

自動レジの製作

恵美 圭輔 岸本 一希
倉田 晴 川島 駿作

1. 研究概要

私たちのメンバの一人の実家がケーキ屋を営んでいて、自動レジを導入出来たら楽で効率的だと思い、興味が沸き、自動レジの製作を行った。

2. 構造と動作

(1) 構造

今回、製作した自動レジの写真を写真-1に示す。また、構造は次のとおりである。

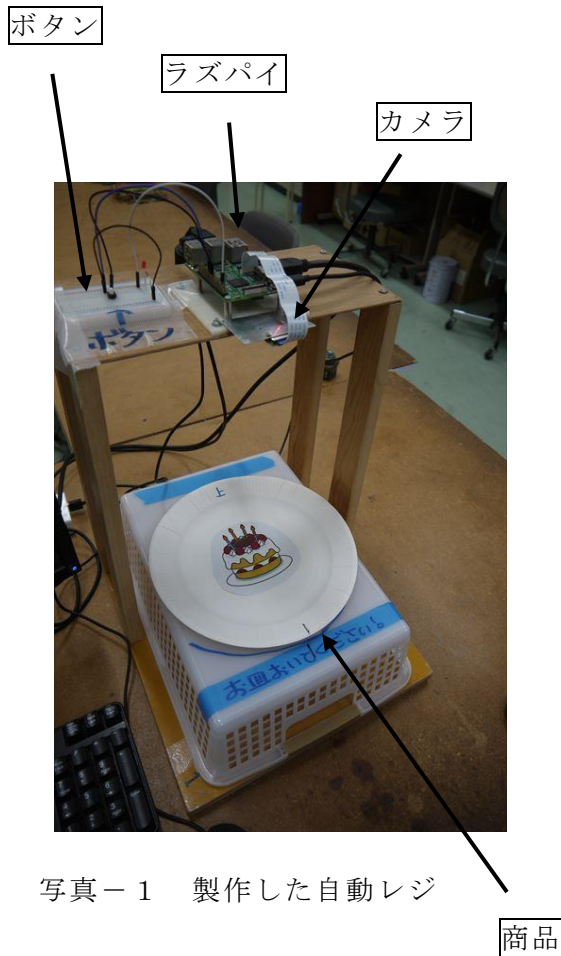


写真-1 製作した自動レジ

(2) 動作

動作は次の順序で行う。

① 写真撮影

お皿の上ののったケーキを撮影するためにボタンを押す。LED が点灯しているときボタンを押すとカメラが起動する。ブレッドボード上に配線したボタン2に示す。

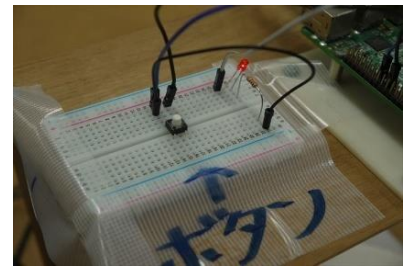


写真-2 ボタン

② ボタンが押されるとカメラにより5秒後に対象物（台の上の商品）を撮影する。



③ トリミングをする

OpenCV により写真の不要な部分をトリミングする。写真-3の左は撮影された全体、右は、それをトリミングしたデータである。

<トリミングとは>
写真画像やオブジェクトの不要な部分を切り取り、必要な部分のみを表示させる加工のことである。



写真-3 トリミングしたデータ

④画像認識

トリミングしたものを判別する。写真-4は、あらかじめ登録しておいたデータである。これから、トリミングした出たと一致する座標を openCV によって出力し、その座標から、どの商品であるかを判別している。



写真-4 登録しているデータ

⑤出力された座標から商品名と値段を表示する。図-1は表示される画面である。



図-1 表示される画面

⑥何かキーを押すと、写真撮影から繰り返される。

3. 構成部品

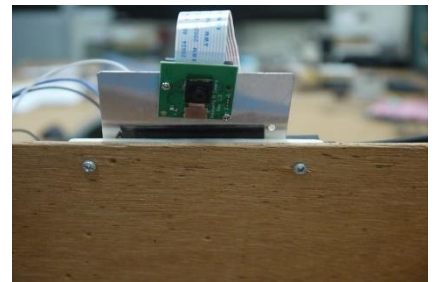
今回製作するために使用した部品とプログラムの詳細を示す。

(1) ラズベリーパイ
ラズパイ (Raspberry Pi) とは、英ラズベリーパイ財団が作ったマイコンボードのことである。一般的なパソコンと同様に CPU を搭載し、OS を利用できる機能を持ちながらも、1 万円以下という低価格で購入できることから人気を集めている。



(2) カメラモジュール

カメラモジュールは小型のものではスマートフォンやパソコンのカメラとして利用され、ATM または自販



機や券売機の顔認証用として利用される場合もある。比較的大型のものでは防犯カメラとしての利用や、車載用カメラ、検査・測定目的での産業用機器に搭載しての利用などがなされている。ラズパイ用も多数販売されている。

(3) ボタン

写真を撮るためにスイッチボタンを設置した。ブレッドボード上に配線を行った。図-2に回路図を示す。プルアップ回路とした。

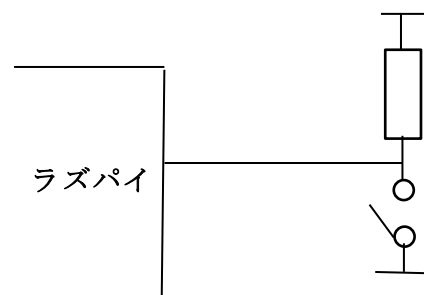


図-2 プルアップ回路

4. ソフトウェアについて

(1) Python

Pythonとは、人工知能を搭載したソフトウェアで、とりわけ機械学習を用いたソフトウェアの開発の分野で広く活用されている。コードが簡潔であるだけでなく、初めて機械学習を学ぶ人にとっても習得しやすい言語であるため、機械学習を学ぶ際には基本のプログラミング言語といえる。

(2) openCV

画像や動画の処理や、プログラムが画像や動画の中の特定の物の位置情報や動き、パターンなどを識別でき、画像の作成や編集、保存なども行えるオープンソースのコンピューター・ビジョン・ライブラリである

(3) 作成したプログラム類

作成したプログラムなどの説明を行う。

① ボタンを押すと LED が消える (LED の点灯時は起動状態)

```
GPIO.setmode(GPIO.BCM)
GPIO.setup(4,GPIO.OUT)
GPIO.output(4,GPIO.LOW)
while True:
    if(GPIO.input(9) == 0):
        GPIO.output(4, GPIO.LOW)
            ↑
        (ボタンが押されると LED が消える)
    sys.exit()
else:
    GPIO.output(4,GPIO.HIGH)
            ↑
        (それ以外の場合 LED が点く)
```

上記のプログラムを実行する (ボタンを押す) と LED が消え、カメラが起動する。

② 5 秒後に解像度を指定して撮影する命令

```
Sudo raspistill -w 800 -h 600 -o
test10.jpg
```

(上記のコマンドではカメラで撮影する解像度を指定する) -w...横,-h...縦の解像度

③ 写真の不要な部分をトリミングする

```
import cv2
img = cv2.imread ('test10.jpg')

# トリミング
img = img[70:300, 200:450]
    (トリミングをする座標の設定)
cv2.imwrite('test11.jpg', img)
cv2.waitKey(0)
```

上記のプログラムによって撮影した写真の不要な部分を切り取る。

④ トリミングした写真データを判別し座標を検出、登録した写真データの品物を判断する。

```
import cv2
import numpy as np
from matplotlib import pyplot as plt

img = cv2.imread('moto1.jpg',0)
img2 = img.copy()
template = cv2.imread('test11.jpg',0)
w, h = template.shape[::-1]

# All the 6 methods for comparison in a list
methods = ['cv2.TM_CCOEFF']

for meth in methods:
    img = img2.copy()
    method = eval(meth)

    # Apply template Matching
    res = cv2.matchTemplate(img,template,method)
    min_val, max_val, min_loc, max_loc = cv2.minMaxLoc(res)

    # If the method is TM_SQDIFF or TM_SQDIFF_NORMED, take minimum
    if method in [cv2.TM_SQDIFF, cv2.TM_SQDIFF_NORMED]:
        top_left = min_loc
    else:
        top_left = max_loc
    bottom_right = (top_left[0] + w, top_left[1] + h)
    print(top_left)
    print(bottom_right)

    cv2.rectangle(img,top_left, bottom_right, 255, 2)

    plt.subplot(121),plt.imshow(res, cmap = 'gray')
    plt.title('Matching Result'), plt.xticks([], plt.yticks([]))
    plt.subplot(122),plt.imshow(img, cmap = 'gray')
    plt.title('Detected Point'), plt.xticks([], plt.yticks([]))
    plt.suptitle(meth)
```

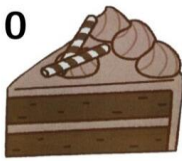

⑤ 図一 3 のように商品名と値段を表示する。

```
im = cv2.imread('decore2.jpg')
cv2.imshow('test',im)
cv2.waitKey()
cv2.destroyAllWindows()

#print(im.shape)
```

チョコケーキですね！

¥ 3 5 0



何か**キ**ーを押してください

図一 3

判別後写真がマッチングしたら商品名と値段を表示する。

⑥ 命令とプログラムを自動実行させる。

```
sudo python led.py
sudo raspistill -w 800 -h
600-o test10.jpg
sudo python tori.py
sudo python bunkasai.py
```

5. 製作風景と完成写真

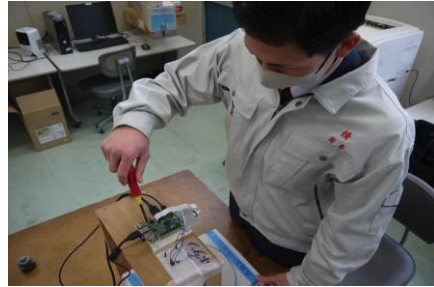
製作している過程の写真 5～8 に、完成した自動レジを写真一 9 に示す。



写真一 5 台の木材を切断



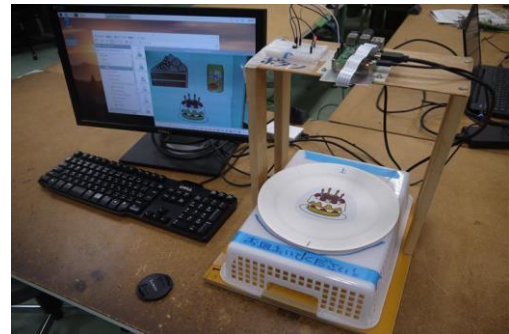
写真一 6 ブレッドボードに配線



写真一 7 台にラズパイを取付け



写真一 8 プログラム入力



写真一 9 完成した自動レジ

6. まとめ

自動レジを作ろうとしたときは、本当にできるのか不安であったが、なんとか完成できてホッとしている。Python でボタンの反応するプログラムや画面に表示するプログラム、カメラで撮影する命令など、インターネットを調べながら進めていった。製作を通して何かを作ることの難しさを学んだと共にとっても楽しかった。