

영상처리를 이용한 PC 내부구조 학습 어플리케이션 설계 및 구현

Design and Implementation of PC-Mechanic Education Application System Using Image Processing

김 원 진*, 김 형 옥*, 조 성 은*, 장 수 정*, 문 일 영*

Won-Jin Kim*, Hyung-Ook Kim*, Sung-Eun Jo*, Soo-Jeong Jang*, Il-Young Moon*

요 약

영상처리 기술을 이용한 PC정비사 자격증 취득을 돕기 위한 어플리케이션을 소개한다. 요즘 사람들은 컴퓨터의 마우스나, 키보드를 직접 이용하기 보다는 손의 제스처, 즉 흥미를 유발시켜 많은 사람들에게 이미지와 그래픽적인 부분을 추가시켜 소개한다. 요즘 화두가 되고 있는 멀티터치 테이블에 3D Maxs, C#을 이용해 각 부품의 Scale 및 Drag를 통해 카메라 뷰를 이동함으로써 효과적으로 자격증 취득을 돕는다. 또한 PC 정비사 국가 자격증에 출제되는 문제를 수록하고, 각 부품을 세부적으로 학습함으로써, 자격증 취득의 기틀을 마련한다. 이에, 본 논문에서는 멀티터치 테이블(MultiTouch-Table) 기반의 학습용 어플리케이션을 제안하고 그 성능을 시험하였다.

Key Words : MultiTouch- Table, Scale, Drag, Camera View, Pc-Mechanic

ABSTRACT

We introduce the application what using the MultiTouch-Table of the PC-mechanic Certification. Thesedays, People does't use the Mouse and Keyboard and use people gesture. We introduce Graphic and Image by addition. Theseday, MultiTouch-Table is so famous. We use it the multitouch-table to on 3D Maxs and C#. We help them to get the certification using the component Scale and Drags through the camera view and then include the PC-Mechanic question of domestic.

* 한국기술교육대학교 컴퓨터공학부(echo12@kut.ac.k, sn0opy@kut.ac.kr, hellocommi@kut.ac.kr, smilesj@kut.ac.kr, iymoon@kut.ac.kr)

제1저자 (First Author) : 김원진

교신저자 : 문일영

접수일자 : 2011년 11월 21일

수정일자 : 2011년 12월 02일

확정일자 : 2011년 12월 12일

I. 서론

최근에는 사용자의 감성과 편의성과 같은 질적인 측면이 제품과 서비스를 판단하는 새로운 기준으로 자리 잡게 되면서, 직관적인 제어가 가능한 하드웨어 및 UI에 대한 관심 증가와 함께 멀티 터치 기술과 터치 UX가 중요한 화제가 되었다. 하지만 현재 상용화된 멀티 터치 기술은 휴대폰과 같은 소형기기가 대부분이다. 물론 대형 멀티 터치 기기로 MS의 Surface, 스페인 Pompeu Fabra 대학의 Reactable 등이 있으나 높은 가격으로 인해 보급되기에는 어려운 상황에 있다. 이에 넓은 화면에서 보다 많은 수의 터치를 동시에 인식하는 하드웨어를 저렴한 가격으로 제작하여 터치 UX를 구현하고, PC 내부 구조 및 각 부품을 한눈에 쉽게 알아볼 수 있게 하기 위하여 PC 내부 구조 어플리케이션을 개발하게 되었다[1].

제작 과정의 핵심적인 요소는 터치를 인식하는 하드웨어를 구성하고, 이 하드웨어에서 발생한 터치 입력을 소프트웨어 내에서 유효한 멀티 터치 이벤트로 연결하는 부분이다. 실제 구성은 아크릴 패널에서 발생한 터치 입력을 적외선 카메라로 검출한 뒤, 이를 TUIO 프로토콜 통신을 통해 ID와 좌표를 부여하여 WPF 어플리케이션에 전달하고, 이 과정에서 일어나는 화면 갱신을 Project로 Rear Screen에 영사하는 형태로 되어있다. 어플리케이션에 전달된 터치 점의 ID와 좌표 값의 변화를 토대로 내부적으로 구현된 터치 이벤트와 연결되며, 이 이벤트를 이용하여 어플리케이션의 제어가 가능하도록 하였다. 결과적으로 Windows 7 환경이 아니더라도 멀티 터치 이벤트를 사용할 수 있다는 특징이 있다[2],[3].

이에, 본 논문에서는 직접 제작한 멀티터치 테이블을 이용한 PC 정비사 어플리케이션을 소개한다. 본 논문의 구성은 다음과 같다. 제 II장에서는 멀티터치 테이블의 하드웨어 기술, Zam 3d와 3ds 포토샷을 이용한 소프트웨어에 대하여 소개 하도록 한다.

II. MultiTouch-Table 기술

MultiTouch-Table 기술은 영상처리 기술, 광학 기술, 전자회로 기술, 컴퓨터 하드웨어 적인 기술을 담고 있으며, MultiTouch-Table 내부의 C#, Zam 3d, 3D maxs, 포토샵, Touch Frame Work등의 기술을 가지고 있다.

1. MultiTouch-Table alliance

시스템은 PROJECTOR > TABLE > 적외선 카메라 > TBETA > TUIO > WPF > PROJECTOR로 구성되어 있다. 아크릴 패널에서 발생한 터치 입력을 적외선카메라로 검출한 뒤, 이를 TUIO 프로토콜 통신을 통해 ID와 좌표를 부여하여 WPF 어플리케이션에 전달하고, 이 과정에서 일어나는 화면 갱신을 Project로 Rear Screen에 영사하는 형태이다.

2. MultiTouch-Table 개발을 위한 소프트웨어

MultiTouch-Table 개발을 위해서는 다음과 같은 프로그램이 사용된다.

(1) C# 이용

무료이며 UI적인 부분을 표현할수 있으며, Touch Frame work를 사용하기 위해서는 C#을 기반으로 작업을 해야 한다.

(2) 3D Maxs 이용

컴퓨터 본체 내부의 각각의 하드웨어를 그림으로 보여주기 위해서는 3d maxs를 이용한 각 부품을 모델링 해야 한다.

(3) Zam 3d 이용

3D max로 모델링 한 각각의 좌표를 C#으로 불러오기 위해서는 좌표를 잡아주는 Zam 3d를 사용해야 한다.

III. 멀티터치 테이블 하드웨어

1. 시스템 구성도(기술 구성도)

테이블은 DSI 방식을 이용해서 제작하였고 PS3 EYE 카메라에 인화된 필름을 사용해서 적외선 카메라로 개조하여 사용하였다. PROJECTOR는 OPTOMA사의 ES526을 사용했다. 광 터치 방법은 터치하는 부분의 빛의 양을 체크하여 터치를 인식하고 구동할 수 있도록 만드는 방법이다. 광 터치 방법은 일반적으로 사람의 눈에 보이지 않는 적외선 빛을 주로 사용한다. 적외선 빛의 변화를 인식하는 방법에는 적외선 센서를 이용하는 방법과, 적외선 카메라를 통해 인식하는 방법이 있다. 여기서 우리는 멀티 터치 테이블을 만드는데 적합한 방법인 적외선 카메라를 이용한다. 적외선 카메라를 이용하여 광 터

치를 구현하는 방법은 다음 아래와 같다.

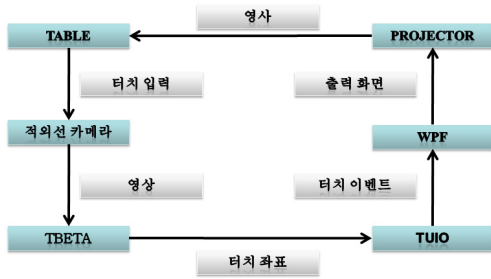


그림 1. 시스템 구성도
Fig. 1. System Diagram

(1) FTIR (Frustrated Internal Reflection)

빛이 밀한 매질(굴절률이 큰 매질)에서 소한매질(굴절률이 작은 매질)로 입사할 때, 입사각이 일정 각도 이상이면 그 경계면에서 빛이 모두 반사되는 현상인 빛의 전반사를 이용하는 방식이다. 아래 그림 2와 같이 적외선이 흐르는 스크린에 손가락이 닿게 되면 적외선이 아래로 난반사되며 반사된 빛을 적외선 카메라로 인식하게 되는 원리이다. 따라서 기본적으로 스크린이 되는 바탕 판이라고 할 수 있는 아크릴 판과 이 아크릴 판의 터치 인식률을 높이기 위한 처리에 이용될 실리콘, 전반사를 이용하기 위한 점광원이 되는 적외선 LED가 사방에 배치된 IR-LED 프레임, 적외선 카메라, 영사지 등의 재료가 요구된다. FTIR 방식에는 밀폐된 별도의 프레임이 필요하지 않음에도, 압력 변화 감지가 가능한 강한 터치 감을 제공한다. 또한 손가락뿐만 아니라 펜이나 다른 물건으로도 유효한 터치가 가능하다는 장점을 가지고 있다. 하지만 빛을 이용하는 만큼 LED 프레임, 실리콘 표면처리 등의 하드웨어 프레임이 필요하며, 반드시 유리 표면이 아닌 아크릴 표면 위에서 사용해야 한다는 제한을 가진다. 또한 객체를 인식할 수 없다는 단점을 가지고 있다. FTIR 방식에서 좋은 터치 점을 얻기 위해서는 아크릴과 영사지 사이에 실리콘을 0.5~1mm 정도의 두께로 실리콘을 고르게 도포하는 이 작업이 매우 중요한 역할을 한다. 실리콘이 도포되고 난 후에 터치를 하게 되면 실리콘이 늘리면서 느낄 수 있는 터치감이 있어 좋은 터치감도 얻을 수 있다. 하지만 고르게 도포를 하는 작업이 어렵고, 만약 고르게 도포 하더라도 나중에 빔 프로젝트를 사용하여 화면을 영사해보면 화질이 떨어지는 현상이 빈번히 발생한다.

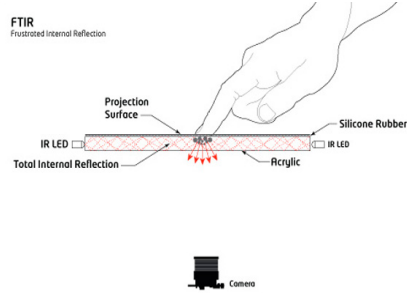


그림 2. FTIR 개념도
Fig. 2. FTIR Diagram

(2) DSI(Diffused Surface Illumination)

기본적으로 FTIR 방식과 일치하며 특수한 아크릴을 사용한다. 이 아크릴 안에는 수 천 개의 작은 입자가 거울과 같은 효과를 내어 적외선 빛이 스크린의 위 아래로 반사된다. 스크린을 터치할 시 터치 점 주변으로 빛이 반사되어 이를 적외선 카메라로 인식한다. FTIR과 DSI 방식은 서로 쉽게 교체가 가능하다. DSI 방식을 사용하게 되면 FTIR 방식에서 필요했던 표면처리과정이 불필요하게 되며 객체 인식이 가능해지고 압력변화에 민감하게 된다. 단점으로는 특수한 아크릴을 사용해야 되기 때문에 보통 아크릴을 사용하는 FTIR 방식보다 비싼 가격을 요한다. 또한 FTIR 방식과 다르게 실리콘으로 표면처리를 하지 않아 사용자가 터치를 하게 될 경우 터치감이 약간 떨어진다.

(3) DI (Diffuse Illumination)

DI 방식은 적외선이 비추는 방향에 따라 Rear 방식과 Front 방식으로 나누어진다. 먼저 적외선을 스크린 아래에서 비추며 이 빛이 터치된 점의 근처에서 반사되는 것을 인식하는 방법인 Rear DI 방식은 FTIR 방식이 오직 실리콘 표면처리를 마친 아크릴 판 위에서만 유효했던 것과 달리 표면처리 과정이 필요하지 않고, 아크릴뿐만 아니라 유리와 같이 투명하고 판판한 재질 위에서도 사용이 가능하다. 또한 LED 프레임도 필요하지 않으며 비교적 간단하게 설치할 수 있다는 장점을 가지고 있다. 더불어 손가락 및 객체의 인식도 가능하지만 터치감이 낮아 터치 오류가 발생할 수 있으며 밀폐된 프레임 안에서 사용해야 한다는 한계점을 가지고 있다.

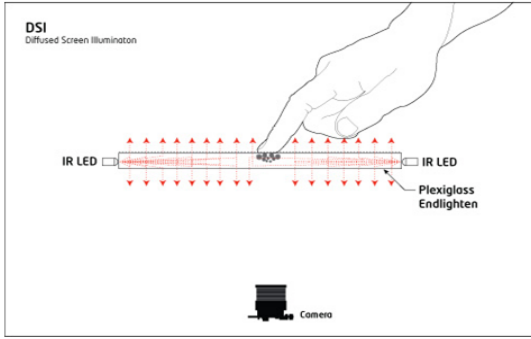


그림 3. DSI 방식의 개념도
Fig. 3. DSI Diagram

Front DI 방식은 스크린의 위에서 적외선을 비추며, 터치 시 발생하는 그림자를 적외선 카메라로 인식하는 방식으로 Rear 방식과 같은 장점을 가지지만 객체의 인식은 불가능하다. 또한 외부의 적외선 환경을 이용할 수 있기 때문에 (조명, 햇빛 등) 편리할 수도 있지만, 주변 환경에 의존하여 사용하는 것이므로 신뢰성이 다소 떨어진다는 단점을 가지고 있다. DI 환경을 만들기 위한 재료는 투명하고 강한 재질의 패널과 적외선 발광기, 영사지, 적외선 카메라로 Rear와 Front 방식 모두 동일하다.

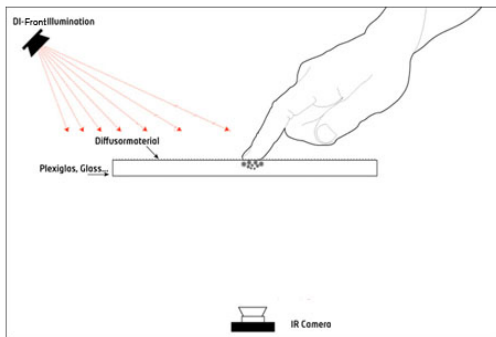


그림 4. Front Diffuse Illumination 개념도
Fig. 4. Front Diffuse Illumination Diagram

2. 하드웨어 구성

하드웨어에는 크게 두 부분으로 구성할 수 있다. 화면이 보이고 그 화면에 터치를 하는 상단부와 적외선 카메라와 프로젝터 거울 등이 있는 하단부로 나눌 수 있다. 우선 하단부에는 프로젝터에서 나오는 영상이 스크린 부분에 영사 될 수 있어야 된다. 빛을 거울에 반사시켜 스크린 부분에 영사가 되도록 구성

한다. 그 다음 스크린에서 터치가 발생되면 인식하기 위해서 적외선 카메라를 설치하는 것인데 아래 그림처럼 스크린 하단부분에 카메라를 설치하여 빛의 변화 모습을 찍을 수 있도록 한다. 그리고 IR-LED Frame, PC, Project 등 전원을 연결할 수 있도록 한다.

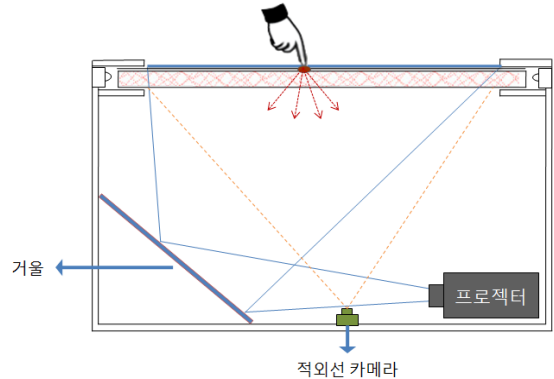


그림 5. 하드웨어 구성
Fig. 5. Hardware Composition

IV. 멀티터치 테이블 개발을 위한 소프트웨어

소프트웨어는 영상처리를 통해 터치입력을 받아서 동작하는 프로그램으로 영상처리에는 TBETA 프로그램을 사용하였다. 영상처리를 통해서 추출된 좌표는 소프트웨어에 전달되어 터치이벤트를 발생하게 한다. 프로그램을 실행하면 PC 내부 구조 학습 어플리케이션의 메인화면이 나타난다. 이 어플리케이션 여러 개의 손가락을 사용해 조작하기 때문에 손가락이 화면에 찍는 좌표 값을 분석해 터치이벤트를 연결시켜 주어야 동작할 수 있다.

1. 영상처리 프로그램(TBETA)

(1) 시뮬레이션 프로그램

TBETA라는 이름으로 더 많이 알려져 있는 Community Core Vision(CCV)[4]는 컴퓨터 비전 및 하드웨어 센싱을 위한 오픈소스 플랫폼으로, 다양한 웹캠 및 비디오 장치 비디오와의 인터페이스 생성이 가능할 뿐만 아니라 TUIO[5], OSC, XML과의 연동이 가능하며, FTIR, DI, DSI, LLP 등의 미래형 멀티터치 비전을 위한 조절환경도 제공한다. 좀 더 구체적인 기술로 설명하자면, 입력 스트림과 출력되는 트래킹 데이터(각 조정 값에 의하여 조절된) 및 멀티터치 환경에서의 이벤트(finger down, moved,

released 등)를 가져와 TUJO 등의 프로토콜 통신을 통해 전달할 수 있다.

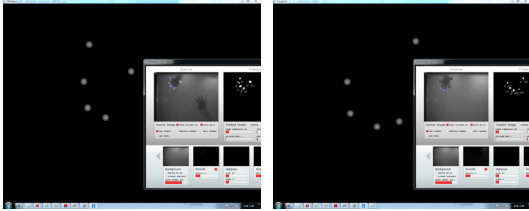


그림 6. TBETA 출력 화면
Fig. 6. TBETA OUTPUT SCREEN

(2) 클래스 구성도

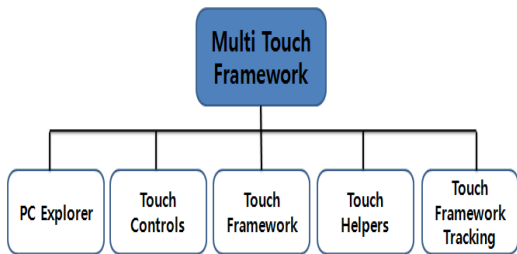


그림 7. MultiTouch Framework 구성도
Fig. 7. MultiTouch Framework Diagram

구성은 5개의 프로젝트로 구성되어 있는 Mindstorm사에서 지원하는 Multi-touch framework 인 Breeze V1.5[6]를 사용하였다. 구체적으로는 CC와 IR이 구현되어 있는 PC Explorer와 컴포넌트(Button, CheckBox, ListBox 등)의 터치 이벤트와 통신 이벤트를 정의해 주는 TouchControls와 TUJO를 통해서 들어오는 터치 정보를 트래킹해서 전달해주는 역할을 하는 TouchFrameworkTracking, 그리고 터치가 작동하는 객체에 대한 정보를 가지고 있는 TouchFramework와 기본적인 터치 이벤트와 각종 기타 계산을 해주는 TouchHelpers로 구성되어 있다. 이 Framework는 TouchFrameworkTraking 프로젝트가 OSC를 통해 터치 좌표를 전송받는 것을 시작으로 해서 작동한다. 전송 받은 좌표는 TouchFrameWork 프로젝트에서 이벤트가 정해지고 TouchControl 프로젝트에서 각 객체에 대한 이벤트가 발생한다. PC Explorer 프로젝트는 모든 객체를 등록하고 인터페이스를 구성해 발생하는 이벤트를 사용자에게 보여주는 역할을 한다.

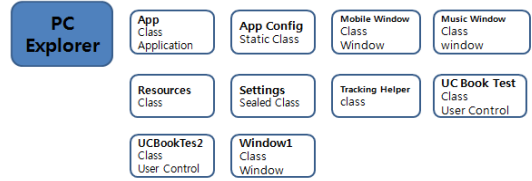


그림 8. Pc Exploler 구성도
Fig. 8. Pc Exploler Diagram

PC Explorer 프로젝트는 앞에서 말한 대로 PC Explorer의 기본 구조를 정의한다. Window1.class는 기본 메인 화면을 구성하며 각 Button을 통해서 각 프로그램인 CC와 IS로 이동 할 수 있다. MobileWindow.class는 IS을 구현한 클래스이며 MusicWindow.class는 CC를 구현한 클래스로 각 기능에 대한 작동이 정의 되어 있다. 이 PC Explorer 클래스는 이미지나 버튼이 추가 되면 TouchFramework 프로젝트에 있는 MITContainer 객체를 생성하여 등록을 하여 터치 이벤트가 입력 될 시 MITContainer.class를 통해서 이벤트가 작동 되도록 한다.

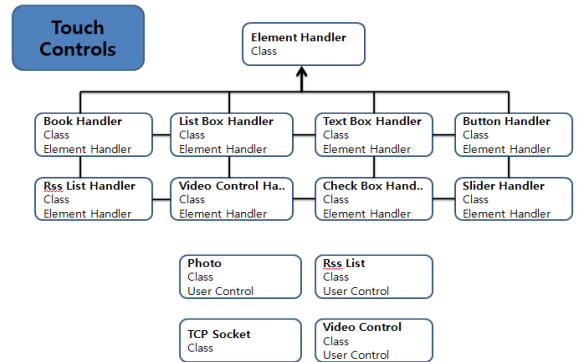


그림 9. Touch Controls 클래스 다이어그램
Fig. 9. Touch Controls Class Diagram

TouchControls는 각 컴포넌트(Button, CheckBox, ListBox 등)에 대한 이벤트를 정의하고 통신을 위해서 추가한 TCPsocket.class로 이루어져 있다. ElementHandler는 Button, ListBox, TextBox, RssList, VideoControl, CheckBox, Slider에 대한 이벤트가 정의되어 있고 각 컴포넌트에서 따로 정의되어야 할 이벤트들은 각 클래스에서 재정의 되고 있으며 이미지를 위한 Photo.class나 동영상을 위한

VideoControl.class로 이루어져 있다. 이 프로젝트는 TouchFramework로부터 작동되는 이벤트에 대한 동작이 정의 되어 있어 실제로 각 컴포넌트에 이벤트가 발생 시 직접 동작하는 프로젝트이다.

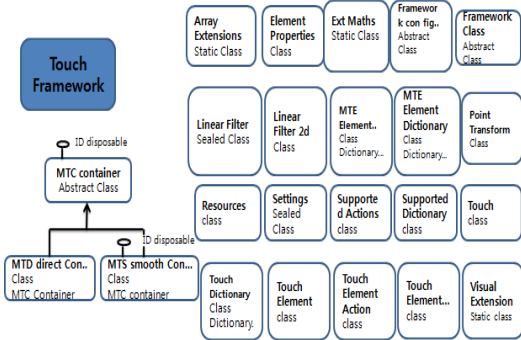


그림 10 TouchFramework 클래스 다이어그램
Fig. 10 TouchFramework Class Daigram

TouchFramework 프로젝트는 각 객체들에 대한 정보를 가지고 있으며 Tracking 프로젝트로부터 들어오는 터치 좌표를 받아 각 객체에서 발생될 이벤트를 TouchControl 프로젝트로 넘겨준다. 이 프로젝트에서 발생될 이벤트가 정해지며 전송된 터치좌표가 의미를 가지는 부분이라고 할 수 있다. TouchUp, slide, Drag, TouchDown, Scroll, Tap, Move, Scale, Rotate, Tick 이벤트가 정의 되어 있고 MIDirectContainer.class와 MTSmoothContainer.class에서 필요한 부분은 재정의 되어 지고 있다. MIDirectContainer.class는 Button이나 ListBox와 같은 기본 컴포넌트에 대한 정보를 가지고 있고 MTSmoothContainer는 이미지나 동영상에 대한 정보를 가지고 있다.

TouchFrameworkTraking 프로젝트는 터치 좌표가 OSC를 통해 들어온 터치 좌표를 TUIO를 통해서 좌표를 인식하고 TouchFramework로 좌표를 넘겨준다. 이 프로젝트에서 터치 좌표의 추가와 삭제가 이루어지고 드래그에 대한 판별이 이루어진다.

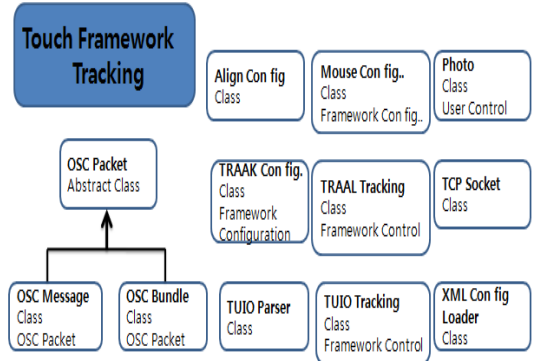


그림 11 TouchFrameworkTracking 클래스 다이어그램
Fig. 11 TouchFrameworkTracking Class Daigram

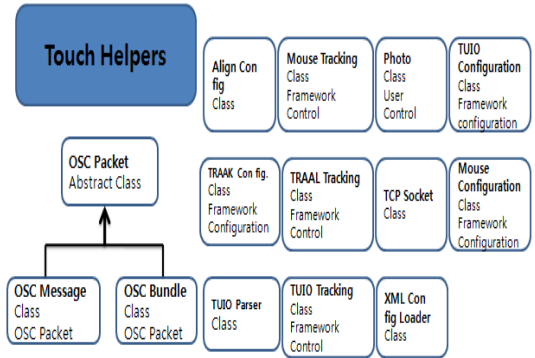


그림 12 TouchHelpers 클래스 다이어그램
Fig. 12 TouchHelpers Class Daigram

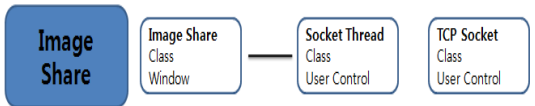


그림 13 ImageShare 클래스 다이어그램
Fig. 13 ImageShare Class Daigram

단말기에 들어가는 ImageShare의 클래스 다이어그램으로 통신을 담당하는 TCP Socket과 주 메인 인터페이스를 구성하는 ImageShare 클래스로 구성되어 있다. ImageShare 클래스는 단말기의 센서 변화를 측정하는 sensorManager와 Thread 작동을 통해 통신을 하는 SocketThread 클래스로 구성되어 있다. TCPSocket은 SendImage, ReceiveImage를 통해서 데이터를 주고 받도록 되어있다.

본 논문은 터치 이벤트에 대한 선언과 회전 각도나 객체의 중심 좌표를 구하는 기능들을 가지고 있

