

1. 작성요령

1) 참가신청서

- 한이음 사이트 로그인 후 공모전(한이음 공모전) 선택 또는 팀블로그 보고서에서 공모전 개발 보고서 작성 선택하여 공모전 참가신청서 페이지로 이동
- 프로젝트명 영문, 작품명, 작품소개, 타대회 참가 신청 및 수상여부 등 작성 및 개인정보 수집·이용 동의 후 개발보고서(Hwp), 제작설계서(PPT) 업로드
- * 프로젝트명은 국문을 영작 하여 영문으로 기재해야 함
- 작품명은 "알람몬", "카카오톡"과 같이 작품의 특성을 반영한 이름으로 작명하여 기재
- 작품소개는 2, 3줄 정도로 요약하여 어떤 기능을 하는 작품인지 명확히 기재
- 프로젝트 명 변경을 원하는 경우 팀블로그 - 신청서 - 수행계획서 변경을 통해 심의를 통과한 경우에만 변경 가능(프로젝트 수행변경 신청 기간은 실습장비 신청 기간과 동일)

2) 개발보고서(Hwp)

- 개발보고서에 표지에 프로젝트명(국문/영문), 작품명 기재
- 개발보고서 본문 내용 작성 후 동일 항목의 핵심내용을 요약본에 요약 기재
- 작품내용이 작성항목 및 세부항목별로 충실하게 반영 될 수 있도록 작성
- 각 항목별로 제시된 평가항목을 고려하여 작성
- 사진은 용량을 최대한 축소하여 삽입하되 평가가능한 해상도 유지
- 세부 작성요령은 개발보고서 상의 작성요령을 참고하여 작성
- 신청접수 시 개발이 완료되지 않은 프로젝트는 최종목표를 기준으로 작성하되 현재까지의 진도 및 추후 개발계획이 명확히 드러나도록 작성

3) SW개발 및 HW제작 설계서(PPT)

- 첨부된 PPT의 양식에 맞추어 설계서 작성
- 슬라이드 2페이지 '수행 단계별 주요 산출물' 내용 숙지 후 권장, 선택사항에 맞는 페이지 작성

3) 공통적용사항

- 문서 전체의 작성 수준으로 수행능력(문서완성도) 평가
- 모든 한글 문서 글씨크기 **제목은 14, 내용은 12, 표는 10** 으로 통일(글꼴 변경 금지)
- 블라인드 평가로 모든 서류(개발보고서hwp, SW개발/HW제작 설계서ppt)내 팀명, 팀원명 등 기재 금지

2. 제출요령

1) 접수방법

- **2020. 9. 15(화) 오후 1시까지**, 한이음 사이트(www.hanium.or.kr)에서 접수(팀블로그 → 보고서)
- * 접수기한 경과 시 시스템이 자동 종료되어 접수가 불가하며, 임박하여 접수 시 오류 수정이 어려우므로 가급적 1, 2일전 접수완료 바람
- 해당 프로젝트의 멘티(팀장)만 접수 가능

2) 분량

- 분량이 명시된 부분을 제외하고는 분량 제한이 없음

3. 개조식 작성예시 * 항목별 핵심내용만 작성

1) Best Case

본 문

I. 작품 개요

※ 평가항목 : 기획력 (필요성, 차별성)

1. 작품 소개

○ Oscilloscope

- Oscilloscope란 Analog 신호를 눈으로 볼 수 있도록 표시해 주는 장치임
- Atmega나 ARM board를 이용한 Embedded 프로젝트 진행 시 하드웨어 부분의 문제점을 파악하는데 사용됨

(MCU에 Clock이 제대로 들어가는지, PWM 파형이 정상적으로 출력되고 있는지 등)



< 그림1. Oscilloscope >

○ 기획 의도

- 상용 Oscilloscope의 경우, 가격이 비싸고 휴대성이 없어 학생들이 사용하기에 불편함
- 저가의 소형 Oscilloscope는 낮은 주파수(수십khz 정도)만 측정이 가능한 경우가 대부분임
- 일반적인 하드웨어 프로젝트 진행에 무리가 없는 SPEC의 휴대용 Oscilloscope를 제작하고 모든 회로도나 소스를 Open Source로 공개하여 다른 사람들이 쉽게 제작할 수 있도록 하는 것을 목표로함

○ 작품 내용

- Android와 MCU의 통신을 통해 측정한 Analog신호를 스마트폰 화면에 출력
- 스마트폰 화면의 Button, Knob Button을 사용하여 원하는 Volt/div, Time/div, Trigger 및 모드 등을 설정

2) Worst Case

본 문

I. 작품 개요

※ 평가항목 : 기획력 (필요성, 차별성)

1. 작품 소개

가. 기획의도

사물인터넷(IoT) 시대가 예고되면서 흑색가전, 백색가전 분야에서도 관련 준비가 한창이라고 한다. 하지만 어떤 방식으로 IoT 생태계를 구축할지에 대해서는 여전히 명확한 표준이 없다.

스마트와 관련된 제품들은 이미 2000년대 초반부터 등장했으나 가격만 비싸고 쓸 만한 기능을 선보이지 못했고, 소비자들은 어떤 스마트 제품이 필요로 하는지 별로 느끼지 못했을 것이다. 하지만 IoT시대가 꿈틀대면서 가전제품도 스마트폰과 홈 디바이스가 결합한 스마트 홈(Smart Home)으로 진화하며 앞으로 내다본 업계 간의 경쟁이 치열해 지고 있다.

또한 업계 간의 개발을 진행하면서 여러 가지 개발 방식을 취할 수가 있다. 예를 들어 통신 방법으로는 지그비, 지웨이브, Wi-Fi 등이 있는데 여기서 나타나는 문제점은 스마트 홈과 관련된 표준이 정해진 것이 아니어서 여러 가지 가능성을 열어두고 있다고 한다. 이와 마찬가지로 센서, 처리 방법, 개발 언어 등 여러 가지가 있다.

IoT산업이 이제 현실로 다가왔을 뿐만 아니라, 현실 세계에 근접한 다양한 신개발 제품속의 창의적인 시도들이 프로젝트의 결과물로 제작하였다. 인간과 교감하는 대표적인 센서 기기와 비인간적인 컴퓨팅 능력을 활용하여 Smart Home의 색다른 감성 서비스를 구축하는데 이바지 하고 싶다.

그리하여, 이 프로젝트에서는 IoT 서비스 통합을 위한 근거리, 원거리 통신 서비스의 통합과 기하급수적으로 존재하는 다양한 기기들의 인터페이스 통합뿐만 아니라 향후에 시장에 진입하는 인터페이스 기기들의 통합을 고려한 중인지 서비스 플랫폼을 구현하였다. 그리하여, IoT 서비스 구현을 위한 서로 다른 기술적 개발 요구에 따른 비용을 줄이면서 사용성 증대를 위한 효율성을 높여 차별화된 Smart Home Service를 구축하였다.

2020 한이음 공모전 개 발 보 고 서

2020. 8.

프로젝트명	국문	라즈베리파이를 활용한 졸음운전 경보기
	영문	Sleeping Alert Using Raspberry pie
작 품 명	NoCarPain(교통사고 안돼!)	

요 약 본

글꼴 맑은고딕, 사이즈 10, 글자색 검정, 줄간격 160, 판넬/책자 등 활용 예정

작품 정보			# 필히 1장으로 작성
프로젝트명	국문	라즈베리파이를 활용한 졸음운전 경보기	
	영문	Sleeping Alert Using Raspberry pie	
작품명	NoCarPain(교통사고 안돼!)		
작품 소개	<p>졸음운전은 자신뿐만 아니라 타인의 생명에도 해를 끼칠 수 있는 매우 위험한 상황을 연출합니다. 따라서 이런 상황을 캐치하고 운전자의 졸음에 경고를 줄 수 있는 시스템으로써 운전자의 눈을 중점으로해서 졸음을 인식하고 다양한 아두이노 기기를 이용해 졸음 중임을 인식합니다. 이로써 운저자의 졸음 운전을 방지할 수 있는 프로젝트입니다.</p>		
작품 구성도			
작품의 개발배경 및 필요성	<p>지난 10년간 고속도로 교통사고 발생원인 1위는 무려 전체의 22.5%를 차지하는 졸음 운전입니다. 매년 2,000건 이상의 졸음운전이 발생하고 그 중 사망자는 100여 명이 넘어갑니다. 따라서 졸음 운전을 경고하고 방지해줄 수 있는 시스템이 필요합니다.</p>		
작품의 특징점	<p>영상처리 기법을 통해 운전자의 졸음 상태를 실시간으로 감시한다. 아두이노를 이용해 졸음을 깨워주는 장치들을 구현하므로 용이 및 보수가 쉽고 추후 아이디어를 추가시키기 쉽다.</p>		
작품 기능	<ul style="list-style-type: none">- 이산화탄소감지 센서를 이용한 이산화탄소 농도 측정- 눈 깜빡임을 인식하여 졸음 식별- 차선 이탈 감지- 심장 박동 감소 인식하여 졸음 식별- 자동 창문 개방, 졸음 인식 경보음, 자동 에어컨 off 기능- 가까운 졸음쉼터나 휴게소의 위치를 음성으로 알려줌- 졸음 인식하면 비상등을 이용하여 주변 차들에게 특이사항 알림		
작품의 기대효과 및 활용분야	<p>작품의 상용화로 졸음운전을 직접적으로 예방함으로써 경제적 관점에서는 사고의 감소를 통한 비용을 줄일 수 있고 더 나아가 사회적 관점으로는 법리분쟁, 인명피해 등을 예방할 수 있다.</p>		

본 문

본문 제목은 임의로 수정하거나 삭제할 수 없음

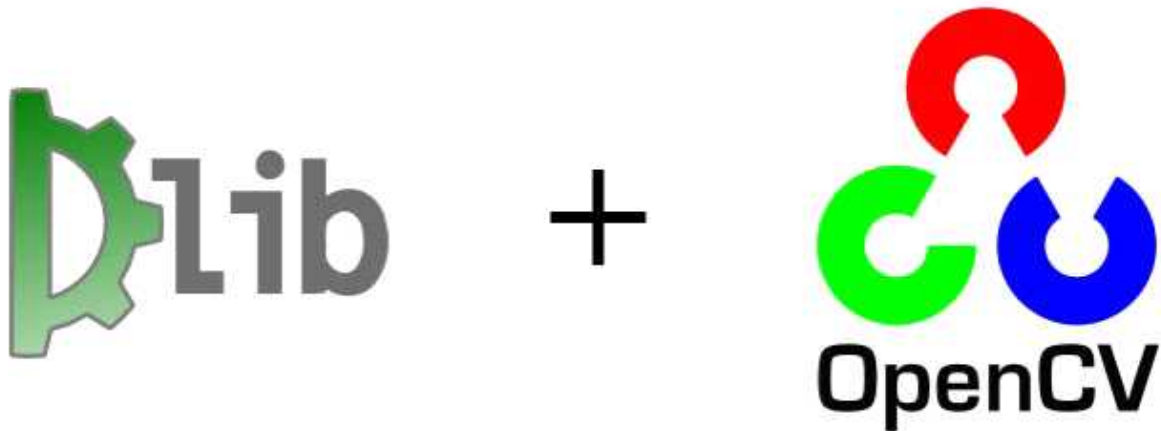
I. 작품 개요 # 작품개요는 2장 이내로 작성

※ 평가항목 : 기획력 (필요성, 차별성)

1. 작품 소개

1) OpenCV, Dlib

- 두 라이브러리를 이용해 눈깜빡임을 인지하여 졸음 식별에 중점을 둠.
- OpenCV에 있는 차선인식 라이브러리를 이용해 졸음을 식별할 때 더욱 정확한 판별이 가능함.



2) 기획 의도

- 국내 현대 모비스 기업에는 DSW라는 운전자의 얼굴을 알아보고 시선 추적까지 가능한 ‘운전자 부주의 경보시스템’을 개발하였지만 아직 대중적으로 상용화되지는 않음.
- 일반적으로 쉽게 접하고 구할 수 있는 라즈베리파이와 아두이노를 이용해 소비자가 합리적인 가격과 작동하기에 어려움이 없는 제품을 만드는 것을 목표로 함.

3) 작품 내용

- 라즈베리파이를 이용해 각종 디바이스를 컨트롤.
- Co2 센서, 심장박동 등을 이용해 정확성을 높임.
- 졸음 인식 후 창문 개방, 경보음 등을 활용하여 졸음 방지

기획의도, 작품 내용, 정의 등 그림을 포함하여 개조식으로 작성 (항목별 핵심내용만 작성)
- 참가신청서(개발보고서 포함) 작성 및 제출요령 '3. 개조식 작성예시' 참조

2. 작품의 개발 배경 및 필요성

1) 작품 제작 동기

- 대부분의 운전자들이 졸음운전에 대한 위험성은 인지하지만, 이에 대해 적절한 대처를 하는 운전자의 비율은 그리 높지 않다.
- 한국도로공사 고속도로 사고 통계에 따르면 지난 10년간 고속도로 교통사고 빈도 1위는 졸음운전 그러나 그에 비해 해결방안이 다소 적다.
- 운송업 종류의 운전을 많이 하는 직종은 야간운행이 빈번하고 보다 큰 차량을 운행하기 때문에 사고가 났을 때의 위험도가 상대적으로 큼.
- 졸음은 생리적인 현상으로 자신 스스로의 인지 및 해결이 힘들.

2) 목적 작성

- 개인뿐 아니라 사회적으로 많은 피해를 끼치는 졸음운전을 줄이기 위하여 졸음 쉼터와 같은 간접적인 방법보다는 운전자를 직접적으로 깨워줄 방법을 연구
- 졸음인식을 캐치하고 최종적으로 졸음을 깨우는 기능들을 아두이노를 이용하여 운전 집중을 할 수 있는 디바이스를 개발하는 것이 목표이다.

작품 제작 동기 및 목적 작성

3. 작품의 특징 및 장점

1) 차별성

- 단순히 눈 깜빡임만을 가지고 졸음운전을 감지하는 것이 아니라 차선 인식, Co2 센서, 심작박동 감지센서 등을 이용해 감지능력에 정확성을 높임.
- Co2 센서를 통해 인지한 Co2 농도가 평균보다 높다면 자동으로 창문 개방을 함으로써 졸음을 미리 예방하는 기능
- 비상등을 이용해 자신뿐만 아니라 주변 운전자에게도 경고할 수 있는 기능

2) 장점

- 눈 깜빡임과 차선 인식을 융합해 인식의 민감도를 조절할 수 있다.
ex) 차선 인식을 통해 차선 방향으로 가지 않고 휘청거린다면 눈 깜빡임에 비해 위험한 상황이므로 민감도를 급격히 올려 눈 깜빡임은 거의 개입하지 않고 경보를 울림
- 타 기업의 비싼 졸음 운전 감지 기능에 비해 비교적 저렴한 가격으로 구현 가능

작품의 주요 기능을 중심으로 기존 제품, 유사 제품과의 기능적/기술적 차별성을 작성

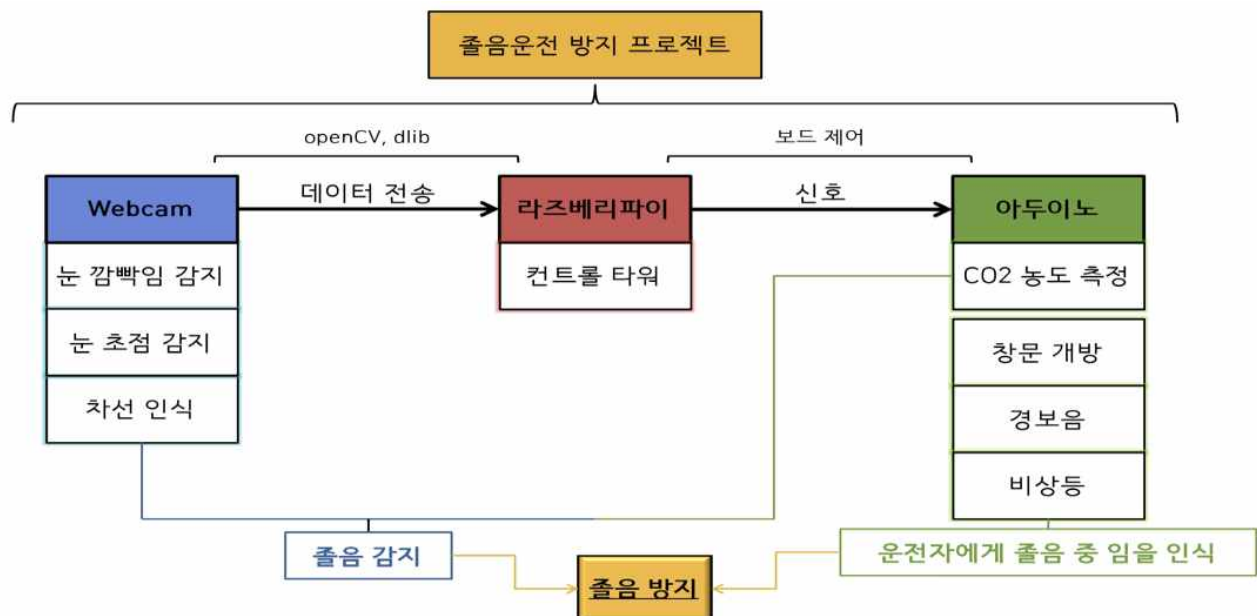
- 기존 작품을 Upgrade한 경우 기존 작품과 비교하여 차이점 및 차별성을 작성
- 유사 작품이 있는 경우 그 내용 및 유사 작품과 비교하여 본 작품의 특이점 및 차별성을 작성
- 신규 창작 작품인 경우 독창성을 충분히 설명하여 작성

II. 작품 내용

※ 평가항목 : 기술력 (기능구체성, 난이도, 완성도)

1. 작품 구성도

1) 전체 구성도



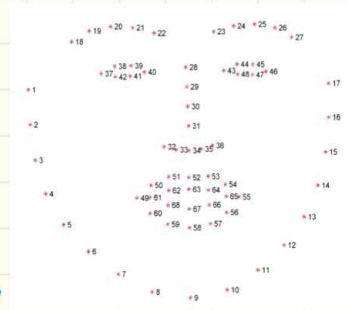
- 웹 카메라를 통해 눈 깜빡임과 초점, 차선을 인식하고 해당 정보를 라즈베리파이에서 입력받는다.
- 라즈베리파이에서 가지고 있는 입력값과 Co2 센서의 수치에 따라 졸음인지 아닌지 인식하고 졸음이라고 판단되면 졸음을 깨워주는 장치들이 작동한다.

2) S/W 구성도

얼굴 인식

<얼굴 인식>

- opencv와 dlib를 이용하여 특징점 추출
- 특징점(face landmarks)은 얼굴의 특정 부분마다 점을 찍어 나타내는 것
- 대표적으로 아래와 같은 68 points markup이 많이 사용



<눈 깜빡임 인식>

- r_eye_points = [42, 43, 44, 45, 46, 47]
- l_eye_points = [36, 37, 38, 39, 40, 41]
- 특징점들끼리 이어 선을 긋고, 가로/세로 비율을 추출할 함수를 만들어 줌
- 눈의 가로/세로 비를 ratio로 잡고 ratio가 어느 임계값을 넘어가면 눈을 감은 상태
- 연결된 카메라로부터 이미지를 가져온 뒤
- opencv를 이용하여 컬러 영상을 담고 있는 프레임을 흑백으로 변환
- 이미지를 흑백으로 변환하고 얼굴을 찾음
- 얼굴에서 입, 눈에 대해 사전에 선언한 함수를 이용하여 선을 긋습니다.
- 연결된 선의 길이를 기반으로 상태를 파악

```
#include <opencv/cv.h>
#include <opencv/highgui.h>

void main() {
    IplImage *frameImage, *blackImage;
    //카메라로부터 프레임(영상)을 받아서 출력
    CvCapture *video = cvCaptureFromFile("D:/study/testmedia.mov");

    cvNamedWindow("video", CV_WINDOW_KEEPRATIO);

    while (1) {
        //프레임(영상)을 하나씩 가져옴
        cvGrabFrame(video);

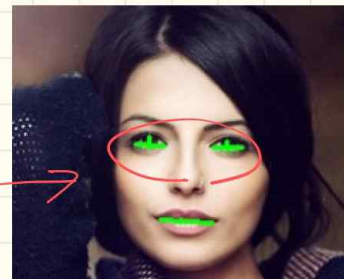
        //프레임(영상)을 하나씩 가져옴
        frameImage = cvRetrieveFrame(video);

        if (!frameImage) { break; }

        blackImage = cvCreateImage(cvGetSize(frameImage), IPL_DEPTH_8U, 1);
        //프레임(영상)을 흑백으로 변환
        //원본으로 : CV_CVCOLORCODE, 흑백으로 : CV_CVCGRAY
        //color code : RGB, GRAY, YUV, YCrCb, XYZ, Lab, Luv, HLS
        cvCvtColor(frameImage, blackImage, CV_BGR2GRAY);
        cvShowImage("video", blackImage); //윈도우에 이미지 표시

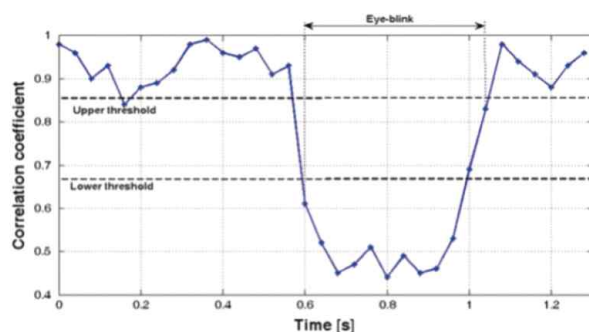
        if (cvWaitKey(10) > 0) {
            break;
        }
    }

    //카메라로부터 프레임(영상)을 받아서 출력
    cvReleaseCapture(&video);
    cvReleaseImage(&blackImage);
}
```



<졸음 인식>

- 눈깜빡임이 250밀리초에서 2초 사이일때 자발적인 눈 깜빡임이라고 간주
- 그 시간보다 더 감는 시간이 길 경우 졸음으로 판단



차선 인식

차선 인식

→ Gray scale, hough

1. 인식(고속도로 특정)

차선의 특징 -> 흰색(일반), 황색(중앙, 끝), 청색(버스 전용)

흰색영역 추출 -> 다른 부분도 추출됨 -> 데이터를 바탕으로 차선으로 추정되는 객체를 특정

2. 이탈 감지

2-1) 차선 내부에 있지만 주행 방향과 차선의 방향이 일치하지 않음

차선과 차량 중심의 연장선이 크게 어긋나있으면 차선이탈

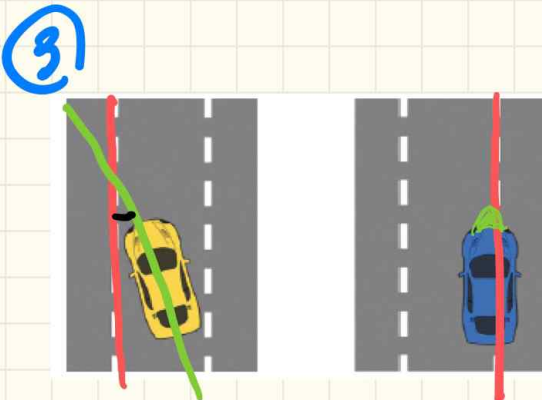
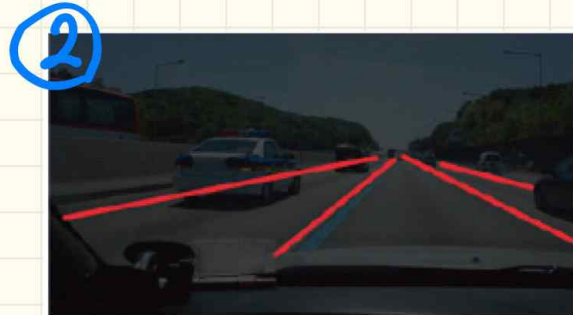
2-2) 차선을 밟고 주행하는 경우

차선이 차량 중심 주변에 걸쳐있으면 밟은 경우

3. 검출 속도 향상을 위한 관심영역 지정

하단을 0%라고 가정했을 때 10%~70% 사이를 지정

카메라 각도를 조절해야한다는 단점

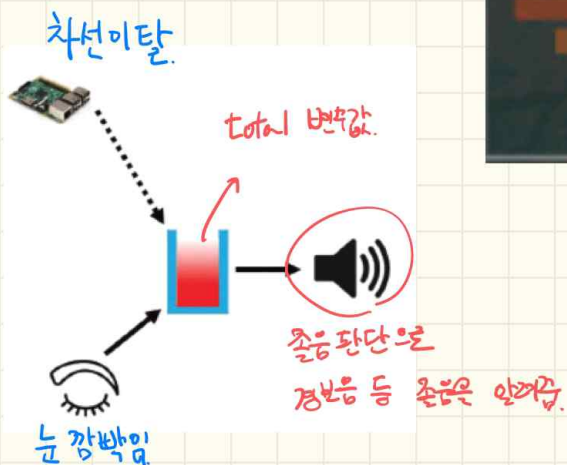


눈 깜빡임 + 차선이탈 융합

두 가지의 정보가 생길 때 마다 total 변수 값을 증가시킨다.

ex)

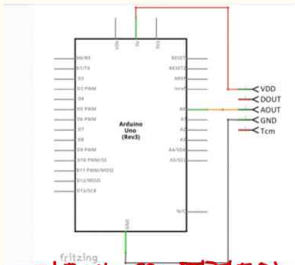
- 눈 깜빡임으로 인해 졸음운전이라 판단되면 total 변수값을 증가
- 차선 이탈도 마찬가지로 증가되는 값이 큼
- total 변수값이 max에 도달하면 졸음운전이라 판단해서 경고
- 경고 후 일정 수치를 깎는다. 이후에 졸음운전이 판단되지 않으면 점차 깎인다.



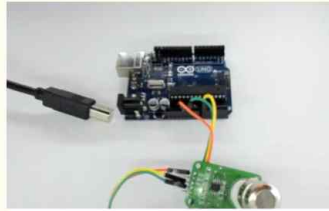
3) H/W 구성도

CO₂ 센서

(1) 읽어 들이는 방법



아두이노, CO2 면별도만



아두이노, CO₂ 연결
실제 사진

[illegible]

CO2센서-라이브러리코드

(2) 파일에 쌓아두기

-> 이산화탄소 데이터를 이용하지 않고, 임계값이 넘어가는 것만 체크하면 됨으로 저장해둘 필요성이 없다.

디스플레이 활용

(3) 주기적으로 읽기

-> 1분에 한번씩 이산화탄소 센서에 값을 가져와서 2000ppm이상일때
자동차 외기버튼이 눌러거나, 환기 실시 이때 환기는 10~15분정도 실시해야한다.

(4) 정보

차량 내부 CO₂ 농도 2500PPm => 위험임계값

농도

차량 내부 CO₂ 농도 700PPm 이하 = 7 안전임계값

~450 건강한 환기 관리 된 레벨

~700 장시간 있어도 건강에 문제가 없는 실내 레벨

~1000 건강 피해는 없지만 불쾌감을 느끼는 사람이 있는 레벨

~2000 줄림을 느끼는 등 컨디션 변화가 나오는 레벨

~3000 어깨 결림이나 두통을 느끼는 사람이 있는 등 건강 피해가 생기기 시작하는 레벨

3000~ 두통, 현기증 등의 증상이 나오고, 장시간으로는 건강을 해치는 레벨

// 이산화탄소 농도가 높을때 충분한 환기 시간

-> 주행중 위험임계값 넘으면 안전임계값에 도달할때까지 창문개방

참고

//SBS 졸음운전 연속기획

창문이 모두 닫힌채 운전하는 차량 내부 -> 승객도 운전자도 졸게된다.

-> 충분히 수면을 취해도 이산화탄소 농도가 높아지면 졸음이 온다

4명이 탄 승용차 기준

20여분에 5000 ppm 돌파

외기버튼 누르기 (창문 개방 대신)

경보음 and 비상등

운전자가 자신의 졸음상태를 인식하지 못하는 경우에 사용

-> 눈 영상 처리를 통해 졸음을 인식하지 못하는 경우, 비상등과 경보음 발생

-> CO2 농도가 높다고 조는것은 아님으로 경보등이나 비상등 장치는 연결하지 않음

운전 차량에서 비상등이 켜지고, 운전자가 인식하고 비상등을 끄면 경보음도 같이 꺼지는 방식으로 구현.

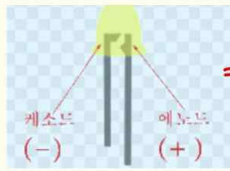
경보음과 비상등은 Set.

경보음 -> 피에조 부저를 이용하여 표현

비상등 -> LED 깜빡거리는 방법으로 표현

비상등 구현 방법

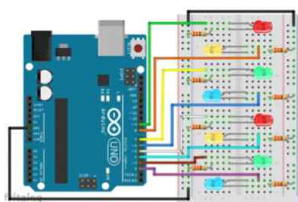
(LED 사용)



=> LED 전류제한용 저항 사용.

순방향 바이어스 LED 연결

브레드보드



아두이노, LED 연결 도면

```
int ledpin[] = {6,7,8,9};
void setup() {
  int index;
  for (index = 0; index<4;index++)
    pinMode(ledpin[index], OUTPUT);
}

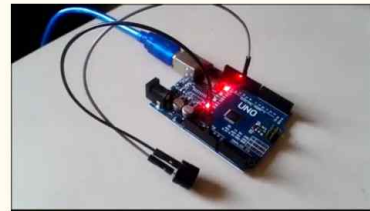
void loop() {
  start();
}

void start() {
  for(i = 0; i<4;i++)
  {
    digitalWrite(ledpin[i],HIGH);
    delay(100);
  }
  for(i=4;i<4;i--)
  {
    digitalWrite(ledpin[i],LOW);
    delay(100);
  }
}
```

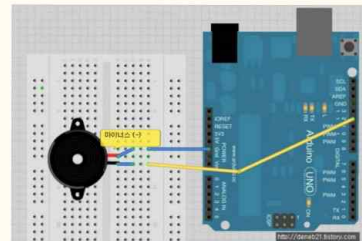
LED 구현 코드

경보음 구현 방법

(피에조 부저 사용)



아두이노, 피에조 부저 연결 사진



아두이노, 피에조 부저 연결 도면

##소스 1##

BEEP 음을 1초 마다 내는 소스, tone() 의 파라미터를 조절하면 다양한 음을 만들 수 있다.

```
1 int speakerpin = 12; //스피커가 연결된 디지털핀 설정
2
3 void setup() {
4 }
5
6 void loop() {
7   tone(speakerpin,500,1000); //500: 음의 높낮이(주파수), 1000: 음의 지속시간(1초)
8   delay(2000);
9 }
```

피에조 부저 구현 코드

S/W구성도, H/W구성도, 서비스흐름도 등을 전체적으로 작성(그림을 포함한 도식 또는 흐름으로 표현)

2. 작품 기능

1) 전체 기능 목록

구분	기능	설명	현재진척도(%)
S/W	눈 깜빡임(졸음) 감지	openCV로 운전자의 눈 폭이 사전 정의된 임계값 미만일 시에 졸음을 감지.	100%
	차선 인식	영상처리 기법으로 차량이 깜빡이를 켜지 않고 차선을 이탈했을 때를 감지. 완성가능시점(9/25)	30%
	심장 박동수 인식	센서를 활용하여 심박수 측정하여 사전 정의값(운전자의 평균 심박수)의 미만일 시에 졸음 의심. 완성가능시점(10/10)	0%
구매 H/W	Co2 농도 측정, 창문 개방	CO2농도가 높아지면 운전자의 졸음을 야기할수 있음. 이에 자동으로 창문을 개방, 환기. 완성가능시점(10/20)	50%
	경보음, 비상등	졸음 감지되면 경보음으로 운전자에게 인식, 주변 차량에게 비상등으로 특이사항 알림. 완성가능시점(10/20)	80%

- # 진척도는 신청일 기준 현재 시점의 진척도(%)를 작성하며 S/W는 개발진행 정도, H/W는 제작진행 정도를 기재하며 구매한 H/W 또는 S/W의 경우 "구매H/W", "구매S/W",라고 표시, 필요 시 줄 추가
- # 현재 진척도(%)가 100이 아닌 경우 '설명'란에 완성가능시점(월/일)을 작성

2) S/W 주요 기능

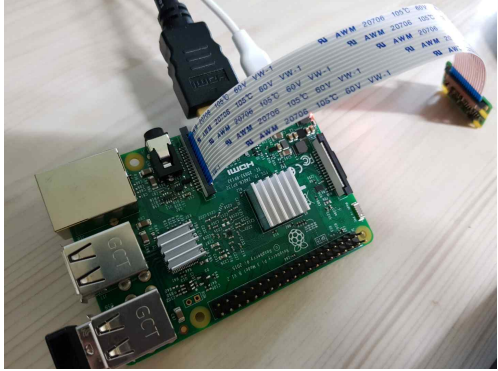
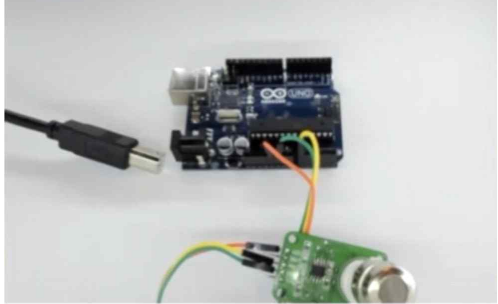
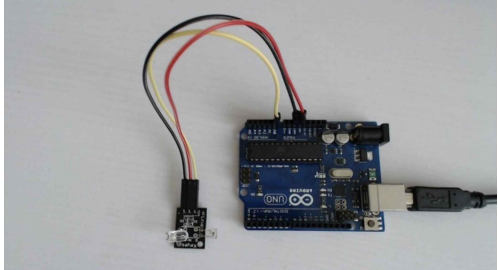
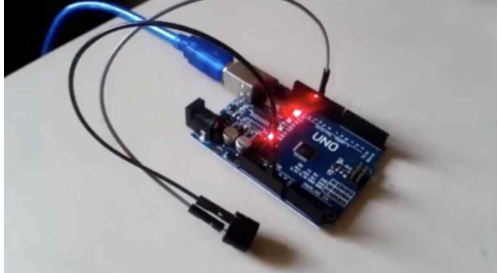
기능	설명	작품실물사진
눈 깜빡임 감지	openCV를 이용하여 운전자의 눈을 실시간으로 감지하고, 눈의 가로와 세로의 비율을 통해 눈이 감겨있는지 확인한다. 그 후 사전 정의된 조건에 따라 졸음이라고 판단.	
차선 인식	영상처리 기법으로 차량이 비정상적으로 차선을 이탈했을 때를 감지. 예를 들어 깜빡이를 넣지 않고 차선 이탈을 할 경우 졸음운전으로 인한 차선 변경이라고 판단.	
졸음 감지의 정확성을 높이기 위한 추가적 요소	각종 센서를 이용하여 눈 깜빡임이나 차선인식만으로는 충분하지 않을 수 있는 오작동 비율을 낮춘다. 눈 깜빡임, 차선 이탈, 각종 센서마다 임계값을 넘으면 경보음이 울리기 위한 수치를 설정해놓고, 이 수치의 총합이 일정 값 이상을 넘을 시 졸음운전이라고 판단.	

작품 실물사진을 반드시 첨부하여 실제 작품 완성도 확인할 수 있도록 작성, 필요 시 줄 추가

- 사람 사진에 눈 깜빡임 알고리즘을 적용시켜서 임계값을 보여주고 졸음인지 아닌지 판단한다.
- 추가적인 요소는 코드로 조절하는 부분이어서 실물사진을 그림으로 대체함.

표는 필수로 작성하고, 부연 및 추가설명 사항이 있는 경우 작성

3) H/W 주요 기능

기능/부품	설명	작품실물사진
졸음 감지 / 카메라 2대	첫 번째 카메라는 운전자의 눈 깜빡임을 인식 두 번째 카메라는 차량 전방의 차선을 인식	
졸음 감지 / Co2 센서(창문 개방)	차량 내부의 CO2농도를 주기적으로 측정하고 그 측정값이 임계값을 넘으면 창문을 개방하여 졸음을 방지.	
졸음 감지 / 심장박동 센서	차량 운전자가 탑승한 후 10분간 운전자의 심장박동수를 주기적으로 측정하고 그 측정값이 일정값이 넘으면 경보음이 울리기 위한 수치의 하한값을 증가시킴	
졸음 퇴치 / 피에조 부저	각종 센서와 졸음 감지에 대한 조건을 충족하여 졸음이라고 감지되면 경보음을 발생시킴	

- # 사진으로 진척도를 확인 할 수 있도록 작성, H/W가 없는 경우 작성 안함(표는 유지)
- # 작품 실물사진을 반드시 첨부하여 실제 작품 완성도 확인할 수 있도록 작성, 필요 시 줄 추가

- 현재 카메라 1대로 눈 깜빡임 인식과 차선 인식을 돌아가면서 테스트하고 있는 상황이고 나머지 카메라 1대는 주문한 상태.

- # 표는 필수로 작성하고, 부연 및 추가설명 사항이 있는 경우 작성

3. 주요 적용 기술

- dlib를 이용해 얼굴에 68 points markup 특징점을 나타내고 이를 이용해 얼굴을 인식한다. (관련한 알고리즘들이 있는 고성능의 라이브러리를 사용 -> face detection, face landmark detection)
- 실시간 컴퓨터 비전을 목적으로 한 프로그래밍 라이브러리인 opencv를 이용 -> dlib의 학습된 랜드마크 모델 데이터 입력
- opencv와 dlib를 활용할 수 있는 언어인 python을 사용하고 opencv에서 제공하는 트레이닝 코드를 이용할 수 있음.
- 줄음을 깨워주는 장치들인 각종 센서와 이 센서들을 관리하는 arduino를 연동하고 라즈베리파이에서 장치들을 작동시킬 값을 받아옴.
- 라즈베리파이로 프로젝트 개발환경을 구축하고 라즈베리파이의 터미널을 이용해 각종 리소스와 사용하는 알고리즘들을 적용시키고 실행
- os는 우분투를 사용해 개발물들을 먼저 테스트한 후 라즈베리파이에 적용시켜 개발하기 편한 환경을 구축

S/W, H/W 기능 구현을 위한 시나리오, 적용 알고리즘 및 주요 적용 기술 등을 상세히 작성
(이론적인 부분 포함)

4. 작품 개발 환경

구분		상세내용
S/W 개발환경	OS	Ubuntu
	개발환경(IDE)	PyCharm, terminal
	개발도구	PyCharm
	개발언어	Python
	기타사항	노트북을 이용
H/W 구성장비	디바이스	Raspberry, Arduino
	센서	Co2 센서, 피에조 부저, 심장박동 센서
	통신	와이파이
	기타사항	라즈베리파이와 아두이노를 연동해주는 모듈 존재
프로젝트 관리환경	형상관리	Git
	의사소통관리	애자일 방법론의 데일리 스크럼
	기타사항	멘토와 월 1회 이상 오프라인 미팅, 카카오톡을 통한 상시 멘토링

상세내용에 사용 툴 등 자세히 기재, 필요 시 항목 수정 또는 추가(협업도구 등)

5. 기타 사항 [본문에서 표현되지 못한 작품의 가치(Value)] 및 제작 노력

- 눈 깜빡임 알고리즘과 차선 인식 알고리즘을 융합해 정확성을 높이는 창작성을 보임
- Co2 센서를 이용해 줄음이 올 수 있는 상황을 미리 인지하고 예방할 수 있는 기능을 추가함.
- Co2 센서와 마찬가지로 심장박동 감지센서를 이용해 차에 탑승한 후 10분간 평균 심박수를 측정하고 그 측정값보다 낮아지면 줄음이 올 수 있는 상황으로 인지하고 예방할 수 있는 기능들을 추가함.
- 비교적 저렴한 가격을 통해 쉽게 접할 수 있다.
- 교통사고를 예방한다는 점에서 법리 분쟁과 인명피해를 줄이고 더 나아가 불필요하게 쓰이는 비용을 효율적으로 사용할 수 있다.

차별화된 성능, 신뢰성, 사용성, 보안, 알고리즘의 창작성, 데이터와 프로그램의 가치, 작품을 통한 가치창출 등을 상세히 작성
(이론적인 부분 포함)

III. 프로젝트 수행 내용

※ 평가항목 : 수행능력 (문제해결능력, 수행충실성)

1. 프로젝트 수행일정

프로젝트 기간 (한이음 사이트 기준)		2020.05.13. ~ 2020.11.30.											
구분	추진내용	프로젝트 기간											
		1월	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월	12월
계획	프로젝트 수행 계획서 작성												
분석	유즈케이스 다이어그램 및 시스템 아키텍처												
설계	opencv, db 이용한 줄음에 대한 얼굴 인식 설계												
	Co2 농도측정센서를 이용한 구현 설계												
	아두이노를 이용한 기타 부수장비 설계												
개발	얼굴인식, 줄음 식별 코드 구현												
	Co2 농도측정 코드 구현 차선이탈 감지 코드 구현												
	아두이노를 통한 모형 제작												
테스트	줄음감지 데이터를 라즈베리파이에 전달												
	라즈베리파이를 활용한 아두이노 모형 구동 테스트												
종료	프로젝트 완료 보고서 2020년 한이음 공모전 준비												

기획부터 결과물 도출까지의 수행절차, 일정 및 현황 기재
프로젝트 기간은 노란색 셀 색상으로 표시, 필요 시 줄 추가

2. 프로젝트 추진 과정에서의 문제점 및 해결방안

1) 프로젝트 관리 측면

- 개발환경(Ubuntu)에서의 줄음 식별 코드를 운영환경(Raspberry-Pi)로 구현시켰을 때 몇몇 문제를 발견 -> 라즈베리파이에서는 파이썬 터미널을 사용할 수 없어서 문제가 발생했던 것. 따라서 라즈베리파이의 자체 터미널을 이용해 소스를 실행시키는 방법으로 문제 해결
- dlib를 이용한 줄음인식 코드 구현에서 몇몇 문제를 발견 -> 윈도우에서 사용했던 폴더 그대로 넘어와서 경로가 엉켜있어서 문제가 발생했던 것. 폴더를 재지정해줌으로써 문제 해결

프로젝트 진행 중 발생한 문제점 및 해결 경험 작성

2) 작품 개발 측면

- 기존의 차선이탈 감지 코드는 딥러닝을 사용하는 방식인데 학습 시간의 차질을 빚을 것이라 대비해 opencv로 구현하는 것을 목표로 둬.
- 눈 깜빡임의 인식이 빠르게 깜빡여도 졸음이라고 판단하는 경우가 있어서 수치 조정이 필요함. -> 2~3 사이의 값으로 조절.
- Co2센서의 정확한 임계값을 설정하기 위해 밀폐된 공간에서 측정하는 방식을 사용해야 함.
- 아두이노가 측정한 수치를 라즈베리파이에 보내 정보로 사용할 때 적절한 오차범위 설정이 필요함.

기술적으로 발생한 문제점 및 해결 경험 작성

3. 프로젝트를 통해 배우거나 느낀 점

- 멘티의 교육적 기대효과

<기술 이해 및 지식 습득>

- 1) 라즈베리파이를 이용함으로써 아두이노 등의 H/W 와 OpenCV, dlib 등의 S/W 부분을 통합하는 법에 능숙해짐.
- 2) OpenCV, dlib를 활용한 영상처리 기법의 기본 개념 함양 및 실습을 통한 실무적 능력 증가.
- 3) 아두이노를 활용함으로써 임베디드 분야 학습에 도움이 되고 더 나아가 하드웨어적 설비 능력 향상.

<체계적인 프로젝트 수행>

- 1) 프로젝트 완성을 통해 프로젝트가 개발만이 중요한 것이 아닌 계획과 설계 또한 매우 중요한 것임을 알 수 있음.
- 2) 지도자의 풍부한 경험과 지식을 토대로 한 멘토링을 통해 멘티로서 프로젝트 수행 능력을 향상.

IV. 작품의 기대효과 및 활용분야

※ 평가항목 : 기획력 (활용가능성)

1. 작품의 기대효과

<경제성>

- 1) 도로교통공단에 따르면 교통사고 처리비용은 국가 예산(400조)의 약 5.9%에 해당하고, 4인 가족 기준 약 147만여 가구의 연간 최저생계비에 이르는 규모다.
- 2) 고속도로 사고의 주요 원인인 졸음 운전을 방지한다면 비용을 대폭 줄여 조금 더 효율적인 곳에 사용할 수 있을 것이다.

<사회성>

- 1) 졸음 운전에 대한 관심을 높힘으로써 인식을 개선
- 2) 법리 분쟁과 인명피해를 줄임으로써 GNH(국민 행복 지수) 증가.

해당 작품을 통한 기존 서비스와의 차별성과 시장성(가격 우위성) 등 작성

2. 작품의 활용분야

- 사회적 측면으로 졸음 운전을 보다 직접적인 방법으로 예방함으로써 졸음운전의 감소와 졸음운전에 대한 인지도를 높일 수 있음.
- 추후 이용자의 측정 데이터를 저장해 더 정교한 모델을 만드는데 사용할 수 있다.
- 경제적 측면으로 인명피해가 아니더라도 교통사고가 줄어들어서 개개인 뿐만 아니라 국가에서도 들어가는 비용이 줄어든다.

해당 작품을 통한 서비스 활용분야에서의 실질적인 효과 작성



내가 기획한 ICT가
세상을 바꾼다면?





SW개발/HW제작 설계서

프로젝트 명 :

한이음 ▶ 프로그램 설계서

수행 단계별 주요 산출물



단계	산출물	일반	응용 소프트웨어	응용 하드웨어
		·모바일 APP ·Web 등	·빅데이터 ·인공지능 ·블록체인 등	·IoT ·로봇 ·드론 등
환경 분석	시장/기술 환경 분석서	△	△	△
	설문조사 결과서	△	△	△
	인터뷰 결과서	△	△	△
요구사항 분석	요구사항 정의서	○	○	○
	유즈케이스 정의서	△	△	△
아키텍처 설계	서비스 구성도(시스템 구성도)	○	○	○
	서비스 흐름도(데이터 흐름도)	△	○	△
	UI/UX 정의서	△	△	△
	하드웨어/센서 구성도	-	-	○
기능 설계	메뉴 구성도	○	○	○
	화면 설계서	○	○	△
	엔티티 관계도	○	○	△
	기능 처리도(기능 흐름도)	○	○	○
	알고리즘 명세서/설명서	△	○	○
	데이터 수집처리 정의서	-	○	-
	하드웨어 설계도	-	-	○
개발 / 구현	프로그램 목록	○	○	○
	테이블 정의서	○	○	△
	핵심 소스코드	○	○	○

※ ○ 권장, △ 선택