

Επεξεργασία Εικόνας

Εργασία 1^η



Η εργασία θα πραγματοποιηθεί από τον φοιτητή:

Όνομα: Κωνσταντίνος

Επώνυμο: Σταθακόπουλος

ΑΜ: 161041

Τμήμα: IP Lab Group 1 (Τρίτη 11:00-13:00)

Περιεχόμενα

Ερωτήμα 1 ^ο	3
Ερωτήμα 2 ^ο	4
Ερωτήμα 3 ^ο	5
Ερωτήμα 4 ^ο	6
Ερωτήμα 5 ^ο	7
Ερωτήμα 6 ^ο	11
Ερωτήμα 7 ^ο	14
Ερωτήμα 8 ^ο	16

Ερωτήμα 1^ο

Ακολουθεί ο κώδικας της συνάρτησης `solarize` καθώς και ο κώδικας για την δοκιμή της και στην συνέχεια ακολουθούν οι έξοδοι που δίνει.

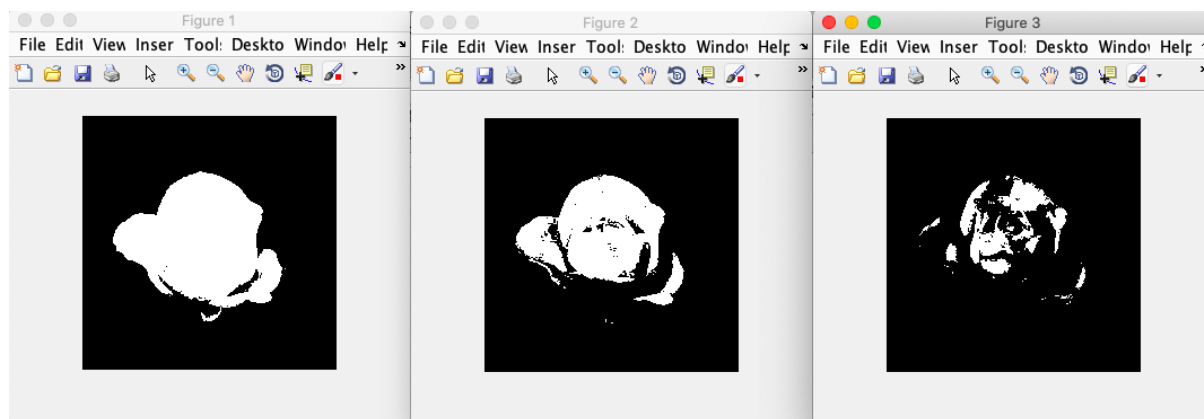
Κώδικας :

```
function solarize(I, T) %I is the image and T is the threshold
figure,imshow(I>T)
end
```

```
RGB = imread('rose.jpg'); %the rose.jpg image is being read
I = rgb2gray(RGB);
solarize(I, 64) %testing the function
solarize(I, 128)
solarize(I, 192)
```

%Τρέχουμε την συνάρτηση ως εξής: `solarize(image, 'threshold value')`

Έξοδοι :



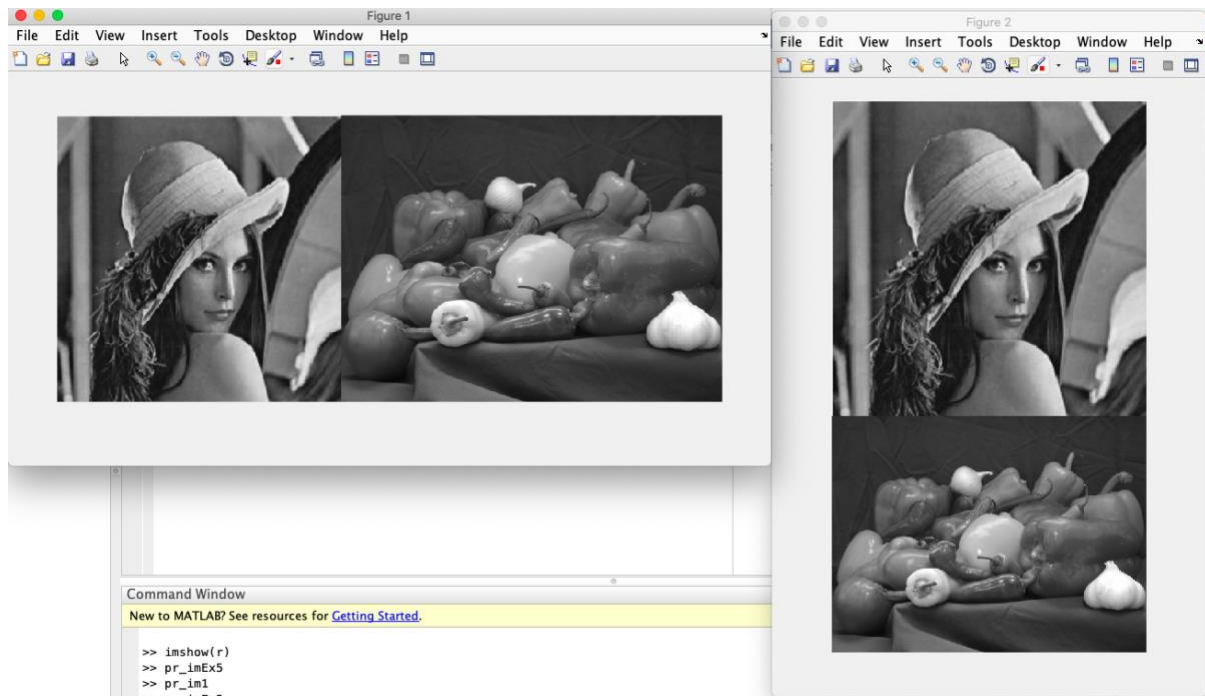
Ερωτήμα 2^ο

Ακολουθεί ο κώδικας του προγράμματος που ζητήθηκε και στην συνέχεια ακολουθούν οι έξοδοι που δίνει.

Κώδικας :

```
I1=imread('peppers.png'); %reading the images
I2=imread('lenna.png');
IG1=rgb2gray(I1); %grayscale the images
IG2=rgb2gray(I2);
ig2=imresize(IG2,[384 384]); %adjusting the dimensions of
lenna.png image
concl=cat(2,ig2,IG1); %merging the two images
figure, imshow(concl)
ig22=imresize(IG2,[512 512]); %adjusting the dimensions of
lenna.png image
conc2=cat(1,ig22,IG1); %merging the two images
figure, imshow(conc2)
```

Έξοδοι :



Ερωτήμα 3^ο

Ακολουθεί ο κώδικας για την εκτέλεση των ενεργειών που ζητήθηκαν και στην συνέχεια ακολουθούν οι έξοδοι που δίνει με την απαραίτητη ανάλυση.

Κώδικας :

```
t=imread('text.png'); %reading images
c=imread('cameraman.tif');
m=uint8(double(c)+255*double(t)); %combined images
figure,imshow(m>254) %thresholding with text only
(exercise3.1)
title('image after the thresholding')
figure

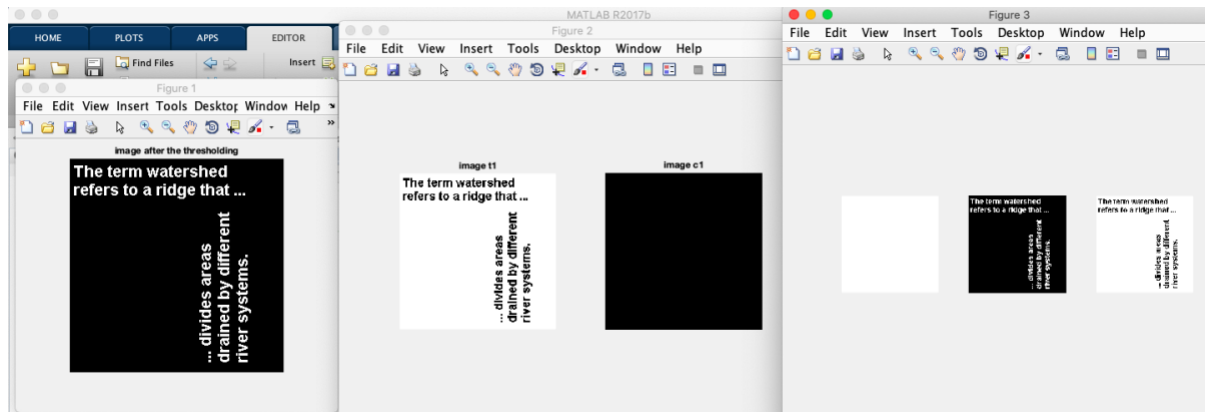
t1 = 1-t; %creating the complimentary images from t and c and
the output is shown (exercise 3.2)
subplot(1,2,1)
imshow(t1)
title('image t1')
c1 = 1-c;
subplot(1,2,2)
imshow(c1)
title('image c1')
figure

m=uint8(double(t1)+255*double(c)); %combined images
subplot(1,3,1)
imshow(m>0.999) %thresholding with text only

m=uint8(double(t)+255*double(c1)); %combined images
subplot(1,3,2)
imshow(m>0.999) %thresholding with text only

m=uint8(double(t1)+255*double(c1)); %combined images
subplot(1,3,3)
imshow(m>0.999) %thresholding with text only
```

Έξοδοι :



Στο **πρώτο παράθυρο (figure 1)** έχουμε την έξοδο από το ερώτημα 3.1 όπου η εικόνα “text.png” είχε τα περισσότερα pixel της 0, πλην τα pixel του κειμένου που ήταν 1, με αποτέλεσμα τα pixel του κειμένου να περνάνε την τιμή κατωφλίσωσης “level” και να παραμένουν άσπρα, εν αντιθέσει με τα υπόλοιπα pixel της εικόνα “cameraman.tif” και της εικόνας “text.png”. Η κατωφλίωση είχε ως αποτέλεσμα την απομόνωση του κειμένου από την υπόλοιπη εικόνα.

Στο **δεύτερο παράθυρο (figure 2)** έχουμε την έξοδο από το ερώτημα 3.2 όπου δημιουργήσαμε τη συμπληρωματική εικόνα t1 της t και τη συμπληρωματική εικόνα c1 της c.

Στο **τρίτο παράθυρο (figure 3)** έχουμε την έξοδο από το ερώτημα 3.2 όπου κάναμε την υπέρθεση και επαναλάβαμε το προηγούμενο ερώτημα για καθένα από τα ζεύγη εικόνων που ζητήθηκαν.

Οι εικόνες $c2 = (t, c1)$, $ct = (t1, c1)$ δεν χρειάζονταν κατωφλίωση, επειδή το κείμενο ήδη ήταν πλήρως ευδιάκριτο στην εικόνα. Πάραυτα έγινε κατωφλίωση και στις 2 εικόνες. Στις δύο αυτές εικόνες που δεν χρειαζόταν κατωφλίωση, οι τιμές της κατωφλίσωσης μπορούν να κινηθούν στα όρια του 0 και του 1. Δηλαδή ανάμεσα στο $0 < level < 1$, όποια τιμή και αν επιλεγεί, πάντα το κείμενο θα είναι ευδιάκριτο.

Ερωτήμα 4^ο

Ακολουθεί ο κώδικας για την συνάρτηση `mygsmed` που θα χρησιμοποιηθεί σε επόμενο ερώτημα.

Κώδικας :

```
%(Excercise 4) %this is an implimentation of Pratt's pseudo-
median filter
function res=mypsmed(im,d)
f=[0.125 0.125 0.125; 0.125 0 0.125; 0.125 0.125 0.125];
imd=im2double(im);
imf=filter2(f,imd);
r=abs(imd-imf)-d>0;
res=im2uint8(r.*imf+(1-r).*imd);
end
```

Ερωτήμα 5^ο

Ακολουθεί ο κώδικας για την εκτέλεση των ενεργειών που ζητήθηκαν και στην συνέχεια ακολουθούν οι έξοδοι που δίνει με την απαραίτητη ανάλυση.

Κώδικας :

```
f=imread('flowers.tif'); %the first three lines are given
fg=rgb2gray(f);
fg=im2uint8(fg(30:285,60:315));
fg_noise=imnoise(fg,'salt & pepper',0.05); %this function adds
5% noise to the image
imshow(fg_noise);
title('initial noisy image')
figure

a=fspecial('average'); %Excercise 5.1 (a)
fg_a=filter2(a,fg_noise);
subplot(1,3,1)
imshow(fg_a)
title('Avarage filter')

fg_b3=medfilt2(fg_noise); %Excercise 5.1 (b)
subplot(1,3,2)
imshow(fg_b3)
title('3x3 median filter')

fg_c=mypsmed(fg_noise,0.3); %Excercise 5.1 (c)
subplot(1,3,3)
imshow(fg_c)
title('pseudo-median filter')
figure
```

```

%Excercise 5.2
fg_noise=imnoise(fg,'salt & pepper',0.1); %this function adds
10% noise to the image
imshow(fg_noise);
title('initial noisy image')
figure

a=fspecial('average'); %(a)
fg_a=filter2(a,fg_noise);
subplot(1,3,1)
imshow(fg_a)
title('Avarage filter')

fg_b3=medfilt2(fg_noise); %(b)
subplot(1,3,2)
imshow(fg_b3)
title('3x3 median filter')

fg_c=mypsmmed(fg_noise,0.3); %(c)
subplot(1,3,3)
imshow(fg_c)
title('pseudo-median filter')
figure

fg_noise=imnoise(fg,'salt & pepper',0.2); %this function adds
20% noise to the image
imshow(fg_noise);
title('initial noisy image')
figure

a=fspecial('average'); %(a)
fg_a=filter2(a,fg_noise);
subplot(2,3,1)
imshow(fg_a)
title('Avarage filter')

fg_b1=medfilt2(fg_noise); %(b first application of the 3x3
filter)
subplot(2,3,2)
imshow(fg_b1)
title('3x3 median filter 1st applic')

fg_b3=medfilt2(fg_noise); %(b second application of the 3x3
filter)
subplot(2,3,3)
imshow(fg_b3)
title('3x3 median filter 2nd applic')

fg_b3=medfilt2(fg_noise,[7,7]); %(b application of the 7x7
filter)

```

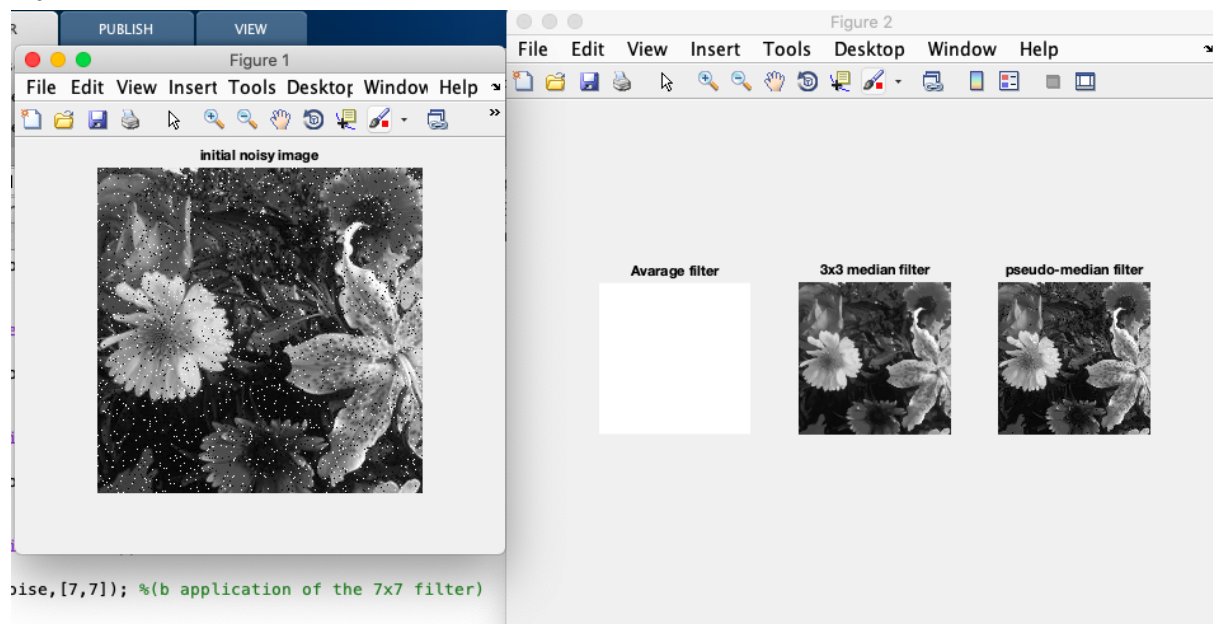


```
subplot(2,3,4)
imshow(fg_b3)
title('7x7 median filter')

fg_c=mypsmmed(fg_noise,0.3); %(c)
subplot(2,3,5)
imshow(fg_c)
title('pseudo-median filter')
```

Ερώτημα 5.1

Έξοδοι :



Παρατηρήσεις

Με θόρυβο salt and pepper 5% έχουμε:

Το average filter (φίλτρο μέσης τιμής) και σε αυτό το ερώτημα και στο επόμενο δεν παράγει τα αναμενόμενα αποτελέσματα πιθανότατα από λάθος που δεν κατάφερα να εντοπίσω.

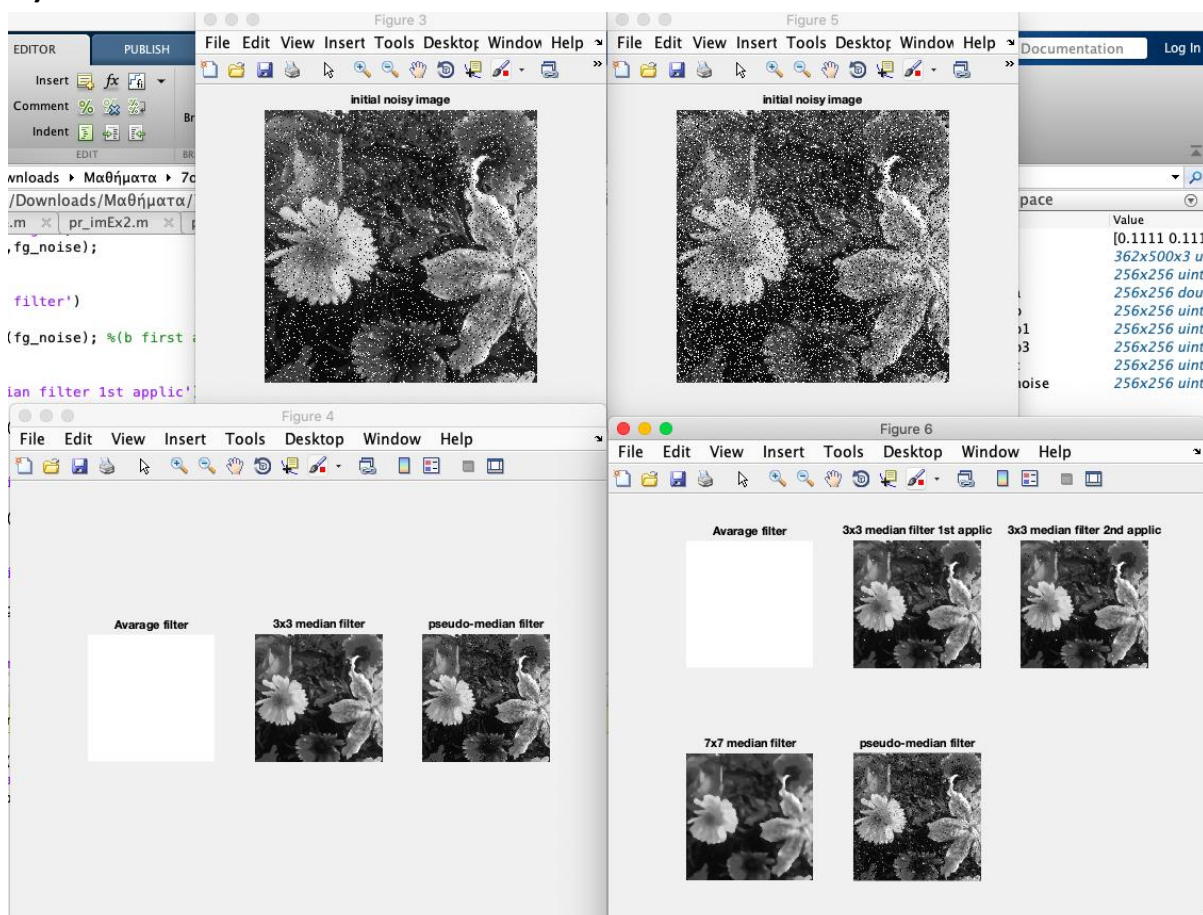
Το median filter (φίλτρο μεσαίας τιμής) καθαρίζει καλύτερα από όλα το θόρυβο αλλά έχει την παρενέργεια ότι μειώνεται η ευκρίνεια της εικόνας (θολώνει) και χάνονται κάποιες λεπτομέρειες (σε αυτή την περίπτωση έχουμε το καλύτερο

αποτέλεσμα).

Το pseudo-median filter (φίλτρο ψευδο-μεσαίας τιμής) δεν καθαρίζει όλο το θόρυβο που έχει η εικόνα αλλά δεν χάνεται η ευκρίνεια της εικόνας.

Ερώτημα 5.2

Έξοδοι :



Παρατηρήσεις

Με θόρυβο salt and pepper 10% (figure 4) έχουμε:

Το median filter (φίλτρο μεσαίας τιμής) καθαρίζει καλύτερα από όλα το θόρυβο αλλά έχει την παρενέργεια ότι μειώνεται η ευκρίνεια της εικόνας (θολώνει) και χάνονται κάποιες λεπτομέριες (σε αυτή την περίπτωση έχουμε το καλύτερο αποτέλεσμα).

Το pseudo-median filter (φίλτρο ψευδο-μεσαίας τιμής) δεν καθαρίζει όλο το θόρυβο που έχει η εικόνα αλλά δεν χάνεται η ευκρίνεια της εικόνας.

Με θόρυβο salt and pepper 20% (figure 6) έχουμε:

Το median filter (φίλτρο μεσαίας τιμής) στην πρώτη εφαρμογή του φίλτρου (3x3) καθαρίζει σε ικανοποιητικό επίπεδο αλλά όχι εντελώς και με την δεύτερη εφαρμογή βλέπουμε μια μικρή βελτίωση (σε αυτή την περίπτωση έχουμε το καλύτερο αποτέλεσμα). Στην εφαρμογή του φίλτρου 5x5 και 7x7 καθαρίζει ικανοποιητικά ο θόρυβος αλλά μειώνεται αρκετά η ευκρίνεια της εικόνας.

Το pseudo-median filter (φίλτρο ψευδο-μεσαίας τιμής) δεν καθαρίζει ικανοποιητικά το θόρυβο που έχει η εικόνα αλλά δεν χάνεται η ευκρίνεια της εικόνας.

Γενικό συμπέρασμα: Το φίλτρο μεσαίας τιμής έχει το καλύτερο αποτέλεσμα (ακόμα και από το φίλτρο μέσης τιμής με βάση το matlab tutorial documentation).

Ερωτήμα 6°

Ακολουθεί ο κώδικας για την εκτέλεση των ενεργειών που ζητήθηκαν και στην συνέχεια ακολουθούν οι έξοδοι που δίνει με την απαραίτητη ανάλυση.

Κώδικας :

```
f=imread('flowers.tif');  
fg=rgb2gray(f);  
fg=im2uint8(fg(30:285,60:315));  
fg_a=imnoise(fg,'gaussian'); %a) image with noise  
subplot(2,2,1)
```

```

imshow(fg_a)
title('a')

fg_b=imnoise(fg,'gaussian',0,0.02); %b) image with noise
subplot(2,2,2)
imshow(fg_b)
title('b')

fg_c=imnoise(fg,'gaussian',0,0.05); %c) image with noise
subplot(2,2,3)
imshow(fg_c)
title('c')

fg_d=imnoise(fg,'gaussian',0,0.1); %d) image with noise
subplot(2,2,4)
imshow(fg_d)
title('d')
suptitle('Images with gaussian noise')
figure

%now the noise is removed by using two different filters
%average filter:
b=fspecial('average'); %the result is not the expected and i
get a white image
fg1=filter2(b,fg_a);
subplot(2,2,1)
imshow(fg1)
title('a')

b=fspecial('average');
fg2=filter2(b,fg_b);
subplot(2,2,2)
imshow(fg2)
title('b')

c=fspecial('average');
fg3=filter2(c,fg_c);
subplot(2,2,3)
imshow(fg3)
title('c')

d=fspecial('average');
fg4=filter2(d,fg_d);
subplot(2,2,4)
imshow(fg4)
title('d')
suptitle('Cleaned images with average filter')

%Wiener filter
f1=wiener2(fg_a);
subplot(2,2,1)

```

```

imshow(f1)
title('a')

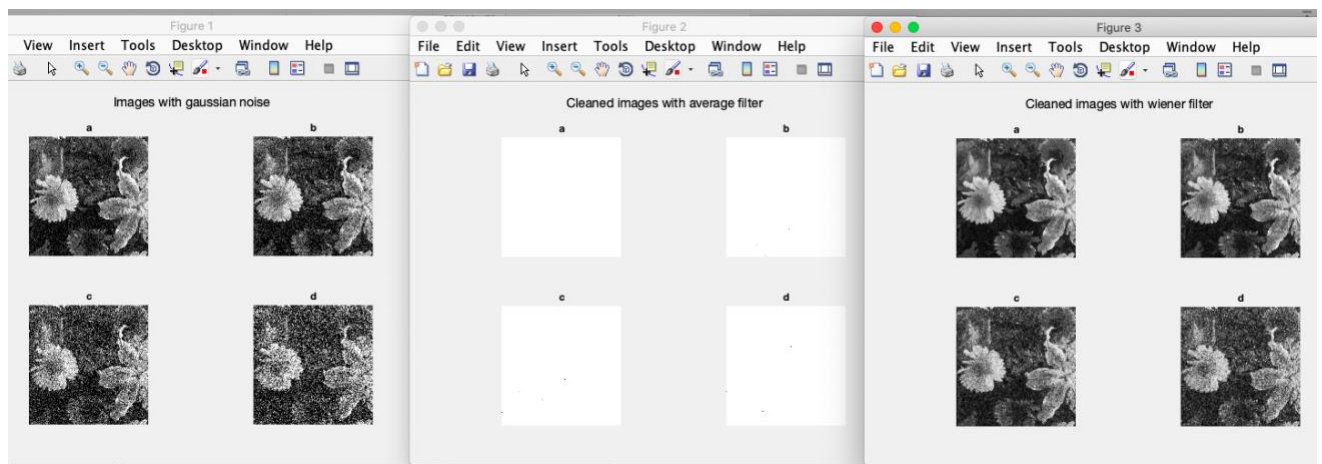
f2=wiener2(fg_b);
subplot(2,2,2)
imshow(f2)
title('b')

f3=wiener2(fg_c);
subplot(2,2,3)
imshow(f3)
title('c')

f4=wiener2(fg_d);
subplot(2,2,4)
imshow(f4)
title('d')
suptitle('Cleaned images with wiener filter')

```

Έξοδοι :



Παρατηρήσεις

Στο πρώτο παράθυρο (figure 1) φαίνεται η εικόνα με τον γκαουσιανό θόρυβο για όλες τις περιπτώσεις που ζητήθηκαν.

Στο δεύτερο παράθυρο (figure 2) φαίνεται η εικόνα που “καθαρίστηκε” με το φίλτρο μέσης τιμής (δεν δούλεψε ούτε σε αυτή την περίπτωση και δεν κατάφερα να εντοπίσω το λάθος μου) με βάση το υλικό στο εργαστήριο για να καθαριστεί ο θόρυβος η εικόνα γίνεται πιο θολή με αποτέλεσμα να χάνουμε ευκρίνεια και να μην έχουμε το επιθυμητό αποτέλεσμα.

Στο τρίτο παράθυρο (figure 3) φαίνεται η εικόνα που καθαρίστηκε με το φίλτρο wiener όπου ο θόρυβος δεν εξαλείφεται εντελώς αλλά είναι πολύ πιο κοντά στο επιθυμητό αποτέλεσμα και ενώ καθαρίστηκε ο θόρυβος σχετικά καλά δεν έχει θολώσει η εικόνα και φαίνεται ξεκάθαρα αυτό που απεικονίζει.

Ερωτήμα 7^ο

Ακολουθεί ο κώδικας για την εκτέλεση των ενεργειών που ζητήθηκαν και στην συνέχεια ακολουθούν οι έξοδοι που δίνει με την απαραίτητη ανάλυση.

Κώδικας :

```
function diffmaxmin (I, n)
%if (n<5) && (n>25) %checking if the value is odd and if is
between 5 and 25
    %if(diff(mod(n,2),0))
        I_min=nlfilter(I,[n,n],'min(x(:))'); %calculation of
min filter for the given image
        I_max=nlfilter(I,[n,n],'max(x(:))'); %calculation of
max filter for the given image
        I_diff=I_max-I_min; %calculation of difference
        imshow(I_diff)
        title(' NON-LINEAR MAX - MIN FILTER')
        figure
    %end
%end %with these checks the function does not give any output
end

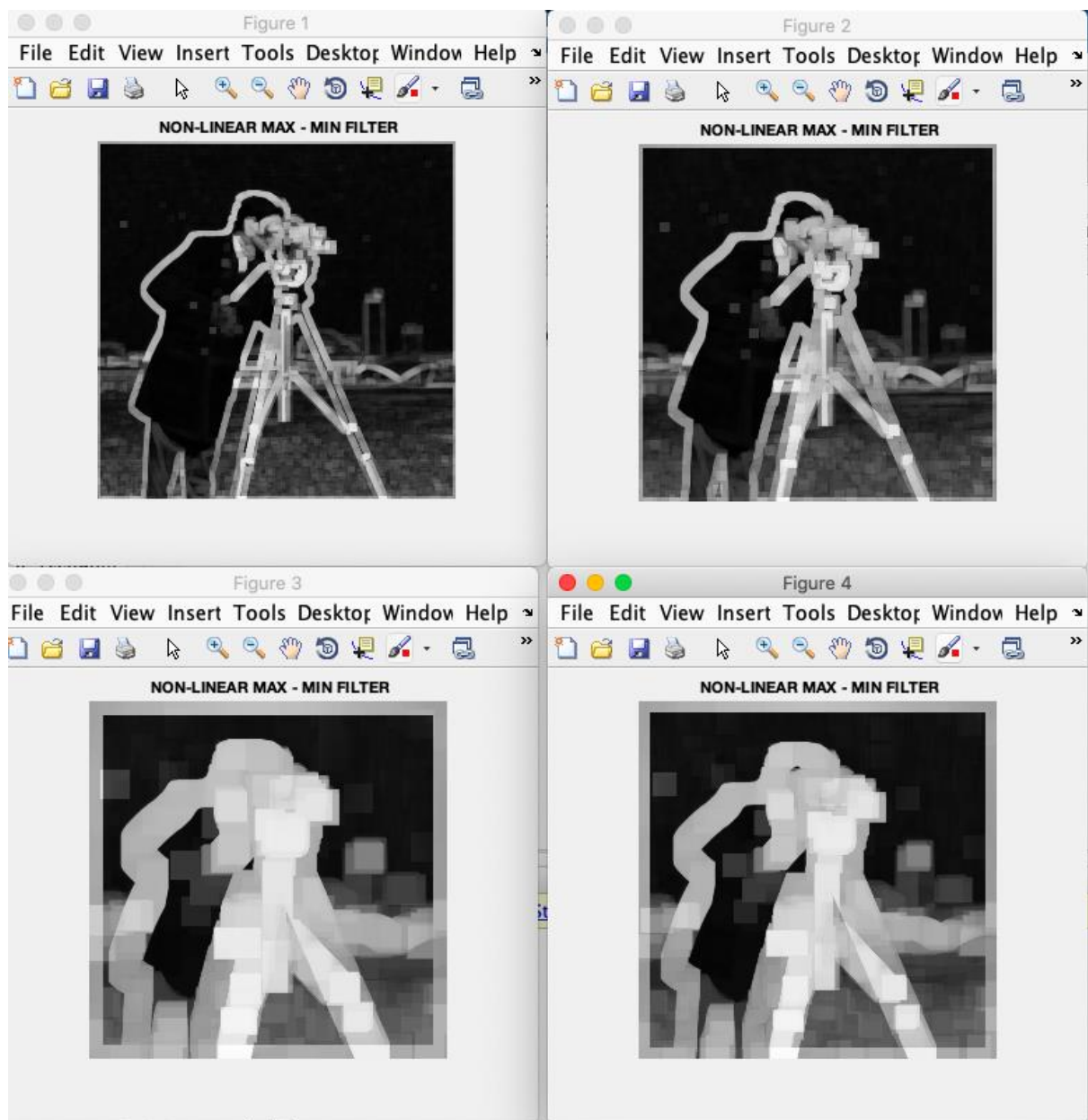
im=imread('cameraman.tif');
n1=5; n2=7; n3=21; n4=17;
diffmaxmin(im,n1) %testing the function
diffmaxmin(im,n2)
diffmaxmin(im,n3)
diffmaxmin(im,n4)
```

Παρατηρήσεις

n1=5; n2=7; n3=21; n4=17; %οι τιμές όπου δοκιμάζεται κάθε φορά η συνάρτηση

Στο πρώτο παράθυρο (figure 1) φαίνεται η εικόνα με τιμή $n=5$ όπου τονίζεται με άσπρο χρώμα το περίγραμμα των στοιχείων της εικόνας με άσπρο και παρατηρούμε ότι όσο μεγαλύτερη είναι η τιμή του n τόσο πιο παχύ είναι το «περίγραμμα» όπως φαίνεται στα υπόλοιπα παράθυρα (figure 2, figure 3, figure 4).

Έξοδοι :



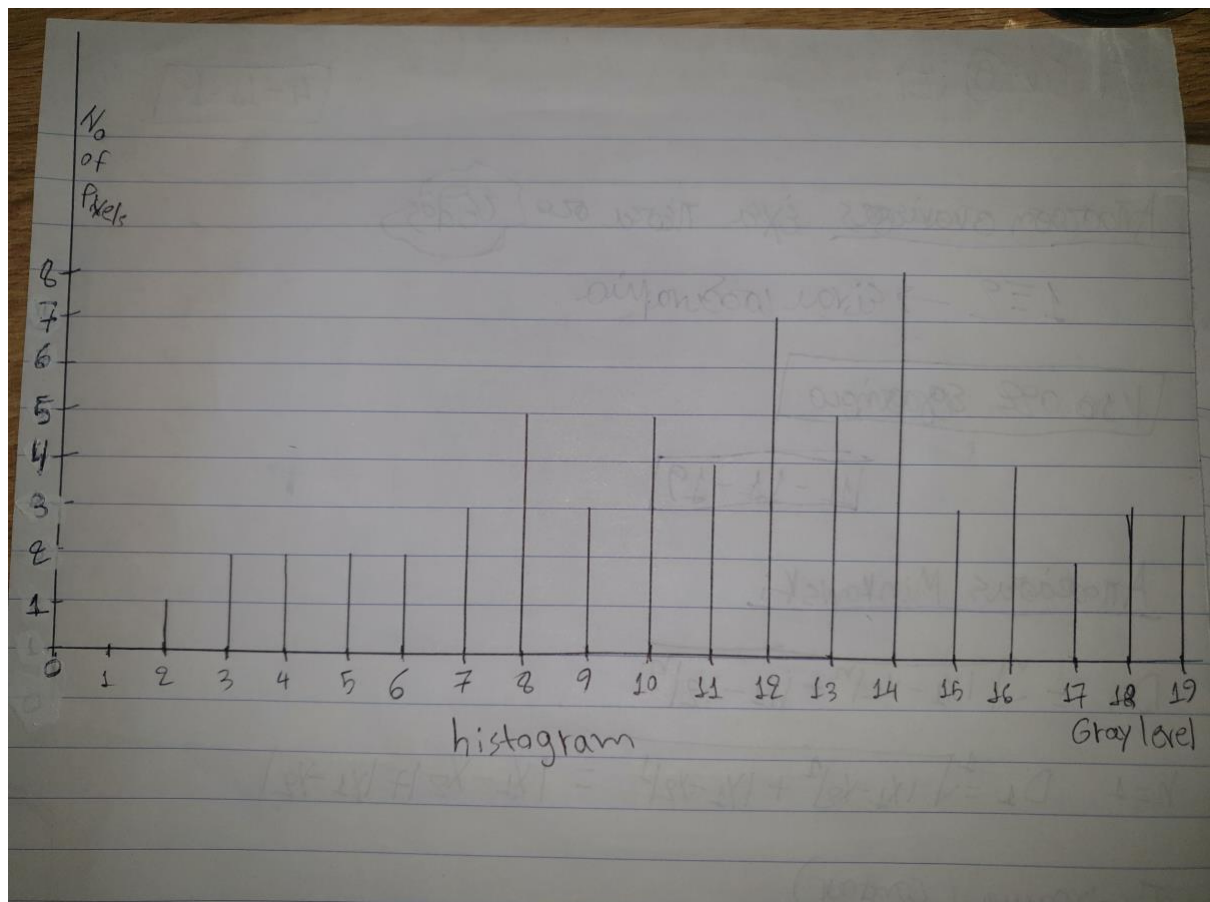
Ερωτήμα 8°

Ακολουθούν οι υπολογισμοί για το ιστόγραμμα στον παρακάτω πίνακα:

Gray level	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
No. of pixels	0	0	1	2	2	2	2	3	5	3	5	4	7	5	8	3	4	2	3	3
$k \sum_{j=0}^n n_j$	0	0	1	3	5	7	9	12	17	20	25	29	36	41	49	52	56	58	61	64
$k \sum_{j=0}^n n_j / n$	0	0																		
sx19	0	0	~0	~1	~1	~3	~3	~4	~5	~6	~7	~9	~11	~12	~15	~15	~17	~17	~18	19

- Η γραμμή που δεν είναι συμπληρωμένη εννοείται ότι προκύπτει από τις από πάνω τιμές δια 64.

Ακολουθεί το ιστόγραμμα της εικονας:



Τιμές των pixels της νέας εικόνας:

11	3	3	12	15	15	17	15
9	7	5	3	5	9	15	15
6	5	1	1	4	11	18	19
7	4	1	0	7	11	12	17
17	6	12	12	17	19	19	17
11	7	15	15	18	18	17	15
9	5	7	11	15	12	15	15
5	3	1	4	6	9	11	11