

Laboratorio Sperimentale di Matematica Computazionale / / Mimosa con rumore (F3)

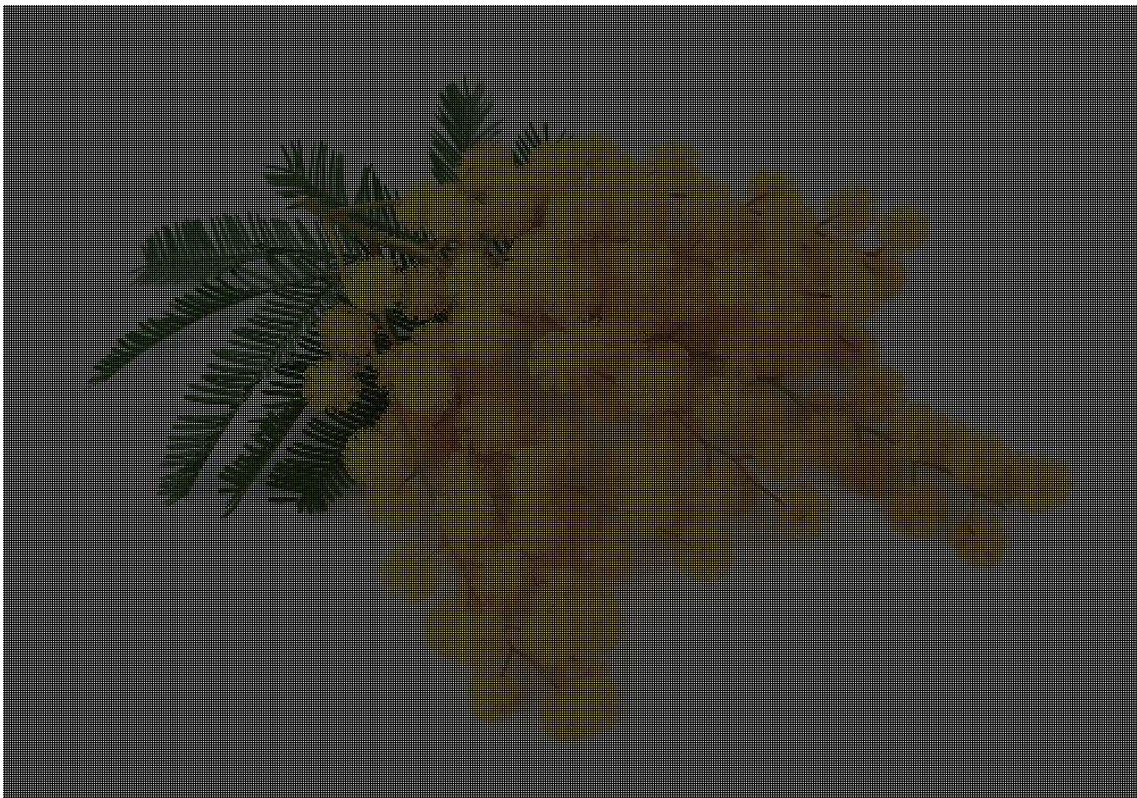
Dario Rancati - 539365

14 giugno 2018

Lo scopo della sperimentazione corrente è quella di applicare il metodo di filtraggio immagini mediante FFT studiato a lezione con i filtri f_j seguenti:

$$f_j = 1 - \cos\left(\frac{3\pi}{2} + \frac{2j\pi}{n}\right) \quad j = 0, \dots, \frac{n}{2}$$

Che, come si vede dal loro grafico riportato nelle slide, sembrano dei filtri "passa medio", che filtrano sia frequenze alte che basse. L'immagine a cui vogliamo applicare il filtraggio è la seguente:



La function che effettua il filtraggio in funzione dei generici filtri f_1 e f_2 è la seguente: l'abbiamo ottenuta modificando leggermente la `filtra` delle slides, facendo sì che il filtraggio avvenisse per ciascuna delle tre componenti dello standard rgb:

```
1 function W = filtra_rgb(V, f1, f2)
2   % calcolo fft di righe e colonne
3   V = double(V);
4   u1 = fft2(V(:,:,1));
5   u2 = fft2(V(:,:,2));
```

```

6   u3 = fft2(V(:,:,3));
7   U(:,:,1)=u1;
8   U(:,:,2)=u2;
9   U(:,:,3)=u3;
10  %qui abbiamo applicato la fft a ciascuna delle tre coponenti dello
11  %standard rgb
12  % filtro e antitrasformo sempre secondo lo standard rgb
13  Z1=diag(f1)*U(:,:,1)*diag(f2);
14  Z2=diag(f1)*U(:,:,2)*diag(f2);
15  Z3=diag(f1)*U(:,:,3)*diag(f2);
16  Z(:,:,1)=Z1; Z(:,:,2)=Z2; Z(:,:,3)=Z3;
17  w1 = ifft2(Z(:,:,1));
18  w2 = ifft2(Z(:,:,2));
19  w3 = ifft2(Z(:,:,3));
20  W(:,:,1)=w1;
21  W(:,:,2)=w2;
22  W(:,:,3)=w3;
23  % ripulisco
24  W = real(W); % tolgo eventuale roundoff immaginario
25  W = uint8(W); % trasformo variabile double in intera
26  % riportando i valori tra 0 e 255

```

Il seguente è lo script che esegue il filtraggio dopo aver calcolato i filtri f_1 ed f_2 . Notiamo che le proprietà di parità del coseno ci consentono una scrittura molto compatta dei vettori, che non richiede nemmeno di separare i casi per parità/disparità. Abbiamo aumentato la luminosità sommando una matrice di tutti 1, in quanto il metodo usato nelle slides schiariva decisamente troppo:

```

1  X=imread('mimosar.jpg');
2  [n,m,l]=size(X);
3  j = [0:n-1];
4  %costruiamo i filtri f1 e f2
5  f1= 1-cos(3*pi/2 + pi*j/n);
6  j = [0:m-1];
7  f2= 1-cos(3*pi/2 + pi*j/m);
8
9  Y = filtra_rgb(X,f1,f2);
10 %schiariamo
11 Y = Y+100*ones;
12 Y = uint8(Y);
13 imshow(Y);

```

Infine, questa è l'immagine filtrata:

