***Facultatea de Automatica Si Calculatoare Specializarea Ingineria Sistemelor***

Disciplina: *Sisteme de Vedere Artificiala*

# Proiect

## Realizarea unui sistem de vedere artificială care să permită evaluarea direcţiei unei săgeţi

### Coteata Corin, 1401A

### Dinte Valentin, 1401A

1. **MATLAB GUI for Image Processing | with source code | Beginners to Advanced features** <https://www.youtube.com/watch?v=5k8RxErq6O8&ab_channel=Elementron>
2. **MATLAB Help Center**

# Cuprins:

## Pasii urmariti – 3

Lucrul in aplicatie – 3 – 5 Modificare cod GUI – 6 Functia “Sageti”- 6 - 10 GUI in functiune – 11-12

### Pasii urmariti:

#### Realizarea unui GUI pentru încărcarea imaginii cu una sau mai multe săgeţi

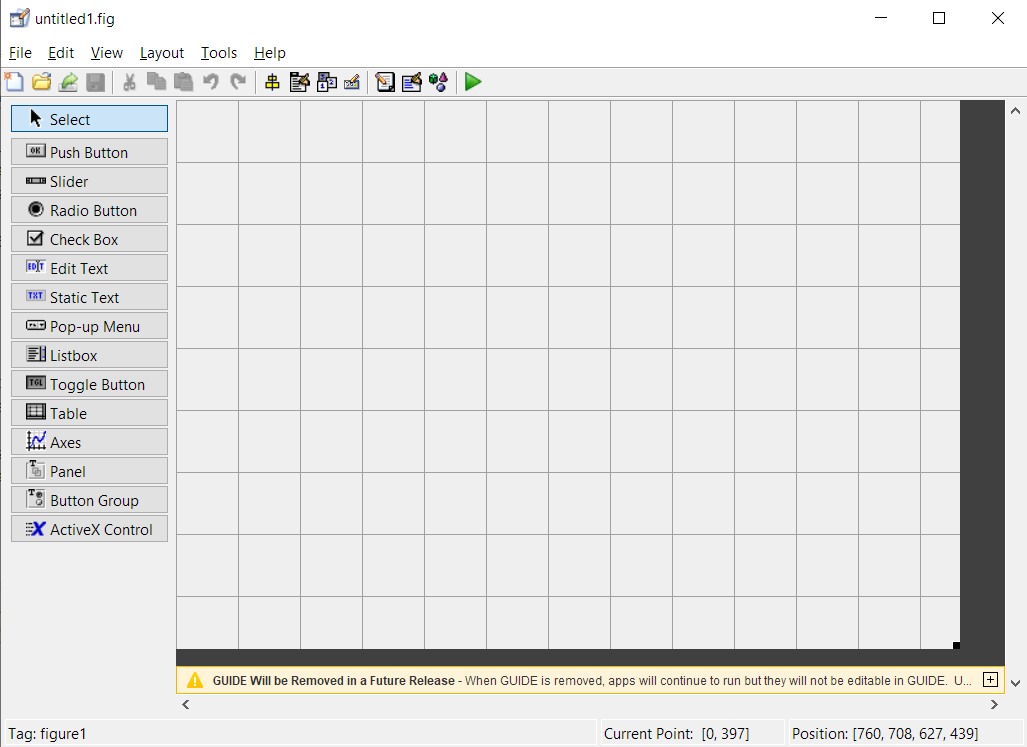
1. Analiza imaginii și proiectarea algoritmului

#### Afişarea direcţiei şi sensului fiecărei săgeţi

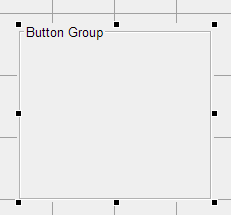
1. Vizualizarea rezultatelor in GUI

*1. Realizarea unui GUI pentru încărcarea imaginii cu una sau mai multe săgeţi*

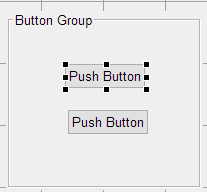
#### Aplicatia folosita:



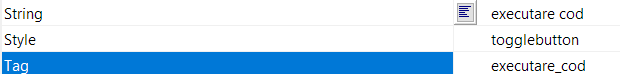
Implementam un grup de butoane:

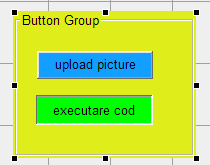


Inseram 2 butoane:

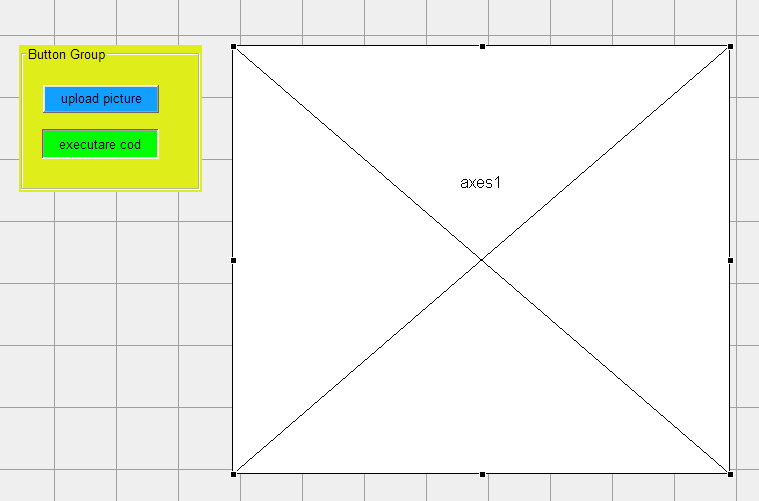


Inspectam fiecare buton in parte si schimbam valoarea string-ului si tag-ului in:





Asignam un “Axis” unde vor fi afisate rezultatele:



### Modificare codului din GUI.m:

1.Functia pentru *Upload Picture***:**

function upload\_picture\_Callback(hObject, eventdata, handles)

a = uigetfile('.png');// va prelua file-uri de tip .png a = imread(a); // dupa ce a gasit-o, o citeste axes(handles.axes1);// setarea handal-ui

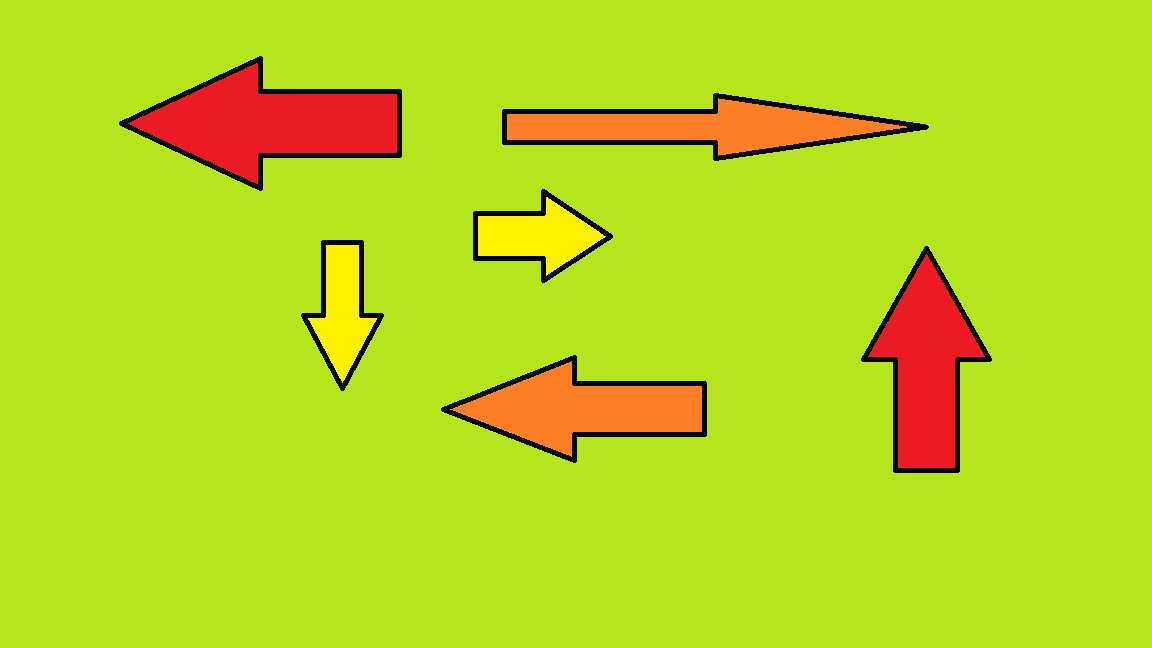
imshow(a); // afisare imagine setappdata(0,'a',a);// stocheaza datele in “a”

### Modificarea codului din Functie:

1.Functia pentru *Executare cod:*

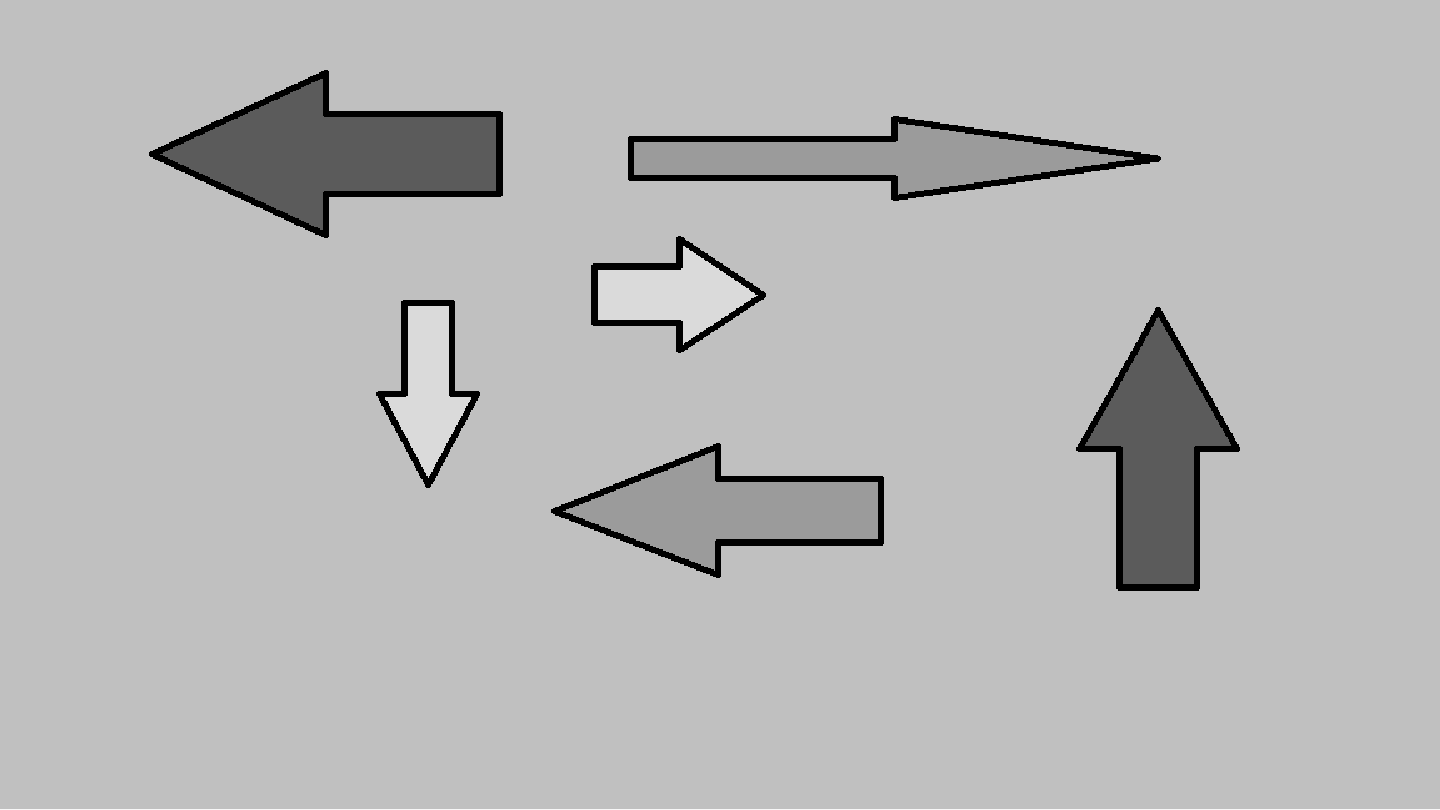
axes(handles.axes1); a = getappdata(0,'a');

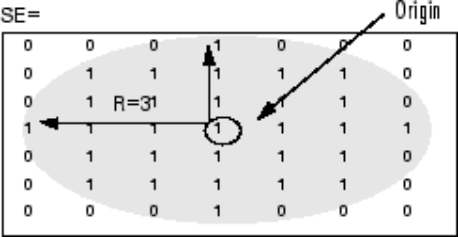
Sageti(a);// executarea functiei sageti

Functia **Sageti:** Imaginea initiala:

function Sageti(II)// initializare functie

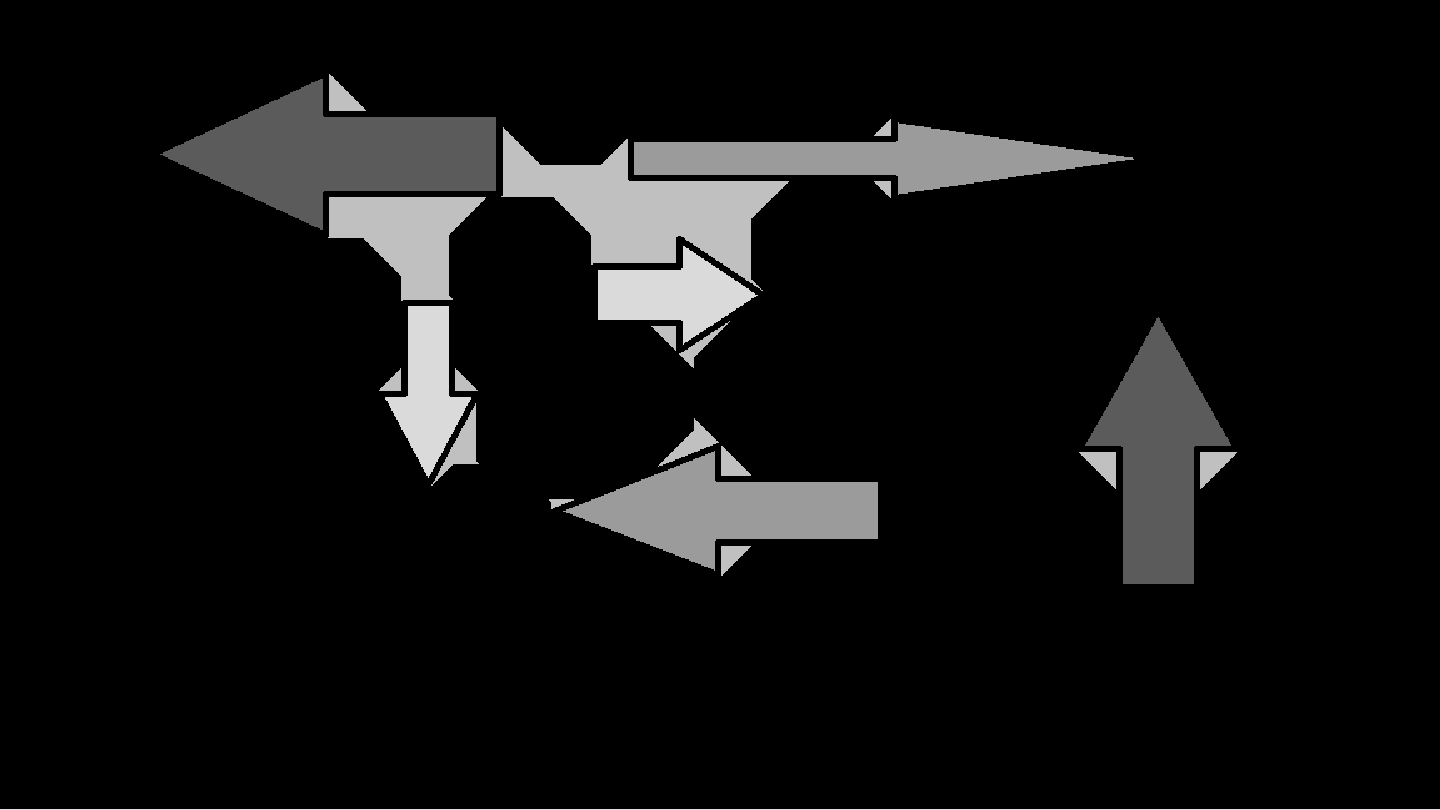
I = rgb2gray(II);// RGB->grayscale



se = strel('disk',55); // structuring element

background = imopen(I,se); // setam fundalul

I2 = subtract(I,background); // scoatem fundalul imaginei

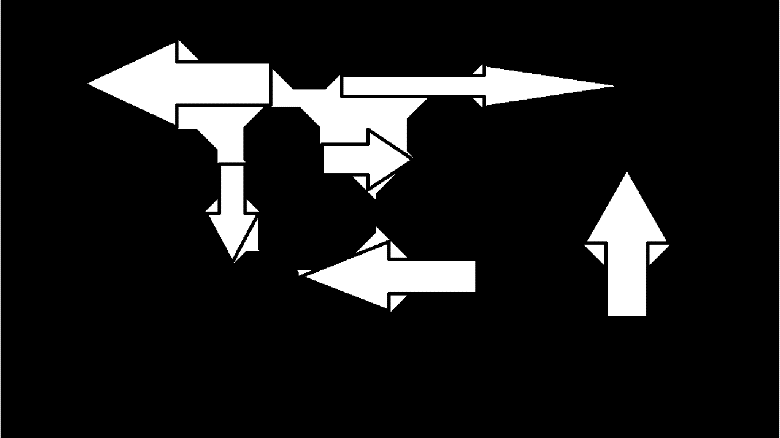


I3 = imadjust(I2, stretchlim(I2), []);// ajustam imaginea cu

// ajutorul lui “stretchlim (care ne ajuta sa setam

// luminozitatea“

level = graythresh(I3); // Binarizarea imaginii bw = imbinarize(I3,level);



[labeled,numObjects] = bwlabel(bw,8); // Determinarea

// numarului de obiecte din imagine si etichetarea

// regiunilor numObjects

arrowdata = regionprops(labeled,'all');// Afisarea numarului

// de obiecte

a=0; b=0;

//Selectarea doar sagetilor

// care fac un unghi cuprins intre -10 si 10 g cu axa X

for i=1:numObjects if(arrowdata(i).Orientation>-10 &&

arrowdata(i).Orientation <10)

// Algoritmul de detectie a sagetilor indreptate spre

// dreapta

if(arrowdata(i).Extrema(4,2)-arrowdata(i).Extrema(3,2)>-3 && arrowdata(i).Extrema(4,2)-arrowdata(i).Extrema(3,2)<3)

r=rectangle('Position', arrowdata(i).BoundingBox);

// Schimbarea conturului dreptunghiurilor in rosu set(r,'edgecolor','r')

// Numararea sagetilor indreptate spre dreapta a=a+1;

hold on

end end

end

//Selectarea doar sagetilor

// care fac un unghi cuprins intre -10 si 10 g cu axa X

for i=1:numObjects if(arrowdata(i).Orientation>-10 &&

arrowdata(i).Orientation <10)

// Algoritmul de detectie a sagetilor indreptate spre

// stanga

if(arrowdata(i).Extrema(7,2)-arrowdata(i).Extrema(8,2)>-3 && arrowdata(i).Extrema(7,2)-arrowdata(i).Extrema(8,2)<3)

r=rectangle('Position', arrowdata(i).BoundingBox);

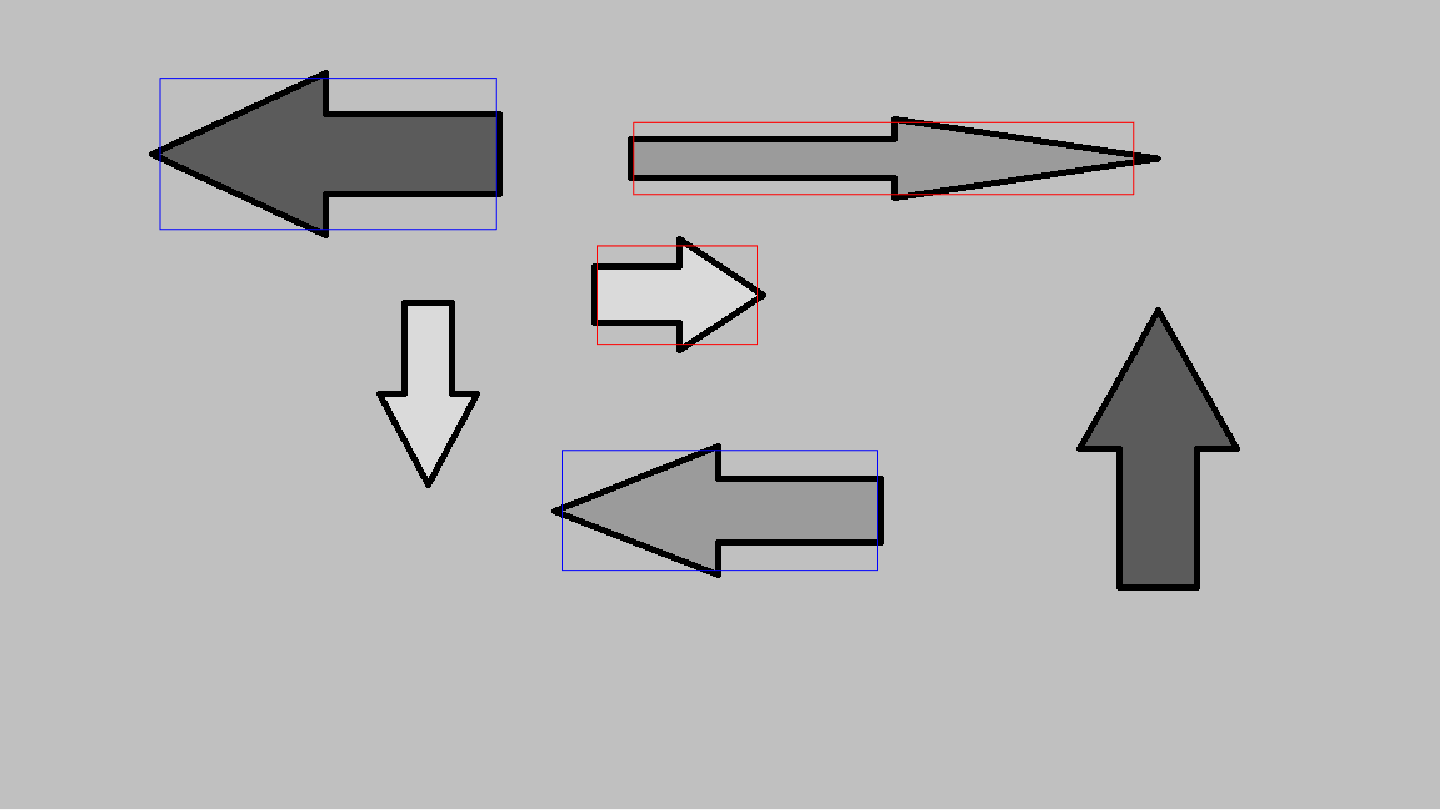
// Schimbarea conturului dreptunghiurilor in albastru set(r,'edgecolor','b')

// Numararea sagetilor indreptate spre stanga b=b+1;

hold on

end end

end



### GUI in functiune:

