

윈도우즈 운영체제 기반의 3D 가상 데스크톱 시스템 설계 및 구현

(Design and Implementation of a 3D Virtual Desktop System based on Windows Operation System)

홍 승 표 [†] 오 명 진 ^{**}
(Seungpyo Hong) (Myeongjin Oh)

신 정 훈 [†] 이 상 준 ^{***}
(Junghoon Shin) (Sangjun Lee)

요 약 개인용 컴퓨터의 성능이 향상되면서 대부분의 컴퓨터 사용자에게 여러 개의 애플리케이션을 구동시키는 다중작업은 일반화되었다. 윈도우즈 운영체제는 다중작업 중인 여러 애플리케이션을 구동할 작업공간을 오직 하나만 제공하고 있기 때문에 많은 사용자들이 불편함을 느끼고 있다. 본 논문에서는 이러한 윈도우즈 운영체제의 단점을 보완하여 윈도우즈 운영체제 환경에서 사용자에게 여러 작업공간을 제공하며, 리눅스 Compiz와 같이 사용자에게 화려하고 직관적인 UI(User Interface)를 제공하기 위해 DirectX를 이용하여 작업공간의 전환 과정을 3D 모션으로 보여줄 수 있는 3D 가상 데스크톱을 제안한다.

키워드 : 3D 가상 데스크톱, 3D 데스크톱, GUI

· 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원 사업의 연구결과로 수행되었음(NIPA-2010-C1090-0803-0006)

· 이 논문은 2010 한국컴퓨터종합학술대회에서 '윈도우즈 운영체제 기반의 3D 가상 데스크톱 시스템 설계 및 구현'의 제목으로 발표된 논문을 확장한 것임

[†] 학생회원 : 송실대학교 컴퓨터학부
hsp0515@gmail.com
coolhoony1@naver.com

^{**} 비회원 : 고려대학교 컴퓨터통신공학부
iamOhMi@gmail.com

^{***} 정회원 : 송실대학교 컴퓨터학부 교수
sangjun@ssu.ac.kr
(Corresponding author)

논문접수 : 2010년 8월 9일

심사완료 : 2010년 10월 22일

Copyright©2010 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨팅의 실제 및 데이터 제16권 제12호(2010.12)

Abstract As personal computers have been recently improved in performance, the multi tasking of running several applications at the same time has been generalized. Since only one task space is provided for performing such tasks on Windows operating system, a number of computer users feel inconvenient. In this paper, we propose a 3D Virtual Desktop System which can show a process of switching task spaces in 3D motion using DirectX. The proposed system can provide a computer user with more task spaces and the intuitive UI(User Interface) such as Compiz of Linux.

Key words : 3D Virtual Desktop, 3D Desktop, GUI

1. 서론

오늘날 개인용 컴퓨터 하드웨어의 성능 향상과 운영체제의 발전으로 컴퓨터를 이용하는 사용자들에게 여러 개의 프로그램을 실행하는 다중작업은 운영체제의 기본적인 기능 중 하나가 되었다. 이와 더불어 컴퓨터 사용자들은 여러 가지 작업을 동시에 하는데 있어서 좀 더 편의성이 뛰어나면서 비주얼적인 GUI(Graphic User Interface)를 원하게 되었다. 이러한 요구를 충족시키기 위해 많은 소프트웨어 관련 기업들과 연구자들은 기능과 편의성이 뛰어난 UI를 제공할 수 있는 기법에 대한 연구들을 진행하고 있다. 특히 최근에 주목받고 있는 3D를 이용한 GUI 관련 그래픽 기법들에 대한 연구가 많이 진행되고 있다[1].

전 세계 대부분의 사용자들은 마이크로소프트의 윈도우즈 기반의 운영체제를 사용하고 있다[2]. 하지만, 윈도우즈 운영체제 환경에서 다중작업을 할 경우 사용자 입장에서 다소 불편한 점이 있다. 예를 들어 그림 1과 같이 작업표시줄에 있는 창 목록들의 제목이 보이지 않아 실행 중인 프로그램을 구분하기 어렵고, 여기저기에서 실행 중인 작업 창들로 인해 사용자가 효율적인 작업환경을 유지하기가 어렵다. 본 논문에서는 이러한 윈도우즈의 작업환경의 단점을 보완하기 위해 리눅스 운영체제에서 자체적으로 지원하는 3D 가상 데스크톱 기능을 윈도우즈 환경에서 지원할 수 있도록 하는 윈도우즈용 3D 가상 데스크톱에 대한 기존의 연구[3,4]를 확장하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대한 내용에 대해 살펴본다. 3장에서는 본 논문에서 설계하고 구현한 3D 가상 데스크톱에 대해 살펴보고, 4장에서는 기존 시스템과의 비교 및 평가를 하였다. 마지막으로 5장에서는 결론을 맺는다.

2. 관련 연구

가상 데스크톱 시스템은 현재까지 여러 프리웨어 및 상용 프로그램으로 개발되었다. 본 절에서는 기존에 개

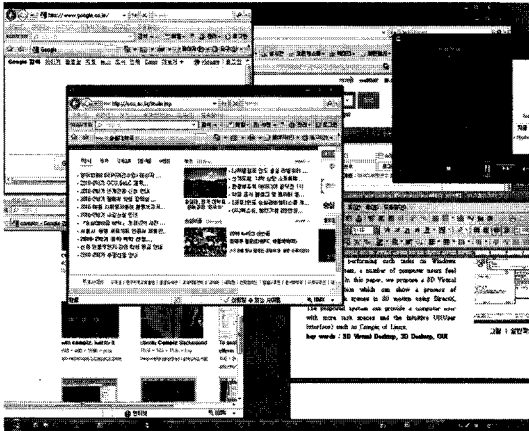


그림 1 일반적인 윈도우즈 데스크톱 환경

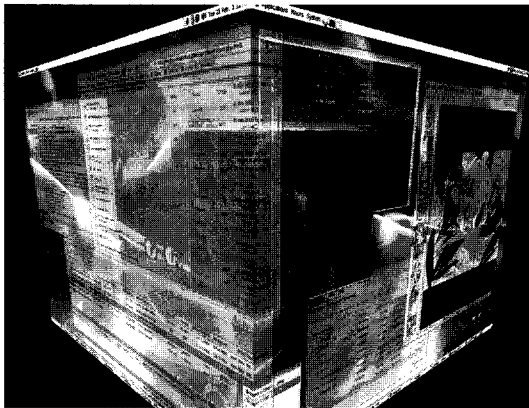


그림 2 리눅스의 3D 가상 데스크톱 Compiz

발된 가상 데스크톱 시스템 중 대표적인 시스템인 리눅스의 compiz[5]와 마이크로소프트의 Desktops[6] 그리고 Otaku Software의 DeskSpace[7] 등을 소개한다.

리눅스의 Compiz는 그림 2에서 볼 수 있듯이 3D를 지원하는 가상 데스크톱 시스템으로 리눅스 기반의 운영체제에서 기본적으로 지원하는 소프트웨어이다. Compiz는 다른 어떤 3D 가상 데스크톱 보다 뛰어난 성능을 제공하고 있다. 이런 이유로 많은 리눅스 사용자들이 만족하고 있는 시스템 기능 중 하나이다. 심지어 Compiz의 3D 가상 데스크톱 기능을 사용하기 위해 리눅스라는 운영체제를 선택하는 사용자들도 있다. 리눅스 Compiz가 높은 성능의 서비스를 제공 가능한 이유는 운영체제에 종속적인 GUI 방식의 윈도우즈와는 다르게 리눅스 계열의 X윈도우는 운영체제와 독립적인 GUI 구조로 되어 있기 때문이다. 따라서 다른 부분에 영향을 미치지 않고 디스플레이에 관련된 X윈도우만을 수정함으로써 개발자가 원하는 GUI로 손쉽게 변경시킬 수 있다. 이러한 이

유로 Compiz는 3D 가상 데스크톱 기능뿐만 아니라 바탕화면을 볼로 태운다든가 그 밖에 윈도우 창이 휘어지는 화려한 이미지 효과를 보여주는 등 많은 기능을 제공하고 있다. 하지만, Compiz는 리눅스 운영체제에 종속적이기 때문에 전 세계 가장 많은 사용자를 보유하고 있는 윈도우 운영체제에서는 동작할 수 없다. 본 논문의 시스템은 윈도우 사용자에게도 Compiz의 유용한 기능들을 공유해 주고자 개발하게 되었다.

마이크로소프트의 Desktops는 다른 부가적인 기능은 모두 없애고 단순히 가상 데스크톱 기능만을 제공하는 소프트웨어이다. 따라서 앞의 리눅스 Compiz에서처럼 작업 공간 전환 시 3D 효과나 다른 GUI 기능들은 전혀 기대할 수 없다. 또한 가상 데스크톱을 제공하기 위해 explorer를 작업공간의 개수만큼 실행시킴으로써 메모리 낭비를 유발하고 작업 공간 전환 시 화면전환이 느리다는 단점을 가지고 있다.

마이크로소프트의 Desktops 외에도 윈도우용 가상 데스크톱 기능을 제공하는 프리웨어나 상용 소프트웨어가 나와 있지만, 3D 효과를 지원하지 않으며 지원하더라도 리눅스 Compiz에 비해 훨씬 높은 사양을 요구하고, 그래픽 카드 성능이 낮으면 다운되는 현상도 발생한다. Otaku Software에서 개발한 DeskSpace는 상용 소프트웨어로 기존 윈도우용 3D 가상 데스크톱 중 가장 최적화되어 있는 소프트웨어이다. 하지만, 리눅스의 Compiz와 같이 부가적인 효과가 없으며, 바탕화면의 개수도 4개로 고정되어 있다는 단점 등이 존재한다.

3. 제안 시스템의 설계 및 구현

3.1 시스템 개요

리눅스 계열의 운영체제와는 달리 마이크로소프트 윈도우의 GUI는 운영체제에 종속적이기 때문에 이미 마이크로소프트에서 설계한 GUI 구조를 개발자가 마음대로 변경시킬 수 없다. 이에 본 시스템은 윈도우즈의 GUI 구조를 지키면서 작업공간의 전환을 3D 효과로 표현하기 위해 그림 3과 같이 풀 스크린 모드로 프로그램이 구동되도록 하였다.

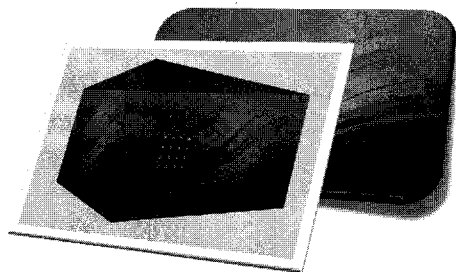


그림 3 풀 스크린 3D 애니메이션

프로그램 실행 시 트레이 아이콘으로 동작하다가 사용자가 작업공간을 전환하려 할 때에 그림 3과 같이 3D 애니메이션을 출력하는 새로운 윈도우 창을 전체화면으로 띄워 윈도우즈의 바탕화면을 가리게 된다. 사용자가 작업공간이 전환되고 있는 3D 애니메이션을 보는 동안에 이전 작업공간에서 사용자가 실행하고 있던 응용프로그램은 숨기고 전환될 작업공간의 응용프로그램들을 복원시킨다. 이러한 작업을 통해 사용자는 마치 여러 개의 작업공간을 보유하고 있는 것처럼 인식하게 된다.

3.2 윈도우 핸들 관리

본 시스템은 다수의 윈도우 창을 숨기거나 복원시키기 위하여 해당 윈도우의 핸들을 이용하여, 핸들 관리를 위해 작업공간의 리스트를 원형 이중 연결 리스트를 사용하여 연결한다. 각 작업공간에서 실행되고 있는 윈도우의 핸들 리스트는 해당 작업공간에 등록된다.

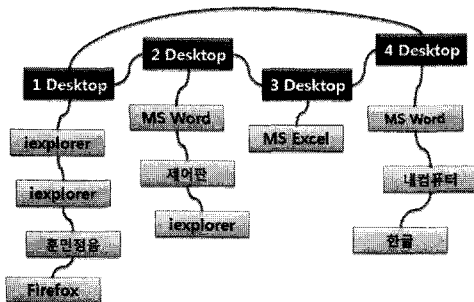


그림 4 윈도우 핸들 관리

그림 4는 네 개의 작업공간을 만들고 여러 응용프로그램을 서로 다른 작업공간에서 실행시켰을 때의 자료구조 예시이다. 만약 사용자에 의해 작업공간의 개수가 변경 될 경우 연결 리스트의 삽입 및 삭제를 수행하여 해당 개수에 따라 연결 리스트 구조를 재조정한다.

윈도우 핸들 연결리스트의 데이터를 갱신하는 시점은 사용자가 작업 공간을 전환하여 3D 애니메이션을 보고 있는 순간이다. 현재 보이고 있는 모든 윈도우 창들의 핸들을 얻어와 현재 작업 공간에 리스트로 추가하여 갱신한 후 해당 윈도우를 숨긴다. 그 다음, 전환되어질 작업공간에 유지되고 있는 윈도우 핸들을 이용하여 기존에 숨겨두었던 윈도우들을 복원시킨다.

본 시스템에서 관리해야 하는 윈도우 핸들은 작업표시줄에 활성화되어 있는 윈도우만 해당된다. 내부적으로는 윈도우 창 없이 실행되는 프로그램들이 대부분이며 각각 윈도우 핸들을 갖고 있다. 문제는 화면에 출력되는 윈도우의 핸들만 별도로 얻어오는 API가 존재하지 않는다는 점이다. 따라서 필요한 윈도우 핸들만 추출하는 부분을 구현할 필요가 있다.

본 시스템에 유효한 윈도우 핸들 관리를 위하여 다음 Win32 API를 사용한다.

```
BOOL WINAPI EnumWindows(_in
    WNDENUMPROC lpEnumFunc, _in LPARAM
    lParam);
```

현재 윈도우 시스템에 등록된 모든 윈도우 핸들을 순차적으로 얻어오는 API이다. 첫 번째 인자에 윈도우 핸들을 필터링하는 함수의 포인터를 넣어주면, 모든 윈도우 핸들 각각에 대해 등록된 함수를 호출해준다. 호출되는 콜백 함수는 인자로 넘어온 윈도우 핸들을 이용하여 해당 윈도우가 화면에 출력되는 종류의 윈도우라면 현재 작업공간의 윈도우 핸들 목록에 추가한다. 이러한 윈도우 핸들 목록 유지를 통해서 숨겨둔 윈도우 창을 복원시킬 수 있다.

3.3 작업 공간 개수에 따른 3D 애니메이션

본 시스템은 기존의 윈도우용 3D 가상 데스크톱 프로그램들과는 달리 작업 공간의 개수를 2개에서 6개까지 유동적으로 변경할 수 있도록 구현하였다. 작업 공간의 개수가 변경되면 그 개수에 맞게 그림 5와 같이 3D 애니메이션도 변화되어야 하며 회전하는 양도 각각 달라야 한다. 예를 들어 2개의 작업 공간을 갖게 되면 전환시 180도 만큼 회전하는 반면, 6개의 작업 공간을 지니고 있다면 그림 5와 같이 60도 만큼 회전하게 된다.

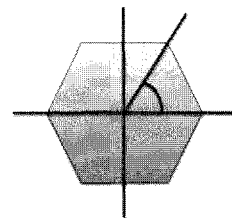


그림 5 작업공간의 개수에 따른 회전각 설정

이 내용을 소스코드로 구현한 내용은 다음과 같다. 작업 공간에 따른 회전각을 구하고 그 각에 따라서 각기둥의 거리를 조절함으로써 2개에서 6개까지 다음의 수식이 모두 적용된다.

```
angle = PI*2/surfNum; // 회전각 계산

// 하나의 면과 원점과의 거리를 계산한다.
x = cos(angle*n);
z = sin(angle*n);
y = sqrt(pow((cos(angle*0)-cos(angle)), 2) +
    pow((sin(angle*0)-sin(angle)), 2));
ViewZPoint=(-1*y*(zoomRate-1/sqrt(two)))-yMid;
```

또한 작업공간의 개수에 따라 그림 6과 같이 각각에 맞게 렌더링 되어야 한다. 개수 변환 시 새로운 폴리곤을 생성 또는 제거하기 위해 Index와 Vertex 정보를 다시 설정해 준다.

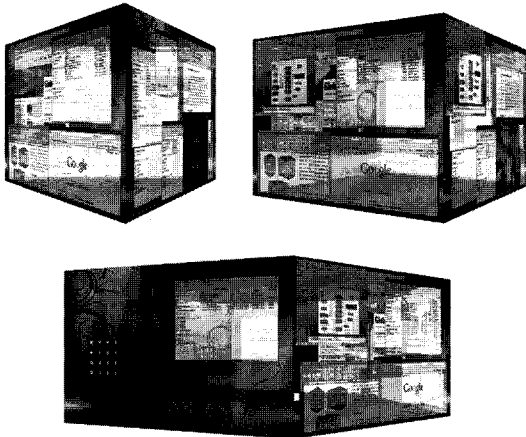


그림 6 다양한 작업 공간의 개수

3.4 사용자 인터페이스

본 시스템의 모든 기능은 그림 7과 같은 방식으로 동작하게 된다. 프로그램 실행 시 트레이 아이콘으로 동작하며 사용자의 입력(Hotkey)을 기다린다. 특정 이벤트가 발생하면 해당하는 기능을 수행 후 다시 입력 대기 상태를 유지한다. 또한 트레이 아이콘 메뉴를 통해 작업 공간의 개수 설정 및 프로그램을 종료할 수 있도록 구현하였다.

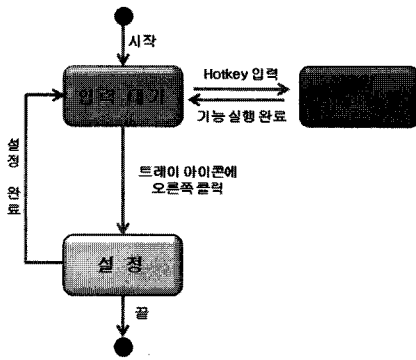


그림 7 시스템 상태도

본 시스템은 작업 공간이 전환되는 UI 방식을 세 가지 방법으로 제공하고 있다. 두 가지 종류의 3D Motion을 보여주며, 간편한 동작을 원하는 사용자를 위하여 3D Motion 없이 작업 공간만 변경시켜주는 모드 또한 지원하고 있다. 사용자가 입력하는 Hotkey의 종류에 따라 여러 가지 모드를 구분하여 동작한다. 즉, 똑같은 오른쪽 방향키를 사용하더라도 Alt키와 Ctrl키, Win키 중 어떤 키를 같이 누르는지에 따라서 작업 공간이 전환되는 애니메이션을 서로 다르게 보여준다.

4. 시스템 평가

본 논문에 제안된 시스템은 기존에 윈도우용 3D 가상 데스크톱 프로그램에서 지원하지 못했던 여러 기능을 제공하고 있다. 운영체제 종속적인 문제로 리눅스의 Compiz처럼 매우 화려한 3D 효과는 보여주지 못하지만, 해당 기능들은 모두 제공된다. 특히 가상 데스크톱이라는 프로그램이 갖추어야 할 점 중 가장 중요한 것은 부프로그램으로서 사용자가 느끼기에 가벼운 소프트웨어여야 한다. 본 시스템은 PC의 사양 및 부하 상태에 따라 동적으로 프레임 수를 조절하여 본 시스템에 할당되는 CPU 부하량을 5퍼센트 미만으로 줄인다. 즉, 저사양의 PC일수록 초당 출력하는 프레임 수를 줄여 애니메이션이 부자연스러울 수는 있겠지만, 다른 프로그램에 거의 영향을 미치지 않으며 작업공간의 전환 속도 또한 PC의 사양에 관계없이 동일한 속도를 유지하게 된다. 또한, 작업공간의 개수를 유동적으로 변경할 수 있어서 기존 출시된 윈도우용 프로그램보다 사용자에게 더 편리한 서비스를 제공할 수 있다. 표 1은 본 논문에서 제안한 시스템과 기존의 시스템에 대한 비교 분석한 내용을 보이고 있다.

표 1 제안 시스템 비교 분석

구분	Compiz	Desktops	Desk Space	제안 시스템
가상데스크톱 기능	○	○	○	○
3D 애니메이션	○	×	○	○
프레임 수의 유동적 조절	○	×	×	○
작업 공간 개수 조절	○	×	×	○
MS 윈도우 설치 여부	×	○	○	○

5. 결론

본 논문에서는 마이크로소프트 윈도우즈 운영체제를 사용하는 사용자들이 작업 시 좀 더 편안하고 쾌적한 환경이 될 수 있도록 여러 개의 가상 데스크톱을 제공하는 시스템을 설계하였다. 리눅스 환경에서 강력한 3D 가상 데스크톱 기능을 제공해주는 Compiz와 유사하게 여러 가지 편의 기능을 윈도우즈 환경에서도 제공받게 됨을 기대할 수 있게 되었다. 운영체제와 독립적인 GUI 구조로 되어 있는 리눅스의 가상 데스크톱 기술을 운영체제에 종속적인 GUI 구조를 가진 윈도우에서 적용시킬 수 있음을 보이고 있다.

참 고 문 헌

- [1] Aaron Weiss, "Desktops in 3D," netWorker, vol.11, Issue 1, pp.26-33, 2007.
- [2] OS Platform Statistics, http://www.w3schools.com/browsers/browsers_os.asp
- [3] Seungpyo Hong, Myeongjin Oh, Junghoon Shin, Sangjun Lee, "Design and Implementation of a 3D Virtual Desktop System based on Windows Operating System," *Proc. of the KIISE Korea Computer Congress 2010*, vol.37, no.1(A), pp.117-118, 2010. (in Korean)
- [4] Seungpyo Hong, Donsu Lee, Sangjun Lee, "Efficient and Flexible 3D Virtual Desktop System in the Windows Environment," *Journal of Measurement Science and Instrumentation, Special Issue for the International Conference on Smart IT Applications*, vol.1 Supplement, pp.178-181, 2010.
- [5] Linux Compiz, <http://www.compiz-fusion.org/>
- [6] Microsoft 『Desktops』, <http://technet.microsoft.com/ko-kr/sysinternals/cc817881%28en-us%29.aspx>
- [7] Otaku Software 『DeskSpace』, <http://www.otakusoftware.com/deskpace>