import processing.video.\*;

Capture video;

int l = 0; //Länge der gesamten Möhre

int s = 0; //anzahl gespeicherter x koordinaten

int [] x = new int [3000]; //array für x koordinaten

int p =3;

//Setup für die Kamera und den Bildschirm

void setup() {

size(600, 200);

//Erstellt eine Liste der Namen der verfügbaren Kameras und gibt diese aus

String[] cameras = Capture.list();

if (cameras.length == 0) {

println("Keine Kameras verfügbar.");

exit();

} else {

println("Verfügbare Kameras:");

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

println(cameras[i]);

}

//Nimmt das Bild einer gewählten Kamera auf

video = new Capture(this, cameras[1]);

video.start();

}

}

//Funktion wird ausgeführt, sobald die linke Maustaste gedrückt wird.

//Nimmt sämtliche Messungen der Möhre vor, berechnet einzelne Werte und gibt die Ergebnisse anschließend aus

void mousePressed() {

println("");

println("");

println("----- Neuer Durchgang -----");

println("");

int a = 50; //deklariere länge der abschnitte, in denen breite gemessen wird, hier 20 pixel

int w = 50000; //deklariere gewünschtes Volumen (in Pixel^3, nicht cm^3)

//Länge der Möre

//prüft von links nach rechts auf einer bestimmten Höhe y im Bild, welche Farbe die Pixel haben und bestimmt somit die Gesamtlänge der Möhre und gibt diese danach aus

for (int i = 0; i < width; i++) {

if (get(i, 100) != -1) {

l += 1;

} else {

break;

}

}

println("Länge der Möhre: ", l, "Pixel");

int c = (l + a/2)/a; //anzahl der abschnitte

println("Anzahl der Abschnitte: ",c);

//Breite der Abschnitte

int [] d = new int [c];

int [] r = new int [c];

int ln = l-a/2; //stelle der ersten messung, kürzer als die anderen, erster schnitt nämlich bei a/2

for (int i = 0; i<c; i++){ //messe die breite in jedem abschnitt; hole messungen bis es so viele wie abschnitte sind

for(int k = 0; k < height; k++){

if (get(ln-a\*i, k) != -1) {

d[i] += 1;

}

}

r[i] = d[i]/2; //aus breite der werte nun den radius für leichtere rechnung

println("Radius",i,": ", r[i]);

}

int lr = l; //kopie der länge der möre, verwendet bei rechnung

double e = l-a/2; //entfernung des ersten schnittes von Halterung

for (int i = 0; i < c;i++){ //selber durchgang in jedem abschnitt

int l1=0;

int ls = w / p /r[i] / r [i]; //länge der schnitte

int as = a - l1/ ls; //Anzahl schnitte

//ort der schnitte

while (lr > e) {

lr = lr - ls;

x[s] = lr;

s++;

}

lr = l-a\*i; //sicher gehen, dass lr für nächsten abschnitt bereit ist

int rl = a - ( as \* ls ); //Restlänge

int vr = p \* r [i] \* r[i]; //Restvolumen

int v1 = w - vr; //Volumen 1. schnitt nächster abschnitt

l1 = v1 / p / r[i] \* r[i]; //länge 1. schnitt nächster abschnitt

lr = lr-l1;

x[s] = lr;

s++;

//double an = a - l1; //neue länge

}

}

//Gibt den einen Frame der Kamera aus, sobald dieser verfügbar ist und erzeugt somit ein flüssiges Bild auf dem Ausgabebildschirm

void captureEvent(Capture video) {

video.read();

}

//zeichnet sämtliche Ausgaben auf den Bildschirm

void draw() {

background(0);

image(video, 0, 0); //stellt das Video auf die gleiche Größe wie die Bildfläche

filter(THRESHOLD); //Filter für die Erkennung der Möhre auf dem Brett (Möhre wird schwarz und Hintergrund weiß dargestellt)

//Hilfslinie zum Ausrichten der Kamera

//stroke(204, 102, 0);

//line(0, 100, 600, 100);

}