

## 1. 과제 계획서

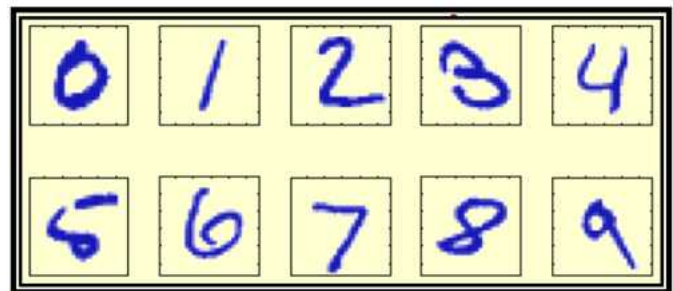
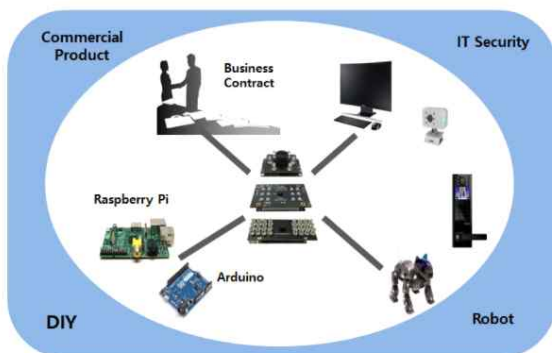
### □ 과제 목적 및 필요성

#### ■ 개요

- 신용카드 크기의 저가 및 저전력 컴퓨터 장치인 라즈베리파이(Raspberry Pi)와 오픈소스 소프트웨어를 활용하여 얼굴인식 및 숫자인식과 같은 객체인식을 수행하고자 함
- 라즈베리파이에 카메라를 부착하고, 파이썬(python) 언어를 기반으로 텐서플로우(Tensorflow)와 같은 딥러닝 오픈소스 프레임워크 또는 컴퓨터 비전 Opencv 라이브러리 등을 이용하여 객체인식 기술 구현
- 라즈베리파이를 동작하기 위한 임베디드 하드웨어와 리눅스 운영 학습을 통한 개발환경 구성

#### ■ 필요성

- 기계학습/딥러닝을 활용한 영상 인식 기술은 인공지능 기술을 기반으로 인간의 시각적인 인식 능력까지 재현하는 단계까지 가능하게 하고 있으며, 영상 데이터 기반 위치 식별도 가능한 객체 인식까지 확장되고 있음
- 산업용 기기 및 소규모 스마트팜, 사물인터넷(IoT)과 같은 저가의 컴퓨팅 자원이 요구되는 하드웨어 환경에 대해서도, 수집된 데이터를 전처리하고, 인공지능 기반의 추론을 수행하여 데이터 기반의 유용한 정보를 추출하기 위한 요구가 증대되고 있음.
- 따라서, 본 프로젝트에서는 초소형 컴퓨터로 활용될 수 있는 라즈베리파이에 대한 하드웨어 및 소프트웨어 구성을 이해하고, 오픈소스 소프트웨어를 활용한 얼굴인식 및 숫자인식과 같은 객체인식 기술을 구현하고자 한다.



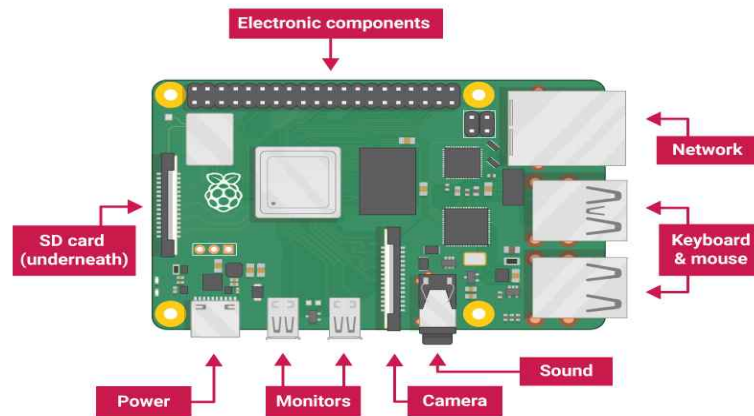
### □ 과제 수행내용

#### ■ 수행내용

- 라즈베리파이 하드웨어 구성과 운영체제(OS) 설치
- 라즈베리파이에 객체인식을 위한 카메라 설치 및 소프트웨어 환경 설정
- 라즈베리파이에서 python 가상환경 설치와 OpenCV를 이용하여 카메라를 통한 실시간 객체인식 수행

#### ■ 라즈베리 파이(Raspberry Pi)

- 라즈베리파이는 영국의 라즈베리 파이 재단이 기초 컴퓨터 과학의 교육을 위해 개발한 싱글 보드형 컴퓨터이며, 마이크로프로세서, 메모리, 입출력 연결단자 등을 하나의 회로로 구성한 초소형 컴퓨터라고 생각할 수 있음.
- 본 프로젝트에서 고려하는 라즈베리파이 사양은 Paspberry PI 4 model B로써, 4GB 메모리를 가짐
- 라즈베리파이 운영체제를 설치하기 위한 SD card와 SD card 리더기, 카메라 장치 준비 - Windows 10이 설치된 PC에 연결



<라즈베리파이 구성요소>

#### ■ OpenCV 컴퓨터 비전 라이브러리

- OpenCV는 오픈소스로 개발된 컴퓨터 비전 라이브러리임.
- 인텔 CPU에서 사용되는 경우 속도의 향상을 볼 수 있는 IPP를 지원

#### ■ 과제 구성 및 방법

##### ① 라즈베리파이 OS 설치 및 접속

- Raspberry PI Imager를 이용하여 설치
- SSH 접속, wpa\_supplicant.conf 파일 만들어서 WIFI 설정
- 공유기를 통해 라즈베리파이로 접속 할 수 있도록 공유기 포트포워딩

##### ② 라즈베리파이에 카메라 연결

- 라즈베리파이 전원을 OFF
- 라즈베리파이 보드에 CAMERA라고 표시된 인터페이스 양쪽 끝을 PUSH 함
- 흰색 면이 HDMI 포트 쪽에 오도록 연결

##### ③ Python 가상환경과 OpenCV설치 후 카메라 동작 확인

- 다양한 환경의 버전에서 동작하기 위한 파이썬 가상환경 설치 및 설정
- OpenCV 라이브러리를 이용한 카메라 인식 동작 확인

##### ④ 카메라를 통한 실시간 객체인식 수행

- OpenCV 라이브러리를 이용한 실시간 얼굴인식 모델 구현 및 실습
- 라즈베리파이에서 딥러닝 오픈소스 프레임워크인 텐서플로(Tensorflow)를 이용하여 손으로 쓴

## □ 기대효과 및 활용방안

### ■ 활용방안

- 본 프로젝트에서 수행한 객체인식 분야의 얼굴 인식 기술을 적용하여 범죄 용의자 검색 기능을 추가하여 범죄 수사에 활용할 수 있음
- 본 프로젝트에서 수행한 객체인식 분야의 얼굴 인식 기술을 적용하여 마케팅 분야에서 등록된 사진과 매칭하여 동일한 인물일 경우 맞춤형 마케팅을 이용 할 수 있음
- 본 프로젝트에서 수행한 객체인식 분야의 얼굴 인식 기반의 인증 프로그램을 이용하여 보안 분야에서 활용 할 수 있음
- 또한, 객체인식 기술은 어디에도 응용할 수 있기에 로봇, 교육, 교통 등 다양한 분야에 활용하여 다방면적인 인공지능 기술의 발전을 기대할 수 있음.

### ■ 기대효과

- 얼굴 인식 기술을 적용하여 CCTV와 얼굴 검색을 하여 범죄 수사에 사용할 수 있음
- 얼굴 인식 기술을 이용하여 마케팅 기술을 만들어 경제적으로 큰 기대를 줄 수 있음
- 휴먼 인증 및 검색의 핵심 기술로 활용 가능
- 얼굴 인식 기술을 적용한 자체 거울로 졸음을 방지하는 프로그램을 이용하여 사고를 예방할 수 있음

## 2. 과제 추진방법

### 가. 팀원별 역할 분담

연번	성명	담당	수행역할	비고
1	배채정	과제관리	과제관리 및 보고서 작성	팀장
2	김민기	자료수집	자료 수집 및 보고서 작성	팀원
3	박수빈	과제계획	과제계획 및 수행	팀원
4	한승희	자료수집	자료수집 및 보고서 작성	팀원
5				
6				
7				

※ 팀 구성원과 세부적인 역할 분담내용을 기술하고 지역사회 및 기업체 연계과제일 경우 참여기업의 역할을 구체적으로 기재

나. 과제추진 일정

추진 내용	수행기간(월)				비 고
	4월	5월	6월	월	
Brain stoming					
자료수집					
실험 및 보고서 작성					

※ 세부추진 내용 및 추진 시기는 구체적으로 기재

다. 참여기관의 제안 내용

없음
----



# 캡스톤 디자인 과제 수행 결과보고서



## 1. 과제수행 및 결과보고

<b>작 품 명</b>	라즈베리파이와 OpenCV 컴퓨터 비전 라이브러리를 활용한 얼굴 인식
<input type="checkbox"/> 작품 개요 및 필요성	
<div> <div>■ 작품 개요</div> <ul style="list-style-type: none"> <li>· 객체인식을 위한 저전력 컴퓨팅 PC인 라즈베리파이 하드웨어 구성 및 소프트웨어 운영</li> <li>· 파이썬(Python) 가상환경에서 라즈베리파이(Raspberry Pi)와 OpenCV 컴퓨터 비전 라이브러리를 활용한 얼굴인식 프로그래밍</li> <li>· 라즈베리파이 모듈에서 딥러닝 오픈소스 프레임워크 텐서플로(Tensorflow)를 이용하여 손으로 쓴 숫자를 카메라로 인식하는 객체인식</li> </ul> </div> <div> <div>■ 과제 필요성</div> <ul style="list-style-type: none"> <li>· 객체인식은 사람이 가장 많은 정보를 받아들이는 시각정보를 컴퓨터가 대신하여 분석하고 해석할 수 있는 연구분야로써, 현재 산업 전반에 빠질 수 없는 핵심기술로 각광받고 있음</li> <li>· 특히, 생체 인식 중 하나인 얼굴 인식은 비밀번호의 대안으로 떠오르고 있으며, 최근 다양한 분야에서 활용이 기대되는 얼굴인식 기술이 각광받고 있음</li> <li>· 얼굴 인식으로 성별과 연령대를 구분하고, 광고를 본 시간 등 다양한 데이터를 축적할 수 있음.</li> <li>· 딥러닝 오픈소스 소프트웨어를 활용하여 기존 기계학습 모델보다 정확도 성능이 우수한 객체인식 수행이 필요함</li> </ul> </div>	
<input type="checkbox"/> 작품의 개발 방법 및 과제 수행 과정	
<div> <div>■ 작품 개발 방법</div> <p>라즈베리파이의 저전력 컴퓨팅 하드웨어 환경에서 딥러닝 기술의 핵심인 합성곱 신경망(CNN: Convolutional Neural Network) 모델을 설계 및 구현을 수행할 수 있는 딥러닝 오픈소스 소프트웨어를 활용하여 다음과 같은 객체인식을 수행</p> </div> <div> <ol style="list-style-type: none"> <li>1. OpenCV 컴퓨터비전 라이브러리를 활용한 얼굴인식           <ul style="list-style-type: none"> <li>- OpenCV를 이용한 카메라 모듈 동작 확인</li> <li>- OpenCV 라이브러리를 활용한 얼굴 객체 인식</li> </ul> </li> <li>2. 손으로 쓴 숫자 인식 모델 학습하기           <ul style="list-style-type: none"> <li>- 숫자 인식을 위한 신뢰성 있는 MNIST(Modified National Institute of Standards and Technology) 데이터 셋을 활용</li> <li>- 구글 클라우드 환경인 코랩(COLAB)에서 합성곱 신경망(CNN) 모델 기반 숫자 인식 모델 설계 및 구현</li> <li>- 딥러닝 오픈소스 소프트웨어로써, Tensorflow 2.0 &amp; Keras 소프트웨어를 사용</li> </ul> </li> </ol> </div>	

### 3. 손으로 쓴 숫자 인식 모델 추론하기

- 학습된 모델을 Load하여 입력값에 대한 모델의 출력값을 반환
- 추론만을 위한 그래프 노드 및 정보만 포함된 Tensorflow Lite model을 사용
- 카메라를 이용하여 실시간으로 손으로 쓴 숫자를 추론하기

## ■ 과제 수행 과정

#### 1. Raspberry Pi OS 설치

- SD card와 SD card 리더기를 이용하여, Raspberry PI Imager를 이용한 OS 설치

#### 2. Raspberry Pi 원격 설정하여 OS에 접속

- SSH 접속 및 Wi-fi 설정, 공유기 포트포워딩을 통하여 공유기를 통해 라즈베리파이로 접속할 수 있도록 환경 구성

#### 3. Raspberry Pi 카메라 연결 및 설치

- 카메라 기본 동작 확인을 위한 OpenCV 라이브러리 설치

#### 4. OpenCV 라이브러리를 이용한 Raspberry Pi 카메라 동작 확인

- TestImage가 정상적으로 출력되는지 확인

#### 5. Raspberry Pi에서 pip 명령어를 이용하여 딥러닝 오픈소스 프레임워크 Tensorflow 설치

#### 6. 구글 클라우드 플랫폼인 코랩(COLAB) 환경에서 MNIST 데이터셋으로부터 숫자 인식을 위한 딥러닝 모델 학습

#### 7. 학습된 모델을 이용하여 입력 값에 대한 모델의 추론 결과를 카메라로 실시간 출력

## □ 결과

. OpenCV 컴퓨터 비전 라이브러리와 카메라를 이용한 얼굴인식 수행 결과



- 라즈베리파이에서의 학습은 저사양의 컴퓨팅 사양으로 인하여 매우 느리고 시간이 많이 걸리기 때문에, 구글 클라우드 환경을 이용해서 모델을 학습한 다음, 학습된 모델을 다운하였음
  - tensorflow.keras.models.load\_model : 학습한 모델 추론을 위한 Keras 모델 Load
  - Tensorflow Lite로 학습 모델 최적화를 통한 추론 수행
  - tensorflow.keras.models.load\_model보다 Tensorflow Lite의 추론 속도가 약 30배 빠름

	<pre>import tensorflow as tf import time  import numpy as np  random_array = np.random.rand(1,28,28,1) random_array = random_array.astype(np.float32) interpreter = tf.lite.Interpreter(model_path= './models/mnist_model.tflite') interpreter.allocate_tensors()</pre>
tflite 모델 불러오기	
모델에 정의된 입력/출력 텐서 정보 불러오기	<pre>input_details = interpreter.get_input_details() output_details = interpreter.get_output_details()</pre>
모델에 입력값을 입력하여 출력값 받아오기	<pre>s_t = time.time() interpreter.set_tensor(input_details[0]['index'], random_array) interpreter.invoke() output_data = interpreter.get_tensor(output_details[0]['index'])  e_t = time.time() print(e_t - s_t)</pre>

- 카메라 장비를 이용한 실시간 딥러닝 모델 추론을 위해서는 Tensorflow Lite 사용을 권장
  - 복잡한 모델 혹은 Tensorflow 최신 버전에서 새롭게 정의된 연산일 경우 Tensorflow Lite 사용에 어려움이 있음

## 2. 활용방안 및 기대효과

### □ 활용방안

- OpenCV와 얼굴 인식 카메라를 사용하여 게임과 같은 'Google T-rex game'을 실행하여 다른 분야에 적용할 수 있음
- 코로나19와 같은 상황에서 사람들의 출입을 통제하여야 할 때, 해당 모델과 적외선 카메라를 사용하여 출입이 가능한 사람과 그렇지 않은 사람을 구분하는 데 활용할 수 있음.
- 눈과 코 주변의 생김새를 집중적으로 분석하여 마스크 착용 여부를 판단할 수 있으며 마스크를 턱과 입에 걸쳐서 착용한 사람이나, 마스크를 착용하지 않은 사람들에게는 경고 알림을 내는 방안으로도 활용할 수 있음.

- 은행 정보나 개인 신상 정보, 전자상거래 등의 보안이 중요한 분야에서는 얼굴인식을 활용하여 신분 확인을 안전하면서도 원활하게 할 수 있음
- 또한, 지문이 흐려 지문인식으로는 신분 확인이 어려웠던 사람들도 얼굴인식을 활용하여 신분 확인이 가능한 기술로 확장될 수 있음

#### □ 기대효과

- 근접촉 생체인증 시스템에 비해 원거리에서도 생체 정보 획득이 가능하며 거부감이 적음
- 객체인식 기술을 적용한 마스크 착용 여부를 판단하여 코로나19 확산을 예방하는데 도움이 될 수 있음
- 실시간 얼굴 인식을 통해 찾고자 하는 사람의 위치 정보를 빠르게 파악 가능
- 범죄자 검색 등 치안 방법 분야에 활용하여 안전한 사회적 환경 구축 가능
- 본 프로젝트에서 수행한 객체인식 기술을 활용하여 저사양 컴퓨팅 환경으로 건물 출입자의 얼굴을 인식하여 빠른 신분확인 가능
- 실시간으로 얼굴을 인식함으로써, 인건비 절약 및 시간 절약 가능

#### □ 기타 (한승희)

