UNIVERSITAT POLITÈCNICA DE CATALUNYA

Sessió 2

INFORME

VISIÓ PER COMPUTADOR

Autors

DAVID LATORRE
ADRIÀ AUMATELL

EXERCICI 2:

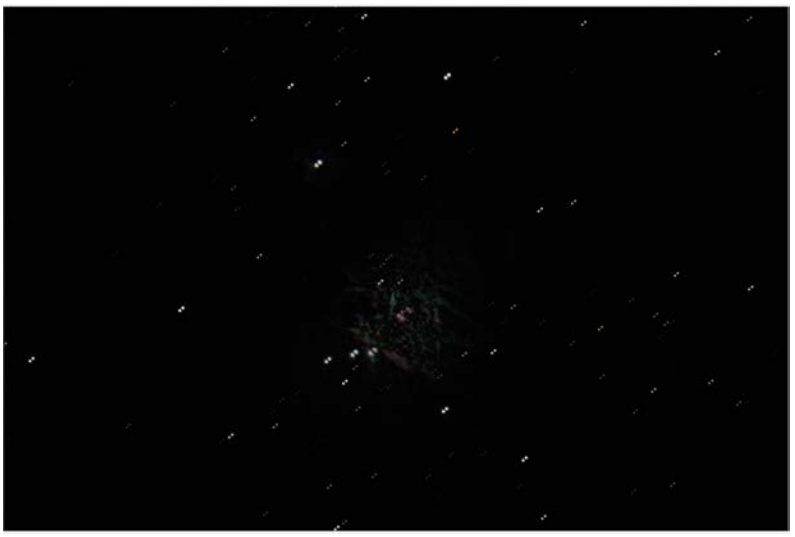
En aquest exercici, tal com s'explica a l'enunciat, fusionarem dos imatges que representen la nebulosa d'Orion, i aplicarem diverses millores per tal d'obtenir una imatge resultant que tingui un bon contrast i poc soroll.

Primer de tot executem els primers 3 apartats, comprovant que observem el que l'enunciat ens diu:

```
%% 1
A = double(imread('_MG_7735.JPG'))/255;
B = double(imread('_MG_7737.JPG')) /255;
```

```
%% 2
DIF = abs(A-B); % imatge diferencia
maxim = max(DIF(:));
DIF = DIF/maxim; % dividim pel seu valor màxim
imshow(DIF);
```

Output apartat 2:



Efectivament, com que les dues imatges es troben mogudes una respecte l'altre, sinó

apliquem cap translació observem que a la imatge diferència per exemple dades de color que representen les estrelles en les dues imatges a la diferència també es troben representades de forma doble, ja que no es troben a la mateixa posició per a les dos imatges.

```
%% 3
Bd = imtranslate(B,[20, -20]);
DIF = abs(A-Bd);
maxim = max(DIF(:));
DIF = DIF/maxim;
figure
imshow(DIF);
```

Output apartat 3:

Un cop aplicada la translació de forma correcta, ara ja sí que a la imatge diferència només observem les dades que una imatge no té respecte l'altre.

Un cop arribats a aquest punt, ja estem preparats per fusionar les dues imatges.

Primer de tot calculem la **imatge sumatori** de les dues imatges (en la que en una li hem aplicat la translació correcte), per tal que en aquesta hi tinguem les dades de les dues imatges.

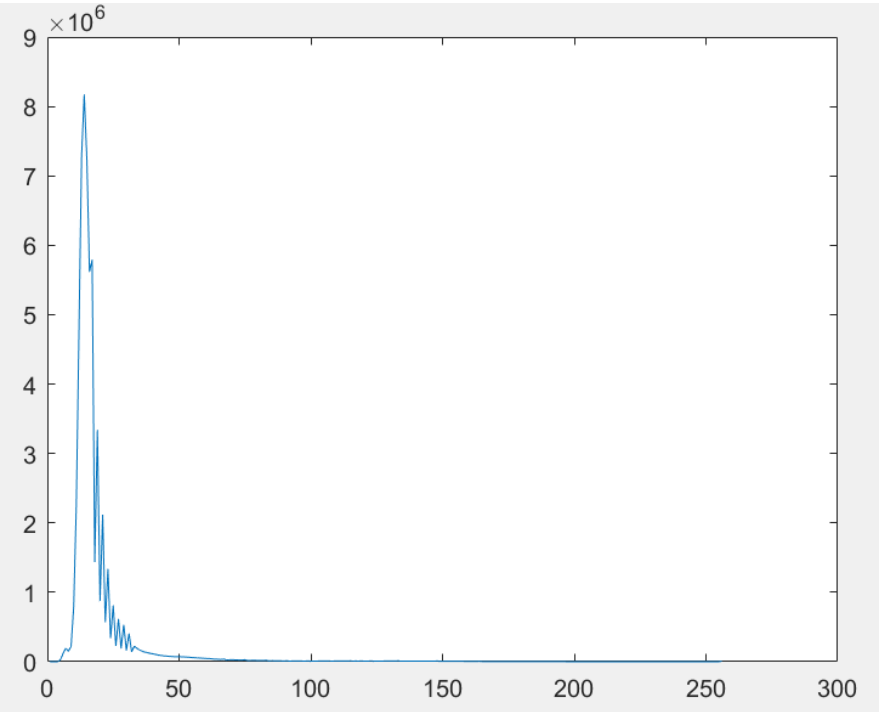
Un cop arribats aquí, començarem a aplicar una sèrie de millores a la imatge per augmentar-ne la seva qualitat.

Per fer això, mirem com tenim el histograma un cop acabem de crear la imatge sumatori:

Imatge sumatori:

Histograma:

```
hist1 = imhist(Ac);
figure
plot(hist1);
```



La clara conclusió que podem extreure d'aquest histograma és que gran part dels píxels de la imatge representen valors molt obscurs. Per tant, el primer que hem de fer és realçar el detall de totes aquestes regions obscures. L'objectiu no és que totes aquestes regions passin a ser regions mitjanes-clares, ja que al cap i a la fi l'espai és de color negre, sinó realçar aquelles que representin algun altre tipus d'informació que no sigui només el background de l'espai.

Al trobar-nos davant d'una imatge en format rgb, per realçar les regions obscures d'una manera sencilla, això ho podem fer passant la imatge a format HSL Per fer això, necessitem definir les funcions que passen rgb a hsl, i hsl a rgb:

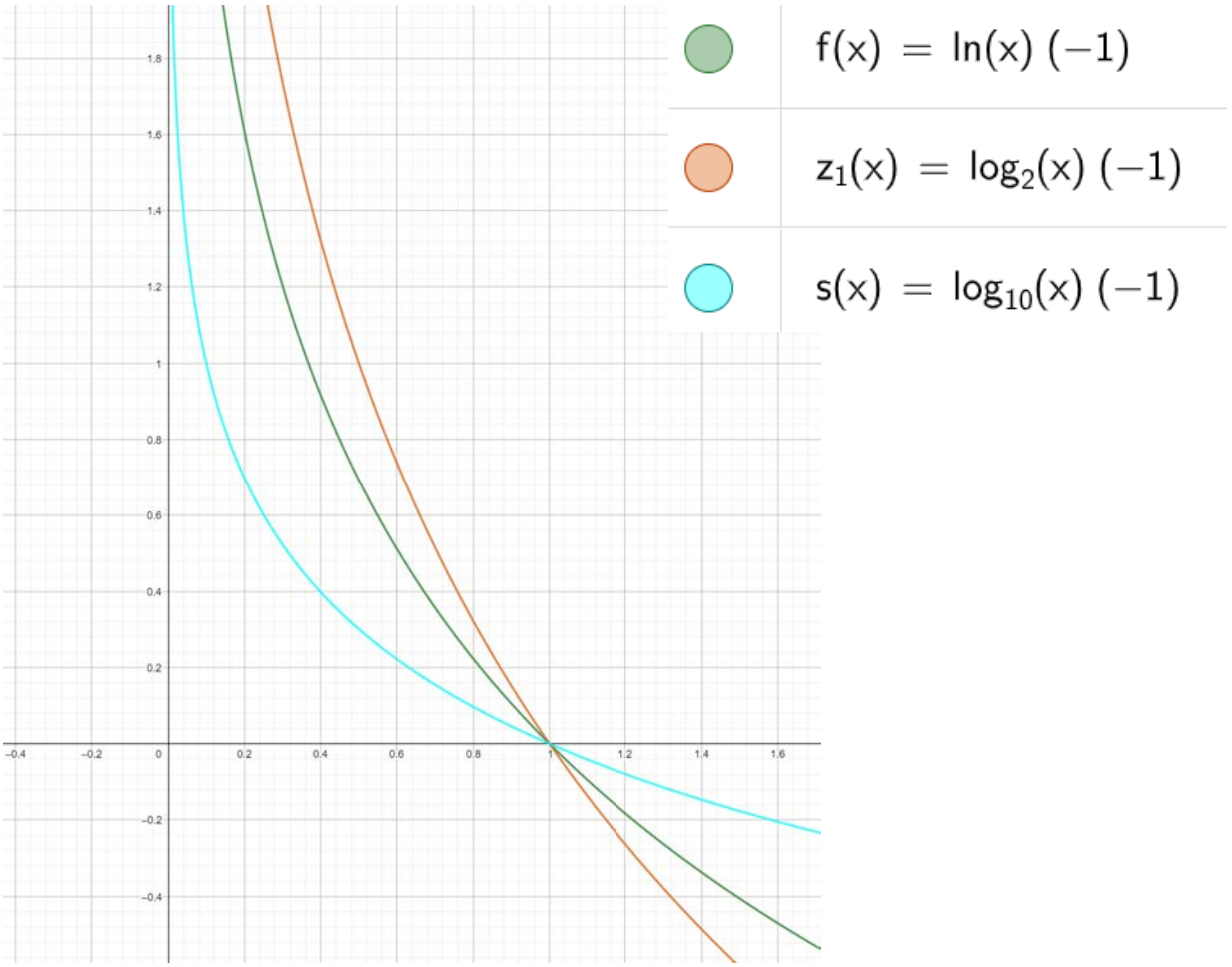
```
function [H,SL,L] = convertToHSL(RGB)
%convertToHSL Converts image from RGB to HSL colorspace
% https://en.wikipedia.org/wiki/HSL_and_HSV

HSV = rgb2hsv(RGB);

% Convert from HSV to HSL
% HL = HV = H;
H = HSV(:,:,1);
SV = HSV(:,:,2);
V = HSV(:,:,3);

L = V-(0.5.*V.*SV);

SL = (V-L) ./ min(L, 1-L);
SL(L==0 | L==1) = 0;
end
```

```
function RGB = convertFromHSL(H,SL,L)
% convertFromHSL Converts image from HSL to RGB colorspace
% https://en.wikipedia.org/wiki/HSL_and_HSV

% Convert from HSL to HSV
% HV = HL = H;
V = L+SL.*min(L, 1-L);
SV = 2*(1-L./V);
SV(V==0) = 0;

HSV = cat(3,H,SV,V);

RGB = hsv2rgb(HSV);

end
```

Un cop tenim aquestes funcions, ja podem executar la següent comanda:

```
[H,S,L] = convertToHSL(Ac);
```

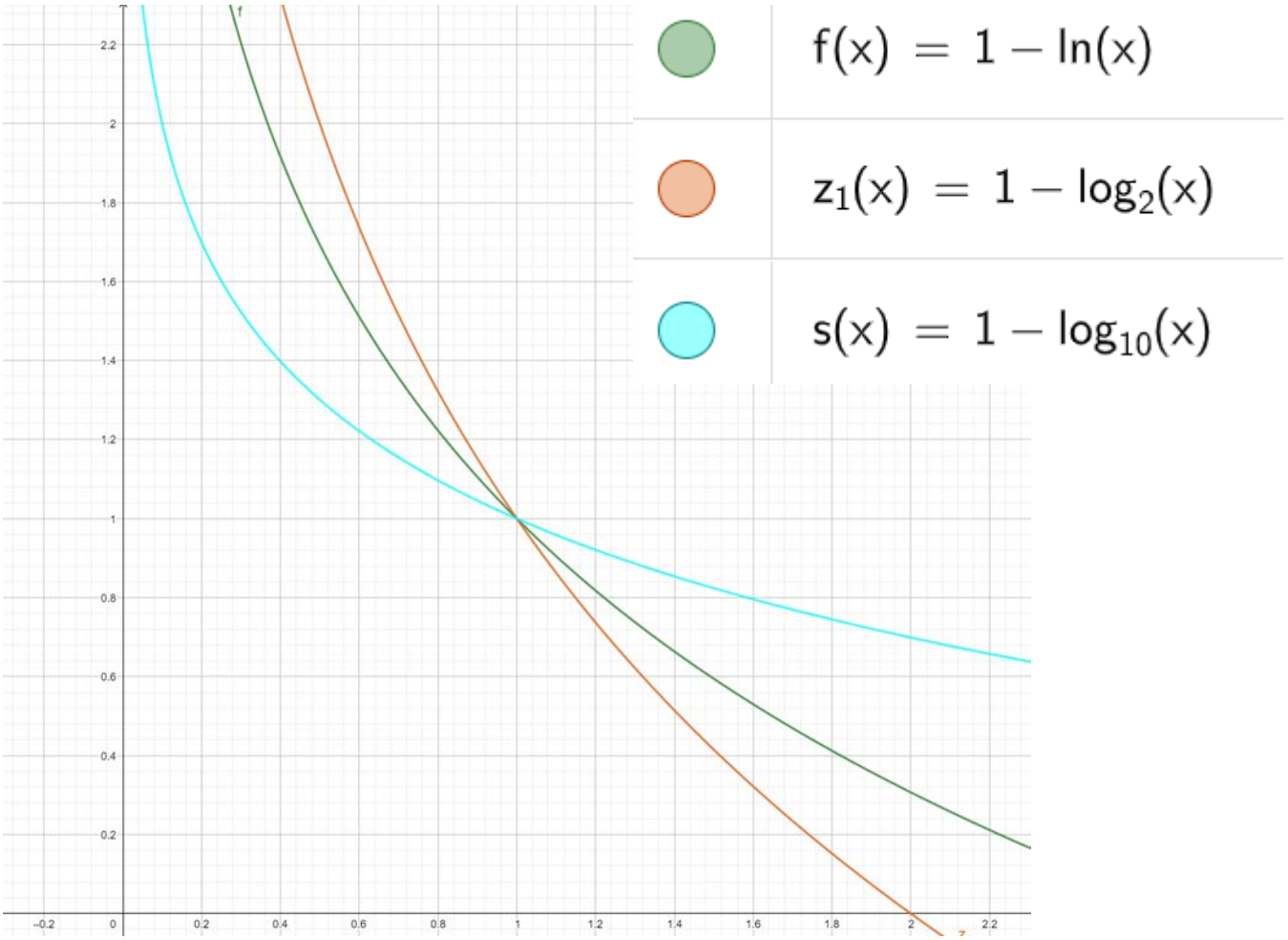
Degut a que els valors de les matrius es troben entre 0 i 1, per realçar les zones obscures el que podem fer és multiplicar el vector L pel negatiu d'un logaritme. Observem les diverses possibilitats:



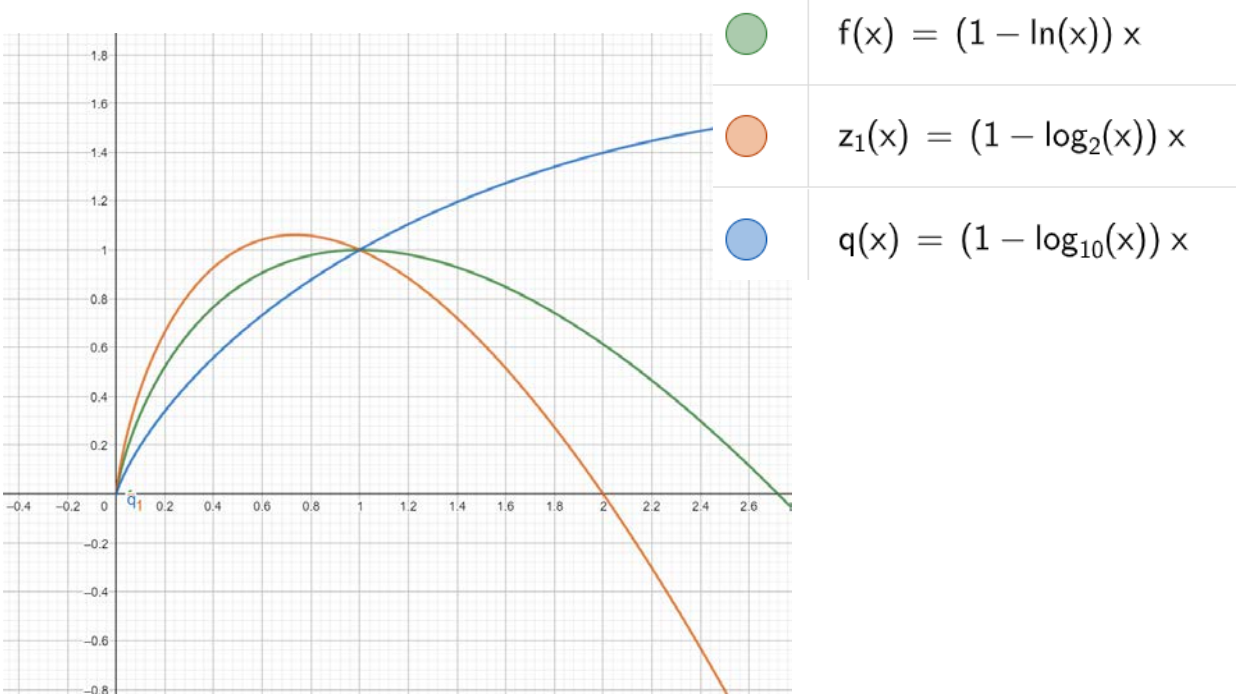
A la gràfica superior observem el comportant del logaritme natural, el logaritme en base dos, i el logaritme en base 10. A primera vista, podriem extreure la conclusió que multiplicar el vector L pel negatiu del logaritme en base 10 és la millor opció, ja que és aquell que eleva d'una forma més exagerada aquells valors propers a 0 respecte els valors que representen regions clares, però això ho haurem de veure amb la imatge que estem tractant.

Abans de fer això, però, no podem multiplicar directament el vector L per les funcions que hem definit en la anterior imatge, això suposaria obtenir una iluminació dels colors similar a la inversa que tenim ara, ja que llavors aquells pixels que representen colors molt clars (proper al blanc) és multiplicarien per un valor proper a 0, i per tant acabarien essent negres.

Aquestes funcions, per tant, les hem de convertir en aquestes:



D'aquesta forma ens assegurem que els valors clars segueixint representant blancs. Aquestes 3 funcions de la gràfica anterior son les que considerem per multiplicar el nostre vector L, produint així els següents resultats (graficament):



Observant aquesta última gràfica comprovem que el logaritme en base 2 no ens acaba d'anar bé, ja que amb aquest podem tenir rangs de la zona de tons mitjans que tinguin una lluminositat superior a rangs de la zona de tons alts, mostrant així unes lluminositats no correctes.

Calculem per tant el vector L d'aquestes de les formes del logaritme natural i el logaritme en base 10, i les normalitzarem a 0-1:

```
% Logaritme natural
Lnatural = L.*(1-log(L));
figure
Lnatural = rescale(Lnatural, 0, 1);
imglognatural = convertFromHSL(H,S,Lnatural);
imshow(imglognatural);
```

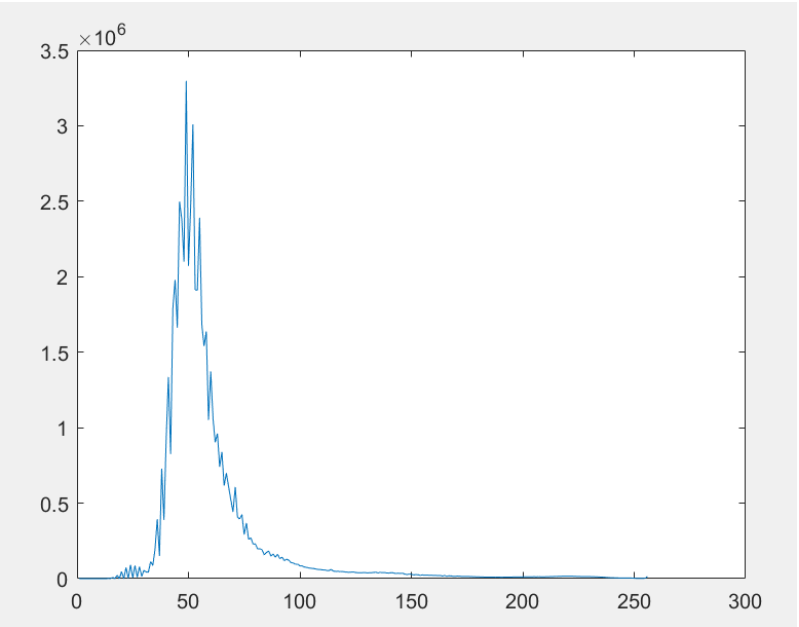
```
% Logaritme base 10
Lb10 = L.*(1-log10(L));
figure
Lb10 = rescale(Lb10, 0, 1);
imglogb10 = convertFromHSL(H,S,Lb10);
imshow(imglogb10);
```

Si comparem aquests dos resultats amb l'original, obtenim el següent:

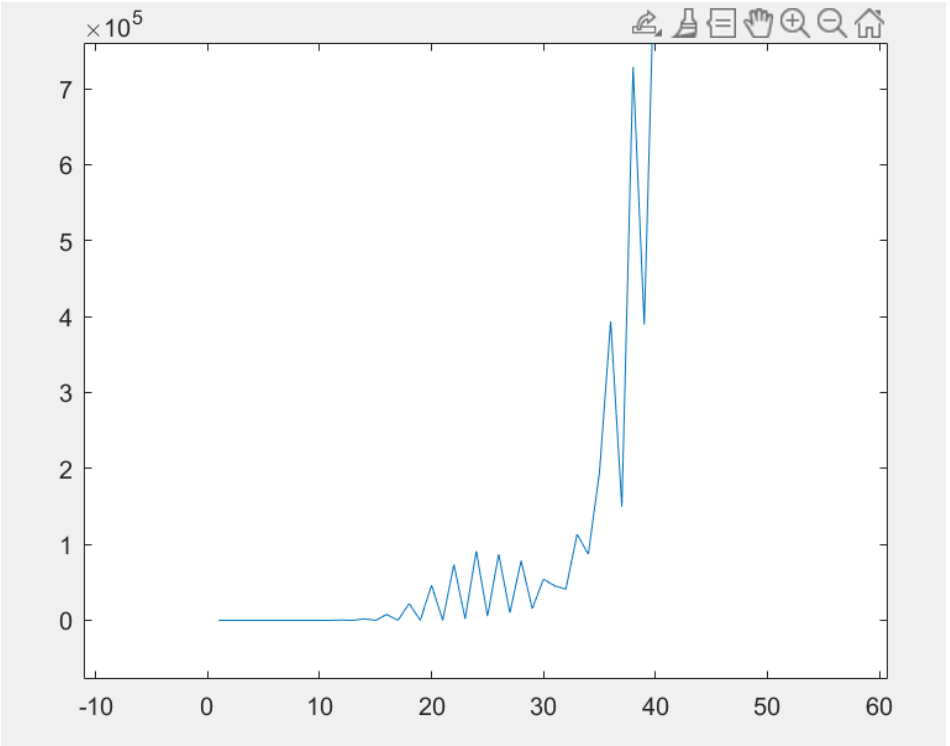
```
montage({A, imglognatural, imglogb10});
```

Observem que el logaritme natural (dalt a la dreta) compleix millor amb les nostres expectatives de pujar els tons baixos respecte el logaritme en base 10. Per tant ens quedem amb el logaritme natural:

Un problema que podem observar que ens acaba de sorgir a conseqüència de pujar els tons baixos, és que hem perdut aquells negres que ens representen el background de l'espai. Necessitem recuperar-los. Mirem el histograma actual:



Mirant el histograma, observem que hi ha una regió a les tonalitats més baixes on el nombre de píxels és 0. Aquesta és la raó per la qual no observem cap negre pur a la imatge. Fem zoom per observar a partir de quina lluminositat comença a haver-hi pixels:



Observem que els pixels amb tonalitats més baixes són representats per aproximadament els valors de 20 a 30 (en el rang de 0-255). Per tant, modifiquem els valors de la intensitat de la imatge perquè el límit inferior comenci a 25 (és a dir, aquells pixels que ara tenen una intensitat 25 passaran a tenir 0):

```
% seguim amb logaritme natural
histlognatural = imhist(imglognatural);
figure
plot(histlognatural);
% 25/255 = 0.098 (aprox)
imglognatural = imadjust(imglognatural, [0.098, 1]);
figure
imshow(imglognatural);
```

Ouput:

Ara ja hem recuperat els negres, i ja comencem a tenir una imatge que va complint amb les nostres expectatives.

Els valors d'intensitat de la foto ja el podem considerar arreglats, no obstant creiem que a la imatge li falta una mica més de vivacitat en el color, necessitem pujar la saturació dels colors. Fem-ho (necessitem tornar a canviar a l'espai HSL):

```
[H,S,L] = convertToHSL(imglognatural);
%multipliquem saturació per: 1.65
S = S.*1.65;
imglognatural = convertFromHSL(H,S,L);
figure
imshow(imglognatural);
```

Output:

Ara ja tenim una imatge que es veu encara millor.

Finalment, però, com a conseqüència dels retocs de intensitat i saturació, s'ha acabat generant a la foto una quantitat de soroll bastant gran, el qual necessitem eliminar-lo. Si fem zoom a la foto podem observar això:

Per tal de reduïr el soroll de la imatge utilitzarem un filtre. D'entrada, considerem tres candidats: el filtre mitjana, el filtre gaussià i el filtre mediana. Per poder decidir quin és el més adient els provarem tots tres.

Per aplicar un filtre de mitjana, definim una matriu de ponderació i posterorment cridem a la funció *imfilter()* amb els paràmetres especificats. Hem triat adient una matriu de ponderació que consideri els 8o veins mes propers d'un pixel.

```
mitjana = imfilter(imglognatural, ones(9)/81, 'conv');
figure
imshow(mitjana);
```

Output:

Amb aquest filtre el que observem és que a l'hora de calcular el valor d'un píxel, com que

estem donant-li la mateixa importància a tots els píxels veins per fer el calcul, perdem informació a la imatge resultant. A més, per eliminar notablement el soroll de la imatge, la foto resultant acabaria essent molt borrosa.

Provem ara amb el gaussià.

Pel que fa al filtre gaussià, només cal usar la funció *imgaussfilt()* i indicar una desviació. En aquest cas hem aplicat una desviació de 2.

```
gauss = imgaussfilt(imglognatural,2);
figure
imshow(gauss);
```

Output:

Aquest filtre ens redueix millor el soroll que l'anterior, ja que som capaços de reduir el soroll sense perdre tants detalls en comparació amb el filtre mitjana. Això és pel fet que a la funció que s'aplica a la convulsió és gaussiana, i a l'hora de calcular un pixel, la influència dels pixels veins disminueix amb la distància al centre.

Vegem una millor comparació de la imatge amb filtre gaussià i sense: (esquerre sense filtre, dreta amb el filtre)

Finalment, per aplicar el filtre mediana, només cal fer ús de la funció *medfilt2()*. L'haurem d'aplicar per a cada color separadament, ja que la funció medfilt2() només accepta una matriu de dos dimensions. Hem triat un veïnatge de 8 pixels en horitzontal i 8 vertical:

```
mediana = imglognatural;
for c = 1 : 3
    mediana(:, :, c) = medfilt2(imglognatural(:, :, c), [8, 8]);
end
figure
imshow(mediana);
```

Output:

Comparem aquest filtre amb la imatge sense filtres, i la imatge amb el filtre gaussià: (dalt esquerre imatge sense filtres, dalt dreta imatge amb filtre gaussià, baix imatge filtre mediana):

Amb els paràmetres que hem escollit, tot i que amb els filtres gaussià i mediana s'assemblen molt, amb el filtre mediana som capaços de reduir més soroll conservant el mateix detall que amb el filtre gaussià. És per això que escollim el filtre mediana.

Un cop fet això ja tenim la imatge final. Comparem-la amb la del principi (la imatge sumatori): (dalt imatge del principi, baix la imatge final)