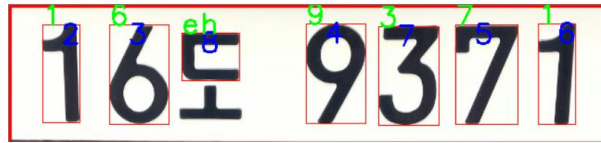


Machine Vision Projects (1)

- 템플릿 매칭과 라벨링 기법을 이용한 번호판 숫자 및 글자 해독 (연두색 글자)
 - ✓ 한 덩어리의 물체를 각각의 객체로 나누어 정보를 파악하는 역량을 기를 수 있었습니다.



- 허프 트랜스폼과 소실점을 이용한 라인 트래킹 및 신호 파악 알고리즘 구상
 - ✓ 가변적인 영상 환경에서도 지속해서 값을 유지하는 방법, fps를 높이기 위한 효율적인 알고리즘을 구상하는 방법을 모색할 수 있었습니다.



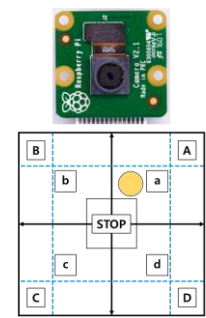
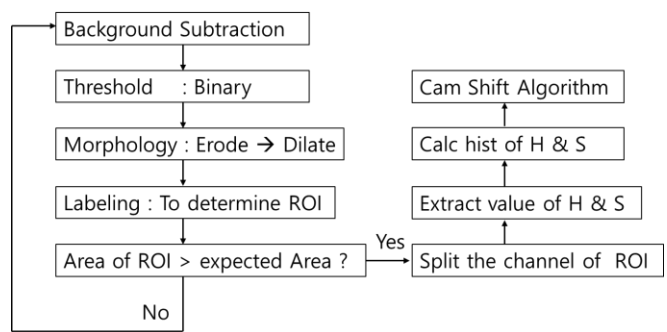
<라인 트래킹 및 소실점 파악>



<신호등 신호 분석 및 알람 시스템>

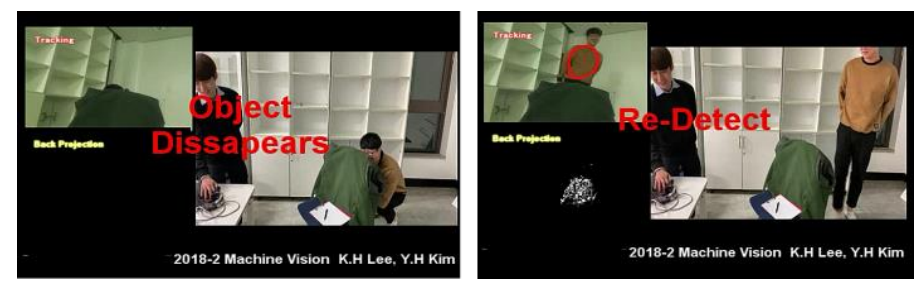
Machine Vision Projects (2)

- 캠 시프트 기법과 서보모터 제어를 이용한 물체 추적 (스마트 CCTV 구현)
 - ✓ MCU를 이용하여 비전 센서를 통해 HW를 제어하는 알고리즘을 구현 할 수 있었고, 색 정보만으로는 물체 추적에 한계가 있다고 느껴, 딥러닝 적용이 필수적임을 느꼈습니다.



Sending Data

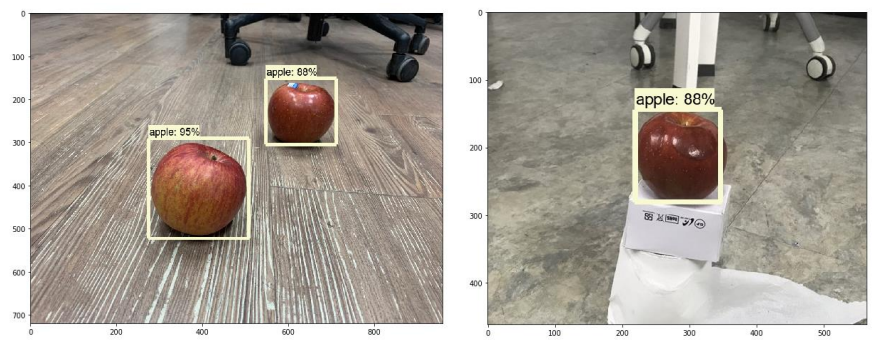
		x Servo	y Servo
A	a	++	++
B	b	--	++
C	c	--	--
D	d	++	--



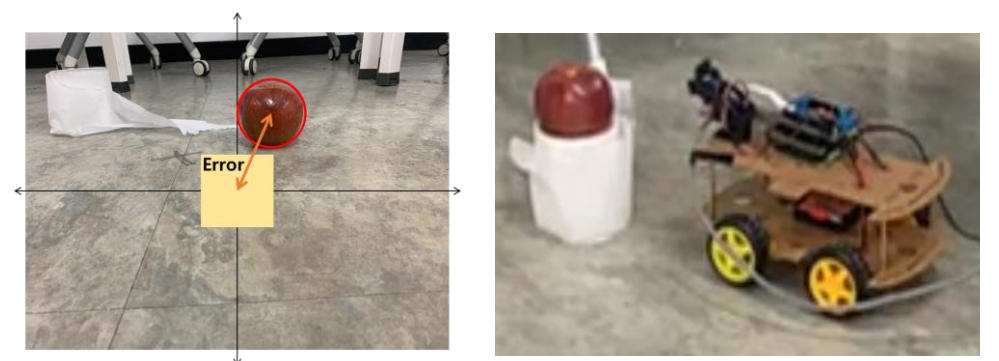
<Cam-shift/HW 제어 알고리즘>

<스마트 CCTV 물체 추적>

- 딥러닝-Fast RCNN 기법과 Cam shift 기법을 이용한 스마트팜 추적로봇 구현
 - ✓ 딥러닝을 적용해볼 수 있었고, 부족한 영상데이터를 보완할 수 있었고, HW 제어를 병행할 수 있었습니다.



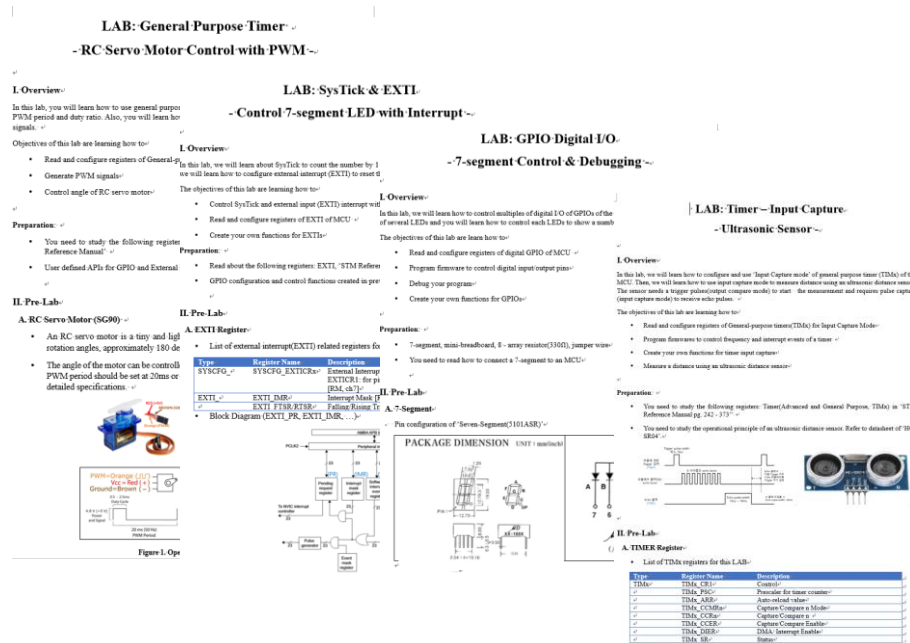
<Fast RCNN을 통한 물체 파악>



<중앙과의 오차를 줄이는 방향으로 물체에 접근하는 모형 자동차>

Embedded Controller

- Atmega128(3학년 1학기) 학습, stm32f411re(현재) 수업 진행 및 수업 자료 제작
 - ✓ 학습 내용: GPIO(LED 제어), A/D converting(가변저항), USART(블루투스), Timer(초음파 센서), PWM(DC 모터), Interrupt(스위치 제어), I2C(자이로 센서), RTOS(태스크 매니지먼트)
 - Atmega128 파이널 프로젝트 : 초음파 센서와 스테핑 모터를 이용한 자율주행자동차 구현
 - Stm32F4RE 파이널 프로젝트 : 라인 센서와 초음파 센서, DC 모터를 이용한 자율주차자동차 구현 중



myEXTI.c	2019-10-01 오후 10:47	C 소스
myEXTI.h	2019-10-01 오후 10:14	C/C++ 헤더
myGPIO.c	2019-09-25 오후 2:00	C 소스
myGPIO.h	2019-10-01 오후 10:09	C/C++ 헤더
myPWM.c	2019-09-22 오후 2:39	C 소스
myPWM.h	2019-09-22 오후 2:39	C/C++ 헤더
myRCC.c	2019-09-23 오후 1:07	C 소스
myRCC.h	2019-09-23 오후 1:09	C/C++ 헤더
mySysTick.c	2019-10-01 오후 5:13	C 소스
mySysTick.h	2019-10-01 오후 10:06	C/C++ 헤더
myTIM.c	2019-09-22 오후 2:55	C 소스
myTIM.h	2019-09-22 오후 2:55	C/C++ 헤더
myUART.c	2019-08-23 오전 12:17	C 소스
myUART.h	2019-07-02 오후 1:28	C/C++ 헤더

<조교로서 제작한 Embedded Controller 수업 자료들>

<개발 중인 stm32f411re의 BSW Library>

Post Capstone

- ADAS 카메라 및 라이다 장착 검사를 위한 센서 캘리브레이션 알고리즘 시뮬레이션 연구.
(자동차공학회 학술대회 제출)

I. Introduction

연구 배경 및 목표

- 카메라와 라이다 센서는 ADAS의 핵심부품으로, 센서 융합 기술에 관한 연구 개발이 활발히 진행되고 있음
- 차량에 탑재된 다양한 센서들의 정보를 융합하여 환경을 인식하기 위해 센서 간 캘리브레이션 과정은 필수적임
- 본 연구에서는 큐브 형태의 타겟을 사용하여 라이다와 카메라 간의 캘리브레이션 알고리즘을 시뮬레이션 상에서 제안하고자 함



기존 연구 대비 차별성

- 일반적인 planar calibration은 라이다 해상도에 따라 평면의 모서리 검출이 달라지므로 변환 행렬 도출에 큰 오차가 발생
- 본 연구에서는 평면이 아닌 큐브를 타겟으로 사용함으로써 각 면의 수직성을 이용하여 특징점 추정 정확도 향상^[1]



Fig1. 다이아몬드 평면^[2]



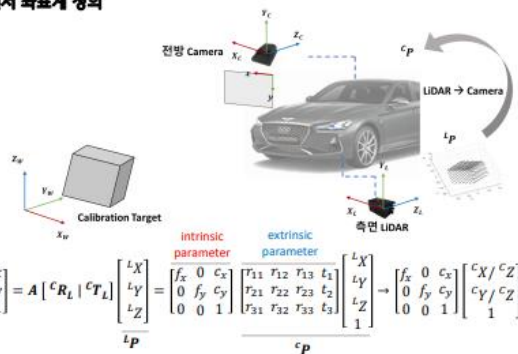
Fig2. 체커 보드^[3]



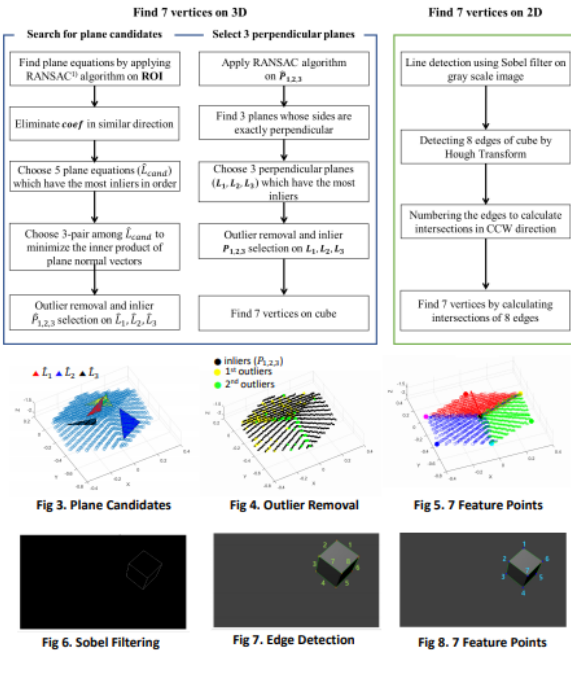
Fig3. 다양한 캘리브레이션 평면^[4]

II. Algorithm

센서 좌표계 정의



알고리즘 흐름도



EPnP^[6]를 이용한 라이다 - 카메라 상대위치 추정

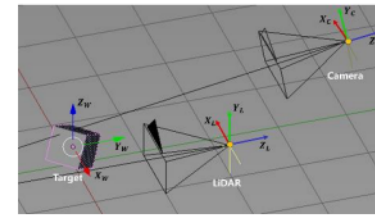
- Focal length와 이미지 크기로부터 카메라 내부 변수 행렬 A 추출

$$focal\ length[px] = \frac{focal\ length[mm]}{sensor\ width[mm]} \times image\ width[px] = 1050$$

$$center\ point\ c_x = \frac{image\ width}{2} = 480, c_y = \frac{image\ height}{2} = 270$$
- EPnP 알고리즘에 특징점 7개의 라이다 좌표와 카메라 좌표 대입 후 변환 행렬 c_{R_L}, c_{T_L} 도출
- c_{R_L}, c_{T_L} 로부터 라이다 - 카메라의 상대위치 계산

III. Simulation Result

시뮬레이션 환경 구성



Sensor Specification	
LIDAR (Velodyne HDL 32E)	
Scan resolution	0.17 [deg]
Noise mean	0.00 [m]
Noise σ	0.02 [m]
Camera (RGB)	
Sensor size	32 x 32 [mm]
Resolution	176 x 144 [px]
Focal Length	35 [mm]

시뮬레이션 결과

- 타겟: 50 x 50 x 50[cm] 정육면체 (라이다로부터 2m 거리에 위치)
- 시뮬레이션 프로그램: Blender
- 10개 스캔 프레임에 대해 라이다 - 카메라 간의 상대 위치 평균을 산출함

Simulation Result	Rotation[deg]			Translation[m]		
	roll	pitch	yaw	dx	dy	dz
Ground Truth	-15.00	0.00	0.00	-0.90	0.60	2.00
Estimation	-15.61	-0.01	-0.76	-0.89	0.66	1.98
Error	0.61	0.01	0.76	0.01	0.06	0.02

- 라이다 포인트 클라우드와 카메라 이미지 매핑

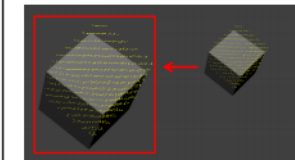
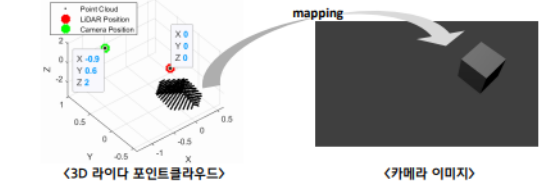


Fig 9. Result by other method^[3]

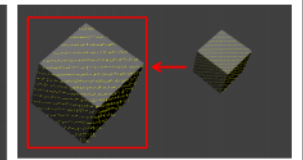


Fig 10. Result by proposed method

결론

- 본 연구의 알고리즘으로 라이다-카메라 상대위치를 추정한 결과, 시뮬레이션 상에서 최대 0.76[deg], 0.06[m]의 오차가 발생함을 확인함
- 추후 실험환경을 구축하여 실제 카메라와 라이다 장비를 통해 본 연구의 알고리즘을 검증할 것임