



국민대학교
소프트웨어융합대학
컴퓨터공학부

창업연계공학설계입문 AD 프로젝트

과제명	색깔 인식을 이용한 xycar 자율주행
소속	국민대학교 소프트웨어학부
학번	20143050, 20191661
성명	이 진, 이진호, 조규현, 전하은, 신영환
문서 제목	창업연계공학설계입문 AD 프로젝트
교수	이시윤

Version	1.3
Date	2019-12-18

 국민대학교 컴퓨터공학부 소프트웨어프로젝트I			
	과제명	색깔 인식을 이용한 xycar 자율주행	
	분반/조	5분반/4조	
	Confidential Restricted	Version 1.3	2019-December-18

목 차

1	개요	3
1.1	프로젝트 개요	3
1.2	추진 배경 및 필요성	3
2	개발 내용 및 결과물	4
2.1	목표	4
2.2	연구/개발 내용 및 결과물	4
2.2.1	연구/개발 내용	4
2.2.2	활용/개발된 기술의 알고리즘	7
2.2.3	현실적 제한 요소 및 그 해결 방안	9
2.2.4	결과물	9
2.3	기대효과 및 활용방안	12
3	참고 문헌	12

 국민대학교 컴퓨터공학부 소프트웨어프로젝트I		
	과제명	색깔 인식을 이용한 xycar 자율주행
	분반/조	5분반/4조
	Confidential Restricted	Version 1.3 2019-December-18

1 개요

1.1 프로젝트 개요

자율 주행 자동차는 운전자 또는 승객의 조작 없이 자동차 스스로 운행이 가능한 자동차를 말한다. 현재 자율주행 자동차는 상용화를 목표로 구글을 비롯한 여러 기업들이 자율주행자동차, 즉 무인 자동차의 개발에 뛰어들고 있으며 업계는 2022~2023년 미국 공학회 기준, 부분 자율주행인 레벨 3 자율주행 자동차가 상용화 될 것이라 될 것으로 보고 있다. 본 프로젝트에서는 이러한 자율 주행 자동차를 구현하기 위해 창업연계 공학설계입문에서 xycar 차선 인식 자율 주행 방식이 아닌 다른 방식으로 자율주행 자동차를 구현해보고자 하였고 구현 방식에 있어 수업시간에 접해보고 학습하였으나 실제 차선인식 자율주행을 하는 동안 적용해보지 못했던 RGB,HSV를 선택하게 되었다.

1.2 추진 배경 및 필요성

<그림 1>과 같이 고정적 형태로 수행하는 프로젝트가 아닌 창의성을 요구하는 AD 프로젝트의 중요성이 대두되고 있다. AD 프로젝트를 수행하면서 문제해결 능력 함양 및 학습 진로 설계를 기대할 수 있으며 본 프로젝트는 opencv, 영상처리기술 및 특정 색깔을 인식하는 방법, findcontour를 이용한 이미지 컨투어를 심화적으로 학습하고 이를 통한 자율주행 코스를 제작하며 자율주행을 체험할 수 있다.

□ 주요 추진 내용

- (교육과정 분야) 기존 정규 교육과정의 개편 등 교육과정 혁신 및 혁신적 교육과정의 개방을 통한 공유 학습 체제 확산
 - 미래 인재 핵심역량 배양을 위한 4차 산업혁명 기초교과 및 전문교과, **Adventure Design 도입** 등 혁신적 교육과정 운영
 - 전공 학점 인정 등 혁신적 교육과정의 실질적 개방·공유*를 통해 학생의 필요와 의지에 따른 자발적·선택적 융합 학습 기회 확대
 - * 사업 참여 학과 간 4차 산업혁명 기초·심화교과 공유 및 전공 학점인정 의무화

주요 개념 정리

- ▶ (4차산업혁명 기초 교과) 4차 산업혁명 유망 분야 기술과 비즈니스 모델에 관한 이해 및 미래 인재 핵심역량(관련 : [붙임2]) 강화를 위한 일반적 교육과정 (대학 전체 대상)
- ▶ (4차산업혁명 전문 교과) 대학이 제시한 4차 산업혁명 특화분야의 산업선도형 전문 인력 양성을 위한 전공 심화교육과정 (사업참여학과 대상)
- ▶ (Adventure Design) 문제해결능력 함양 및 학습진로 설계를 위한 1, 2학년 학생 대상 자기주도형 프로젝트 교과 (사업참여학과 대상)

<그림 1> 혁신적 교육과정 운영

출처= 소프트웨어 프로젝트 2 PPT

 국민대학교 컴퓨터공학부 소프트웨어프로젝트I			
	과제명	색깔 인식을 이용한 xycar 자율주행	
	분반/조	5분반/4조	
	Confidential Restricted	Version 1.3	2019-December-18

2 개발 내용 및 결과물

2.1 목표

xycar 가 빨간색을 따라서 주행할 수 있도록 autodrive 를 구현하며 이때 빨간색을 효과적으로 인식하고 구분할 수 있도록 findcontour 를 활용하여 취득한 영상에서 빨간색을 구분한다. 빨간색을 구분하는 코드에서는 구분한 빨간색 주위로 원을 그리며, autodrive 에서 원의 반지름의 길이에 따라 주행 속도를 다르게 하여 xycar 가 물체에 부딪히지 않도록 한다.

2.2 연구/개발 내용 및 결과물

2.2.1 연구/개발 내용

1.autodrive3.py

Autodrive3에서는 xycar가 주행하기 위해 필요한 여러 가지 모듈들을 불러오고 연결하여 실제로 차가 구동할 수 있도록 한다. 코드는 다음과 같다.

```
#!/usr/bin/env python
import rospy, time, cv2, rospkg, numpy as np
from std_msgs.msg import Int32MultiArray
from motordriver import MotorDriver
from colorDectector import ColorDetector
import time
```

Class AutoDrive에서는 ColorDetector와 MotorDriver를 생성하였으며, 이를 활용하여 trace()에서 영상에 원의 중심점이 존재할 때, 원의 반지름을 기준으로 거리가 가까워질 때마다 속도와 각도의 값을 변경한다.

```
class AutoDrive:
    def __init__(self):
        rospy.init_node('xycar_driver')
        self.color_detector = ColorDetector('/usb_cam/image_raw')
```



```
self.driver = MotorDriver('/xycar_motor_msg')

def trace(self):
    cir_x, cir_r = self.color_detector.run()
    if (cir_x != None):
        if cir_r < 40:
            angle = 90 + float(50) / 320 * (cir_x - 320)
            speed = 125
        else:
            angle = 90
            speed = 90
    else:
        angle = 90
        speed = 90
    self.driver.drive(angle, speed)
```

아래의 코드에서는 메인 함수를 구현하고 있다.

```
def exit(self):
    print('finished')
if __name__ == '__main__':
    car = AutoDrive()
    time.sleep(3)
    rate = rospy.Rate(15)
    while not rospy.is_shutdown():
        car.trace()
        rate.sleep()
    rospy.on_shutdown(car.exit)
```

2.colordetector.py

Colordetector.py에서는 opencv의 여러 라이브러리를 사용하여 빨간색을 검출해낸다. 코드는 다음과 같다.



Class ColorDetector 에서는 빨간색을 검출하기 위해 hsv 의 범위를 정해주고 있으며 ROI 영역을 설정해주고 있다.

```
import rospy
import cv2
import numpy as np
from sensor_msgs.msg import Image
from cv_bridge import CvBridge

class ColorDetector:
    def __init__(self, topic):
        self.lower_red = np.array([120, 100, 0])
        self.upper_red = np.array([255, 255, 255])
        self.cam_img = np.zeros(shape=(480, 640, 3), dtype=np.uint8)
        self.mask = np.zeros(shape=(480, 640),
                                dtype=np.uint8)
        self.edge = np.zeros(shape=(480, 640),
                                dtype=np.uint8)
        self.bridge = CvBridge()
        self.roi_vertical_pos = 0
        self.scan_height = 0
        self.num = [[], []]
        self.idx = 0
        rospy.Subscriber(topic, Image, self.conv_image)
```

run method 에서는 받아온 이미지를 필터링한 후에 inrange 를 통해 이미지에 mask 를 한 후 영상에 원을 그려 반지름과 중심을 반환한다. 이때 원을 하나도 발견하지 못한다면 반지름과 중심에 None 을 넣고 반환한다.

```
def run (self):
    img_color = self.cam_img
    height, width = img_color.shape[:2]
    img_blurred = cv2.GaussianBlur(img_color, (7,7), 0)
    img_hsv = cv2.cvtColor(img_blurred, cv2.COLOR_BGR2HSV)
```



과제명	색깔 인식을 이용한 xycar 자율주행	
분반/조	5분반/4조	
Confidential Restricted	Version 1.3	2019-December-18

```

mask = cv2.inRange(img_hsv, self.lower_red, self.upper_red)
kernel = np.ones((5, 5), np.uint8)
mask = cv2.morphologyEx(mask, cv2.MORPH_OPEN, kernel)
mask = cv2.morphologyEx(mask, cv2.MORPH_CLOSE, kernel)
cnts = cv2.findContours(mask.copy(), cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)[-2]
center = None

if len(cnts)>0 :
    c = max(cnts, key=cv2.contourArea)
    ((x, y), radius) = cv2.minEnclosingCircle(c)
    M = cv2.moments(c)
    center = (int(M["m10"] / M["m00"]), int(M["m01"] / M["m00"]))
    x = center[0]
    cv2.circle(img_color, (int(x), int(y)), int(radius), (255, 0, 0), 2)
    print(int(x), int(y), int(radius))
    return int(x), int(radius)

else:
    x = None
    radius = None
    return x, radius

def conv_image(self,data):
    self.cam_img = self.bridge.imgmsg_to_cv2(data, 'bgr8')
def detect_lines(self):
    return -1, -1

```

2.2.2 활용/개발된 기술의 알고리즘

1.opencv

OpenCV(Open Source Computer Vision)은 실시간 컴퓨터 비전을 목적으로 한 프로그래밍 라이브러리이며 원래는 인텔이 개발하였다. 실시간 이미지 프로세싱에 중점을 둔 라이브러리이다. 인텔 CPU에서 사용되는 경우 속도의 향상을 볼 수 있는 IPP(Intel Performance Primitives)를 지원한다. 이 라이브러리는 윈도우, 리눅스 등에서 사용 가능한 크로스 플랫폼이며 오픈소스 BSD 허가서 하에서 무료로 사용할 수 있다. OpenCV는 Tensorflow, Torch / Pytorch 및 Caffe의 딥러닝 프레임워크

 국민대학교 컴퓨터공학부 소프트웨어프로젝트I			
	과제명	색깔 인식을 이용한 xycar 자율주행	
	분반/조	5분반/4조	
	Confidential Restricted	Version 1.3	2019-December-18

크를 지원한다. 본프로젝트에서는 이를 이용해서 카메라로 취득한 영상을 가공, 분석해서 xycar 자율주행에 적용하였다.



2. cv2.findcontour

contour란 동일한 색 또는 동일한 픽셀 값(강도,intensity)을 가지고 있는 영역의 경계선 정보다. 물체의 윤곽선, 외형을 파악하는데 사용된다. OpenCV의 findContours 함수는 이미지의 컨투어 정보, 컨투어의 상하구조(hierachy) 정보를 출력하며 흑백 이미지 또는 이진화된 이미지만 적용할 수 있다.

images, contours, hierachy = cv2.findContours(image, mode, method) 와 같이 사용하며 이때 image는 흑백이미지 또는 이진화된 이미지, mode는 컨투어를 찾는 방법, method는 컨투어를 찾을 때 사용하는 근사화 방법이다.

3.cv2.inrange

inrange 함수는 image에 있는 색상 중 lower이상이고 upper이하인 색상에 대한 마스크를 만들어 mask에 저장해서 돌려준다. 이때 RGB 세 개를 한꺼번에 비교하는 것이 아니라 R이 범위에 들어가는지, G가 범위에 들어 가는지, B가 범위에 들어 가는지 각각 따로 비교해서 모두 범위에 들어 가는 것만 mask 대상이 된다.

4. cv2.minEnclosingCircle(c)

주어진 카운터에 외접하는 원을 얻기 위해 사용하며 리턴 값은 원의 중심 좌표와 반지름이다. 그렇게 추출한 원의 중심 좌표와 반지름을 기반으로 파란색 원을 화면 위에 그리며 영상을 프레임 단위로 받아들인 후, GaussianBlur로 블러 처리하고 hsv처리해서 hsv화면에서 빨간색은 하얀색으로 나머지 색깔은 검은색으로 이진화하고 morphologyEx를 이용한 opening,closing 변형 방법을 사용한다. 이 방법은 노이즈를 제거하는 데 가장 효과적이다.

5. cv2.morphologyEx()

이 함수는 erosion과 dilation 알고리즘을 이용해서 보다 향상된 이미지 변형 기능을 제공한다. 함수의 인자 3가지인데, 첫번째 인자에는 이미지 변형을 적용할 원본 opening은 erosion을 수행하

 국민대학교 컴퓨터공학부 소프트웨어프로젝트I			
	과제명	색깔 인식을 이용한 xycar 자율주행	
	분반/조	5분반/4조	
	Confidential Restricted	Version 1.3	2019-December-18

고 dialation 수행해서 본래 이미지 크기로 돌려놓는 것이다. losing 기법은 dialation 수행을 한 후, 바로 erosion수행을 하여 본래 이미지 크기로 돌려놓는 것을 지칭한다.

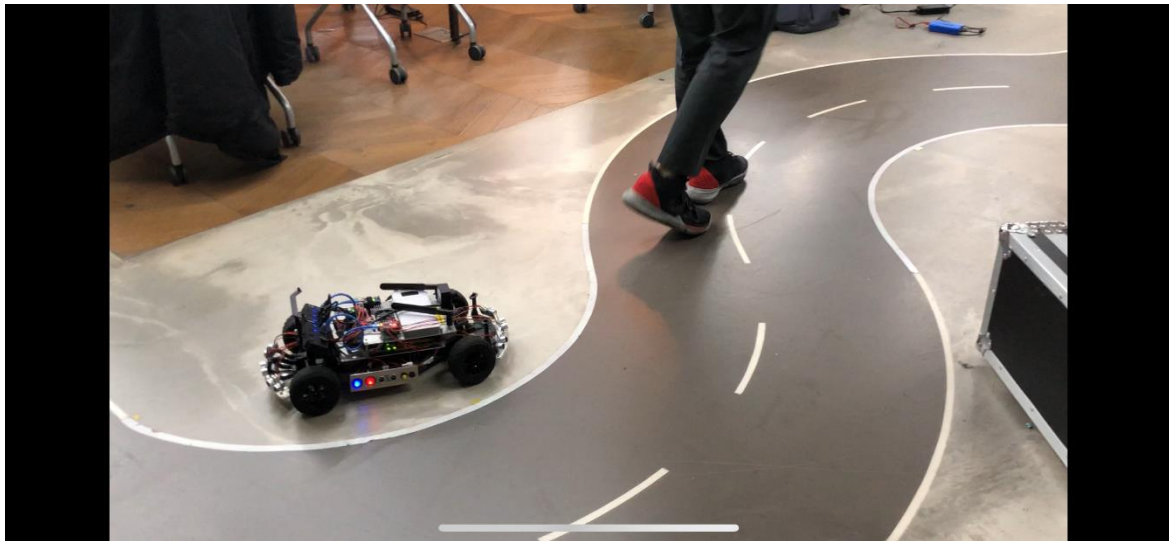
2.2.3 현실적 제한 요소 및 그 해결 방안

빨간색을 따라서 주행할 때 물체가 입체적인 물체가 아니라 종이 위에 인쇄되어 있으면 직선에서는 주행에 문제가 발생하지 않으나 곡선 코스를 주행할 때 빨간색을 인식하지 못하는 문제가 발생한다. 이 문제를 해결하기 위해 xycar가 주행하면서 효과적으로 빨간색을 인식할 수 있도록 종이를 배치하였다.

2.2.4 결과물

다음의 사진들은 위의 코드를 사용하여 다양한 상황에서 빨간색 물체를 효과적으로 검출하고 물체를 따라가는지를 알아본 여러 상황에서의 결과물이다.

1. 움직이는 빨간색 물체를 따라 주행하기



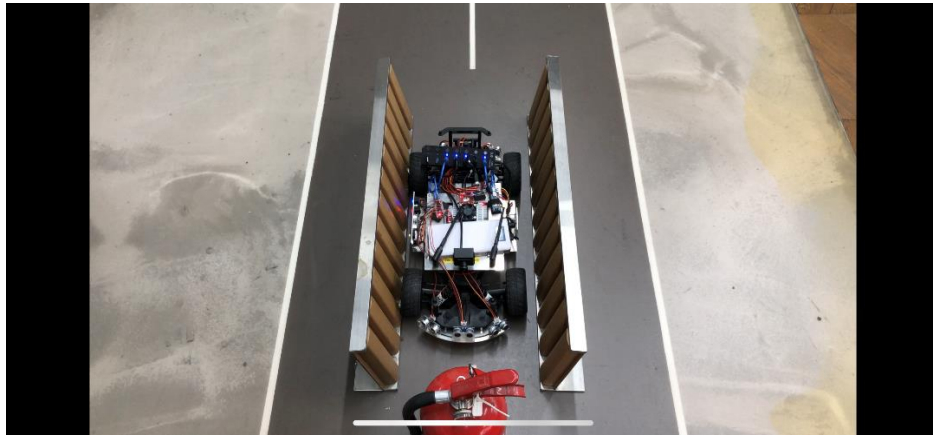
 국민대학교 컴퓨터공학부 소프트웨어프로젝트I			
	과제명	색깔 인식을 이용한 xycar 자율주행	
	분반/조	5분반/4조	
	Confidential Restricted	Version 1.3	2019-December-18

2. 빨간색 물체를 보고 주행하며 일정 거리에 정지하기

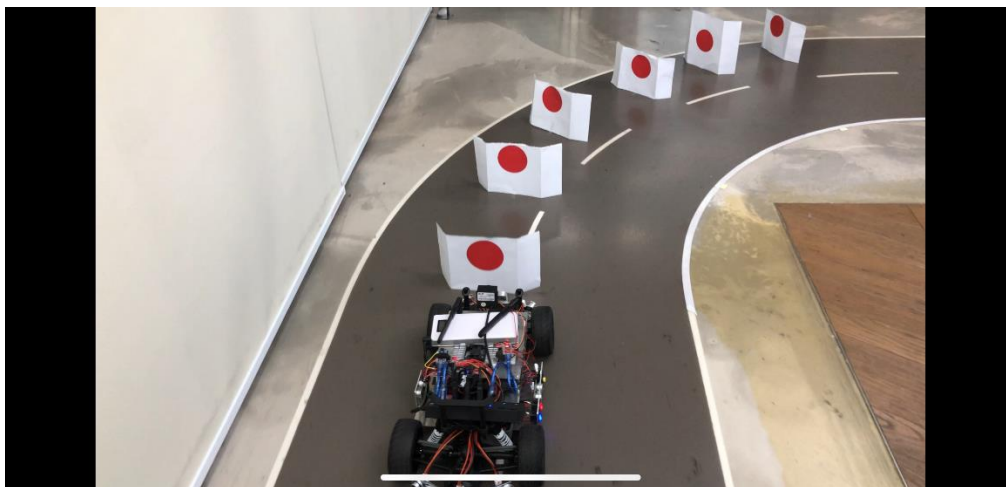
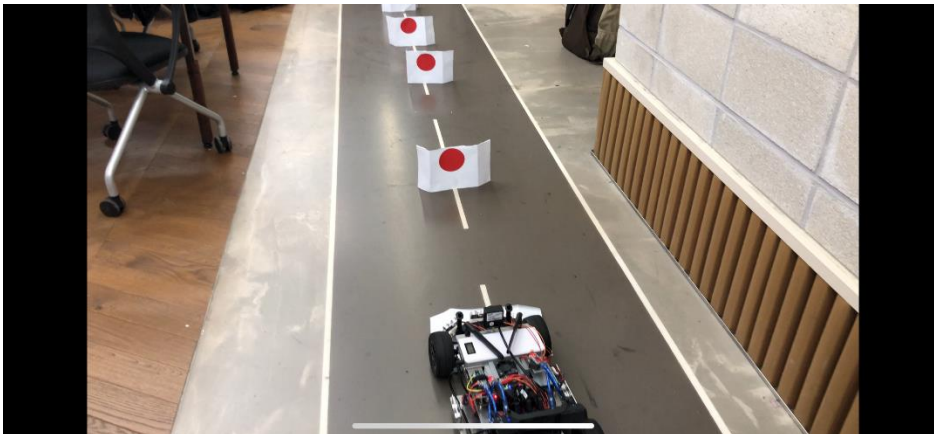




과제명	색깔 인식을 이용한 xycar 자율주행	
분반/조	5분반/4조	
Confidential Restricted	Version 1.3	2019-December-18



3. 빨간색 원을 따라가며 코스 주행하기



 국민대학교 컴퓨터공학부 소프트웨어프로젝트I			
	과제명	색깔 인식을 이용한 xycar 자율주행	
	분반/조	5분반/4조	
	Confidential Restricted	Version 1.3	2019-December-18

2.3 기대효과 및 활용방안

본프로젝트는 한학기 동안 진행한 차선인식 자율주행을 다른 방식으로 접근하는데 초점을 맞추어 색깔을 인식하여 자율주행이 가능하도록 하였다. 이와 같은 프로젝트를 통해 opencv와 영상처리에 대해 보다 명확하게 알 수 있으며 rgb,hsv를 활용하여 특정 색깔을 검출해내고 원하는 값을 얻어내는 것을 보다 심화적으로 학습할 수 있다. 또한 자율 주행을 여러 방면에서 생각해 봄으로써 차선 인식을 이용한 자율주행 뿐만 아니라 다양한 방식으로 자율주행에 접근해 볼 수 있다.

3 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	Web	자율주행자동차의 정의	wikipedia			
2	Web	자율주행 기술 지금은 어디일까	HMG 저널			
3	Web	Opencv	wikipedia			
4	Web					
5	Web					
6	Web					
7	Web					
8	Web					
9	Web					