

## Sessió 6 – Ferran Conde i Sergi Ibàñez

L'objectiu del laboratori és seleccionar un element ajudant-nos d'una discretització basada en una matriu de pesos, aplicant l'algorisme de flux màxim.

### **Imatge original**



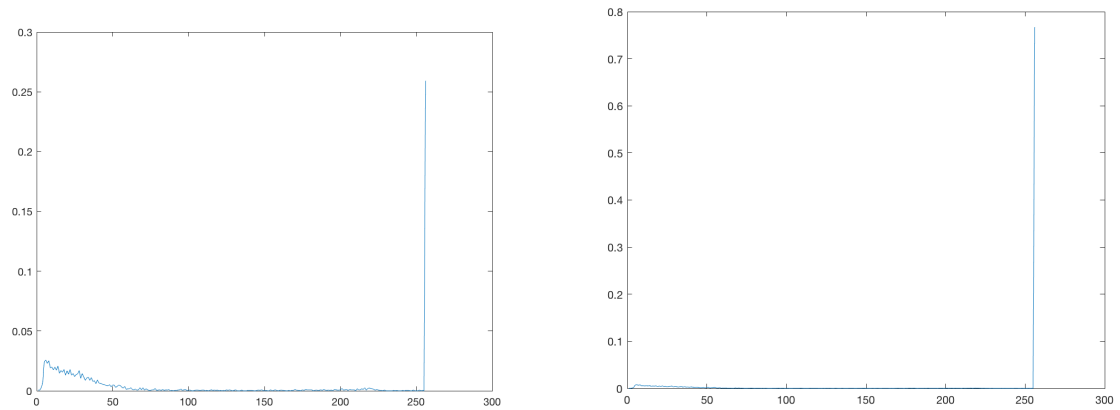
Lavors, seleccionem un rectangle:

### **Selecció**



Després de seleccionar un rectangle amb el gat, hem creat dos histogrames (de la imatge sencera i del rectangle) i els hem normalitzat.

### Histogrames normalitzats (imatge sencera, rectangle retallat)



Posteriorment, hem assignat pesos als píxels basant-nos en la probabilitat que el seu nivell de gris aparegui a l'histograma sencer, o a l'histograma de l'objecte a destacar.

Finalment, hem aplicat l'algorisme de flux màxim sobre la matriu de pesos construïda anteriorment per seleccionar l'objecte a destacar.

### Imatge seleccionada final



## Codi

```
%% Sessió 6 - Ferran Conde i Sergi Ibáñez

%% Seleccionem una imatge i en retallem un rectangle.

[file,path] = uigetfile('*.png'); % obre l'explorador d'arxius
I = imread(fullfile(path,file));
Irgb = rgb2gray(I);
Ir = imresize(Irgb,[100 120]); % ajust a una mida raonable
imshow(Ir);

rect = getrect;
[F, C] = size(Ir);

%% Creació dels histogrames
% Creem histogrames de la imatge sencera, del rectangle, i normalitzem
WM = zeros(F*C,F*C); % creació de la matriu de pesos (lligams entre pixels)
Ihist = imhist(Ir);
Rhist = imhist(imcrop(Ir, rect));
Ihist = Ihist / sum(Ihist);
Rhist = Rhist / sum(Rhist);

%% Assignem pesos
% Assignem pesos a píxels depenent de la probabilitat d'aparèixer
% a un histograma o a l'altre

for i = 1 : numel(Ir)
    Ib(i) = Ihist(Ir(i)+1) > Rhist(Ir(i)+1);
end

If = Ib == 0;

%Ib = els píxels que segurament són del background
%If = els píxels que segurament són del foreground

%% Definim lligams

% Definim (arbitràriament) el node (1,1) com a foreground
% i el node (F*C,F*C) com a background

WM(1,1:F*C) = If(:)'; % lligam entre el foreground (1,1) a tots els pixels
WM(1:F*C,1) = If(:); % la matriu ha de ser simètrica
WM(F*C,1:F*C) = Ib(:)'; % lligam entre el background (1,1) a tots els
pixels
WM(1:F*C,F*C) = Ib(:);
WM(1,F*C) = 0; % no hi ha d'haveri cap lligam entre el foreground i el
background
WM(F*C,1) = 0;
WM(F*C,F*C) = 0;
WM(1,1) = 0;

% lligam entre un pixel i els seus dos veïns
for c=2:C-1
    for f=2:F-1
        %diferència de ndg amb el veí de sota
        v = abs(Ir(f,c) - Ir(f+1,c)); % quant menys diferència més lligam
        v = max(0,min(1, 1 - v/32));
        WM(f+(c-1)*F,f+1+(c-1)*F) = v;
```

```

WM(f+1+(c-1)*F,f+(c-1)*F) = v; % la matriu ha de ser simetrica
%diferencia de ndg amb el vei del costat
v = abs(Ir(f,c) - Ir(f,c+1)); % quant menys diferencia mes lligam
v = max(0,min(1, 1 - v/32));
WM(f+(c-1)*F,f+c*F) = v;
WM(f+c*F,f+(c-1)*F) = v;
%diferencia de ndg amb la diagonal
v = abs(Ir(f,c) - Ir(f+1,c+1)); % quant menys diferencia mes lligam
v = max(0,min(1, 1 - v/32));
WM(f+(c-1)*F,f+1+c*F) = v;
WM(f+1+c*F,f+(c-1)*F) = v;
end
end

G = graph(WM);
[mf,~,foreground,background] = maxflow(G,1,F*C);

%% Imatge final
Ir(background) = 255; % Canviat fons a blanc ja que el gat era negre (i
aixi es veu mÈs facilment)
imshow(Ir,[]);

```