyright ACM, (2013). This is the author's version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in Proceedings of the 2013 conference on Human factors in computing systems (CHI 2013). http://doi.acm.org/10.1145/2470654.2470681

INTRODUCTION
3D designers need perspective and orthographic views to complete precise objects manipulation. In the perspective view (PV), the scene is rendered using a center of projection (the position of the virtual camera). Objects in the distance appear smaller than objects close by. This provides depth perception, and contributes to the realism and the perception of relative 3D positions. This view is mainly used for context. In contrast, orthographic views (OV) uses projection lines which are orthogonal to the projection plane. The point-of-view is then free from parallax and also from depth effects. This allows precise positioning and easier alignment of 3D objects.

The current approach, called the "four-views configuration" (4VC) [8], includes these two projection methods. 4VC displays 4 different views of the scene at the same time, allowing manipulation in each of them. It is composed by three OVs (usually front, top and left views) and one PV. However, it presents the following limitations:

- Reduced display and manipulation space. Workspaces are reduced to the currently selected view. This forces the designer to compensate by frequently adapting the zoom factor of each view.

- Cognitive overload. The use of multiple views increases the load on the users' working memory and requires additional cognitive efforts for comparison and context switching [12, 4] (the effort increases with heterogeneous representations: solid and wireframe).

- Selection difficulty. The selection of a non-lighted wireframe object in the OVs present a high index of difficulty (in terms of Fitts' Law). Moreover, lines of different objects frequently align, resulting in occlusion.

4VC is often used by novices. Because of its workspace issues, more expert users prefer another configuration called here the "Switch View Configuration" (SVC). SVC only offers one view at a time in full-screen resolution, and allows switching between views with keyboard shortcuts. SVC, however, only copes with the first of the three issues introduced above.

We design, implement and evaluate IUCA (Interaction Using Camera Animations), a new technique to fluently switch between views during manipulation. IUCA alleviates these previous limitations by providing: (1) a main PV for context overview in full-screen resolution; (2) easier objects selection in PV before switching to other views; (3) semi-automatic selection of the OVs for manipulation; (4) animated view transitions integrated *into* the manipulation task (see Figure 1). We report on a user study with novice users, and on a presentation of IUCA to 3D designers. The results of the study show that IUCA improves performance compared to SVC and 4VC. Post-study interviews revealed that users, especially professionals, feel comfortable using IUCA.

RELATED WORK

Many studies have been proposed for improving 3D manipulation in a single PV, and a description of most of them can be found in Bowman et al. [2]. These studies mainly aim at suppressing the need of orthographic projections, and focus on providing accurate manipulation in context (all in the PV). However, 3D designers still need orthographic projections that are free from parallax and depth cues, and that better support precise manipulation, object alignments and quick size comparison.

A first step in integrating OV advantages in the PV was presented by Herndon et al. [7] with the "Interactive shadows" technique. The technique allows user to manipulate 3D objects in a PV via their projected shadows on a floor plane and on two walls. The technique, however, can not entirely replace OVs for precise manipulations because it is sensitive to degenerated configurations, when manipulation axes are orthogonal to the viewport projection plan for instance. Moreover, designers of complex scenes do not want to systematically use floor planes or wall planes. The technique has been adapted by Ayatsuka et al. [1] for augmented reality environment, but with the same limitations.

Recently, two studies [5, 10] introduced the use of animated transitions between views, building on the results of Robertson et al. [11]: animated transitions assist in preserving the spatial relationships between views; they shifts some of the user's cognitive load to the human perceptual system, therefore reducing the cognitive overload and overall interaction time. Grossman et al. [5] introduced a technique for constructing 3D curves by drawing a series of 2D curves with the 2D tape drawing technique. In this configuration, working in an OV is essential for precise manipulation of the points defining the curve. Designers also have to frequently switch to PVs to get a global view. The authors chose to animate the transitions between the views, and demonstrated the benefits. However, in contrast with IUCA, animations are not integrated into the manipulation task: designers have to do an extra action for switching, which is time consuming. The same limitation is present in the "ViewCube" technique proposed by Khan et al. [10]: an orientation indicator of the view can be manipulated for fluently changing the view. Moreover, object selection is done in OVs, a high difficulty task as discussed above.

In all these studies, the viewpoint is changed before and after the object manipulation. IUCA introduces a viewpoint switch *during* the object manipulation. IUCA exhibits several benefits that will be detailed in the following paragraphs: 1) selection is easier because it is always performed in the PV; 2) animations maintain the context and help users orient themselves; 3) user performance is improved; and 4) users mainly prefer IUCA because they perceive the 3 previous benefits that improve the comfort in manipulation.

IUCA

IUCA consists of integrating transient animated transitions between views "into" the manipulation task. It is composed of 4 steps:

1. Selection of both the object and the OV. Selecting an object, in the PV, with left or right mouse click selects a group of possible OVs: respectively vertical (including front, left or right views) or horizontal (including top or bottom views). The system then automatically choose one view in the selected group. The selected view is the one which projection plan is the *most parallel* to the current camera's projection plan.
2. Animated transition. Camera parameters are animated for a transition from the current perspective projection to the selected orthographic one. This is done in two steps. First, extrinsic parameters are animated: the direction of the camera is modified to become orthogonal to the selected plan, its position is modified to have the selected object in the middle of the screen at the same distance than before the selection (see Figure 2). Secondly, intrinsic parameters are animated to achieve a parallel projection (see Figure 3). We conducted informal studies that revealed that these two steps are better performed one after the other in this order to offer comprehensible transitions.
3. Object Manipulation in OV. After the animation, the camera keeps the orthographic projection until the manipulation is done (i.e. until the mouse button is released).
4. Animated back transition. Once the manipulation is done, the camera is animated back to its original PV (i.e. original intrinsic and extrinsic parameters).

In our implementation of IUCA, left or right mouse click respectively select the vertical or horizontal OV group. This could be done with a keyboard shortcuts depending on the 3D designers convenience.

The mouse cursor disappears during the manipulation, and comes back after the return animation. This is critical for the pointing movements as the cursor always reappear at the position of the selected object (see Figure 1). The user can

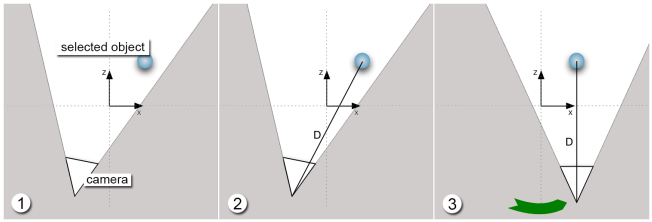

Figure 2. Extrinsic animation. The distance D between the camera and the object is preserved. The camera rotates until becoming aligned with the chosen orthographic projection plane, and having the selected object in the centre.

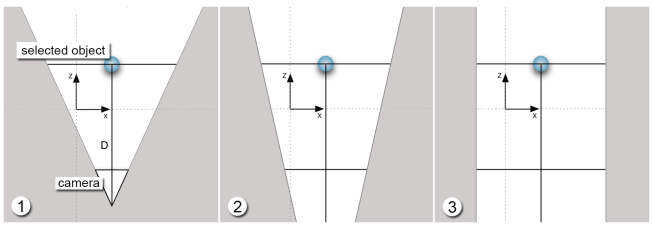

Figure 3. Intrinsic animation. The orthographic proj. surface is defined by the intersection of the camera's clipping volume with the plane defined by the object position and X and Y. Object ratio on screen is kept.

then concatenate the manipulations of a same object, with few mouse movements.

EVALUATION

We conducted a study to evaluate IUCA for object positioning in a 3D environment, in terms of pointing time, usability and user comfort. The study involves the most frequent task in 3D design: translating an object from its initial position to another position. Most of 3D designers need high precision in positioning (3D architectural design for instance), and this cannot be done in a single PV. We compared IUCA to the standard 4VC and SVC techniques. For the quantitative results (pointing time), the proposed task is not ecological: the 3D environment is minimized in order to avoid any bias. Professionals used IUCA in an ecological way, with an Autodesk Maya Plugin, in order to provide more realistic qualitative results.

In the quantitative part, participants had to perform successive 3D pointing tasks. They selected a small sphere with a fixed radius, and brought it into another sphere. Movement time (*MT*) was measured from the participant's click on a start button, until the last mouse button release. Transition animations had the same duration for each task with the first animation at 0.5 s. and the return animation at 0.2 s. These durations seems comfortable.

The difficulty of the task was computed using the adaptation of the Fitts' law proposed by Kabbash et al. for area cursors [9], and adapted for volumes as Hancock et al. did [6].

$$ID = \log_2\left(\frac{D}{2*(Rt-Rs)}+1\right) \quad (1)$$

4 difficulties were presented: 3, 5, 6 and 7. Difficulties are obtained by changing the target size and the distance between the target and the small sphere in 3D. Considering *Rs*: the radius of the small sphere, *Rt*: the target radius, and *D*: distance between both spheres in 3D, the difficulty *ID* is obtained by the formula (1). For each configuration, 80 tasks were computed and each difficulty is equally and randomly presented. Participants had to complete all the tasks of a configuration before changing to another. A training session of 8 trials (2 trials per difficulty) was performed before each configuration. In each configuration, the 4 difficulties were equally and randomly presented, and the randomness was different for each user. The configurations were counterbalanced for each subject using a latin square ordering.

For the qualitative evaluation, we collected participant's opinions with a questionnaire regarding comfort, perceived performances, possible tiring and disturbing effects. We used a Brooke System Usability Scale (*SUS* [3]) to compute a quantitative estimation on the usability of IUCA. We wanted to know if IUCA is usable, and we estimated that 4VC and SVC are, just because they are used of lot.

Participants 15 regular computer user (with ages in the range [25, 35]) participated in the experiment. They were all member of the laboratory and had no 3D modeling experience. We enrolled novice users in 3D modeling to avoid biases on previous 4VC and SVC experience.

Apparatus The experiment was run on a 30" screen with a resolution of 2560x1600 pixels. Participants were in a desktop configuration mode. They sat in front of the screen during the experiment. The only interaction devices were a classical 2D mouse with two buttons, and a keyboard. The questionnaire was filled after the experiment.

RESULTS

Movement Time Figure 4 shows the average *MT* for each configuration and for each *ID*. In all configurations, the average *MT* increases with the difficulty, as suggested by Fitts' law. For each difficulty IUCA is faster than 4VC and SVC. A repeated measures ANOVA shows that the configuration has a statistically significant effect ($F_{1,585} = 226.8$, $p < .0001$), and this is also true for each participant. On average, IUCA is 16.6% faster than the 4VC, and 13.8% faster than SVC. Moreover, thanks to constant animation times (1.4sec for each *MT*) IUCA does not only improve task accomplishment time, but also depth perception and then the perception of relative positioning of 3D objects.

Error Rate For the three configurations, error rate increased with *ID*. This confirms Fitts' law again. The 3 configuration had similar error rate, and a Chi^2 test did not reveal any significant difference.

Novice Opinions The Brooke SUS provided a numerical evaluation of the usability of IUCA. The global mean rate is 3.69/4.0 (= 92%). IUCA was thus assessed as "easy to learn and easy to use". Most of the participants seemed comfortable using it. However, half of participants judged that the system was, or could be, perturbing and tiring *at the beginning*. We discuss this problem in the next section.

Professional Opinions 4 professionals, aged between 20 to 42 years old, tested the Autodesk Maya plugin of IUCA. They were recruited from 4 different companies providing various services around 3D modeling: architecture, 3D models, an-

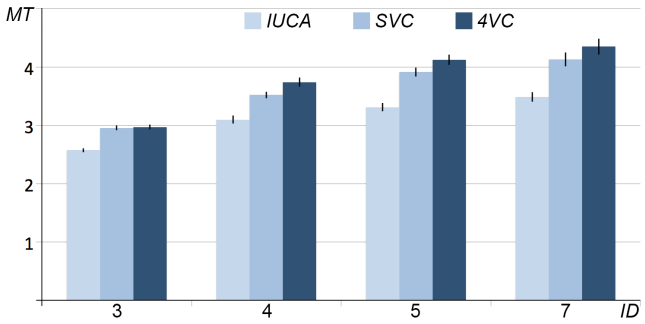

Figure 4. Average Movement Time (*MT*) for each *ID*. For each configuration, *MT* increases as *ID* increases, but in IUCA *MT* is always significantly lower ($p < .0001$). IUCA is 17% average faster than 4VC, and 14% average faster than SVC. (Error Bars: +/- 1SE)

imation, design, ... Two of them used 3DSmax Autodesk Software, and the two others used Blender (open-source and freeware). All were familiar with 4VC and SVC. After some time provided to understand, manipulate and integrate IUCA, the 4 participants filled the Brooke SUS: the global mean rate was $3.62/4.0$ (=91%). IUCA appears usable, useful, easy to learn and not tiring. When asked about their opinions, professionals expressed that they "easily recover their previous habits" (P4), and "have a real feeling of winning time" (P2). P1 "would like to use it every day for modeling". P3 needed more time than others for understanding the technique (this is discussed in the next section), but after 10min he started to model a complex scene and also expressed an interest on using IUCA every day.

LIMITATIONS AND FUTURE WORK

The evaluation provides encouraging results on the benefits and potential of IUCA. However, some limitations exist, and professionals have noticed it when using IUCA in "real-world" tasks. The first one is related to repetitive tasks. An example is the adjustment of the position of several points in a 2D curve. Manipulating each point, one by one, implies many successive animations while the user would rather like to switch once to the appropriate OV at the beginning, and switch back at the end. The current version of IUCA does not allow staying on an OV for a while. Some solutions based on a pseudo-mode, like holding a mouse button, will be investigated.

Another limitation is that multiple selection are not possible. A solution could be the inspired by WIMP interaction: with a click in the background and a mouse drag to select many objects with an elastic box. Then, a click on one of the selected object launches IUCA, and the selection is considered as one object. This will be studied with the Maya plugin.

The unpredictability of the maximum translation distance in OVs is also a limitation. For instance, when the moving object reaches the border of the OV, the user has to release the mouse, to change the camera parameters in the PV, and to re-launch IUCA for finishing the task. A first solution could be "automatic camera translations" when an object reaches the border of the OV. But this requires further studies.

Finally, some users were perturbed by IUCA *at the beginning*, i.e. before they well understood and anticipated animations. Indeed, when the pointing task starts, camera animation implies target movements. The 2D projections of the direction between the starting position and the target are different before and after the animation (as visible in Figure 1-5 and Figure 1-7), and could be totally opposed. During the training phase, some users started their pointing movement in the "before animation" direction, and had to correct it. But after a training time, all users managed to anticipate. This learning effect will be studied by observing the evolution of the cursor movements during training.

CONCLUSION

IUCA improved user performance by integrating transient animated transitions into the manipulation task. The technique proposes an optimal use of the screen resolution, better performances in pointing time than classical configurations, and easier objects selection. We will now focus on the areas of improvement proposed in the previous section, and we will investigate orientation tasks.

ACKNOWLEDGMENT

Thanks to Laurence Nigay, Renaud Blanch and François Bérard (LIG) for their fruitful reviews, and to Nadine Mandran (LIG) for helping with the statistics. This work was supported in part by a BQR grant from Grenoble-INP (INTUACTIVE project).

REFERENCES

1. Ayatsuka, Y., Matsuoka, S., and Rekimoto, J. *Penumbrae for 3D interactions*. in Proc. of UIST (1996), pp. 165 - 166.
2. Bowman, D. A., Kruijff, E., LaViola, J. J., and Poupyrev, I. *3D User Interfaces: Theory and Practice*. Addison-Wesley Professional (2005).
3. Brooke, J. *SUS: a quick and dirty usability scale*. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, and A. L. McClelland. Usability Evaluation in Industry. London: Taylor and Francis (1996).
4. Cockburn, A., Karlson, A., and Bederson, B. B. *A review of overview+detail, zooming, and focus+context interfaces*. in Journal of ACM Computing Surveys (2009), pp. 2:1 - 2:31.
5. Grossman, T., Balakrishnan, R., Kurtenbach, G., Fitzmaurice, G., Khan, A., and Buxton, B. *Creating principal 3D curves with digital tape drawing*. in Proc. of CHI (2002), pp. 121 - 128.
6. Hancock, M., Carpendale, S., and Cockburn, A. *Shallow-Depth 3D Interaction: Design and Evaluation of One-, Two- and Three-Touch Techniques*. in Proc. of CHI (2007), pp. 1147 - 1156.
7. Herndon, K. P., Zeleznik, R. C., Robbins, D. C., Conner, D., Snibbe, S. S., and van Dam, A. *Interactive shadows*. in Proc. of UIST (1992), pp. 1 - 6.
8. Johnson, T. E. *Sketchpad III: a computer program for drawing in three dimensions*. in Proc. of AFIPS (1963), pp. 347–353.
9. Kabbash, P., and Buxton, W. A. S. *The "prince" technique: Fitts' law and selection using area cursors*. in Proc, of CHI (1995), pp. 273 - 279.
10. Khan, A., Mordatch, I., Fitzmaurice, G., Matejka, J., and Kurtenbach, G. *ViewCube: a 3D orientation indicator and controller*. in Proc. of I3D (2008), pp. 17 - 25.
11. Robertson, G. G., Mackinlay, J. D., and Card, S. K. *Cone Trees: animated 3D visualizations of hierarchical information*. in Proc. of CHI (1991), pp. 189 - 194.
12. Wang Baldonado, M. Q., Woodruff, A., and Kuchinsky, A. *Guidelines for using multiple views in information visualization*. in Proc. of AVI (2000), pp. 110 - 119.