
E16 Descripció de regions

Table of Contents

1. Proves inicials	1
2. Descriptors de Fourier	2
3. Observar si es necessari obtenir tots els Descriptors de Fourier	3
4. Provem amb diferents nivells de descriptors	4

1. Proves inicials

```
im=imread('head.png');  
imshow(im),title('imatge entrada')  
  
im=imresize(im, 0.5);  
figure,imshow(im),title('imatge reescalat')  
  
area=sum(im(:));  
  
ero=imerode(im,strel("disk",1));  
cont=imsubtract(im,ero);  
figure,imshow(cont),title('imatge contorns')  
  
perim=sum(cont(:));  
C=4*pi*area/perim/perim;  
  
% Comanda per obtenir dades de la imatge  
Dades=regionprops(im,'all');  
  
[fila,col]=find(cont,1);  
B=bwtraceboundary(cont,[fila,col],'E');
```





2. Descriptors de Fourier

```
% Volem comprovar que la transformada de fourier es completament reversible
mig=mean(B);
```

```
% Primer obtenim els descriptors
```

```
Bc(:,1)=B(:,1)-mig(1);
```

```
Bc(:,2)=B(:,2)-mig(2);
```

```
s=Bc(:,1)+i*Bc(:,2);
```

```
% Aplicam fourier per obtenir l'espectre
```

```
z=fft(s);
```

```
figure,plot(abs(z)),title('espectre')
```

```
figure,plot(log(abs(z))),title('espectre en escala logarítmica')
```

```
% Ara que tenim la info de fourier, apliquem la funció inversa
```

```
ss=ifft(z);
```

```
% Ara que tenim la info original la apliquem a una imatge artificial i
```

```
% comprovem que ha funcionat
```

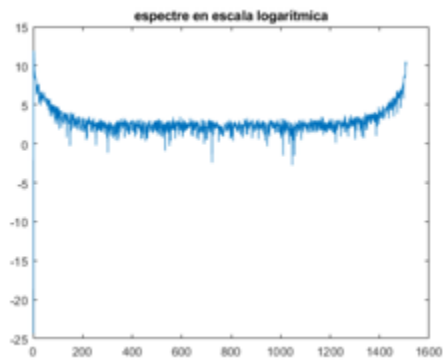
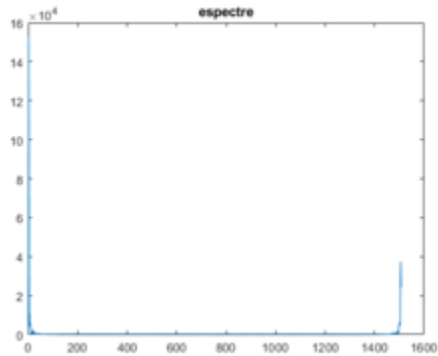
```
files=round(real(ss)+mig(1));
```

```
cols=round(imag(ss)+mig(2));
```

```
aux=zeros(size(im));
```

```
aux(sub2ind(size(aux),files,cols))=1;
```

```
figure,imshow(aux),title('imatge original recuperada')
```

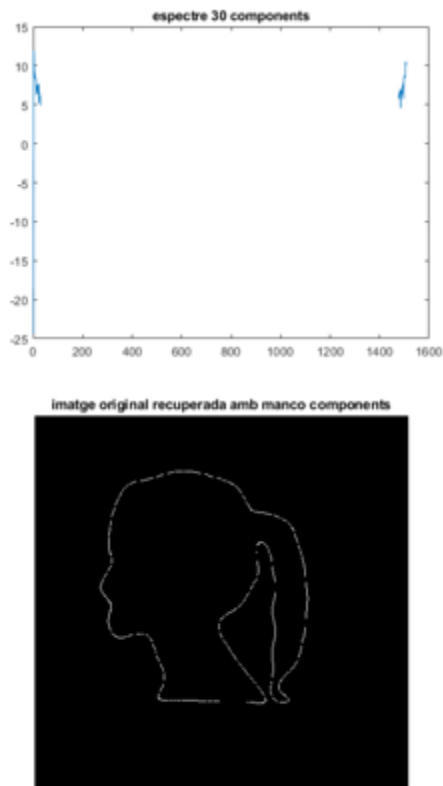


3. Observar si es necessari obtenir tots els Descriptors de Fourier

```
% Volem comprovar que passa si agafem només 30 descriptors  
N=30;  
tmp=z;  
tmp(N+1:end-30)=0;  
figure,plot(log(abs(tmp))),title('espectre 30 components')
```

```
% Ara apliquem la inversa i comprovem que surt
mida=500;
ss2=ifft(tmp);
files2=round(real(ss2)+250);
cols2=round(imag(ss2)+250);
aux=zeros(mida);
aux(sub2ind(size(aux),files2,cols2))=1;

% Podem comprovar que la imatge resultat es semblant però no igual, ja que
% no teniem tota la info
figure,imshow(aux),title('imatge original recuperada amb manco components')
```



4. Proven amb diferents nivells de descriptors

Configuración inicial

```
mida = 500; % Tamaño de la matriz auxiliar
valores_N2 = 20:20:100; % Valores de N2 a iterar
resultados = cell(length(valores_N2), 1); % Para guardar resultados

% Iterar sobre diferentes niveles de N2
for i = 1:length(valores_N2)
    N2 = valores_N2(i); % Valor actual de N2
    tmp2 = z; % Copiar datos originales

    % Reducir los componentes espectrales
```

```
tmp2(N2+1:end-N2) = 0;

% Mostrar el espectro con N2 componentes
figure, plot(log(abs(tmp2))), title(['Espectre amb ', num2str(N2), ' components']);

% Transformada inversa
ss3 = ifft(tmp2);

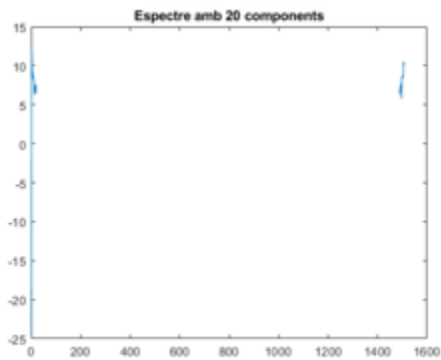
% Reconstruir las coordenadas
files3 = round(real(ss3) + 250);
cols3 = round(imag(ss3) + 250);
aux2 = zeros(mida); % Crear matriz auxiliar vacía

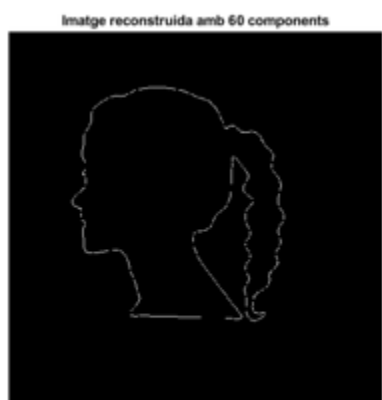
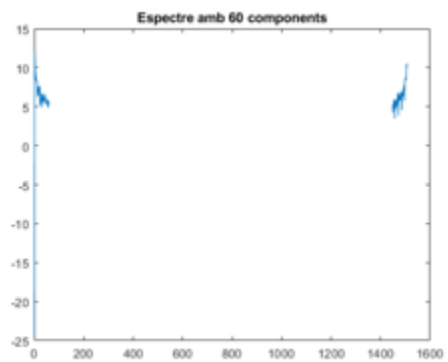
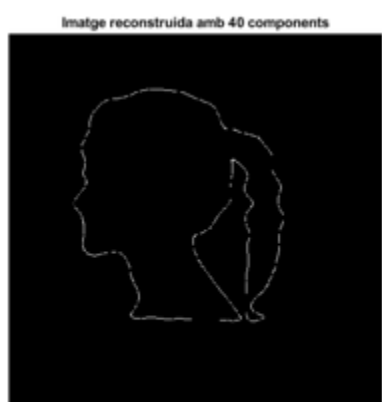
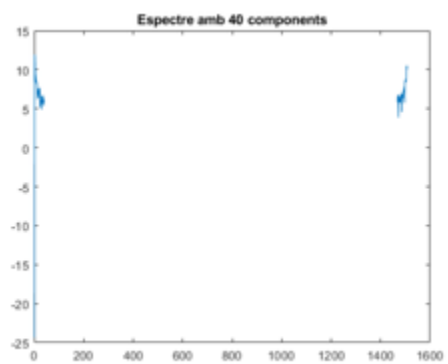
% Asignar valores en la matriz reconstruida
aux2(sub2ind(size(aux2), files3, cols3)) = 1;

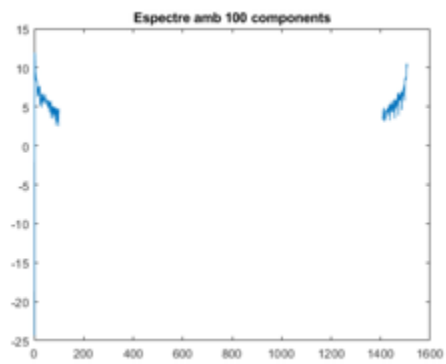
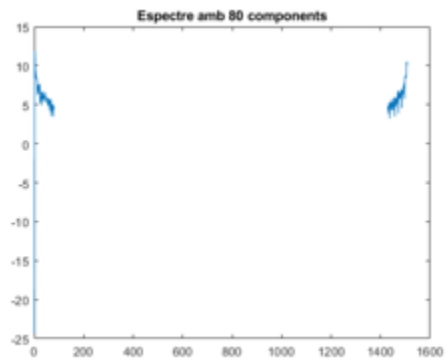
% Guardar el resultado en la matriz de resultados
resultados{i} = aux2;

% Mostrar la imagen reconstruida
figure, imshow(aux2), title(['Imatge reconstruida amb ', num2str(N2), ' components']);
end

% La matriz "resultados" contiene las imágenes reconstruidas para cada N2.
```







Published with MATLAB® R2024b