Proiect la Proicesarea Imaginilor Universitatea Tehnica din Cluj-Napoca

# Tema proiectului:

# Detectarea formei si culorilor unor instrumente

# de scris si folosirea lor pentru desenare

## Studenta:

Nume: Vasilache Maria

Grupa 30238

-2023-

# Descrierea temei

Proiectul are ca scop implementarea unui paint foarte minimalist și simplu. Folosindu-ne de camera web vrem să afișăm o fereastra pe care să desenăm. Algoritmul detectează marker-ele pe care le vom arâta în fața camerei și va desena folosind culoarea marker-ului acele puncte (vor fi mai multe deorece noi vrem să desenăm traseul rezultat în urma mișcării marker-ului în fața ferestrei) care se vor afla în mijlocul liniei care definește muchia superioară a conturului.

1. Materiale necesare:

🡪Markere de diferite culori

🡪Camera web pentru a monitoriza activitatea instrumentelor de scris

🡪lumina buna

# Algoritmi pimplementati:

**Detectie contur**

Pantru detectarea conturului, m-am documentat din sursa [[1]](https://www.researchgate.net/figure/Detection-of-pen-tips-in-a-pencil-case-step-by-step-only-the-last-image-of-the-sequence_fig3_224253029)

Intrucat, cea mai amare parte a lucrarii presupune detectarea instrumentului de scris, in urma caruia se vor face actiunile de desenare, a fost important sa se detecteza bine, indifernt de alte obiecte care se afla in jurul nostru. Pentru acest aspect, m-am documentat din aceasta sursa, pentru **detectarea obiectelor indifferent de conturul din spate** [[2].](https://aws.amazon.com/blogs/machine-learning/streamlining-data-labeling-for-yolo-object-detection-in-amazon-sagemaker-ground-truth/)

**Detectie culoare**

Pentru detectarea culorii unui obiect, am avut drept support, materialul de la suersa [[3]](https://www.researchgate.net/figure/Detection-of-pins-in-a-pencil-case-Top-sequence-with-6-X-ray-images-298-298-pixels_fig5_224253029)

In cazul proiectului meu, am studiat Modelarea si simularea algoritmului de detective a obiectelor in Python, urmand sa fie transpus codul in C, folosind Open CV. Mai multe informatii in materialul din sursa [[4]](https://www.researchgate.net/publication/346003836_Modelarea_si_simularea_algoritmului_de_detectie_a_obiectelor_in_Python_pentru_Raspbery_Pi)

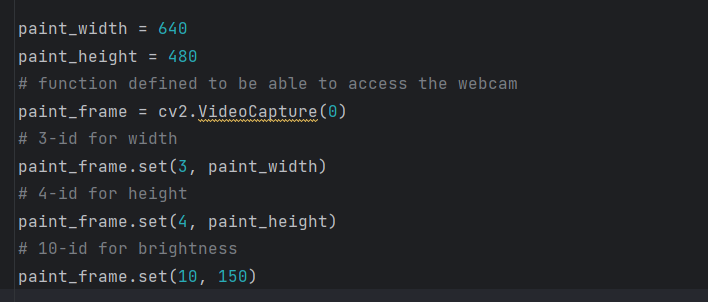
1. Implementare
   1. Activarea camerei

Pentru activarea camerei, se va folosi o functie din biblioteca CV2:



* 1. Creare ferestrei in care se va proiecta camera

Pentru a genera fereastra se va seta dimensiunea ferestrei, setand inaltimea si latimea ferestrei. Pentru acest lucru, se poate folosi functia **.set,**  atribuita variabilei in care s-a create conexiunea cu camera. 😊



* 1. Detectarea culorilor obiectelor

Culorile obiectelor care vor orientate spre camera, vor fi stocate intr-o lista, sub forma formatului HSV specific valorilor minime si maxime, intervak in care poate fi identificata culoarea.

Text

Description automatically generated

* 1. Culorile atribuite pentru colorarea pe ecran

Pentru a fi afisate pe ecran, foermatul acceptat, este BGR, Blue, Green, Red.

Text

Description automatically generated

4.5 Find Color():

Aceasta functie: def find\_color(img, color, color\_draw):

Primeste ca parametru imaginea in timp real care apare pe ecram si este furmizata de camera, vectorul de culori care se pot detecta si culorile cu care se va colora pe ecran, actualizandu-se imaginea rezu;tat.

In functia **draw\_on\_canvas(),** se va apela aceasrta functie, gasindu-se pe ecranul nostrum culoarea caracteristica obievtului.

In aceasta functie, **find color,** se expune algoritmul de gasire a culorii existente in instrumental de scris.

Pas1: Se iau valorile HSV stocate in vectorul color(culorile care este capabil sa le detecteze).

Pas 2: Se stocheaza valoarea instrumentului de scris

Pas 3: Se preia punctul in care se afla instrumental de pix( unde s-a gasit culoarea care existe in vectorul stocat)

Pas4: se adauga un nou punct, cu culoarea specifica, in poziti aculorii gasite

Pas 5: se returneaza lista noilor puncte

* 1. Get Contours:

Este o functie care primeste ca parametru imaginea.

Se va apela functia specifica dic cv2, findContours, care va prelua conturul obiectului current.

* 1. Functia Draw\_on\_canvas

Va desena pe panza, pixelii care s-au format anterior, sub forma de cerculete.

Pentru o desenare mai usoara, am folosit functia:

cv2.circle(img\_result, (point[0], point[1]), 10, color\_draw[point[2]], cv2.FILLED)

* 1. Main()

Fiindca acem imagine generate de camera in mod real, iar obiectele care vor fi prezentate camerei nu sunt aduse la perioade fixe de timp, ci vor fi aratate camerei, in mod prompt, aplicatia trebuie sa fie capabila sa raspunda in fiecare moment pentru a satisfice desenarea in timp real, prin urmare, am pus toata logica intr-un **while True,** in care se va cauta culoarea, pixelii de colorat, forma obiectului, si afisarea noii imagini impreuna cu punctele rezultate.

* 1. Exit

Penru terminarea executiri programului, se va aapasa tasta ‘a’

if cv2.waitKey(1) & 0xFF == ord('a'):  
 break

# Video prezentare:

Pentru prezenatra functionalitatii complete a proiectului, se poate urmari inrerigtratea de le uramtorul link:

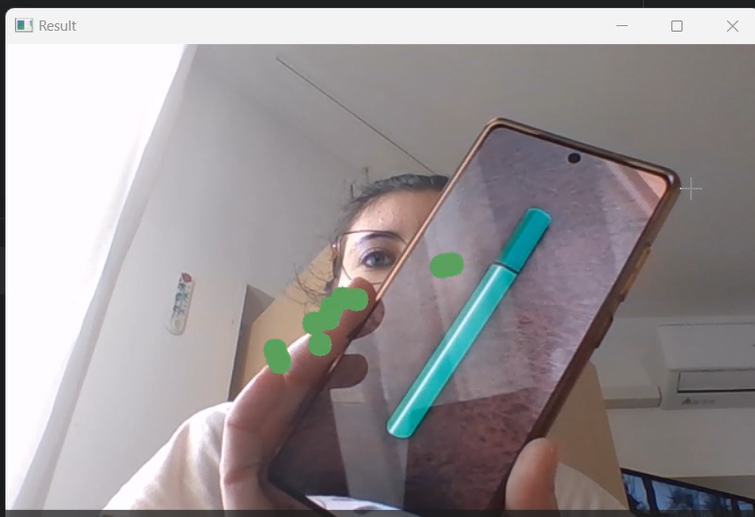
<https://drive.google.com/file/d/1WtxariGSFx3M_PA2dQVI9RjZzVxBWZ79/view?usp=share_link>

# Testare

Pentru testare, este important instrumental de scris utilizat si lumina abientala, astfel va exista situatia in care se va detecta greu obiectul, sau va fi detectat gresit.

In fisierul color detect, am catuata nivelele pentru HSV min si max pentru care se poate detecta culoarea din imaginea data.

1. Detectare marker verde

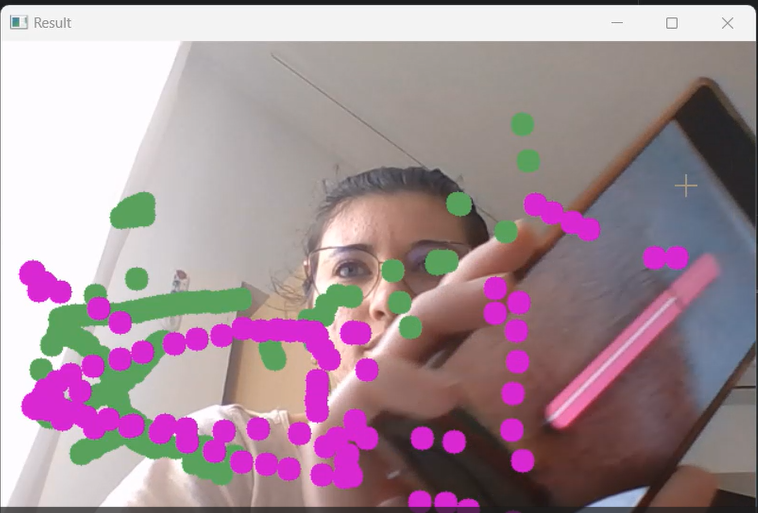


Determinarea valorilor HSV minime si maxime pentru detectareculorii verde:

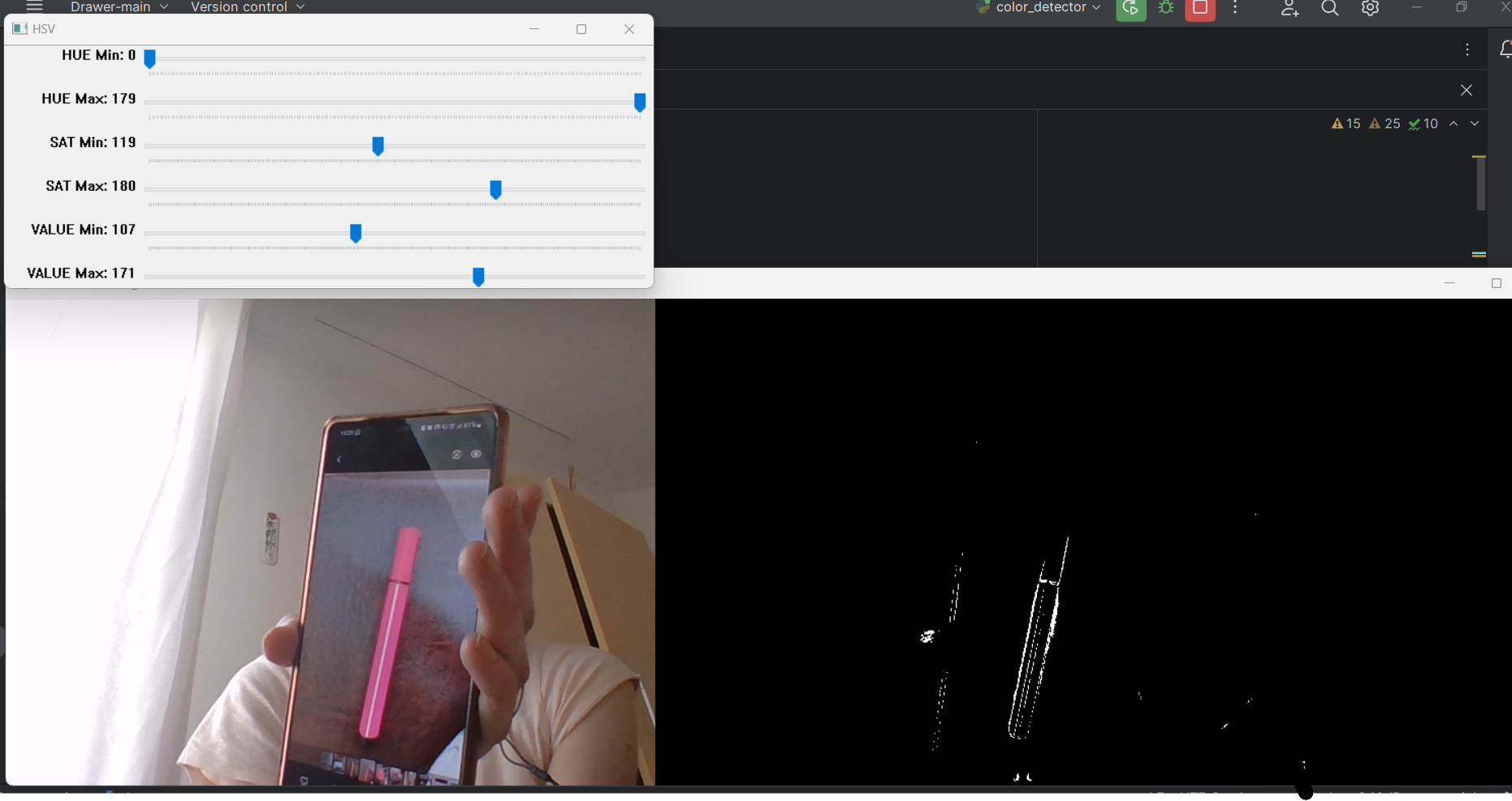
Graphical user interface, application

Description automatically generated

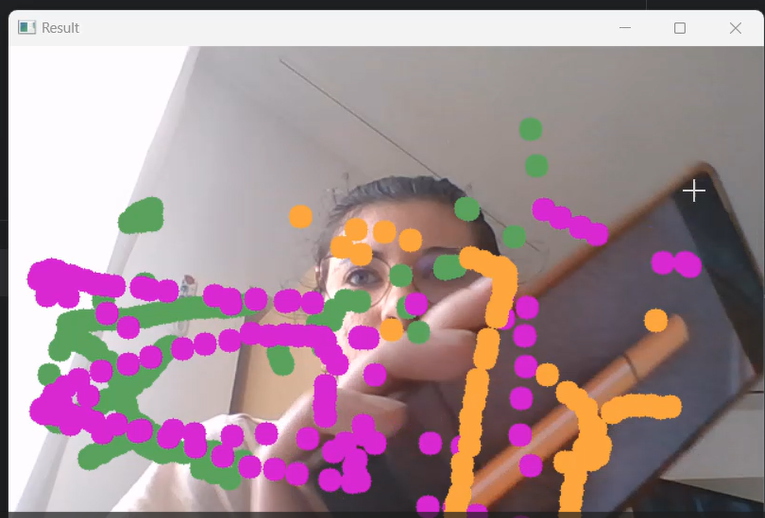
1. Detectare culoare magenta



Determinarea valorilor HSV minime si maxime pentru detectare magenta:



1. Culoare orange



Determinarea valorilor HSV minime si maxime pentru detectare culorii orange:



## 4. Bibliografie:

[1] <https://www.researchgate.net/figure/Detection-of-pen-tips-in-a-pencil-case-step-by-step-only-the-last-image-of-the-sequence_fig3_224253029>

[2] <https://aws.amazon.com/blogs/machine-learning/streamlining-data-labeling-for-yolo-object-detection-in-amazon-sagemaker-ground-truth/>

[3] <https://www.researchgate.net/figure/Detection-of-pins-in-a-pencil-case-Top-sequence-with-6-X-ray-images-298-298-pixels_fig5_224253029>

[4]<https://www.researchgate.net/publication/346003836_Modelarea_si_simularea_algoritmului_de_detectie_a_obiectelor_in_Python_pentru_Raspbery_Pi>

[5] <https://docs.opencv.org/3.4/da/d6a/tutorial_trackbar.html> (pentru reglarea nivelului HSV), in cazul in care vom avea intensitati diferite ale luminii in caera in care testam

[6] <https://docs.opencv.org/3.4/d8/dfe/classcv_1_1VideoCapture.html> [desenare pe ecran]

[7] <https://docs.opencv.org/3.4/d4/d73/tutorial_py_contours_begin.html> [gasire contur]

[8] <https://ro.planetalpha31.com/color-detection-using-python> [detectarea culorilor unui obiect]