

〈금요일 발표〉

전기전자기초설계및소프트웨어실습

〈AI Term Project〉

담당교수 : 조용범

실험날짜 : 2021. 6 . 25

조 : 팀 37

조원 : 201711282 황현
201711247 이우용



1. Title

AI Term Project

2. Name

201711282 황현

201711247 이우용

3. Abstract

SVM 머신러닝을 이용하여 사람의 피부와 피부가 아닌 것을 구별하는 AI를 만들어 AVR보드에 이식하고, UART 통신을 통해 빛의 삼원색, RGB 값을 입력받는다.

입력받은 색이 사람의 피부인지 아닌지를 판별하여 AVR 보드의 7segment와 외부회로를 통해 그 판별값을 사용자가 확인할 수 있도록 한다.

4. Background

SVM 머신러닝은 libsvm 사이트에서 skin_noskin 데이터셋을 이용하여 학습을 수행하였다. 커널은 RBF의 경우 가장 높은 정확도를 보였기에 채택하였으며 감마는 0.001로 설정하였다. 파이썬을 이용하여 수행하였으며 이를 C++의 헤더파일로 변환, 이를 또 다시 C 소스파일로 변환하여 AVR보드에 이식하였다.

컴퓨터를 통한 입력없이 AVR 보드만으로 RGB 변수값을 입력받기 위하여 UART통신을 사용하였다. UART통신은 직렬 통신으로 BPS 9600, 데이터 타입은 1 start bit, 1 stop bit, no parity bit, 8 data bit 으로 설정하였다.

타이머 인터럽트를 사용하여 AI의 판별값을 10초간 AVR이 출력하도록 설정하였다. 타이머 인터럽트의 경우 오버플로우 인터럽트를 사용, 8분주로 scaling 하였다.

외부 인터럽트를 사용하여 putty로부터 RGB 변수값 입력의 안내를 받도록 설정하였다. 4번 인터럽트를 눌러 RGB변수들을 입력, 이를 통해 AVR 내부의 AI알고리즘이 입력받은 색이 사람의 피부인지 아닌지를 판별할 수 있다. 인터럽트의 경우 falling edge에 동작하도록 설정하였다.



5. Experimental Results

1. 실험1

A. Source Code

```
int main()
{
    init_port();           // led 설정 초기화
    init_interrupt();       // 외부 인터럽트 설정 초기화
    timer_setting();       // 타이머 인터럽트 설정 초기화
    init_uart();           // UART 통신 설정 초기화
    init_sound();          // sound 관련 설정 초기화
    int i;
    char tx, rx;
    int cnt;

    sei();                 // interrupt on

    float x_sample[3];

    while (1) {           // 무한 반복
        while (1) {       // getN이 0이면 반복
            wait();        // Tasmart4 --- 출력
            if (getN) {    // getN이 1이면 (RGB 변수를 putty로 부터 받아옴)
                getN = 0;  // getN을 0으로 바꾸고 (초기화)
                break;     // 반복문 탈출
            }
        }

        x_sample[0] = R;
        x_sample[1] = G;
        x_sample[2] = B;    // 받아온 RGB 변수들을 x_sample 배열에 저장

        a = predictLabel(x_sample); // 훈련시킨 알고리즘을 통해 a값 출력 (a가 0이면 사람피부, a가 1이면 사람피부가 아님)
        stopgo = 1;         // 타이머 인터럽트를 동작시키기 위한
        while (1) {

```

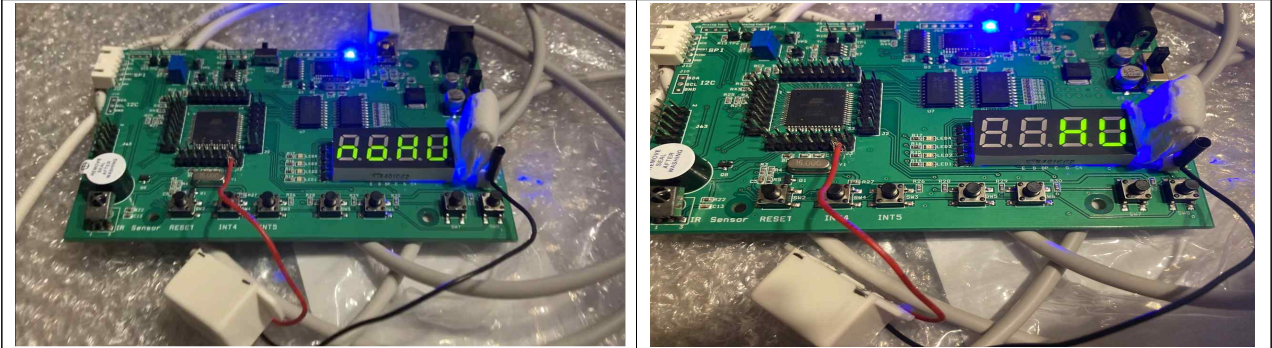
```
ISR(TIMER0_OVF_vect) {    // 타이머 인터럽트
    if (stopgo == 1) {    // stopgo가 1이면 다음의 동작들을 수행한다.
        temp++;           // temp 증가
        if (temp == t / 2) // 0.5초 경우 진행
        {
            if (a == 1)   // no skin 경우 진행
            {
                sound_on(); // 사운드 on
                temp++;
            }
            temp++;
        }
        if (temp == t) {    // temp=t, 즉 1초가 되면
            if (a == 1)     // a가 1(비사람피부)이면 실행
            {
                sound_off(); // 0.5초 뒤 사운드 오프
                temp = 0;
            }
            if (a == 0)     // skin 경우 진행
            {
                if (running_time == 0) // running_time 변수가 0 일경우 진행 (단한번)
                {
                    sound_on(); // 사운드 on
                    temp = 0;
                }
                else if (running_time == 1) // running_time 변수가 1 일경우 진행 (단한번)
                {
                    sound_off(); // 1초 동안 소리가 on 되며 한번만 진행 됨
                    temp = 0;
                }
            }
        }
    }
}
```

```
while (1) {
    if (a == 0)           //predict 결과가 0(사람피부)이면 hu출력 함수 실행
    {
        hu();
    }
    if (a == 1)           //predict 결과가 1(비사람피부)이면 ndhu출력 함수 실행
    {
        ndhu();
    }
    if (running_time == 10) break; // 타이머 인터럽트를 통해 10초가 지나게 되면 display를 중단
}
running_time = 0;        // running_time 변수 초기화
stopgo = 0;              // 타이머 인터럽트 중단
a = 3;                   // a=3으로 설정하여 아무런 동작도 하지 않도록 설정
return 0;
}
```

```
ISR(INT4_vect) {          // 4번 인터럽트 누르면 RGB putty로부터 받아오는 동작 수행
    _delay_ms(2500);      // 스위치 에러를 줄이기 위한 딜레이 함수
    get();                // RGB값을 putty로부터 입력받기 위한 get 함수
    getN = 1;             // 값을 받아왔으면 getN=1 하여 변수를 받아왔음을 알린다.
}
```



B. Data



https://konkukackr-my.sharepoint.com/:v/g/personal/s4291476_konkuk_ac_kr/Ec4L9RAnR1dBh-NGfxBdNLIBq1tIeLiRcgRY0e_zx4JiKg?e=1YBaWE

결과 동영상 링크

C. Discussion

사람 피부와 사람 피부 아닌 것을 구분하기 위해서 구현한 AI프로젝트이며, 사람 피부에 가까운 RGB의 색상을 대입하면 사진 오른쪽과 같이 “HU”휴면이라는 의미의 출력을 내고 사람 피부가 아닌 RGB값이 대입되면 왼쪽과 같이 “noHU”노 휴면이라는 의미의 출력을 세그먼트를 통해 내보낸다.

또한, 사람의 피부일 경우 처음 1초동안 외부회로(부저)가 동작되어 소리를 내고 사람의 피부가 아닐 경우 0.5초 간격으로 부저의 소리가 출력된다.

부저의 사운드를 통해서도 사람의 피부와 아닌 것을 구분할 수 있다.

위 링크의 동영상 결과에서도 볼 수 있으며, 3가지 색 대표적인 색상들을 대입하여 AI의 정확도가 나름 정확한 것을 확인 할 수 있다.



6. Analysis

skin_noskin 데이터셋은 약 25만개의 대용량의 데이터셋이기 때문에 동작 속도가 빠른 파이썬을 이용하더라도 머신러닝을 하는데 굉장히 많은 시간이 소요되었다. 또한 데이터셋을 테스트 하기 위한 test 파일이 따로 주어지지 않아 학습된 알고리즘을 삼중 for문을 통해 모든 RGB값들에 대해 정확도를 일일이 테스트하였다. 그렇기 때문에 수치화된 정확도를 얻지는 못하였으나 대략적으로 Blue 값이 Red 와 Green 보다 상대적으로 높을 경우 사물로 판별하였고, 반대로 Red와 Green 값이 Blue 값보다 상대적으로 높은 경우엔 사람의 피부색으로 판별하였다. 이는 실제로도 인간의 피부색과 비슷하였다. 학습은 RBF 커널, 감마 0.001을 사용한 경우에 가장 높은 정확도를 보였다.

7. Conclusion

외부 회로를 구현하기 위해서 본 과제에서 사용하지 않는 비어있는 GPIO D포트를 사용하여 사운드를 출력할 수 있었다.

타이머 인터럽트를 활용하여 0.5초의 시간동안 사운드가 출력되게 하여 사람이 아닐 경우의 오류 표현을 연출했다.

또한, 사람일 경우 1초 동안 사운드가 한 번 출력되어 반복적인 0.5초와 대비되게 안정적인 느낌을 연출해 주었다.

8. References

https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html#skin_nonskin

-> skin_noskin 데이터 셋을 다운받은 사이트

