

Universitat Politècnica de Catalunya
VISIÓ PER COMPUTADORS

Informe Sessió 3

Carles Llongueras
Curs 2021-2022

Exercici1

Per aquest exercici de laboratori intentarem dissenyar un algoritme simple d'auto-enfoc. Per això utilitzarem una sèrie d'imatges e intentarem decidir quina es la que esta millor enfocada, entenent per enfocada la que contingui més “derivada” en la seva part central.



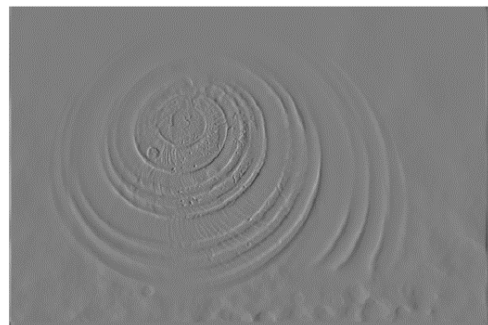
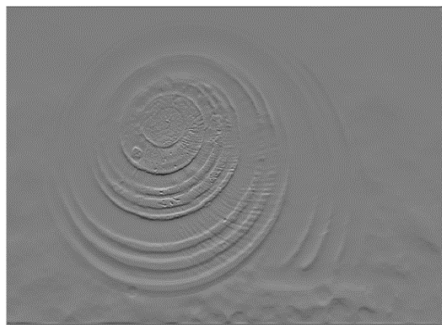
Implementació:

1. Primer llegirem les imatges i les convertirem a nivell de grisos

```
A = double(rgb2gray(imread('_61A5845.jpg')));  
B = double(rgb2gray(imread('_61A5855.jpg')));  
C = double(rgb2gray(imread('_61A5861.jpg')));
```

2. Passarem un filtre derivatiu de tipus “Sobel”, i el seu transposat per poder veure tant el contorns horitzontals com verticals. I l’aplicarem a les imatges.

```
hy = fspecial('sobel');  
hx = hy';  
AY = imfilter(A,hy);  
AX = imfilter(A,hx);  
BY = imfilter(B,hy);  
BX = imfilter(B,hx);  
CY = imfilter(C,hy);  
CX = imfilter(C,hx);  
imshow(AY,[]);  
imshow(AX,[]);
```



3. Com que ens es indiferent si el contorn es positiu o negatiu fem el valor absolut i el sumem.

```
AA = abs(AY) + abs(AX);  
AB = abs(BY) + abs(BX);  
AC = abs(CY) + abs(CX);  
imshow(AA, []);  
imshow(AB, []);  
imshow(AC, []);
```



Il·lustració 1 AA



Il·lustració 2 AB



Il·lustració 3 AC

Com es pot veure a simple vista la imatge AA té molt més detall en el seu centre que les altres dues, per tant segurament serà la millor enfocada.

4. L'últim pas serà crear un simple algorisme per veure quines de les fotografies esta mes enfocada.

```
Arow = size(AA,1);
Acol = size(AA,2);
Brow = size(AB,1);
Bcol = size(AB,2);
Crow = size(AC,1);
Ccol = size(AC,2);
Asigma = max([Arow Acol])/10;
Bsigma = max([Brow Bcol])/10;
Csigma = max([Crow Ccol])/10;
AG = fspecial('gaussian',[Arow Acol],Asigma);
BG = fspecial('gaussian',[Brow Bcol],Bsigma);
CG = fspecial('gaussian',[Crow Ccol],Csigma);
```

Primerament guardarem el nombre de files i de columnes que te cada fotografia. Aquestes dades les utilitzarem per dues coses, primerament per calcular la sigma la qual serà el mida de la finestra dividit entre deu (ja que volem que els pesos disminueixin ràpidament a partir que s'allunyin del centre de la imatge) i la pròpia finestra, que en aquets cas tindrà la mida sencera de la fotografia.

Per aquets exercici utilitzarem un matriu gaussiana ja que ens interessa donar mes importància als píxels que hi ha al centre de la imatge.

```
AA = imfilter(AA,AG);
AB = imfilter(AB,BG);
AC = imfilter(AC,CG);
ar = 0;
for i = int16(Arow/3):int16((Arow-Arow/3))
    for j = int16(Acol/3):int16((Acol-Acol/3))
        ar = ar + sum(sum(AA(i,j)));
    end
end
br = 0;
for i = int16(Brow/3):int16((Brow-Brow/3))
    for j = int16(Bcol/3):int16((Bcol-Bcol/3))
        br = br + sum(sum(AB(i,j)));
    end
end
cr = 0;
for i = int16(Crow/3):int16((Crow-Crow/3))
    for j = int16(Ccol/3):int16((Ccol-Ccol/3))
        cr = cr + sum(sum(AC(i,j)));
    end
end
```

Per últim apliquem el filtre a les imatges i recorrem cada píxel suma'n el pes de cada píxel, però per donar més pes als píxels interiors en comptes de recórrer tota la matriu de valors el que fem serà recórrer només els píxels que estiguin més en el centre, per exemple en aquest cas només s'han recorregut un 11% del píxels, aquells que corresponen a la part mes central de la imatge.

AM	BM	CM
3.2639e+06	3.1505e+06	1.7618e+06

Com es pot veure la imatge A es la guanyadora (cosa que és pot veure a simple vista) ja que es la que te més detall en la part central, sen així la fotografia millor enfocada.

Codi del exercici

```
A = double(rgb2gray(imread('_61A5845.jpg')));
B = double(rgb2gray(imread('_61A5855.jpg')));
C = double(rgb2gray(imread('_61A5861.jpg')));
hy = fspecial('sobel');
hx = hy';
AY = imfilter(A,hy);
AX = imfilter(A,hx);
BY = imfilter(B,hy);
BX = imfilter(B,hx);
CY = imfilter(C,hy);
CX = imfilter(C,hx);
imshow(AY,[]);
imshow(AX,[]);
AA = abs(AY) + abs(AX);
AB = abs(BY) + abs(BX);
AC = abs(CY) + abs(CX);
imshow(AA,[]);
imshow(AB,[]);
imshow(AC,[]);

Arow = size(AA,1);
Acol = size(AA,2);
Brow = size(AB,1);
Bcol = size(AB,2);
Crow = size(AC,1);
Ccol = size(AC,2);
Asigma = max([Arow Acol])/10;
Bsigma = max([Brow Bcol])/10;
Csigma = max([Crow Ccol])/10;
AG = fspecial('gaussian',[Arow Acol],Asigma);
BG = fspecial('gaussian',[Brow Bcol],Bsigma);
CG = fspecial('gaussian',[Crow Ccol],Csigma);

AA = imfilter(AA,AG);
AB = imfilter(AB,BG);
AC = imfilter(AC,CG);

AM = 0;
for i = int16(Arow/3):int16((Arow-Arow/3))
    for j = int16(Acol/3):int16((Acol-Acol/3))
        ar = ar + sum(sum(AA(i,j)));
    end
end
BM = 0;
for i = int16(Brow/3):int16((Brow-Brow/3))
    for j = int16(Bcol/3):int16((Bcol-Bcol/3))
        br = br + sum(sum(AB(i,j)));
    end
end
CM = 0;
```

```
for i = int16(Crow/3):int16((Crow-Crow/3))  
    for j = int16(Ccol/3):int16((Ccol-Ccol/3))  
        cr = cr + sum(sum(AC(i,j)));  
    end  
end
```