//Kütüphaneleri import ediyoruz...

import processing.serial.\*; //Arduino uno için haberleşme kütüphanesi

import hypermedia.video.\*; //Kamera görüntüsünü almak için kütüphane

import java.awt.Rectangle; //dikdörtgen sınıfı --- yüz koordinatları için

OpenCV opencv; // OpenCV classından opencv adında bir nesne oluşturduk

//Kameranın görüntüsü için ekran ebatları

int genislik = 480;

int yukseklik = 360;

Serial port; //bağlantı noktası

char y\_motor = 95; //servo konumları için değişkenler

char x\_motor = 90;

char y\_motor\_ID = 0;// servoların kimlik IDleri

char x\_motor\_ID = 1;

int y\_yuz\_konum=0;//algılanan yüz konumları

int x\_yuz\_konum=0;

int y\_ekranOrta = (yukseklik/2);// dikeyde ekran orta noktası

int x\_ekranOrta = (genislik/2); // yatayda ekran orta noktası

int orta\_hata\_payi = 10; // kabul edilen hata payı

int adim=1;// adım sayısı

void setup() {

size( genislik, yukseklik );//oluşturulacak pencere ebatları

opencv = new OpenCV( this );

opencv.capture( genislik, yukseklik ); //video akışı

//frontal face kütüphanesinin adresi

opencv.cascade( "C:/Users/CodeREng/Documents/Processing/libraries/opencv/haarcascades/haarcascade\_frontalface\_alt.xml" );

port = new Serial(this,"COM7", 57600);//arduinonun bağlı olduğu port no, ve band genişliği

port.write(y\_motor\_ID); //y motorun kimlik nosu gonder

port.write(y\_motor); //y motorun pozisyonu gonder

port.write(x\_motor\_ID); //x motorun kimlik nosu gonder

port.write(x\_motor); //x motorun pozisyonunu gonder

}

public void stop() {

opencv.stop();

super.stop();

}

void draw() {

opencv.read();// çerçeve yakala

Rectangle[] faces = opencv.detect( 1.2, 2, OpenCV.HAAR\_DO\_CANNY\_PRUNING, 40, 40 );

image( opencv.image(), 0, 0 );//video resimlerini goster

noFill();// yüz bolgesini çerçeve içine almak için özellikler => dortgenin içi boş

stroke(255,0,0); //dörtgenin çerçeve rengi kırmızı

//çerçeve çizer

for( int i=0; i<faces.length; i++ ) {

rect( (faces[i].x)+5, (faces[i].y)+5, faces[i].genislik, faces[i].yukseklik );

}

if(faces.length > 0){//yüzün olup olmadığını kontrol etmek için

y\_yuz\_konum = faces[0].y + (faces[0].yukseklik/2);// bulunan yuzun y orta noktasını bul

x\_yuz\_konum = faces[0].x + (faces[0].genislik/2);// bulunan yuzun x orta noktasını bul

if(y\_yuz\_konum < (y\_ekranOrta - orta\_hata\_payi)){// bulunan yuzun ekranın yukarısında olup olmadığını kontrol et

if(y\_motor >= 5 || y\_motor <= 85 )y\_motor += adim; //ekranın yukarısındaysa açıyı arttır

}

else if(y\_yuz\_konum > (y\_ekranOrta + orta\_hata\_payi)){// bulunan yüzün ekranın aşağısında olup olmadığını kontrol et

if(y\_motor <= 175 || y\_motor >= 95)y\_motor -=adim; //aşağısındaysa açıyı azalt

}

if(x\_yuz\_konum < (x\_ekranOrta - orta\_hata\_payi)){ //bulunan yuzun ekranın solunda olup olmadığını kontrol et

if(x\_motor >= 5 || x\_motor <= 85)x\_motor += adim; //x motoru sola hareket ettirmek için 1 adım arttır.

}

else if(x\_yuz\_konum > (x\_ekranOrta + orta\_hata\_payi)){//bulunan yuzun ekranın sağında olup olmadığını kontrol et

if(x\_motor <= 175 || x\_motor >= 95)x\_motor -=adim; //x motoru sağa hareket ettirmek için 1 adım azalt.

}

}

port.write(y\_motor\_ID); //arduinoya y motor kimliğini gonder

port.write(y\_motor); //arduinoya y motor derecesini gonder

port.write(x\_motor\_ID); //arduinoya y motor kimliğini gonder

port.write(x\_motor); //arduinoya y motor derecesini gonder

delay(1); //saniyenin 1000 de 1 inde bunu tekrarla

}