


캡스톤 디자인 I

종합설계 프로젝트

프로젝트 명	Depth map 을 활용한 후방카메라의 한계 개선 D - CarMera
팀 명	천 리 안
문서 제목	결과보고서

Version	1.4
Date	2015-05-26

팀원	최승혁
	신태섭
	박성우
	신동호
	최성현
	박민욱
지도교수	한재일

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26


CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인I 수강 학생 중 프로젝트 "**Depth map을 활용한 후방카메라의 한계 개선**"를 수행하는 팀 "**천리안**"의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 "**천리안**"의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역


Filename	중간보고서-Depth map을 활용한 후방카메라의 한계 개선 - D - CarMera
원안작성자	최승혁, 신태섭, 박성우, 신동호, 최성현, 박민욱
수정작업자	최승혁, 신태섭, 박성우, 신동호, 최성현, 박민욱

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2015-05-22	신동호	1.0	최초 작성	
2015-05-23	최성현 신동호 최승혁	1.1	내용 추가	영상처리 수행 내용
2015-05-24	신태섭 박민욱 박성우	1.2	내용 추가	자동차 제어 수행 내용
2015-05-24	신태섭 최승혁	1.3	내용 추가	향후 계획, 오타 검토
2015-05-25	팀원 전원	1.4	최종 검토	수정 및 최종 검토

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

목 차

1	개요	4
1.1	프로젝트 개요	4
1.2	추진 배경 및 필요성	4
1.2.1	후방카메라 및 후방초음파센서(BWS)의 시스템 현황	4
1.2.2	후방카메라 및 후방초음파센서(BWS)의 문제점	4
1.2.3	Depth map을 활용한 후방카메라(BWS)의 필요성	6
2	개발 내용 및 결과물	7
2.1	목표	7
2.2	연구/개발 내용 및 결과물	8
2.2.1	연구/개발 내용	8
2.2.1.1	Depth map	8
2.2.1.2	Calibration	8
2.2.1.3	영상처리	9
2.2.1.4	자동차 제어	9
2.2.2	수행내용	11
2.2.2.1	Depth map	11
2.2.2.2	Calibration	13
2.2.2.3	영상처리	14
2.2.2.4	자동차 제어	17
2.2.3	시스템 기능 요구사항	19
2.2.4	시스템 비기능(품질) 요구사항	19
2.2.5	시스템 구조 및 설계도	20
2.2.6	활용/개발된 기술	21
2.2.7	현실적 제한 요소 및 그 해결 방안	22
2.2.8	결과물 목록	24
2.3	기대효과 및 활용방안	24
1.	기대효과	24
2.	활용방안	24
3.	참고 문헌	25
4.	부록	25
	<사용자 매뉴얼>	25

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

1 개요

1.1 프로젝트 개요

현재 모든 차량에는 운전자에게 후진 시 후방의 정보를 제공하기 위해 후방카메라 및 후방초음파센서(BWS)가 장착되어 있다. 자율 주차, 무인 주행 등 차량 산업의 발전에 맞춰 발전된 부가기능이 개발 중 이다. 하지만 현재 상용화된 후방 카메라 및 후방 초음파센서(BWS)는 측정 범위 및 물체에 많은 문제점을 가지고 있다.

본 프로젝트는 기존 후방카메라에서 image 영상만을 제공하여 발생하는 문제점을 보완하기 위해 Kinect 의 depth map을 활용하였다.

Kinect의 Depth map 영상과 Image 영상을 합성하여 운전자에게 장애물까지의 거리정보를 시각적으로 제공하는 영상소프트웨어를 개발한다.

이 영상소프트웨어를 통해 운전자가 보는 화면에 가까운 물체를 하이라이트 처리하여 운전자가 후방 정보에 대해 좀더 확실한 정보를 얻을 수 있도록 도와준다.

또한 본 프로젝트는 추가적으로 영상소프트웨어를 모형차량에 탑재하여 후방의 장애물에 근접하게 접근되었을 때 구동이 제어되는 테스트 환경을 구축한다.


1.2 추진 배경 및 필요성

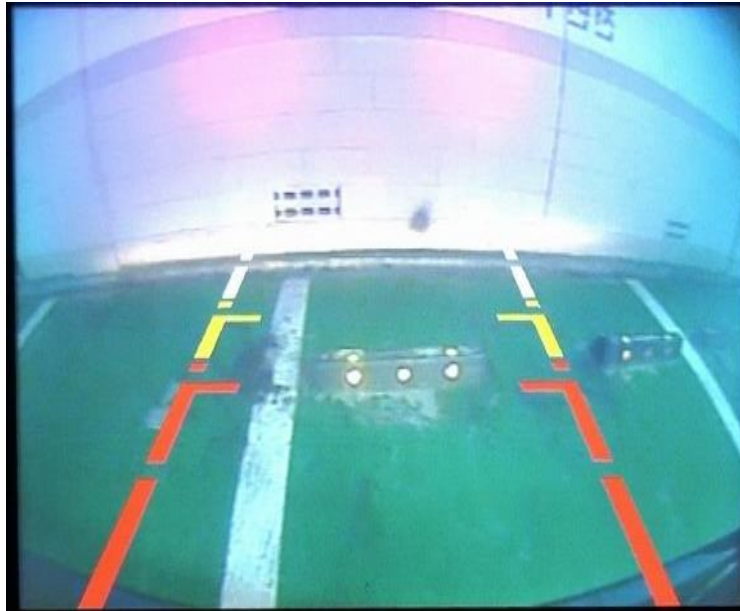
1.2.1 후방카메라 및 후방초음파센서(BWS)의 시스템 현황

- 1) 후방카메라는 현재 image 영상만을 제공한다.
- 2) 후방초음파센서(BWS)는 시야 사각지대를 초음파센서를 사용, 감시하고 해당지역에 장애물이 있으면 운전자에게 경보를 통해 청각적 정보 제공한다.
- 3) 후방초음파센서(BWS)는 40kHz의 초음파를 발산, 초음파가 물체에 부딪쳐 되돌아 올 때까지의 시간을 측정하여 물체까지의 거리를 측정한다.

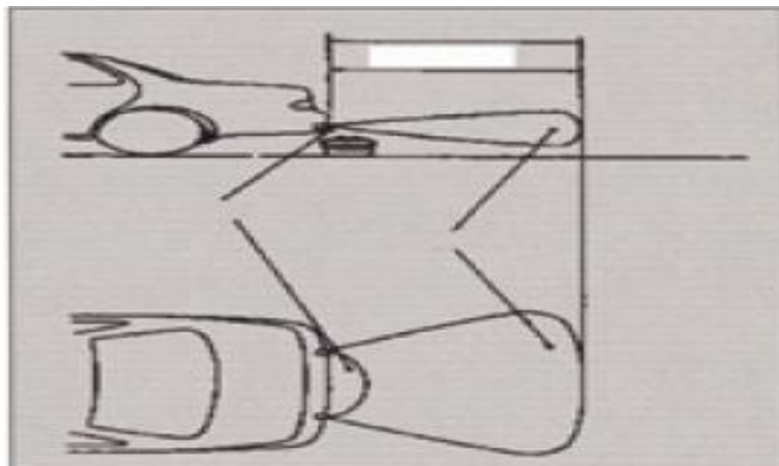
1.2.2 후방카메라 및 후방초음파센서(BWS)의 문제점

- 1) 후방카메라는 image 영상만을 제공하기 때문에 물체까지의 정확한 거리측정이 불가능하다.


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26



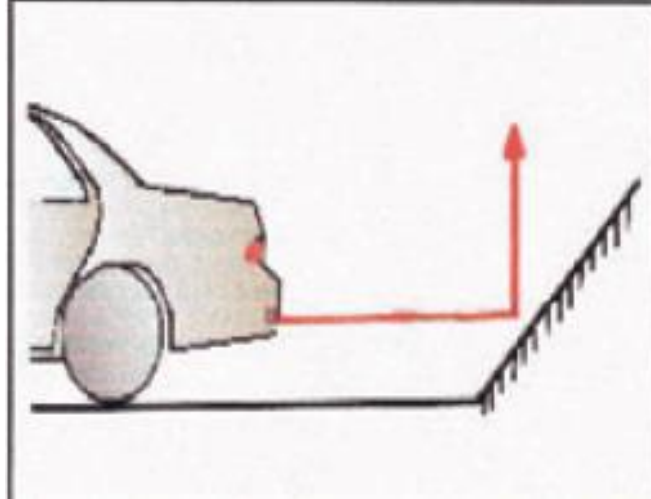
- 2) 후방초음파센서(BWS)는 자동차 범퍼와 가까운 지역은 감지 못하는 영역이 존재한다.



- 3) 후방초음파센서(BWS)는 초음파의 특성 때문에 철사, 체인, 자갈, 눈에 덮여 있는 소화전등 감지 못하는 물체가 존재한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

- 4) 후방초음파센서(BWS)는 주변환경에 따라 초음파 반사로 인해 불규칙한 경보 음이 발생해 정확한 정보를 제공할 수 없다.



1.2.3 Depth map을 활용한 후방카메라(BWS)의 필요성


현재 상용화 되어있는 후방카메라 및 후방초음파센서(BWS)는 일반 image 영상과 초음파센서만을 이용하여 운전자에게 후방 정보를 제공하기 때문에 인식되는 범위(사각지대, 거리) 및 장애물(모양, 재질)에 따라 한계가 있다. 이러한 한계 때문에 후방 진입 시 많은 사고가 발생하고 있다. 미국의 한 연구조사기관(IIHS)에 따르면 매년 후진 관련 사고로 평균 210명이 사망하고, 1만5천여 명이 부상당하는 것으로 집계된다. 이러한 조사 결과처럼 현재 상용화된 후방카메라 및 후방초음파센서(BWS)는 운전자에게 보다 정확하고 현실감 있는 정보를 제공하고 있지 못하다. 이러한 문제점을 해결하기 위해 Depth map 영상과 image 영상을 합성하는 소프트웨어 개발을 통해 기존 시스템의 인식한계를 극복하고 더 현실감 있는 시각적 정보를 운전자에게 제공하여 후방 진입 및 주차 시 발생하는 사고를 절감한다. 또한 이렇게 개발한 소프트웨어를 모형차량에 테스트 환경을 구축하여 후방 장애물에 차량이 접근하였을 때 구동을 제어하는 테스트 환경을 구축하여 충돌사고를 막는다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2 개발 내용 및 결과물

2.1 목표

- 1) Depth map 정보를 이용하여 자동차 운전자에게 자동차 후방과 장애물 사이의 직관적인 깊이 정보를 제공할 수 있는 영상처리 알고리즘을 구현한다.
- 2) 후방 영상카메라와 Depth map을 측정하는 Kinect의 Depth map 카메라의 영상을 Calibration하여 두 영상 사이의 이질감을 최소화한다.
- 3) 모형 자동차를 통하여 Depth map을 활용한 후방카메라의 기능 테스트를 가능하게 한다.
- 4) 기존의 후방카메라와 초음파 센서의 한계점인 거리감의 부재와 사각지대의 존재를 Depth map을 이용하여 개선한다.
- 5) Depth map 정보를 활용하여, 모형차량과 장애물 사이의 충돌 가능성이 높아진 경우 사용자에게 위험을 알리고, 모형차량에 제동을 하는 제어 알고리즘을 구현한다.

 <div> 국민대학교 컴퓨터공학부 캡스톤 디자인 I </div>	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2.2 연구/개발 내용 및 결과물

2.2.1 연구/개발 내용


2.2.1.1 Depth map

해상도 512 * 424 정도의 비교적 고화질의 depth 영상을 출력하는 것을 목표로 Kinect부터 depth 영상을 제공받는다. Kinect를 pc와 연결하고, Kinect를 구동하기 위한 SDK를 설치한다. Kinect studio를 실행시키면, 기존의 카메라 영상 (color image)와 depth image를 함께 얻을 수 있다. 이 두 영상을 가지고 정확히 일치하도록 합성해야 한다.



2.2.1.2 Calibration

위에 제시된 그림이 Kinect 를 pc에 연결하고, Kinect studio를 실행하여 얻은 depth 영상과 color 영상을 캡처한 모습이다. 이제 이 두 개의 영상을 합성하는 작업을 해야 한다. 그런데 kinect의 카메라 중에서 color 영상을 촬영하는 카메라와 depth 영상을 촬영하는 카메라의 위치가 약간 떨어져 있기 때문에, 같은 곳을 촬영하더라도 두 영상의 위치가 완벽히 일치하지 않게 된다. 따라서 두 영상을 정확히 일치하도록 Calibration 알고리즘을 적용하여 깔끔하게 합성하는 작업이 필요하다. 그리고 depth 영상에서 나온 물체의 테두리가 깔끔하게 촬영되지 않는(데이터가 튀는) 현상이 종종 발생하게 된다. 이를 해결하기 위해서도 보정하기 위한 알고리즘을 정확하여 좀 더 정밀하고 정확한 영상을 획득할 수 있도록 해야 한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2.2.1.3 영상처리

합성한 영상에서 물체와 카메라 사이의 거리에 따라 하이라이트 표시를 해야 한다. 운전자에게 물체와 자동차 사이의 거리 정보를 확실하게 제공할 수 있고, 시각정보를 기존의 카메라보다 눈에 띄게 나타낼 수 있기 때문이다. Visual Studio 2012를 이용하여, 합성된 영상에서 일정 거리 이하의 물체에 도형이나 색깔로 표시를 하여 영상 처리를 한다. 그리고 필요하다면, 하이라이트 표시 이외에 경고음을 발생시켜서 운전자에게 확실히 각인시켜 줄 수 있는 방법도 생각해 볼 것이다.

2.2.1.4 자동차 제어

Kinect로부터 획득한 Depth정보를 이용하여 모형차량 제어를 해야 한다. 따라서, Kinect와 연결되어 있는 PC와 모형차량 간의 통신이 가능해야 하는데 그러기 위해선 먼저

모형차량의 전체적인 제어를 할 수 있는 Raspberry Pi를 설정해야 한다. Raspberry Pi 설정을 단계별로 나타내면 다음과 같다.

1. SD카드에 OS 설치

Raspberry Pi에서 사용하는 운영체제는 일반적으로 Raspbian을 사용한다. Raspbian은 Debian linux가 Raspberry Pi용으로 포팅 된 운영체제이다.

2. Wireless 네트워크 환경 설정

운영체제가 설치되면 네트워크 설정을 해야 한다. Wireless환경을 설정하기 위해서 Wifi동글을 이용하거나 블루투스 모듈을 사용해야 한다. 데이터 전송속도를 고려하여 Wifi동글을 이용하여 설정한다.

3. Client/Server 모델을 적용할 언어 선정

통신을 위하여 네트워크모델을 정해야 한다. PC를 Server로, Raspberry Pi를 Client로 하여 설정하고, 이 모델을 적용할 언어를 설정한다. 개발언어는 C언어, Java등을 사용할 수 있다.

또한, Raspberry Pi와 모형차량의 각 모듈의 실질적인 제어를 하는 Micro controller와 UART를 통신을 위해 GPIO핀을 사용하기 위한 설정을 해야 한다. 이를 단계별로 나타내면 다음과 같다.

1. Wiring Pi 모듈 설치

Wiring Pi 모듈은 Raspberry Pi의 GPIO핀에 접근할 수 있게 하는 C언어로 작성된 라이브러리들을 사용할 수 있게 한다. Wiring Pi는 아두이노에서 사용하는 Wring system과 유사하기 때문에, 쉽게 사용이 가능하다.

wiringPi		3.3V	① ②	5V		wiringPi
8	I2C_SDA	GPIO02	③ ④	(5V)		
9	I2C_SCL	GPIO03	⑤ ⑥	GND		
7	GPCLK	GPIO04	⑦ ⑧	GPIO14	UART_TXD	15
		(GND)	⑨ ⑩	GPIO15	UART_RXD	16
0		GPIO17	⑪ ⑫	GPIO18	PCM_CLK	1
2		GPIO27	⑬ ⑭	(GND)		
3		GPIO22	⑮ ⑯	GPIO23		4
		(3.3V)	⑰ ⑱	GPIO24		5
12	SPI_MOSI	GPIO10	⑲ ⑳	(GND)		
13	SPI_MISO	GPIO09	㉑ ㉒	GPIO25		6
14	SPI_SCLK	GPIO11	㉓ ㉔	GPIO08	SPI_CE0	10
		(GND)	㉕ ㉖	GPIO07	SPI_CE1	11

위의 GPIO그림과 같이 GPIO14, 15번 핀에 Wiring Pi 라이브러리를 이용하여 Controller와 값을 주고 받을 수 있다.

2. Wiring Pi 라이브러리를 이용하여 차량 제어 프로그래밍

앞서 선택한 개발 언어를 이용하여 알고리즘을 구현한다. 이 프로그램에는 Wiring Pi 라이브러리와 서버/클라이언트 라이브러리를 사용한다. 이를 통해 PC에서 Depth map 정보를 이용하여 해당하는 제어를 Raspberry Pi로 전송할 수 있고, 해당하는 제어에 맞도록 차량을 구동할 수 있게 된다.

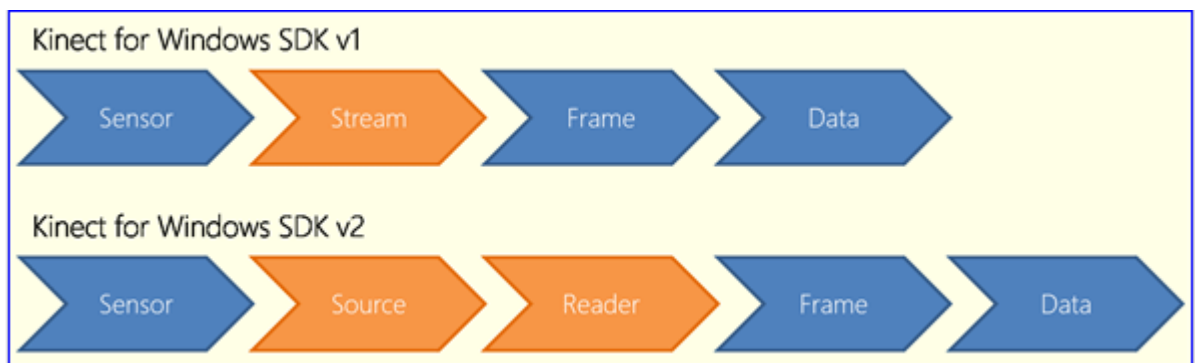
 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2.2.2 수행내용

2.2.2.1 Depth map

<구현 결과>


기존 Kinect v1 for windows를 사용하려고 하였으나 야외에서의 Depthmap 획득이 어렵다는 지적받아 그런 문제점을 해결하기 위해 Kinect v2 for windows로 장치를 교체하였다. 그리고 Kinect v2 for windows를 사용해 보니 Kinect v1 for windows보다 더 선명하고 정밀한 depth map을 획득 할 수 있었다. Kinect v1 for windows 와 Kinect v2 for windows의 윈도우 환경에서 장치를 구동할 수 있게 해주는 SDK의 버전이 다르기 때문에 기존 SDK 1.8에서 2.0 버전으로 교체하였다.

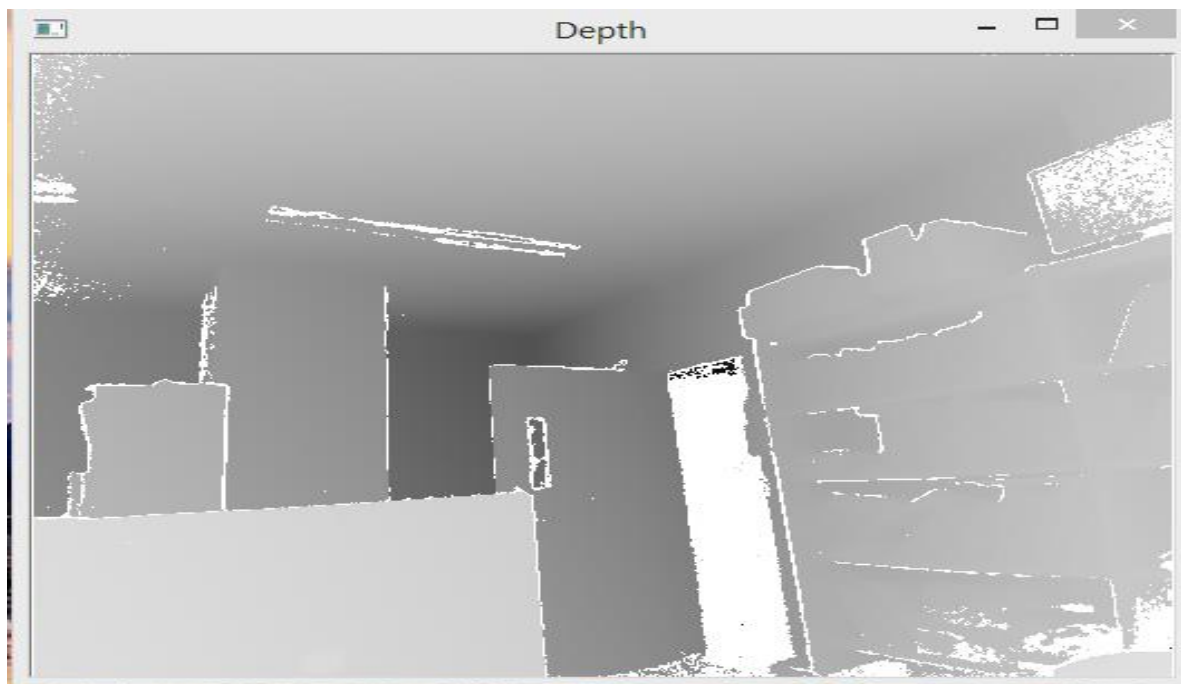


Kinect SDK v1 Sensor 에서 Stream 에서 Frame 을, Frame에서 Data를 검색하는 흐름이다. Kinect SDK v2 Sensor 에서 얻은 Source 에서 Reader 를 열고, Reader 에서 Frame 을, Frame에서 Data를 검색하는 흐름이다. Kinect SDK v2 Source 는 Kinect에서 Color 와 Depth 데이터의 흐름마다 존재 한다. 이것은 Kinect SDK v1 의 Stream 과 동일하다.

		Kinect v1	Kinect v2
색상 (Color)	해상도 (Resolution)	640 × 480	1920 × 1080
	fps	30fps	30fps * 3
심도 (Depth)	해상도 (Resolution)	320 × 240	512 × 424
	fps	30fps	30fps
인물 영역 (Player)		6 명	6 명
인물 자세 (Skeleton)		2 명	6 명
관절 (Joint)		20 관절 / 사람	25 관절 / 사람
손의 개폐 상태 (Hand State)		△ (Developer Toolkit)	○ (SDK)
심도의 취득 범위 (Range of Depth)		0.8 ~ 4.0m (Near Mode 0.4m ~) (Extended Depth ~ 10.0m)	0.5 ~ 8.0m
사람의 감지 범위 (Range of Detection)		0.8 ~ 4.0m (Near Mode 0.4 ~ 3.0m)	0.5 ~ 4.5m
각도 (Angle) (Depth)	수평 (Horizontal)	57 번	70도
	수직 (Vertical)	43 번	60도
틸트 모터 (Tilt Motor)		○	× (수동)
여러 애플리케이션		× (단일 앱)	○
표 1 Kinect v1과 Kinect v2 센서의 사양 비교			

Kinect의 sensor로부터 depth를 얻기 위한 source를 받아오고, depth 구조를 위한 Source 인터페이스를 정의하였다. depth 구조를 위한 Reader 인터페이스를 열었고, depth 데이터를 처리하기 위해 open cv의 cv::Mat 형을 사용하여 프로그래밍 하였다. 여기서 buffer Mat은 16비트로 이루어진 원시 데이터인데 이를 화면에 영상으로 출력하기 위해서 8비트로 나타내었다. 따라서 출력된 depth map 영상은 0~255 사이의 값을 갖게 된다. 0에 가까울수록 검은색을 나타내게 되는데, 이 때는 물체로부터 멀리 떨어져 있음을 의미하고, 255에 가까울수록 흰색을 나타내게 되는데, 이는 물체로부터 가까이에 위치함을 나타낸다. Kinect v2 의 경우에는 최소 0.5m 정도부터 최대 4.5m 떨어진 물체까지 depth를 인식하여 영상에 0~255에 해당되는 색상으로 출력해준다. 출력된 depth 영상을 보면 다음과 같다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26




2.2.2.2 Calibration

<구현 결과>

1차 중간발표 때, 광학 이미지와 depth 이미지를 합치는 부분에 대해서 구체적으로 연구하고 조사해야 할 필요가 있다는 지적을 받았다. 우선 광학이미지 (color영상) 는 depth 이미지를 출력하는 것과 비슷한 방법으로 얻어 올 수 있다. sensor로부터 color 영상 구조를 위한 source 인터페이스를 가져와서 color 이미지 정보를 저장하였다. 그 다음 reader 인터페이스를 열어서 Kinect v2의 이미지 영상을 처리하기 위해 적절한 데이터 크기를 할당하였다. Color 이미지를 처리하기 위해 open cv의 cv::Mat형을 선언하였고, buffer Mat의 원시 이미지 데이터에서 실제 영상 데이터를 얻기 위해서 부호 없는 8비트의 정수로 나타내기 위해 color Mat를 설정하였다. 이로써 4 채널로 나란히 1 화소를 표현하는 데이터 형식을 나타내었다. 프레임에서 color이미지를 얻은 뒤 다루기 쉬운 R,G,B,A 형으로 변환하여 나타내었다. 이렇게 얻어진 광학이미지 영상과 앞서 구했던 depth 영상을 합성하였다. Color 영상의 크기에 depth 영상을 표현하였다. Color 영상의 height와 width만큼에 depth 영상의 x 좌표와 y 좌표를 Mapping하여 MapColorFrameToDepthSpace() 함수를 사용하여 영상을 합성하였다.

지난 발표 이전에는 depth 영상에 color 영상을 Mapping하는 방식을 사용하였는데, 그 때 물체의 일부가 겹쳐 보이는 noise 현상이 발생하여서, 위에서 설명한 방식으로 변경함으로써 깔끔하게 두 개의 영상을 합성할 수 있었다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

Color 영상과 depth 영상을 합성한 모습은 다음과 같이 나타난다.




2.2.2.3 영상처리

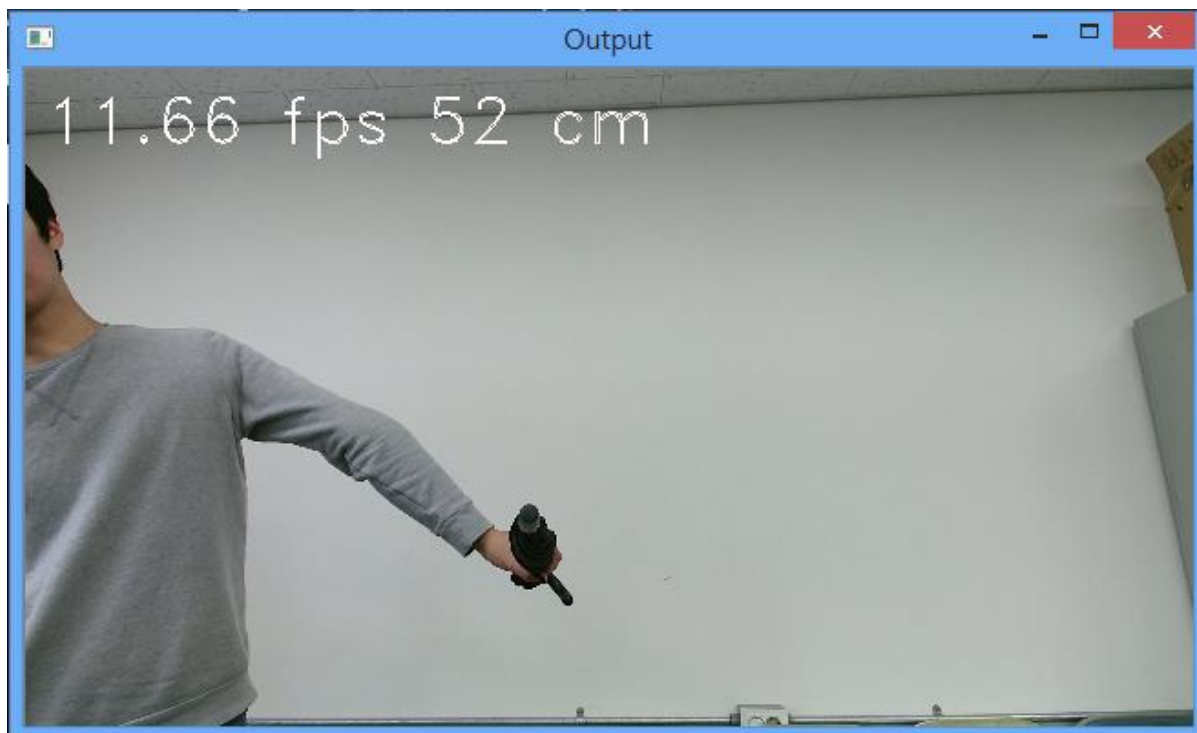
<구현 결과>

1차 구현 당시 depth 영상을 기준으로 image 영상을 calibration 해서 영상이 이중으로 겹쳐 보이는 문제점을 지적 받아서 이번에 그 문제점을 해결하기 위해 image영상을 기준으로 잡고 depth 영상을 스케일해서 calibration을 진행하여 문제를 해결하였다. 이렇게 calibration된 영상에서 운전자에게 보다 확실하게 시각적인 정보를 제공하기 위해 4가지 모드의 영상효과를 처리하였다. 먼저 1)효과처리가 되지 않은 영상과 2) depth가 작을수록 빨간색 농도를 진하게 처리한 영상, 3) depth가 작아질수록 영상에 보이는 물체의 깜박거리는 속도를 빠르게 처리한 영상, 4) 거리가 가까울수록 물체의 밝기가 밝아지게 처리한 영상 으로 구현하였다.

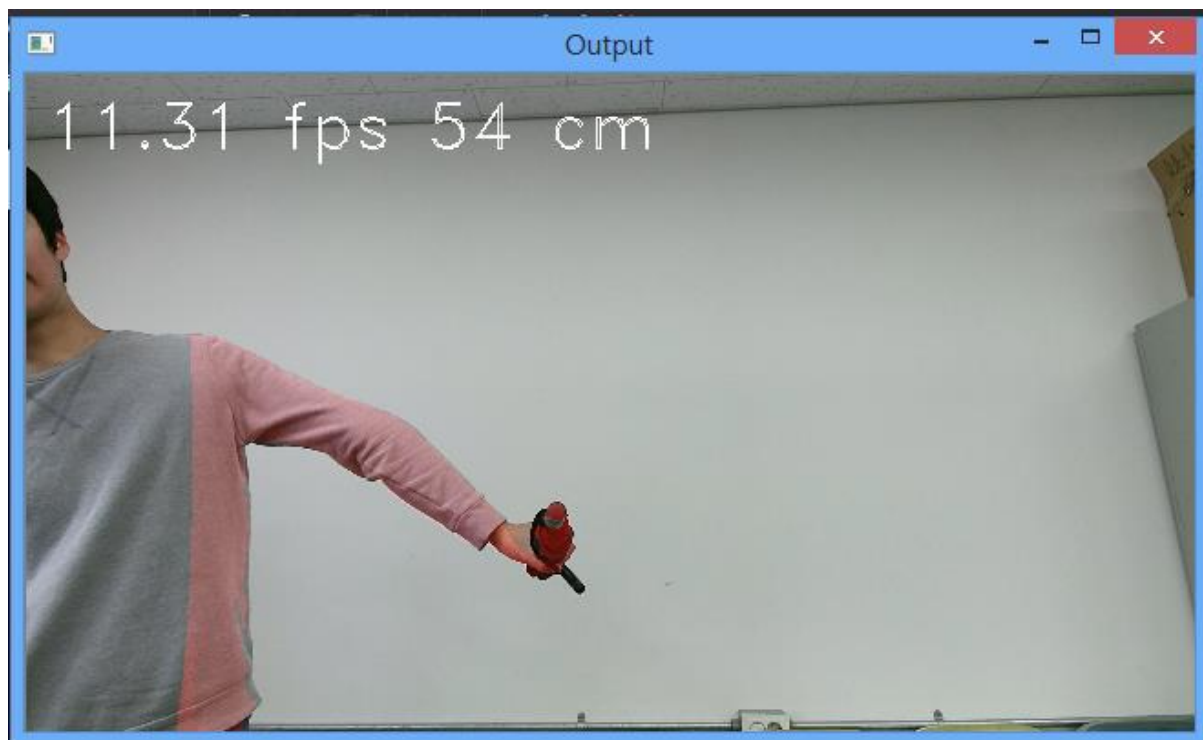
그 영상들을 캡처한 부분을 보면 다음과 같다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

1) 효과 처리가 되지 않은 영상

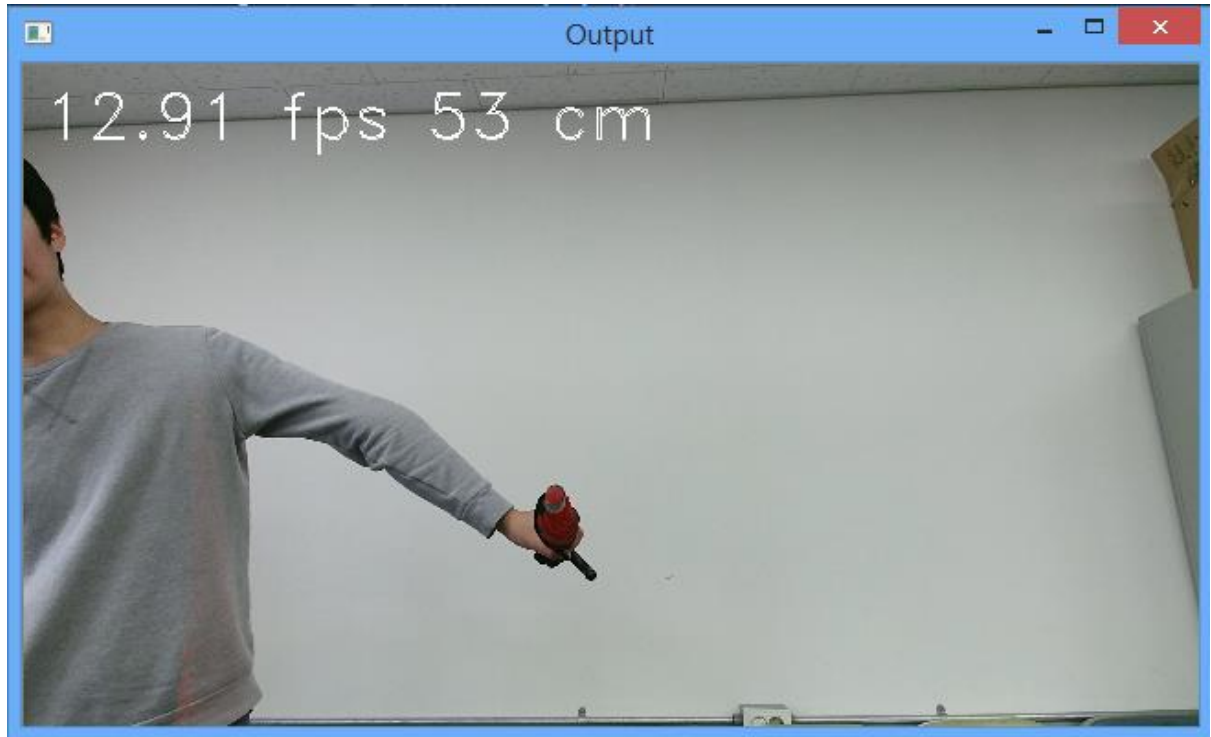


2) depth가 작을수록 빨간색 농도를 진하게 처리한 영상

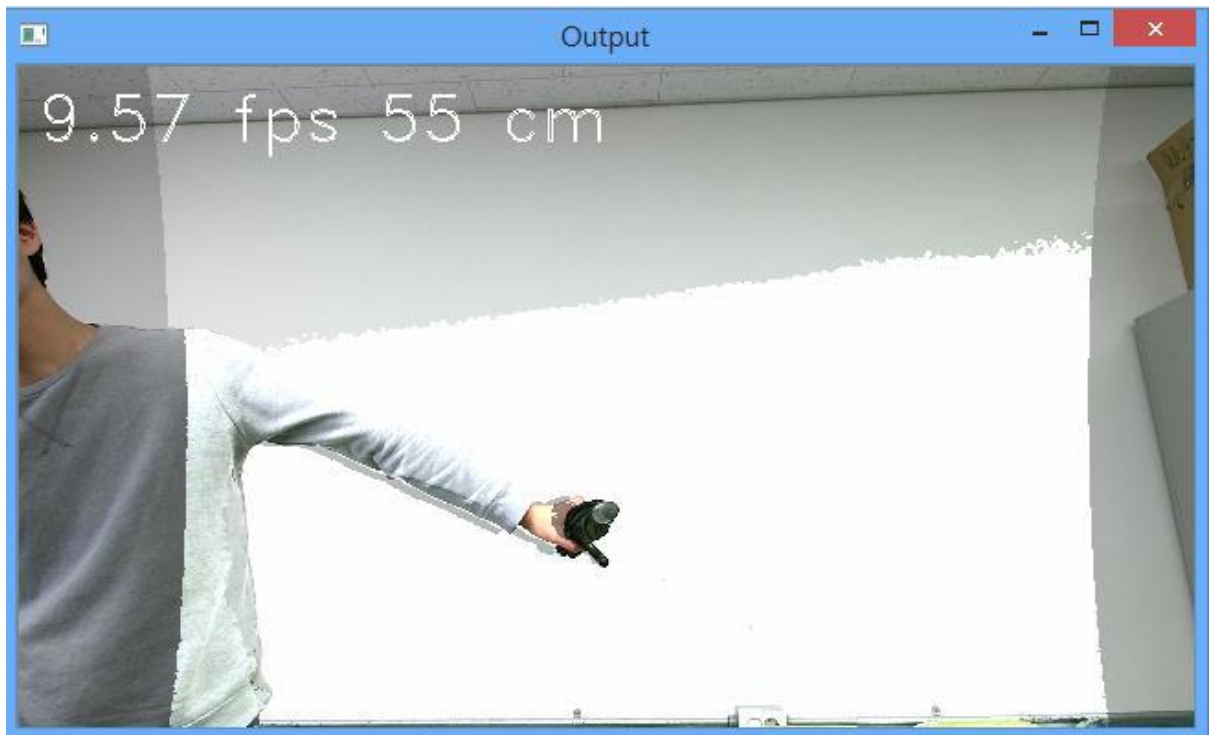



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

3) depth가 작아질수록 영상에 보이는 물체의 깜박거리는 속도를 빠르게 처리한 영상



4) 거리가 가까울수록 물체의 밝기가 밝아지게 처리한 영상



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2.2.2.4 자동차 제어

1. 자동차 핸들 인터페이스

기존의 키보드 디바이스의 input value으로 구동되었던 차량 제어 기능을 Gaming 레이싱 핸들 device의 input value 으로 대체하기 위하여, Windows의 Joystick API를 사용하였다. 하지만 우리가 구입한 핸들 device는 Joystick API를 지원받지 않는 하드웨어이기 때문에, USB에서 input value을 Raw Data로 읽어 들이고 이를 의미있게 Parsing하여 사용하였다. 이를 위해 Microsoft는 Raw Input API 지원하기 위해 Human Interface Devices(HIDs)라는 클래스와 API를 제공한다. 우리는 이러한 API들을 사용하여 USB에 연결되어 있는 핸들 device로부터 값을 Joystick API를 통해 읽어 들이는 것처럼 사용하였다.



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

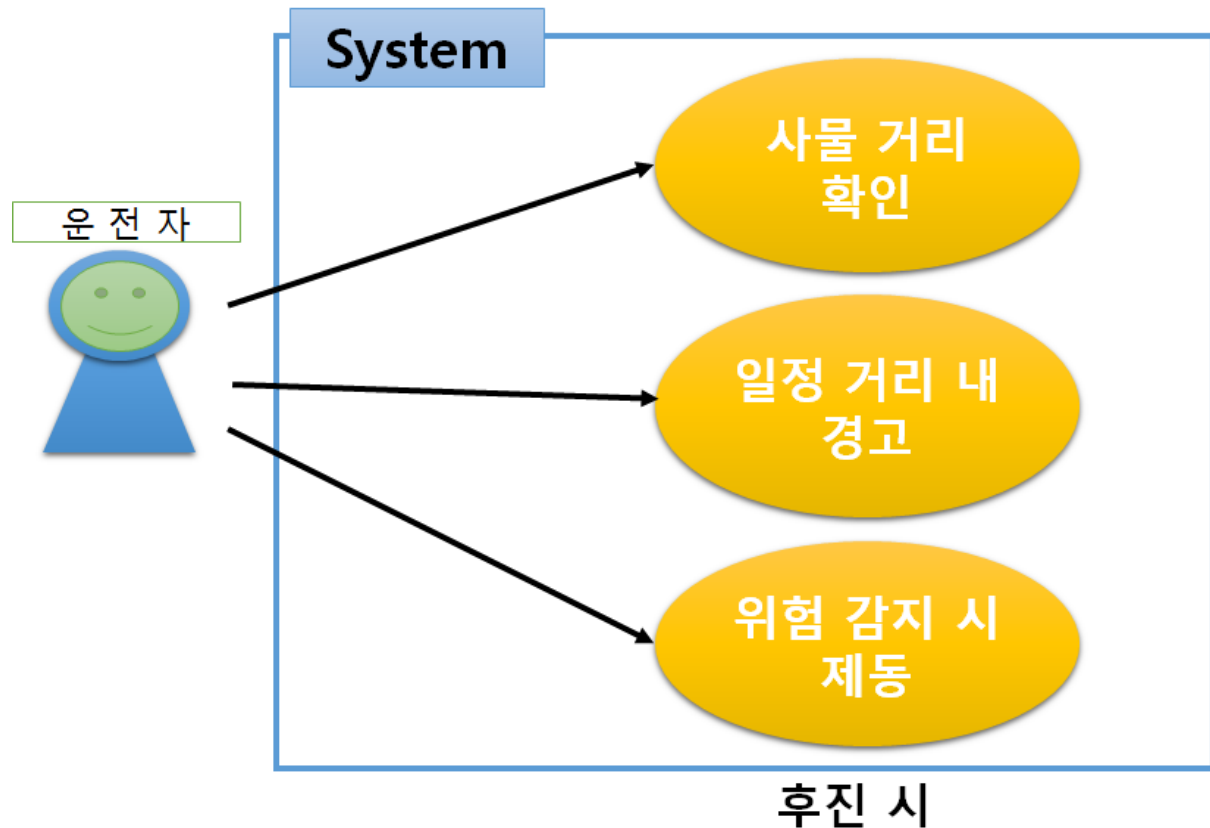
2. 블루투스 통신

레이싱 핸들로부터 읽어들이는 input value에 맞게 모형 차량을 제어해야 하는데, 모형차량의 Raspberry Pi와 레이싱 핸들이 USB를 통해 유선으로 연결되어 있을 수 없으므로, Wireless 통신을 통해 차량의 움직임을 조작하였다. 이를 위해 블루투스 네트워크를 이용하였다. 차량의 Raspberry Pi와 핸들 device가 연결되어 있는 PC에 각각 블루투스 dongle을 삽입하고, 이후 Raspberry Pi에는 linux에서의 블루투스 라이브러리인 Bluez를 이용하여 Client Side 프로그램을 작성하였고, PC에는 Windows에서의 블루투스 라이브러리인 ws2bt(Winsock to Bluetooth)를 사용하여 Server Side 프로그램을 작성하였다. 각각의 프로그램은 Socket 구조를 통해 데이터를 Read / Write하여 데이터를 주고받을 수 있게 된다.

3. 소켓통신을 통한 제어 신호 전송

Kinect 영상처리를 통하여 Kinect와 물체사이의 설정한 임계값(약 65cm~70cm) 이하로 거리가 좁혀질 경우 제어 신호를 Kinect를 구동시키는 PC에서 소켓을 통해 모형 자동차에 전송한다. 전송된 제어 신호는 모형 자동차에 제동을 걸어 자동차의 움직임을 정지상태로 만든다.


2.2.3 시스템 기능 요구사항



- 사물거리 확인 : 완료
- 일정거리 내 경고 : 완료
- 위험 감지 시 제동 : 완료

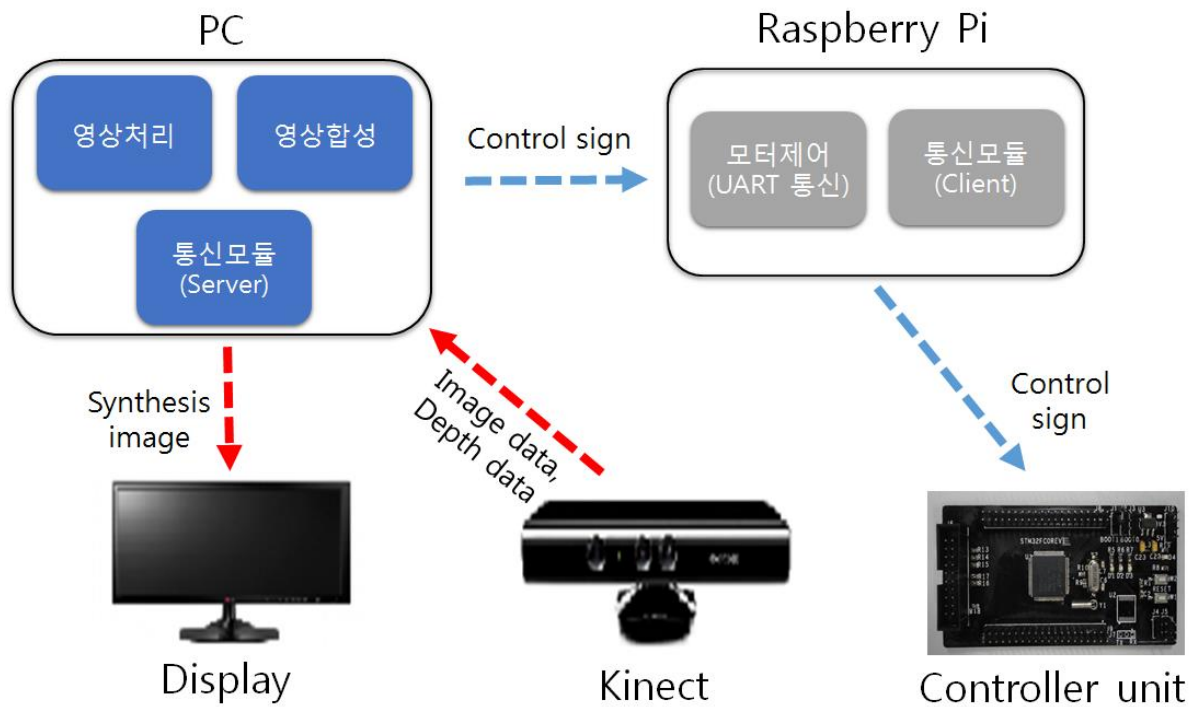
2.2.4 시스템 비기능(품질) 요구사항

- 성능 : 0.5초 이내에 운전자가 물체를 인식 할 수 있을 정도로 영상 합성을 실시간으로 처리한다. 카메라가 1초당 30 frame 정도 촬영할 수 있을만한 해상도를 지닌다.
- ➔ 달성 : 영상의 실시간 처리와 1초당 30 frame정도의 해상도를 지닐 수 있을 정도 출력된다.
- 정확성 : 지름 3CM 이하 정도의 가느다란 물체나 면적이 25CM² 정도의 작은 크기의 물체도 인식이 가능하게 해야 한다.
- ➔ 달성 : 지름 3cm 이하 정도의 우산 끝도 인식이 가능하다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

- 안정성 : 키넥트와 컴퓨터, 모형자동차 간의 통신이 언제든지 끊기지 않을 수 있도록 안정성 있는 통신구조를 갖추어야 한다.
- ➔ 최종 구현하여 동작시켜본 결과 영상뿐만 아니라 모형자동차의 작동과 제동까지 실시간으로 처리되며, 시험용으로 1시간 이상 켜 놓아도 통신이 끊어지지 않았다.

2.2.5 시스템 구조 및 설계도



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2.2.6 활용/개발된 기술

대분류	소분류	기능	형식	비고
파일	<i>Kinect image</i> 영상 저장, 열기	Kinect image 영상을 저장, 열기한다..	C++ 함수	
	<i>Kinect Depth</i> 영상 저장, 열기	Kinect Depth 영상을 저장, 열기한다	C++ 함수	
입력	<i>Kinect image, depth</i> 영상 information	Kinect로부터 image, depth 정보를 얻는다.	Module	
	<i>모형자동차</i> <i>Controller Unit</i>	차량제어	Module	
출력	<i>Display</i>	Image 와 depth 합성영상 출력	Scene	
	<i>모형자동차</i>	차량제어	Module	
알고리즘	<i>Depth interface</i>	Kinect 에서 얻은 depth 정보를 분석하여 사용자가 의도한 interface 를 알아낸다.	C++	
	<i>모형차량</i> <i>interface</i>	차량 제어		
모양	<i>글꼴</i>	정렬		
		폰트 바꾸기		

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2.2.7 현실적 제한 요소 및 그 해결 방안

- 하드웨어


1) Kinect

Depth map을 얻기 위한 카메라의 경우 외부에서 사용 가능한 depth camera는 그 비용이 예산을 초과하므로 차선책인 Kinect를 사용하여 실내를 기반으로 프로그래밍 하며 프로그램이 검증 될 경우 좋은 성능의 카메라와 호환 가능하게 구성한다.

현재까지 알려진 Kinect v1의 사양 중 depth 영상의 해상도는 320 * 240으로, 실제로 실험을 위한 영상을 촬영해 보았더니, 뚜렷하게 인식되지 않는 부분들이 많이 보였다. 특히 테두리 부분에서 그림자가 심하게 진다거나, 테두리가 명확히 드러나지 않는 부분들이 많이 보여서 확실한 depth 정보가 확보되지 않았다. 따라서 Kinect v2를 이용하여 실험을 해 보았는데, 해상도가 512 * 424으로 많이 향상되어서 좀 더 세밀한 부분까지 출력 가능한 것을 확인할 수 있었다.

Kinect v1의 단점을 보완하기 위해 Kinect v2를 사용하면 되지만, 해외 구매대행을 해야 하기 때문에 기기를 확보하는 데까지 시간이 많이 소요된다. 따라서 우선 v1을 통하여 개발을 진행하고, v2를 구매하게 된다면 기기를 바꾸어 카메라의 성능을 향상시킬 수 있도록 할 것이다.

-> 계획서의 내용을 반영하여 영상의 성능을 높이기 위해서 Kinect v1에서 Kinect v2로 교체하여 구현하였다. 그 결과 그 제한요소들을 해결할 수 있었다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2) 모형자동차

Depth map 을 구하기 위한 Kinect 의 요구전력이 모형 차량에 내장된 파워로는 부족하기 때문에 외부에서 전력을 공급해야 된다.

무선으로 차량을 구성할 경우 Raspberry Pi의 무선 네트워크를 통하여 영상을 pc로 전송해야 되는데 무선 네트워크의 경우 거리나 외부 변수에 따라 그 속도가 불안정하고 평균적으로 최대 지원 속도가 54mbps이며, 연결방식이 일반적인 n방식으로 이루어질 경우 속도는 20mbps 에 머문다.

Depth map 의 영상정보와 일반 이미지 영상의 경우 초당 23Mbyte의 용량을 가지고 있으며 이를 실시간으로 pc에 전송함과 동시에 pc에서 차량제어 신호를 보내야 한다. 무선 네트워크로 구성 하였을 경우 좀 더 유동적인 움직임이 가능하지만 정보의 상호교환과 실시간 처리에 있어서 전송속도 문제가 생길 여지가 있으므로 유선 랜 구성을 차선택으로 둔다. Kinect의 시야 각은 수평 57도 수직 43도로 좁은 편이나 시야를 위해 두 개의 Kinect를 모형 차량에 설치하기엔 어댑터의 수나 공간의 협소함 같은 물리적 한계가 있다.

-> 계획서의 내용을 반영하여 블루투스 모듈을 사용하여 개선하려고 하였지만, 실시간 처리가 쉽지 않아서, 인터넷으로 선을 꽂아서 유선 통신으로 해결하였다.

- 소프트웨어

1) Kinect

미흡할 수 있는 Kinect의 영상처리에 부분에 대해서 인터넷 검색 및 전문 서적을 활용하여 개선 방안에 관련된 알고리즘을 찾아봐야 한다. Image 영상과 depth 영상을 합성하였을 때, 오차가 많이 발생할 수 있으므로 이를 대비하여 자연스럽게 영상을 합성할 수 있는 프로그래밍 방안을 찾아보고 적용시켜야 한다. 또한 좀 더 정밀한 depth 영상을 출력하기 위해 무조건 하드웨어를 교체하기보다는 소프트웨어 부분에서 최대한 개선시킬 수 있을만한 방법을 알아보는 것이 중요하다.

-> depthToColor() 함수를 통해서 depth 영상을 image 영상에 맵핑하여 영상이 깨지거나 정밀하지 못했던 부분을 해결하였다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26

2.2.8 결과물 목록

수행 결과물	기술문서
하드웨어(Kinect, 모형차량, 차량핸들)	무
Kinect Calibration 알고리즘	무
차량핸들 Interface	무
차량 시연 도로	무
Kinect와 차량간의 통신 모듈	무


2.3 기대효과 및 활용방안

1. 기대효과

- 초음파센서가 감지하지 못하는 물체까지 인식하여 차량 손상을 방지할 수 있다.
- 일정거리 이내의 물체에 대해 차량을 제어하기 때문에 사고 발생 위험을 줄일 수 있다.
- 운전자의 시야 확대로 안전한 후방 주차가 가능하다.

2. 활용방안

- 실제 차량에도 지원하여 활용 가능하다.
- 차량의 후방뿐만 아니라 측면의 사각지대까지 개선할 수 있도록 활용 가능하다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26


3. 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	웹 페이지	http://darkpgmr.tistory.com/		2013, 2014		영상처리
2	서적	오픈소스 OpenCV 를 이용한 컴퓨터 비전 실무 프로그래밍	홍릉과학 출판사	2008	정성환	영상처리
3	서적	가초 알고리즘부터 라이브러리 활용까지 영상처리 프로그래밍	프리렉	2013	이희석	영상처리
4	웹 페이지	http://www.buildinsider.net/small/kinectv2cpp/03		2014		Kinect v2
5	웹 페이지	https://www.microsoft.com/en-us/kinectforwindows/meetkinect/features.aspx				Kinect v2
6	웹 페이지	http://www.codeproject.com/Articles/758344/Background-removal-using-Kinect-green-screen-effect		2014		Kinect v2
7	웹 페이지	http://www.codeproject.com/Articles/185522/Using-the-Raw-Input-API-to-Process-Joystick-Input		2011		차량핸들
8	웹 페이지	http://wiringpi.com/		2015		차량제어

4. 부록

<사용자 매뉴얼>

- 차량에 Kinect를 탑재하고 Kinect 전원을 켜다.
- 차량을 후방진입 하도록 제어한다.
- 차량의 후방을 보여주는 Display를 통해 차량과 물체간의 거리 정보를 확인한다.
(설계된 4가지 모드를 통해 확인)
- 후방물체가 지정된 거리까지 접근하면 Kinect와 차량간의 통신을 통해 후방진입을 차단한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	Depth map을 활용한 후방카메라의 한계 개선	
	팀 명	천 리 안	
	Confidential Restricted	Version 1.4	2015-05-26