



# **UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA**

---

*Interactiune om calculator*

*Detectie numar degete folosind markere de culoare*

---

Forogau Elena-Ana

Grupa: 30644

Profesor indrumator: Tiberiu Marita

## Cuprins

Descriere proiect: .....	3
Structura proiectului: .....	3
Rezultate .....	4
Implementare .....	6
Bibliografie .....	7

## Descriere proiect:

Proiectul are ca obiectiv dezvoltarea unui algoritm capabil să detecteze numărul de degete din imagini utilizând markere de culoare. Metoda utilizată se bazează pe procesarea imaginilor și segmentarea acestora conform unor modele de culoare create.

## Scopul proiectului:

Proiectul vizează:

- Construirea de modele de culoare care să descrie distribuția valorilor Hue (culoare) pentru markerele roșii, verzi, galbene și albastre.
- Segmentarea eficientă a imaginilor folosind aceste modele.
- Detectarea corectă a numărului de degete prin identificarea componentelor și eliminarea zgomotului.

## Structura proiectului:

### 1. Selectare ROI:

- Utilizatorul selectează manual regiuni reprezentative pentru fiecare culoare.
- Aceste regiuni sunt utilizate pentru a construi un model de culoare specific.

### 2. Construirea modelelor de culoare:

- Se calculează histogramele pixelilor din zonele selectate.
- Histogramele sunt procesate pentru a elimina zgomotul și pentru a calcula media și deviația standard a valorilor de culoare.

### 3. Segmentarea imaginilor:

- În imaginile de testare, algoritmul segmentează pixelii în funcție de modelul de culoare corespunzător.
- Dacă diferența dintre mediile modelelor de culori similare (ex.: două nuanțe de albastru) este mică, acestea sunt fuzionate într-un model unic.

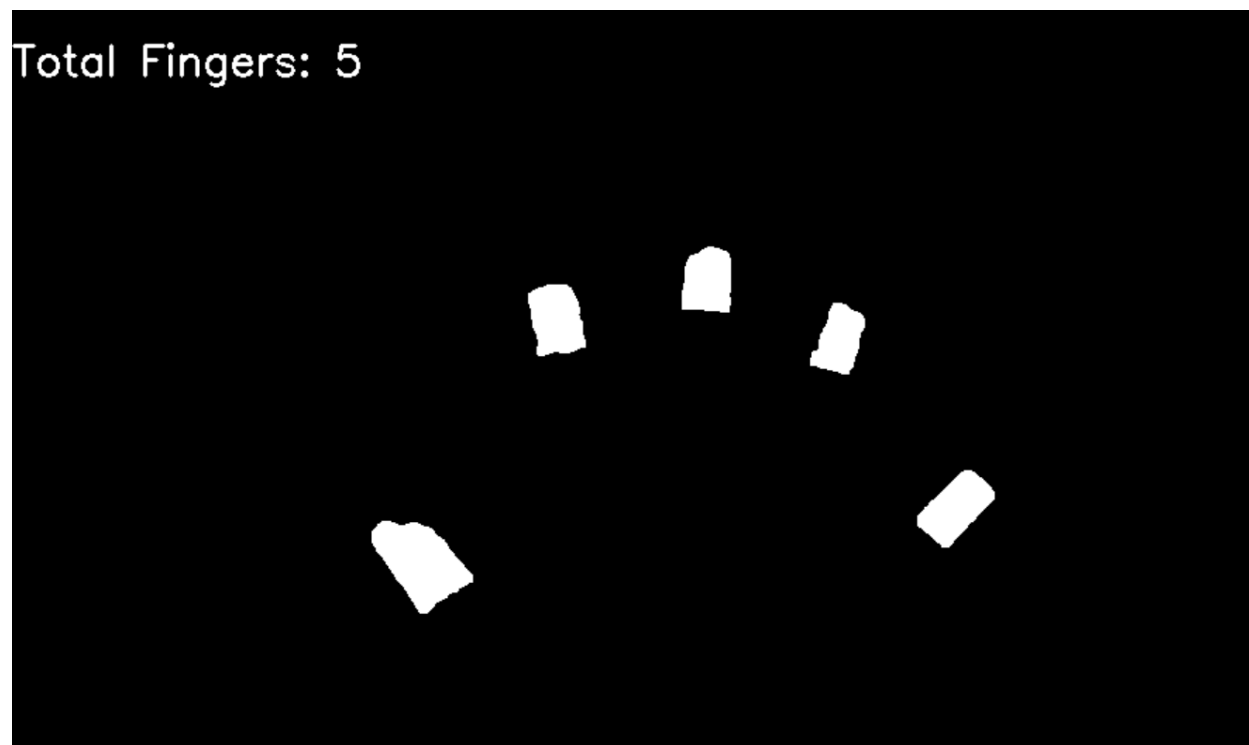
#### 4. Postprocesare:

- Se aplică operații morfologice (eroziune și dilatare) și eliminarea componentelor conexe mici pentru îmbunătățirea segmentării.
- Acest proces reduce zgomotul și elimină detecțiile false.

#### 5. Rezultate și afișare:

- Măștile binare rezultate sunt afișate pentru fiecare culoare.
- Numărul de obiecte detectate este calculat și afișat în imaginea finală.

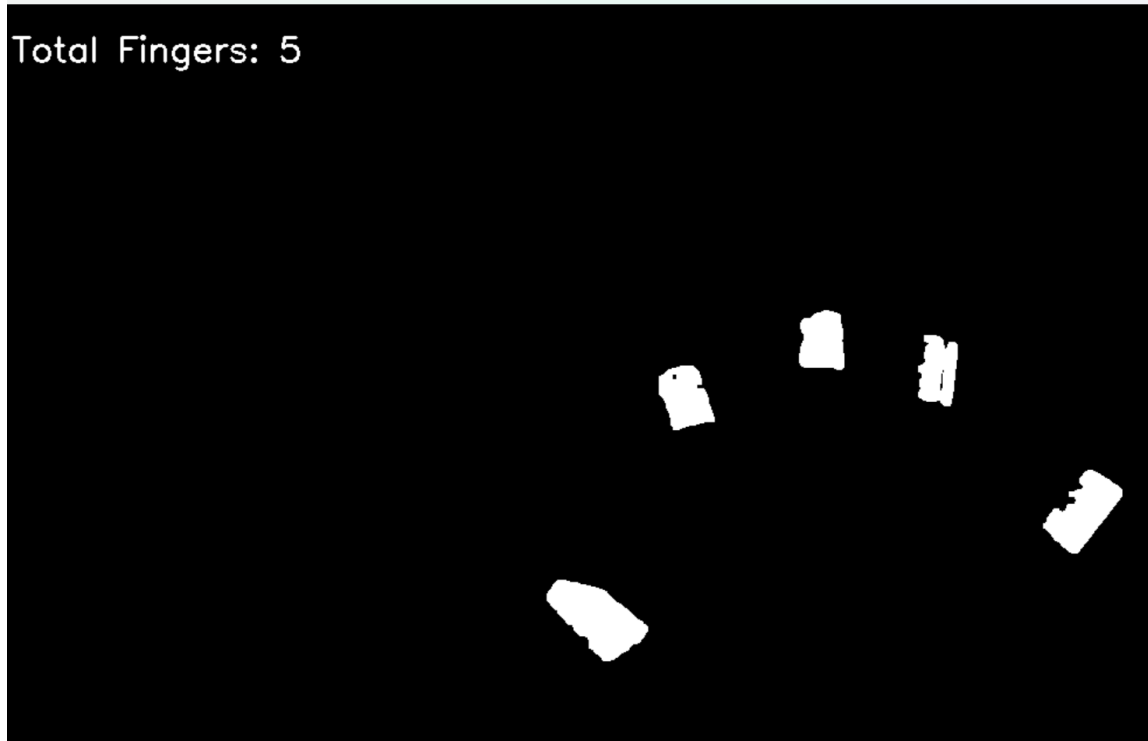
## Rezultate



Total Fingers: 5



Total Fingers: 5



```
Folosim modele separate pentru cele doua nuante de albastru.  
Nr degete detectate:  
Rosu: 1  
Verde: 1  
Galben: 1  
Albastru: 2  
Total degete: 5  
Folosim modele separate pentru cele doua nuante de albastru.  
Nr degete detectate:  
Rosu: 1  
Verde: 1  
Galben: 1  
Albastru: 2  
Total degete: 5  
Folosim modele separate pentru cele doua nuante de albastru.  
Nr degete detectate:  
Rosu: 1  
Verde: 1  
Galben: 1  
Albastru: 2  
Total degete: 5  
Folosim modele separate pentru cele doua nuante de albastru.  
Nr degete detectate:  
Rosu: 1  
Verde: 1  
Galben: 1  
Albastru: 2  
Total degete: 5  
Folosim modele separate pentru cele doua nuante de albastru.
```

Segmentarea s-a dovedit a fi eficientă pentru culorile selectate în imaginile de testare, cu un nivel redus de zgomot.

Algoritmul gestionează bine zgomotul prin filtrarea histogramelor și prin eliminarea componentelor conexe mici.

În imaginile cu zgomot semnificativ sau cu culori care se suprapun în spectrul Hue, algoritmul poate avea erori minore.

# Implementare

## 1. Selectarea manuală a ROI

Se folosește funcția `onMouseSelectROI`, care permite utilizatorului să selecteze o zonă dreptunghiulară cu mouse-ul.

Permite utilizatorului să definească regiuni reprezentative pentru fiecare culoare, care sunt apoi utilizate pentru a construi modele de culoare.

## 2. Construirea modelelor de culoare

Fiecare culoare este reprezentată printr-un model, `ColorModel`, care conține o histogramă globală, media și deviația standard a valorilor de culoare (Hue).

Pasi:

Se adaugă valorile din regiunea selectată în histograma globală folosind metoda `addToGlobalHistogram`.

Se filtrează histograma cu un filtru Gaussian pentru a elimina zgomotul.

Se elimină valorile mici din histogramă folosind metoda `removeNoiseFromHistogram`.

Se calculează media și deviația standard folosind metoda `calculateMeanAndStdDev`.

## 3. Segmentarea imaginilor

Segmentarea atribuie fiecărui pixel din imagine o etichetă (culoare), pe baza intervalului definit de media și deviația standard a fiecărui model.

Pasi:

Imaginea de intrare este filtrată cu un filtru Gaussian pentru a elimina zgomotul.

Intervalul de valori acceptat pentru Hue este determinat de:  $[\text{hue\_mean} - k \cdot \text{hue\_std} \dots \text{hue\_mean} + k \cdot \text{hue\_std}]$ , unde  $k$  este un factor (implicit  $k=2.5$ ).

Pixelii din acest interval sunt etichetați ca făcând parte din segmentul curent.

Se aplică operații morfologice (eroziune și dilatare) pentru a reduce zgomotul suplimentar.

## 4. Fuziunea modelelor de culoare (pentru nuanțele de albastru)

Dacă diferența între mediile celor două nuanțe de albastru este mică (mai mică decât un prag), se creează un model fuzionat pentru a simplifica segmentarea.

## 5. Numărarea obiectelor

Obiectele segmentate sunt numărate prin identificarea conturilor lor, utilizând funcția `findContours`.

## Concluzie

Proiectul a demonstrat că utilizarea modelelor de culoare pentru segmentarea imaginilor poate fi o metodă eficientă și accesibilă pentru recunoașterea gesturilor. Implementarea combină tehnici simple, precum analiza histogramelor și operațiile morfologice, pentru a obține rezultate robuste în detecția degetelor. Deși performanța generală este satisfăcătoare, precizia segmentării poate fi influențată de factori precum iluminarea, zgomotul sau suprapunerea spectrelor de culoare. Acest proiect evidențiază potențialul utilizării abordărilor clasice în procesarea imaginilor pentru diferite aplicații.

## Bibliografie

- . Laboratorul 2, Interacțiune Om-Calculator, Universitatea Tehnica din Cluj-Napoca
- . <https://docs.opencv.org/>
- . [Learn OpenCV by Examples: Applying Gaussian Filter](#)
- . <https://stackoverflow.com/>