BumpTop Desktop

Course: CMPT 811

Instructor: Dr. Gutwin

Student: Faham Negini

NSID: fan780

Dec 2011

Motivation

Computer desktops have been designed as two dimensional environments using concepts like icons and windows to bring about a more user friendly method comparing to text console environments to computer users. Although they have done a good job for more than a decade and helped many users be introduced to computer systems without solid knowledge about operating system's available commands and functionality, but huge improvements to 3D graphics techniques backed by strong hardware accelerators brings about a new potential arena for application user interfaces that is hardly touched yet. BumpTop desktop is a primary step for future GUI’s which uses that one additional dimension.

Many ideas initiated in traditional 2D desktop environments is about to vanish in 3D environments. In BumpTop desktop files and folders will have 3D visual and physical representation and there will be new ideas for sorting and arranging documents that using them users can scale their control over large number of documents they have. The ultimate goal of this project will be to introduce a completely new shell not only a desktop for operating systems like Linux through which users enhance their usage performance and scale by the use of 3D representation of files, folders, windows, widgets and etc. . Main features going to be implemented will be: 3D visual and physical representation for files and folders, selection and transformation of files and folders, creating piles of documents and operations like Fan-Out and Grid for the piles, context menu system and regular operations through the menu.

Background

The registered *BumpTop* is initially designed and developed as a commercial software available under Windows and Mac and on April 30 2010 the developers announced:

> “Today, we have a big announcement to make: we're excited to announce that we've been acquired by Google! This means that BumpTop (for both Windows and Mac) will no longer be available for sale. Additionally, no updates to the products are planned.
>
> For the first week of May 2010, we kept BumpTop Free available for download to give BumpTop fans one last chance to grab a copy. BumpTop is now no longer available for free download.”

As of then this project and its neat idea has not improved one step further. This was the first point of motivation for me to build an application software like that. I decided to do this project under Linux and as a freeware as an answer to people who like to have it under Linux. My other source of motivation was the interest of developing a complete 3D environment instead of what we are familiar today as the GUI of our applications and even develop further interaction methods other than using 2D pointing instruments like Mouse for users to do interaction with computer systems. I'm inspired to extend the boundaries of the project to address many other places other than just Desktop, which computer users interact with their personal computers through 2D graphical user interfaces.

What I've developed is literally titled *BombasticDestop* in its code base, which is assumed to be its temporary codename while its yet under development. According to the commercial BumpTop Desktop I'm inspired to develop this project to a stage which offers services and capabilities which provoke people to pay money for that beside its initial free services which makes it available for many people using personal computers.

Description

Development Environment

The first thing needed to do this project was a good 3D rendering engine which needed to be freely accessible and preferable open-source, I've found OGRE a quite mature and fully featured library which is supported by a very good community under many platforms from iOS to Linux and Windows. Then it was clear for me to choose this library as my rendering engine. The second major library which was needed was something to be able to simulate physical laws between objects in the 3D desktop environment. I've candidate ODE and Bullet for this purpose and after looking at their communities and feature sets soon figured out that Bullet is the best choice because it was supported by a good community and a very much more active project in compare to ODE, in addition to that I found many people complaining with the performance of ODE. In addition to these two major libraries I've used IOS as my input library and OgreBullet as a wrapper which facilitates connecting Ogre entities to Bullet's more easily. After building all of these libraries source codes under my system, I've used KDevelop4 to setup my development environment using CMake as the major scripting language to configure my project files and build system. Detailed version and specification of the libraries I've used can be found at the following.

Ogre	v. 1.7.3
Bullet	v. 2.79
OgreBullet	svn revision 2986
OSI	v. 1.3.0
CEGUI	v. 0.7.5
SILLY	v. 0.1.0

CEGUI and SILLY are two other library which were added to the project after I started to develop the menu system. CEGUI (Crazy Eddie's GUI) is an open-source library which provides many 2D windowing functionality in an abstract layer which then can be presented through plugins in any environment. Then using this library and its related plugin to OGRE I could easily implement stuff like windows, buttons, context menu and etc. In my environment. One of best features of this library is that it separates your GUI definition layer from your C++ code, then you can manipulate your GUI by editing the XML files without the need of recompiling your application. Of course implementing the context menu could be done much more easier using 2D layers in Ogre itself but thinking about further developments of the project encouraged me to embed a powerful windowing system from the beginning.

Software Architecture

It is decided for the project to be mainly created on top of OGRE library that it features lots of basic needed subsystems like resource management, scene management, material scripting language, log system and etc.. I've used Ogre's Log system to generate logs dedicated to my evaluation plan while participants were doing the experiment. Ogre's Log system not only generates a default log file about events which happened in the Ogre itself, you can use it to generate new log files and send your own messages to them to be logged.

Publisher/subscriber relationship is widely used for handling subjects like input events by OIS, frame rendering events by OGRE and window manager events like WindowResized. There is a thin wrapper layer called

OgreBullet, used for synchronization between visual and physical representation of the scene. according to demanded features, this is not only the physical calculations which are applied to visual, but also sometimes new transformation state is decided by visual layer and then is applied to the physics, then this is a 2 way synchronization.

Modules

There is a *Desktop* module which runs the main rendering loop and also is a frame and input listener, it receives different frame and input events (i.e. FrameStarted, FrameEnded, FrameRenderingQueued, MousePressed, KeyPressed and etc.) and implement some general features of the application like the Camera system, selection and transformation of objects. This module in some places is more like a finite state machine which changes its state depending on the order and kind of operations the user choose to do; and this module is responsible to do the initialization of different parts of the application like the bullet world, CEGUI library and the initial graphical scene of Ogre like setting the room and the lights and etc.

Object is another core module which abstracts any kind of objects which are going to be spawned into Desktop, this module encapsulates both visual and physical features of an object and talks to operating system whenever the object is executed (like when it is double clicked). Objects can have different types like a Folder, and Image or etc., which according to their type the will inherit different materials and maintain different behaviours in relation with the OS. Of important functionality which object instances have is scalability, it means you can change the scale (size) of the objects and according to the scale of the objects their weights would be decided, then for example if the user make an object bigger than other objects it would be heavier and probably assumed more important in compare to other objects. Highlighting and Freezing are two other functionality which objects maintain, when an object is Highlighted its material is changed to an other corresponding material from the designed material library which means the object is selected and when it is Frozen the material changes to another material which means the object won't be selectable any more to the user.

One of bests features of the Ogre library is a well support for creating libraries of materials by inheritance and global variables. I have defined about ten materials for different purposes and using these techniques managed to reduce the redundancies of the material scripts to around 180 lines of code.

Pile module provides pile operation capabilities, user create and destroy piles of objects through selecting the menu *pile* item when he right-clicks over a bunch of selected objects. Piles address a few operations which are accessible by the context-menu. Unpile, will destroy the pile entity and returns the objects it contains to the object pool of the Desktop module, but it won't change the position of the objects, then the objects would still be arranged like they are in a pile. Grid, will do the same as the unpile operation do in addition to changing the position of objects and arranging them into a grid, this operation calculates the boundaries of the grid in advance and places the whole grid in a place of the room that doesn't collide with the walls. The Fan-Out is another operation which like grid in addition to unpile the Objects, change their positions but not through a grid like arrangement, it changes their positions to their initial position they had before becoming a pile yet like grid it calculates the boundaries of it in advance and avoid any collision with walls.

Functionality

When the user starts the application it displays the icons placed over the desktop (for evaluation purposes I randomly generate 50 icons, each 10 of them from a different type) and user can select the icons either by clicking or be drawing rectangular selection areas. The second method will generate four planes each mapped to

one edge of the rectangle and then a volumetric intersection query is executed to determine which icons are placed inside the area. After selecting the icons they will become highlighted and users can move them around the room, while moving them around when every MouseMove event is triggered a ray from the camera point of view toward the infinity is generated and then the intersection of this ray with plane XZ is computed, the resulted intersection point will be considered as the target point to move the pivot (average of objects point) of the selected objects to that. Then objects physical positions are manually manipulated and objects are moved according to movement of the pointer. These functionality are maintained for piles too, when there are a bunch of objects (more than on object) selected the user can right click on one of them and select the pile menu item to convert them into a pile. When they are converted into a pile all the objects will be placed on top of each other and thereafter the user won't be able to select or move them one by one.

Evaluation

Purposes

BumpTop Desktop primarily tries to make the desktop environment a more fun place. I need to find out parameters which make BumpTop Desktop more fun rather than a 2D desktop, and how are they going to affect the performance of the user, and figure out if any specific group of people are more interested in BumpTop Desktop. If I manage to specify the group of people who are more interested, and the features they think is more attractive, then I can figure out which aspects of the BumpTop Desktop should be more developed, and what group of people should be targeted as audiences of this product. I need to know how different features of BumpTop Desktop are going to affect the performance of users then I can balance performance and amusement of them to get the best result.

Limitations

Of potential threats to the reliability and validity of this evaluation is the small group of experimenters. The group of experimenters which I plan to use consist of 9 people. Using this small group of experimenters I may not be able to specify exactly what type of computer users are more interested in BumpTop Desktop, and this may lead to wrong conclusions in further development of the application.

Audiences

- Average users like aged parents who occasionally use computers for casual purposes like watching videos, musics and etc.

- Non-gamer experts like Computer Science students and those who work with their personal computers on a daily basis

- Gamers who have a good experience with video games, specially 3D games which are tightly related to the BumpTop Desktop 3D features, like camera, navigation and etc.

I've selected 9 people, 3 from each of these groups to see how they evaluate different features of the application. Then I can address a more specific group of people by this application or define different configuration templates in the application for each society.

Evaluation Scenario

Before the user starts I tried to describe most features of the application for him/her and asked him/her to train himself/herself before starting the evaluation. This took about 5 minutes for each participant. At the evaluation the experimenter was invited to complete a couple of tasks using.

The evaluation environment of the BumpTop Desktop was like a quite crowded desktop containing about 50 icons and 5 sticks. Each 10 icon had a different color corresponding to the color of one of the sticks. The user was asked to select and translate the icons to their corresponding sticks deciding by their colors to freeze them,

this operation was done quite like dragging and dropping an icon unto a folder. When the object become frozen the user won't be able to select or transform that again. The moment user finish putting all icons to their sticks the application would be closed. The log generated by the application collects many operations that user has chosen to do.

For each experimentation session one log file is generated which clearly shows the scenario and chain of operations the participator have chosen, and the amount of time each one of them took him to do.

In the end the user is asked to answer a questionnaire. According to the budget and other limitations of the evaluation some descriptive questions are asked too.

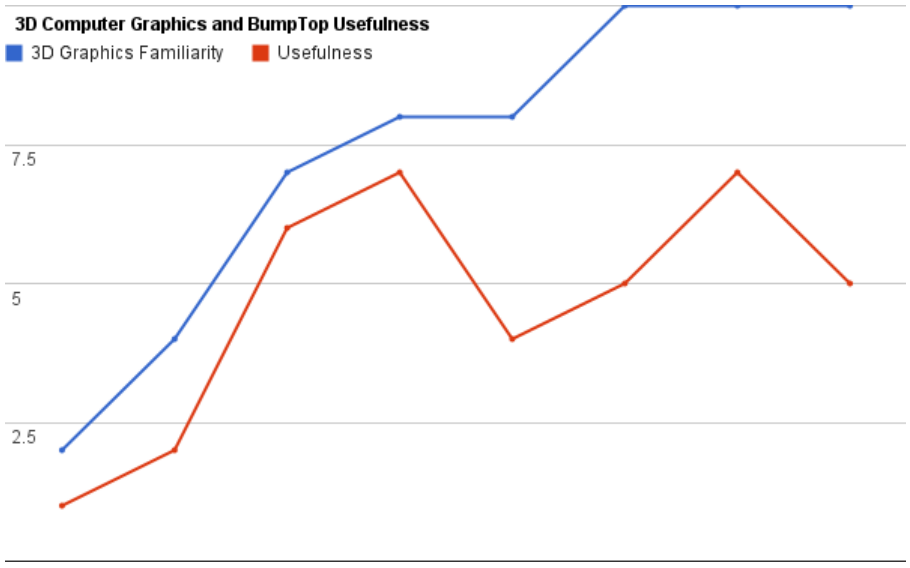
Instrumentation

The user was asked to do the experiment twice, once using the mouse and the other time using the touchpad, both on a Lenovo G570 laptop under Ubuntu 11 using Xfce 4.8 desktop environment.

A Lenovo G570 featuring a very standard interface, 15.6“ monitor and a two button touch-pad, which implements the middle mouse button by pressing both left and right buttons simultaneously.

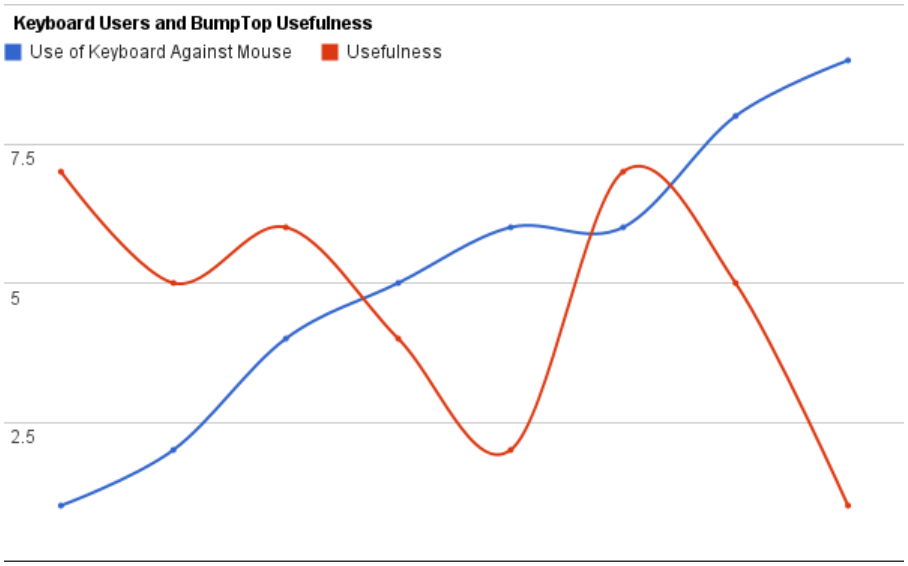
Results

The image below is the comparison between participators familiarity with 3D computer Graphics and how useful they've found the BumpTop Desktop. As you can see in the chart, these two factors are closely mapped to each other and according to that the developers should more continuously target more appropriate societies as the users of the BumpTop Desktop.



The chart below is about how people found BumpTop Desktop useful in compare to their dedication to using computer keyboards to interact with their PCs. As it is obvious in this chart the more people use their keyboards

the less they find BumpTop Desktop useful. Then people like programmers and typists have less chance to be users of BumpTop Desktop. This is yet another illustrative comparison that can clearly help me to decide which society is the the target for this application.



Conclusions

According to questionnaires and logs the results of the evaluation illustrates, for the BumpTop Desktop to meet the proposed requirement the application should be more polished and some tiny flaws should be resolved, other than these shortcomings like icons highlighting method and etc., I found out people who are more interested in 3D video games are more likely interested in the application. And this is what can be quite useful for further development of the application.

Reflection on the Process

Using open-source libraries has proved quite useful in this projects. In the development process I almost had no serious problem in detecting bugs and handing exceptions with the use of being able to debug and monitor everything which is going on inside the libraries I'm using in my project. One of the most benefits of open-source libraries is that you have full access to the source code of the library and then no unintended behaviour by the library would surprise you. I tried to build all the libraries I use from scratch on my system and then connect my debugging symbols of the libraries I linked to my project to these sources then when I needed to debug my application I was fully equipped to detect bugs.

Not all the features of the motivating project I've chosen to follow could be done in a month or so at the end of the term and obviously I needed to omit some features and limit its scale then it would fit in my available resources. On the other hand the purpose of this project couldn't be easily touched by only implementing a few number of the features, then I think the decision of which parts to implement and which parts to avoid should be taken more carefully to meet the other requirements of the class like a more precise and usable evaluation result.

Appendices

BumpTop Desktop Evaluation Questionnaire

Student ID: .....

1 Specify how often you are by your Desktop or Laptop PC in a day (in average)
   1 Less than 1 hour
   2 Between 1-3 hours
   3 Between 3-6 hours
   4 More than 6 hours

2. How often do you play video games (computer or console) in a day (in average)
   1 Less than 10 min
   2 Between 10-60 minutes
   3 Between 1-3 hours
   4 More than 3 hours

3. How much are you introduced to 3D Computer Graphics?

My first time ▢▢▢▢▢▢▢▢▢▢ Quite familiar (Played many 3D Games)

4. How often do you use these devices in compare to each other?

Mouse ▢▢▢▢▢▢▢▢▢▢ Keyboard

5. How useful you found the camera operations?

Useless ▢▢▢▢▢▢▢▢▢▢ Useful

6. How did you find the BumpTop Desktop as your PC desktop?

Boring ▢▢▢▢▢▢▢▢▢▢ Fun
Useless ▢▢▢▢▢▢▢▢▢▢ Useful
Non-User Friendly ▢▢▢▢▢▢▢▢▢▢ User Friendly

7. If this product be distributed as a freeware, how likely is it for you to use it?

I won't use it ▢▢▢▢▢▢▢▢▢▢ I'll definitely use it

8. If this product be distributed as a commercial software, how much do you think it will worth to pay?

1CAD ▢▢▢▢▢▢▢▢▢▢ 10CAD

9. Please write about any feature you would like to be added or modified about BumpTop Desktop:

.....
.....
.....

BumpTop Desktop Evaluation Agreement

I agree for my questionnaire and any kind of collected data during the experiment, to be used for improvement and development of BumpTop Desktop. These data won't be shared with anyone else but the instructor of CMPT 811 Dr. Carl Gutwin and anyone who is involved directly or indirectly with the development of this project or any other project basing on BumpTop Desktop. Collected data will only be used for statistical analysis and development of the application, in addition to providing CMPT 811 term project evaluation plan requirements.

These data won't be used (as mentioned above) under your name, and your identification will always remain hidden to everyone who may use it.

Though according to this agreement your name will be collected as a participanter of the BumpTop Desktop experimentation.

Be notified that the name **BumpTop Desktop** is just the primary codename of this project in this phase, the title of this project may change during any period of time in future.

Participant Name: ..... Signature: .....
Date: mm / dd / yyyy

Experimenter Name: ..... Signature: .....
Date: mm / dd / yyyy

Collected data through questionnaires:

Student	Use PC(Q1)	Play Game(Q2)	3D Graphics Familiarity(Q3)	Use of Keyboard Against Mouse(Q4)	Camera Ops(Q5)	Fun(Q6.1)	Usefulness(Q6 .2)	User Friendly(Q6.3)	Will Use It(Q7)	Price (Q8)
1	4	1	10	2	3	8	5	5	3	4
2	4	1	2	9	1	4	1	3	1	1
3	4	2	8	6	6	8	7	10	6	1
4	4	1	4	6	3	6	2	2	1	1
5	4	2	10	1	10	5	7	7	8	6
6	4	1	7	4	7	7	6	8	7	6
7	3	2	8	5	7	8	4	7	4	2
8	4	1	10	8	3	8	5	3	3	1