사물인식을 위한 open cv

**openCV설치 방법**

open CV설치를 위해서는 16gb의 저장공간이 필요하다.(라즈비안을 설치한 sd카드의 용량)

라즈베리 터미널에서 패키지 매니저 설치

~$ sudo apt-get install pkg-config

Open CV설치확인

~$ pkg-config --modversion opencv

Open CV최신버전 업데이트

~$ sudo apt-get update

~$ sudo apt-get upgrade

OpenCV 설치에 필요한 패키지 매니저 설치

~$ sudo apt-get install build-essential cmake

~$ sudo apt-get install libjpeg-dev libtiff5-dev libpng12-dev

~$ sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libxvidcore-dev libx264-dev libxine2-dev

~$ sudo apt-get install libv41-dev v41-utils

~$ sudo apt-get install libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev

~$ sudo apt-get install libgtk2.0-dev

~$ sudo apt-get install mesa-utils libgl1-mesa-dri libgtkgl2.0-dev libgtkglext1-dev

~$ sudo apt-get install libatlas-base-dev gfortran libeigen3-dev

~$ sudo apt-get install python4.0-dev python3-numpy

Open Cv 설치를 위한 폴더 생성

~$ mkdir opencv

~$ cd opencv

~/opencv $ wget -O opencv.zip <https://github.com/opencv/opencv/archive/3.4.3.zip>

~/opencv $ unzip opencv.zip

~/opencv $ wget -O opencv\_contrib.zip <https://github.com/opencv/opencv_contrib/archive/3.4.3.zip>

~/opencv $ unzip opencv\_contrib.zip

Opencv 설치폴더 이동

~/opencv $ cd opencv-3.4.3

~/opencv/opencv-3.4.3 $ mkdir build

~/opencv/opencv-3.4.3/build $ cmake -D CMAKE\_BUILD\_TYPE=RELEASE \

-D CMAKE\_INSTAL\_PREFIX=/usr/local  ﻿\

-D WITH\_TBB=OFF  ﻿\

-D WITH\_IPP=OFF  ﻿\

-D WITH\_1394=OFF  ﻿\

-D BUILD\_WITH\_DEBUG\_INFO=OFF  ﻿\

-D BUILD\_DOCS=OFF  ﻿\

-D INSTALL\_C\_EXAMPLES=ON  ﻿\

-D INSTALL\_PYTHON\_EXAMPLES=ON  ﻿\

-D BUILD-EXAMPLES=OFF  ﻿\

-D BUILD\_TESTS=OFF  ﻿\

-D BUILD\_PERF\_TESTS=OFF  ﻿\

-D ENABLE\_NEON=ON  ﻿\

-D ENABLE\_VFPV3=ON  ﻿\

-D WITH\_QT=OFF  ﻿\

-D WITH\_GTK=ON  ﻿\

-D WITH\_OPENGL=ON  ﻿\

-D OPENCV\_EXTRA\_MODULES\_PATH=../../opencv\_contrib-4.0.0/modules  ﻿\

-D WITH\_V4L=ON  ﻿\

-D WITH\_FFMPEG=ON  ﻿\

-D WITH\_XINE=ON  ﻿\

-D BUILD\_NEW\_PYTHON\_SUPPORT=ON  ﻿\

../

~/opencv/opencv-3.4.3/build $ sudo nano /etc/dphys-swapfile

~/opencv/opencv-3.4.3/build $ free

~/opencv/opencv-3.4.3/build $ sudo /etc/init.d/dphys-swapfile restart

~/opencv/opencv-3.4.3/build $ free

**Open cv 컴파일**

~/opencv/opencv-3.4.3/build $ time make -j4

~/opencv/opencv-3.4.3/build $ sudo make install

~/opencv/opencv-3.4.3/build $ cat /etc/ld.so.conf.d/\*

 ~/opencv/opencv-3.4.3/build $ sudo ldconfig

~/opencv/opencv-3.4.3/build $ sudo nano /etc/dphys-swapfile

~/opencv/opencv-3.4.3/build $ sudo /etc/init.d/dphys-swapfile restart

~/opencv/opencv-3.4.3/build $ free

~/opencv/opencv-3.4.3/build $ pkg-config --modversion opencv

~/opencv/opencv-3.4.3/build $ cd

~$ python

import cv2

cv2.\_\_version\_\_

~/opencv/opencv-3.4.3/build $ cd

~$ rm -rf opencv

출처 : <http://www.3demp.com/community/boardDetails.php?cbID=235>

**OpenCV 사람인식**

Open CV에서 사람을 인식하는 함수인 detectmultyscale을 매개변수로 이미지를 받는다.

Picamera #웹캠

Img = picamera.caputre\_continues() # 카메라로 찍은 사진

Result = cv.dtectmultyscale(picamera);

다음 이미지에서 사람의 얼굴이 있는 좌표를 구한다.

X = result[0][0]; # 얼굴의 x좌표 시작

maxX = result[0][0] +result[0][2]; # 얼굴의 x좌표 끝

Y = result[0][1]; #얼굴의 y좌표의 시작

MaxY = result[0][1] + result[0][3] # 얼굴의 y좌표 끝

해당 이미지의 얼굴에 사각형을 표시한다.

CV2.rectangle(IMG,(x,y),(maxx,maxy),(0,255,0),2) #매개변수 순서대로 이미지파일,(사각형 시작점x,y),(사각형 끝 x,y),사각형 선의 색,선의 굵기)

CV2.imshow(title, IMG)#2번째 매개변수는 본래의 이미지다.

**화면 정중앙에서** 사람의 얼굴 중앙까지의 거리

toX = ((x + maxx) – x)/2 +x; # 화면 중앙에서 얼굴중앙까지의 x 거리

toY = ((y + maxy) – y)/2 +y # 화면 중앙에서 얼굴중앙까지의 y 거리

전체적인 시스템

초음파 센서로 사람이 가까이 오는걸 파악하고 카메라를 작동 시킨다.

카메라 이미지에서 사람을 인식한 후 구글 비전api를 사용해서 사람의 감정을 분석요청한다.

사람 얼굴의 중앙과 화면중앙의 거리를 파악한 후 해당 x,y값 만큼 모터들을 작동시킨다.

Lcd 패널에 구글 비전 api를 통해 온 감정에 따라 미리 준비한 표정 이미지를 출력한다.