PakPao 3D: The Design and Implementation of a Three-Dimensional User Interface for an Operating System Using a Game Engine

Satidchoke Phosaard and Jessada Tanthanuch
School of Information Technology, Institute of Social Technology
School of Mathematics, Institute of Science

Suranaree University of Technology
111 University Ave., Muang District, Nakhon Ratchasima 30000, THAILAND
s@sut.ac.th, jessada@math.sut.ac.th

Abstract-**PakPao 3D is a novel 3-dimensional user interface. We propose a complement of a 3D user interface desktop metaphor of the operating system. The augmented virtual reality replaces the 2-dimensional desktop user interface, while the launched applications are still in 2D windows. This allows the user to immerse oneself in the screen, viewing as the first person, and travel through the world instead of looking at the screen as from a bird's-eye view over the desktop. To interact with the interface, basic input devices are used. Traveling in the virtual environment utilizes the keyboard, while selecting and manipulating objects employs a mouse. The distinction of this 3D interface is that the application and file icons are also true 3D objects which can be manipulated in addition to the animated and realistic environment elements added to the interface. General functionalities of the desktop can be performed including creating customized application shortcuts. To accomplish this, a state-of-the-art game engine is used to implement the interface. While introduced as an augmented virtual reality environment desktop and maintaining full usage of unaltered launched applications in 2D windows, the users found the interface attractive and demonstrated that using such an interface was enjoyable. The study shows that productivity is reduced since the interface becomes more complex; however the users still prefer to use the interface occasionally.**

Keywords: 3D user interfaces, desktop virtual environments, game engine, operating systems, human-computer interactions

I. INTRODUCTION

It is undeniable that operating system interface is one of the most common interfaces with which computer users interact. It is in almost every single consumer computer everywhere. The most common graphical user interface nowadays has been used for more than two decades and is known as desktop metaphor [3] [21], such as Microsoft Windows, MacOS X, KDE, GNOME, etc. The metaphor has been evolving only a bit over time mostly with the addition of shadows, lighting, transparency, gradients, etc. to the icons, buttons and the interface elements to make them more realistic and attractive. Computer scientists are finding ways to extend the user experience beyond the current two-dimensional (2D) desktop model. Adding three-dimensional (3D) window manager is one of the active methods introduced into the interface. Such a method has not been widely adopted until recently when it has been integrated into the new release of the most popular operation system in the world, Microsoft Windows. Windows Vista, the latest version of Microsoft Windows released in early 2007, has a new eye-catching task switching utility using Windows-Tab keys called Flip-3D. The next well-known 3D interface is Project Looking Glass of Sun Microsystems [24]. Another popular product is 3DNA Desktop [2] which is reaching a million downloads at download.com. The latest one is most similar to this project, PakPao 3D, which represents desktop metaphor of the operating system as a virtual reality environment. There are also several commercial products and research trying to extend the current 2D metaphor with 3D, which shows the active activities of the area.

Figure 1. PakPao 3D: The beach virtual environment.

PakPao 3D is a virtual reality interface of an operating system (see Fig. 1). Our goal is to provide an extension of user experience over the desktop environment called desktop virtual environment with common features available in 2D desktop. The timelines of virtual reality development can be found in [23]. The name "PakPao" is a Thai word derived from the characteristic of the "blowfish" as it has the ability to inflate itself. This is analogous to getting more 3D-sense of the user interface. PakPao 3D provides users with a virtual environment in the context of the user viewing the environment as the first person. Users will find the environment more spacious than the desktop environment. In 2D desktop, when the users put more and more recently used icons of files and application shortcuts on the desktop, it become more difficult to store, locate and categorize icons. In PakPao 3D, icons are modeled to be virtual. 3-dimensional objects placed in a multi-dimensional space correlating to the virtual world reference, e.g., ground, walls or trees. This lets the user have a larger collection of frequently used icons to be accessed at hand. Moreover, our proposed interface encourages the users to combine their recognition abilities to locate and categorize icons more easily. The study in Task Gallery [12] shows that the 3D user interface helps users to remember where they put their documents and windows, respectively. We assumed that the icon finding in 3D will demonstrate the same advantage.

Entertainment is introduced into the interface and we consider it a milestone. Pleasure, enjoyment and fun are essential to life, as well as having fun at work is important [4]. Using a game engine, several effects can be added to the interface with little effort. We decided to start small by bringing in simple but realistic animated elements to the environment as well as letting the Windows task bar handle the running process. By combining all features of our metaphor, we are interested in determining overall usability of the interface. The users' attitudes toward the interface in the areas of entertainment, aesthetics and usage preferences were assessed along with users' performance.

II. Related Works

Many researchers have been exploring techniques to enhance or even replace the existing 2D desktop metaphor using 3D. There are comparisons among each technique in [21] and [17]. In the following, we address the most relevant techniques following from the frequently cited works which have contributed to the development of PakPao 3D.

3DNA Desktop [2] is software available for download. The program searches icons on the desktop and makes them available in the virtual environment. The two-dimensional automatically generated icons are attached onto a transparent billboard. After the user launches the applications, working windows are presented in 2D. The ongoing task windows are tiled on the wall in the same fashion as in the Task Gallery [12]. The program includes fancy 3D graphics and animated elements. The interface does not provide a file or window manager in 3D. The interface is most similar to PakPao 3D. By analyzing the interface, we found that not all objects used to launch the applications are icons. Some of them are real-world objects, such as a telephone, which leads to user confusion. We take that into consideration in our design. Total 3-dimensional icons that look similar to the 2-dimensional icons would prevent the confusion. In 3DNA, the user cannot freely move objects in 3D manner while our icons can be freely moved in the environment, which takes the real ability of the 3-dimensional interface. Another main difference is that in PakPao the 3D user is closer to the dynamic terrain, such as water, than in 3DNA Desktop. It provides the user the ability to be more interactive with the elements and feel tighter with the interface.

The Task Gallery [12] is a 3D window manager developed by Microsoft. The concept of the interface is that the windows of running applications can be hung on the walls or put into stacks. The latter feature has been made into a commercial product as Flip-3D. The metaphor of putting windows on the wall is not compatible with our various kinds of virtual environments, so we leave the window management to the Task Bar and Windows. Task Gallery does not allow the user to employ egocentric navigation, thus not offering full 3D experience.

The next area of related work is 3D file system visualization. Computer users spend large amounts of time browsing and managing files. The classical 3D file system visualizations extend the concept of a conventional file directory tree to that of a 3D file directory cone [26]. Several file visualization and file management interfaces are presented and analyzed in [17]. The MountainView [25] is likely to be applied to our interface. However, there is no single interface that best fits our proposed interface for hierarchical file system browsing. So, we let Windows Explorer deal with the task and allow users to be able to create shortcuts to the desired files and put them in the virtual world.

All stated works were taken into consideration in PakPao 3D design in the areas of the 3D metaphor, the 3D window manager and the 3D file system management. Several other works are presented in [22], [13], and [19]. Table I presents the compilation taxonomy of selected works related to PakPao 3D bases on [21]. It shows a summarized taxonomy of the involved systems based on function and the number of dimensions of the interface. The works closer to the left of the table indicate that they are more connected to our work. The published works are cited; the rest are discussed in [17] and [21].

III. Technical Fundamentals

In this section, we present the essential concepts underlying our interface. An introduction to the 3-dimesional user interface is elaborated in [7]. We explain two main concepts significant to our interface: moving objects in three dimensions and game engine. In-depth correlation discussion between each technique and the implementation is stated in the next section.

TABLE I
Summarized Taxonomy of the Involved Systems Based on Function[a] and the Number of Dimensions of the Interface

Number of dimensions in the interface	Function						
	Operation systems				Applications		
	Metaphor	Window managers	File managers	Process managers	Search	Web Browsers	Miscellaneous
2D	Microsoft Windows, MacOS X, KDE, GNOME				File search	Internet Explorer	Applications
3D	PakPao 3D, Cube, 3DNA Desktop, 3dtop, TDFSB,	Task Gallery[12], Cube, SphereXP, Looking Glass [23]	3DFM, 3DOSX, VFS, FSN, FSV, MountainView [24], Tactile3D, TDFSB, StepTree, XCruiser	psDoom [9], LavaPS, ps3	VxInsight, Pansophica	Clara, SphereXP	OpenCroquet,

[a]The systems closer to left of the table indicate that the systems are more similar to PakPao 3D.

A. Moving Objects in 3D

Since the icons of 3D objects are shown on the monitor screen by the perspective projection, dragging an icon makes the object move parallel to the viewing plane or visible screen (see Fig. 2). In order to calculate the position of the object in the space, these mathematical formulae are used:

$$z_{\text{new}} = z_{\text{old}} + \rho \frac{d_y'}{\rho_{\text{proj}}} \cos\phi, \tag{1}$$

$$y_{\text{new}} = y_{\text{old}} + \rho \frac{d_x'}{\rho_{\text{proj}}} \cos\theta, \tag{2}$$

$$x_{\text{new}} = x_{\text{old}} - \left(\rho \frac{d_y'}{\rho_{\text{proj}}} \cos\phi \right) \frac{\tan\phi}{\cos\theta} - \left(\rho \frac{d_x'}{\rho_{\text{proj}}} \cos\theta \right) \frac{\tan\theta}{\cos\phi}, \tag{3}$$

where (x, y, z) are co-ordinates of the object in the space, ρ is the distance from the camera to the object in the space, ρ_{proj} is the visual distance from the camera to the visible screen, d'_x and d'_y are distances from the old position to the new position on the monitor screen in *x* and *y* directions, respectively, ϕ is the angle of camera tilting and θ is the angle of camera panning.

B. Game Engine

A game engine is a powerful tool for scientific research [15]. Our work demonstrates the capability of a game engine to robustly produce such realistic and advanced graphics. We use 3D Game Studio [1] as our tool since we had it in hand and it has all the graphics and other functions we need. Fig. 3, extended from [20], shows the component schema of a general game engine to illustrate our tool's framework. From the bottom of the diagram, hardware is the layer directly related to hardware devices, such as display adapter, mouse, keyboard, speaker, etc. Directly above the bottom layer is the operating system layer that is tightly interrelated with the low level APIs. The low level standard APIs, such as OpenGL, Direct3D or DirectSound, are utilized by various engines, e.g., a graphics engine, sound engine, network engine, etc. This layer defines the quality of the output, and it is the starting layer of the game engine. The next layer is the game API layer. This layer indicates all functionalities available to be used by the game developer. How realistic the characters, scene and terrain are, as well as how advanced are the effects of the game, is determined in this layer. The topmost layer is where the game is really written.

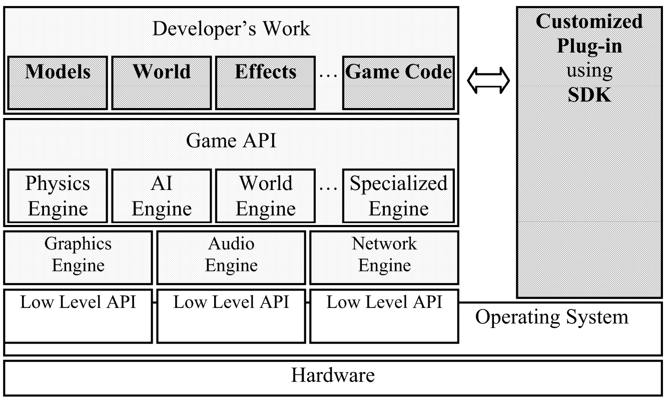

Figure 3. A modern modular game engine architecture.

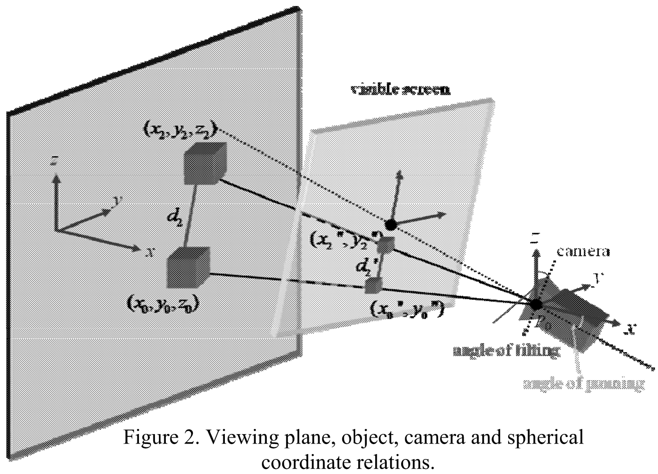

Figure 2. Viewing plane, object, camera and spherical coordinate relations.

Usually game engines come with authoring packages. We implemented the interface using built-in authoring tools and deployed it over the engine they provided.

IV. Application Realization and Implementation

Next, we illustrate the realization of the user interface according to the functional design. We remark on the issues as well as discuss how to apply the techniques introduced in the previous section.

Users should be able to achieve standard desktop operations using PakPao 3D. The requirements are selecting and dragging icons, launching the applications and creating shortcuts. In the 3D environment, navigation and camera control are required. Placing animated parts in the environment is also one of our features. The class diagram derived by the set of features is shown in Fig. 4. We executed the design utilizing the game engine development paradigm introduced in the previous section to create the interface as follows:

A. Navigation & Control

The keyboard and mouse are used to interact with the interface. Studies regarding controlling a 3D user interface using a 2D device are [6], [7] and [14]. Fig. 2 also represents our interface manipulation. The user uses "up" and "down" arrow keys on the keyboard to go backward and forward. By pressing and holding the "left" and "right" arrow keys, the pan properties of the camera are reassigned, which lets the user turn left and right, respectively. The user can use the "shift" key to accelerate the movement. By dragging the mouse up and down while holding the right click, the camera tilts upwards and downwards. The camera is freely positioned in the space. Expecting a shorter time to reach the target icons with the minimal effort of path finding, we did not apply any collision detection and allowed the camera to be passable through all obstacles. The user can fly freely in the space so that they can utilize as much 3D space as possible to store icons.

B. Icon Dragging and Manipulation

Choosing the manipulation scheme of thorough three-dimensional icons in a virtual environment for operating system interface is challenging. According to [18], there are four options to move objects in 3D space: 1) Provide three axes for the user to select the movement axis; 2) Move the object on a plane paralleled to the viewing plane; 3) Use a

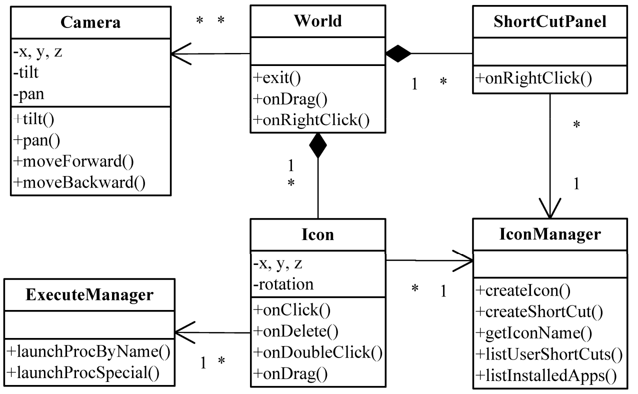

Figure 4. Simplified UML Class Diagram of PakPao 3D software design.

reference plane in the scene to determine the motion plane; and 4) Heuristically guess the movement direction based on the initial movement. We chose the second method, which is the most intuitive for desktop replacement interface. We employ the mathematics foundation given in the topic of “Moving Objects in 3D” from the last section to relocate the new icon positions.

C. *Process Launching & Task Manager*

Launching the application employs a dll plug-in. The icons are embedded with behaviors written in c-script compiled and attached to the icon by the game engine. The application launching method is invoked when the user double-clicks on the icon. The icon executes a system command by calling an outside plug-in that we have written in Delphi implementing ExecuteManager methods as shown in Fig. 4. The auto-generated icons in the virtual environment are created from the list of installed applications. This is handled by IconManager.

Certain proposed interface organizes task windows in 3D [16]. Our interface is not suitable with such a metaphor due to none of referenced walls to hang windows, so we maintain windows taskbar fashion to list application tasks.

D. *Shortcuts*

Users can create shortcuts by right clicking on the create shortcut panel. In the 3D space, using right-click to display context menu and choose to create icons causes a problem. The system does not know the depth position of the icon to be created. So we create a panel or an object, shown in Fig. 5, attached with the method that launches a wizard guiding the user through the process. The wizard imitates Windows wizard style with 3D icons for the user to choose to represent shortcuts. The created icons are placed near the create shortcut panel. Created icons can be deleted by the “Delete” key on the keyboard.

E. *Animation & Effects*

Simple animation models can be created by packing a series of different minor models into an animated model using general 3D tools. Our interface also integrates complex dynamic elements, such as waving sea and dynamic sky. The game engine empowers us to implement this more effectively.

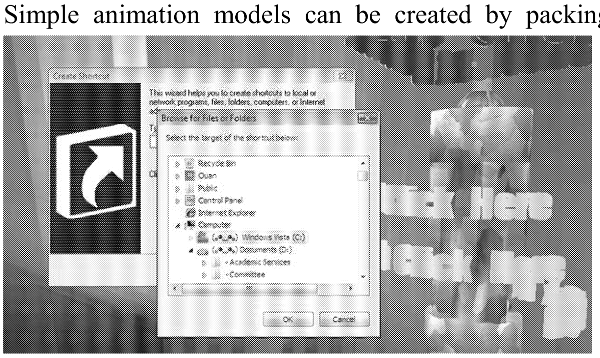

Figure 5. Create short cut wizard in PakPao 3D.

Finally, all of the components, which are models, object behaviors and plug-ins were put together in the 3D space, called level, to create complete PakPao 3D virtual environments.

V. Interface Evaluation

We conducted extensive evaluations to gain insight into our purposed interface. The cognitive walkthrough, summative evaluation, questionnaires and interview are performed to provide data in task performance and user preference metrics [8]. The ability of users’ manipulation, recognition, orientation realization and other observations were noted.

A. *Hardware and Software*

We conducted the tests using Microsoft Windows XP running on Intel Pentium 4 2.8 GHz systems with 512MB of RAM, Intel 865G chipset with Integrated Intel Extreme Graphics 2 with 96MB graphics RAM. We tested the interface with 1024x768 pixel screen resolution on 17-inch flat screen CRT monitors. PakPao 3D run smoothly on the systems.

B. *The Participants*

The participants’ targeted profiles were general computer users. The experiments did not require users to have experience in 3D manipulations. However, the participants were required to have good control of the computer mouse. Twenty four users who participated in the experiments had an average age of 20 plus 8 years of general-use computer experience.

C. *Methods and the Tasks*

Since the design aims to create a metaphor which can really be used in day-to-day operations, the users were given a set of everyday tasks that general computer users do on their computer. The users were asked, for example, to find the icons and launch applications, such as word processor, Internet browser and mail client; switch between applications; manipulate the 3D icons and create application shortcuts. Users were timed when they reached indicated points.

It was important to conduct a practicing session to obtain accurate results. A learning curve was also assessed at the introduction and practicing stage. In the introduction stage, we made it most similar to the ordinary Windows interface. Each practice interface was a plain room with walls attached by Windows desktop background. Three virtual environments were introduced in this practice stage. Users were asked to perform further tasks until they were skilled with the handling.

After the practicing session, users were presented with tasks that had to be executed on Windows XP and five virtual environments or “scenes.” All tasks randomly performed in each scene had the same quantity of work or effort. The orders of the scenes tested were randomized.

ISBN:978-988-98671-5-7

The number of icons put in the interface was controlled. The 3D virtual environment scenes are listed as follows:

Scene 1: Room – Windows XP desktop look-alike
Scene 2: Office
Scene 3: Home
Scene 4: Beach
Scene 5: Outer Space

Scene 1 was similar to the one used in the practicing stage. Scene 1 was a simple room with four walls, each of which was painted by Windows XP desktop wallpaper. Scene 2 was a virtual office. Scene 3 was a virtual home. The majority of the scene was the living room. Scene 4 was a beach. There were rocks, vibrant palm trees, sandy beach, animated sky and dynamic sea water. The last scene was an outer space look-alike. We timed the users, as well as observed, while they were performing the tasks. Attitude questions toward the overall PakPao 3D metaphor were given in the questionnaires using the 1-5 Likert scale, 5-strongly agreeing and 1-strongly disagreeing. The users were asked to rate each interface on whether they enjoyed using the interface; whether the interface was attractive; if the interface reduced the stress of computer users; if the interface was easy to use; and if the icon dragging could be performed intuitively. Other overall preferences were assessed. We also interviewed users for opinions and comments.

D. Results & Discussions

Time spent to finish tasks in Windows XP and in each virtual environment indicating user performance was analyzed using repeated measures ANOVA, as presented in Table II. Since attitude data distribution was not normal, we applied the Friedman test to investigate the differences among test conditions. The Wilcoxon test was used to perform pair-wise comparisons. The results are shown in Table III. Briefly, results and qualitative discussions are addressed as follows.

The practicing stage showed that the users could get familiar with the interface quite easily. After travelling around the first practice virtual environment for 10 minutes, the manipulation on the second and the third practicing scenes showed no significant difference.

As indicated in Table II, as we expected, user performance was reduced significantly ($F_{5, 19} = 3.956$, $p < 0.05$) when the three-dimensional user interface was used. Users spent more time to finish the tasks on the average of 40.16% when they used PakPao 3D instead of Windows XP. Even though scene 1 and 2 had obstacles that users had to perform way-finding task to find targeted icons, the results showed no significant differences in performance between scenes with more obstacles and scenes with much fewer obstacles (scene 4 and 5). The reason behind this was users could navigate through the walls as we did not apply object collision detection.

The user attitudes toward PakPao 3D were mixed among negative, neutral and positive across the aspects. We were successful in introducing enjoyment to the interface as the users ranked scene 4 (beach scene, mean rank 3.94) significantly higher ($p<0.05$) than using Windows XP (mean rank 3.31). Scene 3 (virtual home, mean rank 3.77) and scene 5 (virtual space, mean rank 3.71) were rated high as well but not significantly different from Windows XP. We found that just adding 3D into a 2D interface like in scene 1 (room with wallpaper similar to Windows desktop wallpaper) did not entertain users. Instead, the complexity of navigation annoyed users.

From the aesthetic aspect, users rated the interfaces into two groups. Windows XP and scenes 2 and 3 were in the first group. This group got mean rank scores of no more than 3. Scenes 3, 4 and 5 got high scores, especially scene 4 (mean rank 4.72). The difference between the two groups was statistically highly significant ($p<0.01$). The results suggested that the users like realistic and fancy interfaces. Usual Windows XP and 3D augmented to it in scene 2 as well as the dull office in scene 3 did not attract users.

Users were asked if the interface will reduce the user's stress. Users tend to agree with the statement for Windows XP interface and scenes 3, 4 and 5. It was clear that the hindrance of navigation and unattractive scenes got negative feedback.

There was no surprise on ease of use preference. Windows XP was intuitive, while the 3D interface got a highly significant lower score ($p<0.01$). However, users still rated PakPao 3D positively on ease of use as 4 out of 5 scenes got mean ranks more than 3. Users exhibited a certain degree of loss of orientation especially while tilting the camera and flying. The experiment suggested that tilting the camera should not be implemented. Users should always be on the ground. This confirms the results of the study in [11], and several others referred to therein, that the users need navigation aid in a three-dimensional environment.

Users found it strange when they could not exactly place an icon where they wanted in the three dimensional interface. Several users could manage to do so by a series of camera turnings. Although we could not manage to introduce

TABLE II
MEANS OF TASK COMPLETION TIME OF DIFFERENT USER INTERFACES AND THEIR REPEATED-MEASURE ANALYSIS OF VARIANCE.

	Means (SD)						Repeated Measures ANOVA		
Variable	Windows XP	Scene 1 (Room)	Scene 2 (Office)	Scene 3 (Home)	Scene 4 (Beach)	Scene 5 (Space)	F	df	Sig.
Time	2.54 (0.86) a[1]	3.42 (1.53) b	3.50 (1.38) b	3.64 (1.90) b	3.71 (1.91) b	3.53 (1.47) b	3.956	5	.013

[1]Means followed by the same letter in the same row are not significantly different ($p > 0.05$).

TABLE III
FRIEDMAN TEST RESULTS DERIVED FROM USERS' FEEDBACK (LIKERT SCALE) IN THE SIX CONDITIONS.

	Mean Ranks (SD)						Test Statistics		
Variable	Windows XP	Scene 1 (Room)	Scene 2 (Office)	Scene 3 (Home)	Scene 4 (Beach)	Scene 5 (Space)	Chi-Sqr.	df	Sig.
Enjoyment	3.31 (0.74) a,b[1]	3.00 (0.85) a	3.27 (0.88) a,b	3.77 (1.02) b,c	3.94 (0.88) c	3.71 (0.92) b,c	14.33	5	.014
Attractiveness	2.32 (0.47) a	2.72 (0.84) a	2.80 (0.82) a	4.32 (0.76) b	4.72 (0.76) b	4.12 (0.81) b	50.85	5	.000
Stress	4.06 (1.01) b	2.83 (1.01) a	2.79 (0.87) a	3.67 (1.16) b	4.23 (1.00) b	3.42 (0.93) a,b	19.016	5	.002
Ease of Use	4.98 (0.87) b	3.29 (0.88) a	3.08 (0.78) a	3.33 (0.85) a	3.44 (0.83) a	2.88 (0.65) a	28.224	5	.000
Dragging	4.54 (0.90) b	3.42 (1.16) a	3.42 (1.06) a	3.35 (1.13) a	3.29 (1.01) a	2.98 (0.98) a	17.183	5	.004

[1]Mean ranks followed by the same letter in the same row are not significantly different ($p > 0.05$).

dragging manipulation to be as intuitive as in a two-dimensional interface, most of our proposed scenes got positive rates since the use of dragging in the interface did not require high accuracy in nature. Techniques for more intuitive dragging in the operation system interface are our suggested future works. Our preliminary recommended technique would be heuristically snap-dragging in three-dimensional interface [10].

For overall preferences, users would like to use scene 4 the most, scene 3 the second, scene 5 the third, scene 2 the fourth and scene 1 as the last preferred one among PakPao 3D scenes. Users commented that they preferred realistic scenes with as many details as possible. The details included dynamic objects, lighting, transparency and glossy effects. Most of them preferred relaxed environments like in scenes 4 and 3, while some of them liked exotic environments like in scenes 4 and 5. When the users were asked if PakPao 3D could be used instead of Windows XP for everyday use, they rated this 3.14 (SD=1.05) on average. When they were asked to rate how often they would use the interface, 1-never and 5-always, the average rating was 3.04 (SD=0.79), which meant occasionally. Users agreed that the spacious interface could be used to handle more icons effectively with a 3.92 (SD=0.91) rating. By observation, it was useful to group icons in three-dimensional space by placing them near referenced objects. This could increase the ability to recognize the icon positions and reduce the time to find the icons. Users agreed with the above statement with a 3.96 (SD=0.84) rating.

Heavy dynamic virtual environments like scene 5 could cause a headache if the users stare too close to the monitor. Incorporating such elements has to be carefully designed since the size of affordable monitors is getting larger and larger. One user suggested that the interface should employ only a mouse without a keyboard. Users were satisfied with the functionality of creating shortcuts.

VI. Conclusion and Future Work

We have designed and implemented a novel 3D user interface which provides users with a thorough 3D workplace and objects as well as a fully augmented virtual environment desktop. To accomplish this rich realistic virtual environment, a powerful game engine expedited the fulfillment of the implementation phase. We came up with a complete working interface which can replace the desktop metaphor. We can conclude certain frameworks to contribute to the design of such interface from our extensive evaluations. The desktop metaphor interface let the user interact with the computer with ease. Adding three-dimensions to the desktop metaphor increases working space and enjoyment. In this case, the virtual environments have to be realistic with attractive effects or the interface will annoy the users. Dragging icons paralleled to the viewing pane is preferred. The complexity of the interface opens room for improving icon dragging and user navigation in the operating system interface.

Acknowledgment

This research is supported by Suranaree University of Technology and the National Research Council of Thailand. Thanks to the dedicated graphics and implementation team: Siwadol Sateanpattanakul, Saharat Areeras, Parita Phongphanich and Paramita Athiwet. Special thanks to Dr. Piyada Thipyapong and Dr. Paul J. Grote for great support.

References

[1] 3D Game Studio [Online]. Available: http://3dgamestudio.com.
[2] 3DNA Corp., 3DNA Desktop [Online]. Available: http://www.3dna.com.
[3] A. Marcus, and van Dam, "A. User interface developments for the nineties," *IEEE Comput.,* vol. 24, no. 9, Sep. 1991, pp 49-57.
[4] A. Monk, M. Hassenzahl, M. Blythe and D. Reed, "Funology: designing enjoyment," In *CHI '02 Extended Abstracts on Human Factors in Computing Systems,* 2002, pp. 924–925.
[6] D. A. Bowman, D. Koller, L. F. Hodges, "Travel in Immersive Virtual Environments: An Evaluation of Viewpoint Motion Control Techniques," *vrais*, 1997, p.45.
[7] D. A. Bowman, et al, "An introduction to 3-D User Interface Design," *Presence: Teleoperators and Virtual Environments*, vol. 10, no. 1, 2001, pp. 96-108.
[8] D. Bowman, E. Kruijff, J. LaViola, I. Poupyrev, *3D User Interfaces: Theory and Practice*, Addison-Wesley Professional, CA, 2004.
[9] D. Chao, "Doom as an interface for process management," In *Proceedings SIGCHI'01*, 2001, pp. 152-157.
[10] E. A. Bier, "Snap-dragging in three dimensions," In *Proceedings of the 1990 Symposium on interactive 3D Graphics,* SI3D '90, 1990, pp. 193-204.
[11] E. Haik, T. Barker, J. Sapsford and S. Trainis, "Investigation into effective navigation in desktop virtual interfaces," In *Proceedings of 7th Int. Conf. on 3D Web Technology*, Feb. 2002, pp. 59–66.
[12] G. Robertson, et al, "The Task Gallery: A 3D Window Manager," in *Proceedings of CHI 2000*, pp. 494-501.
[13] H. Regenbrecht, G. Baratoff, and M. Wagner, "A tangible AR desktop environment," Computers & Graphics, Special Issue on "Mixed Realities – Beyond Conventions". 2001.
[14] M. Chen, S. J. Mountford, and A. Sellen, "A study in interactive 3-D rotation using 2-D control devices," In *Proceedings of the 15th Annual Conference on Computer Graphics and interactive Techniques*, SIGGRAPH '88, pp. 121-129.
[15] M. Lewis, and J. Jacobson, Game Engine in Scientific Research: Introduction. *Commun. ACM* vol. 45, no. 1, Jan. 2002, pp. 27-31.
[16] M. van Dantzich, V. Gorokhovsky, and G. Robertson. "Application redirection: hosting Windows applications in 3D," In *Proceedings of NPIVM' 99,* Nov. 1999, pp. 87-91.
[17] N. Höglund, "3D Graphics in the User Interface of a File System Browser," Master Thesis, Royal Institute of Technology, Sweden, 2004.
[18] P. Issacs, J. Shrag and P. S. Strauss, "The design and implementation of direct manipulation in 3D," In *Course notes from the SIGGRAPH 2002 conference*, July 2002.
[19] P. Knees, M. Schedl, T. Pohle, and G. Widmer, "An innovative three-dimensional user interface for exploring music collections enriched," In *Proceedings of the 14th Annual ACM international Conference on Multimedia,* Oct. 2006, pp. 17-24.
[20] R. Andreoli, R. Chiara, U. Erra, V. Scarano, "Interactive 3D Environments by Using Videogame Engines," *iv*, 2005, pp. 515-520.
[21] S. Brunstad and T.F. Eie, "Adding a D to 2D - a better interface?," IFI, UiB, Bergen, Norway / IHA, Man-Machine Interaction, 2002.
[22] S. Di Verdi, D. Nurmi and T. Hollerer, "Mixed and Augmented Reality," in *Proceedings of the Second IEEE and ACM International Symposium*, Oct. 2003, pp. 298- 299.
[23] S. Ellis, *Origins and Elements of Virtual Environments, in Virtual Environments and Advanced Interface Design*, eds. W. Barfield & T.A. Furness III, Oxford University Press, 1995, pp. 14-57
[24] Sun Microsystems, Inc., Project Looking Glass [Online]. Available: http://www.sun.com/software/looking_glass
[25] T. Altom, M. Buher, M. Downey, and A. Faiola, "Using 3D landscapes to navigate file systems: the MountainView interface," in *Proceedings of the 8th International Conference on Information Visualization*, 2004, pp. 645-649.
[26] T. Mummer, "Exploring large graphs in 3D hyperbolic space," *IEEE Computer Graphics and Applications*, vol. 18, no. 4, July/Aug. 1998, pp. 18-23.