

Laboratorium z przedmiotu Systemy wbudowane (SW)			
Karta projektu – zadanie 7			
Nazwa projektu:			
System automatycznego otwierania drzwi za pomocą rozpoznania twarzy			
Prowadzący: dr inż. Przemysław Zakrzewski	Autorzy 148057 150218	Grupa dziekańska:	16
		Ocena:	

Cel projektu

Celem projektu było przygotowanie aplikacji zdolnej do zapisu i późniejszej identyfikacji osoby na podstawie jej twarzy. Aplikacja ta mogłaby być użyta w systemach potrzebujących zabezpieczeń np. jako kontrola dostępu do pomieszczenia. Pozwoliłoby to na zwiększenie kontroli i bezpieczeństwa poprzez identyfikację unikalnych cech człowieka. W naszym projekcie rolę systemu potrzebującego zabezpieczeń pełni prosty Serwomotor, który symuluje działanie drzwi.

Projekt a realizacja

Projekt został zrealizowany za pomocą biblioteki opencv-python, która pozwala na przetwarzanie obrazu w czasie rzeczywistym. Przed uruchomieniem programu tworzymy lokalną bazę danych, która będzie zawierać informacje o cechach twarzy osoby, której chcemy przekazać dostęp do aplikacji. Następnie używając uczenia maszynowego trenujemy program na podstawie zdjęć osób. Tak zbudowaną bazę danych używamy w aplikacji, która analizuje obraz w czasie rzeczywistym korzystając z danych jakie przygotowaliśmy.

Początkowym planem było zbudowanie „ręki” robota, która mogła być sterowana za pomocą strony internetowej przez połączenie WI-FI. Dodatkowo robot byłby w stanie wykonywać prosty algorytm wieży Hanoi. Niestety wydrukowanie elementów wraz z kupnem wszystkich elementów wyniosłoby o wiele za dużo jak na kieszeń studenta :-(

Założeniem było zrealizowanie projektu za pomocą Raspberry PI wraz z kamerą Raspberry PI Camera HD, niestety przez obecną sytuację na rynku nie udało nam się dostać urządzenia w znośnej cenie, dlatego zmuszeni byliśmy do zrealizowania projektu na Arduino.

Najważniejsze fragmenty kodu

Faza uczenia się

```
for root, dirs, files in os.walk(my_face_dir):
    for file in files:

        # Sprawdź czy plik jest zdjęciem
        if file.endswith("png") or file.endswith("jpg"):

            # Pobierz ścieżkę absolutną pliku
            path = os.path.join(root, file)

            # Pobierz identyfikator użytkownika jako imię (ewentualnie nazwisko)
            label = os.path.basename(root).lower()

            # Przypisz identyfikator liczbowy do użytkownika
            if not label in label_id:
                label_id[label] = current_id
                current_id += 1
            ID = label_id[label]

            # Skonwertuj plik do gray scale image
            pil_image = Image.open(path).convert("L")

            # Skonwertuj plik do tablicy
            image_array = np.array(pil_image, "uint8")

            # Zidentyfikuj twarz na zdjęciu
            face = cascade.detectMultiScale(image_array)

            # finding the Region of Interest and appending the data
            for x, y, w, h in face:
                img = image_array[y:y+h, x:x+w]
                cv2.imshow("Test", img)
                cv2.waitKey(300)
                face_train.append(img)
                face_label.append(ID)

# Zapisz dane do pliku
with open("labels.pickle", 'wb') as f:
    pickle.dump(label_id, f)
```

Faza przetwarzania obrazu

```
while True:
    check, frame = video.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    face = cascade.detectMultiScale(gray, scaleFactor=1.2, minNeighbors=5)

    for x, y, w, h in face:
        face_save = gray[y:y+h, x:x+w]

        ID, conf = recognise.predict(face_save)
        if conf <= 30:
            print(ID)
            print(labels[ID])
            cv2.putText(frame, labels[ID], (x-10, y-10), cv2.FONT_HERSHEY_COMPLEX, 1, (18, 5, 255), 2, cv2.LINE_AA)
            frame = cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 255), 4)

    cv2.imshow("Video", frame)
    key = cv2.waitKey(1)
    if key == ord('q'):
        break
```

Faza komunikacji z Arduino

```
port = serial.Serial('COM4', 9600)
```

Konfiguracja połączenia z Arduino

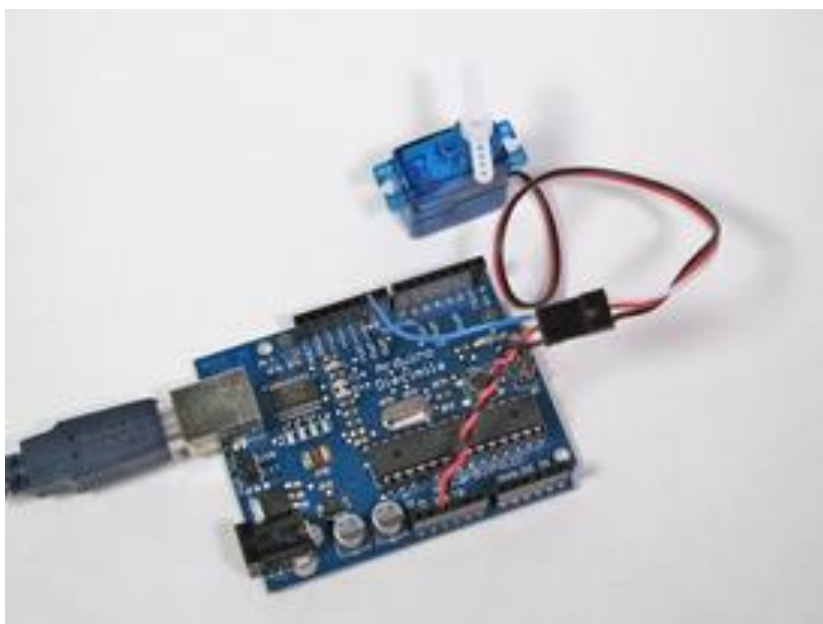
```
if len(face) == 0:
    port.write(str.encode('0'))
```

Gdy twarz nie została znaleziona

```
if ID == 0:
    port.write(str.encode('1'))
    print("ACCESS GRANTED")
else:
    port.write(str.encode('0'))
    print("ACCESS DENIED")
```

Gdy twarz została znaleziona

Połączenie



Podsumowanie

Podsumowując, projekt „System automatycznego otwierania drzwi za pomocą rozpoznania twarzy” pomimo prostoty w swoim wykonaniu może być przydatny dla osób, które poszukują dodatkowego zabezpieczenia dostępu. Dodatkowo program może być rozbudowany do komunikacji z serwerem. Dla większej elastyczności możliwe jest także przetransponowanie aplikacji na Raspberry PI (gdy ceny potanieją :)).