

ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΚΟΝΑΣ

ΕΡΓΑΣΙΑ 1^η

ΟΝΟΜΑ: ΠΙΚΡΙΔΑΣ ΜΕΝΕΛΑΟΣ

ΑΜ: 141291

ΤΜΗΜΑ: ΟΜΑΔΑ ΕΕ1



ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΑΝΕΠΙΣΤΗΜΙΟΥ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Περιεχόμενα

ΕΡΩΤΗΜΑ 1	3
ΕΡΩΤΗΜΑ 1.1	4
ΕΡΩΤΗΜΑ 1.2	4
ΕΡΩΤΗΜΑ 2	5
ΕΡΩΤΗΜΑ 2.1	5
ΕΡΩΤΗΜΑ 2.2	6
ΕΡΩΤΗΜΑ 3	9
ΕΡΩΤΗΜΑ 4	11
ΕΡΩΤΗΜΑ 4.1	11
ΕΡΩΤΗΜΑ 4.2	13
ΕΡΩΤΗΜΑ 5	17
ΕΡΩΤΗΜΑ 6	21

ΑΝΑΦΟΡΑ ΕΡΓΑΣΙΑΣ

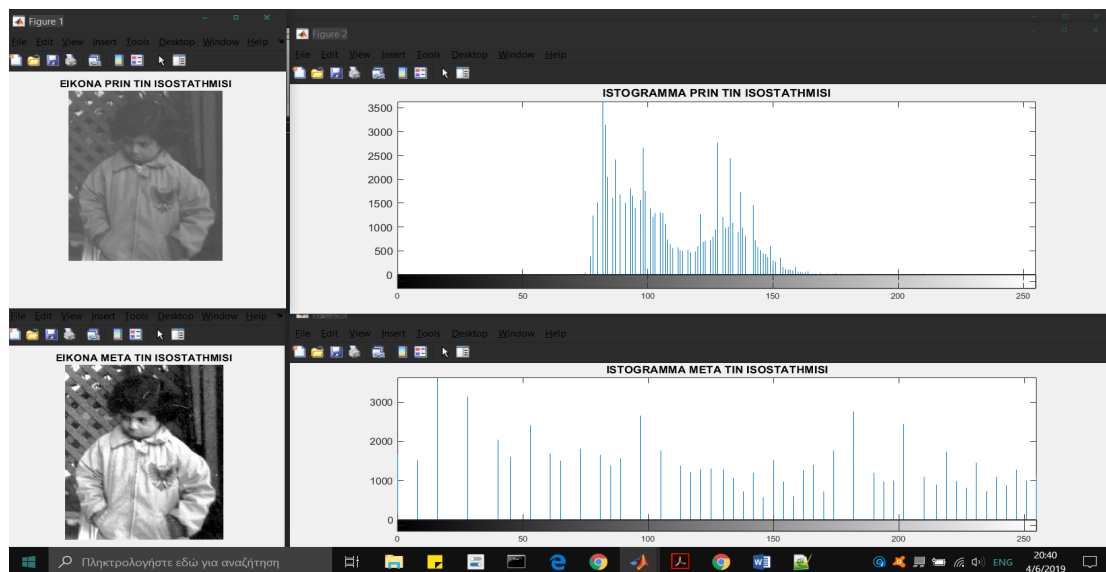
ΕΡΩΤΗΜΑ 1

ΚΩΔΙΚΑΣ ΣΥΝΑΡΤΗΣΗΣ:

```
function [p] = myHeist(im)
p=imread(im);
%ΠΡΙΝ ΤΗΝ ΙΣΟΣΤΑΘΜΙΣΗ
figure,imshow(p);
title('ΕΙΚΟΝΑ ΠΡΙΝ ΤΙΝ ΙΣΟΣΤΑΘΜΙΣΙ');
figure,imhist(p);
title('ΙΣΤΟΓΡΑΜΜΑ ΠΡΙΝ ΤΙΝ ΙΣΟΣΤΑΘΜΙΣΙ');
axis tight;
%ΜΕΤΑ ΤΗΝ ΙΣΟΣΤΑΘΜΙΣΗ
ph=histeq(p);
figure,imshow(ph);
title('ΕΙΚΟΝΑ ΜΕΤΑ ΤΙΝ ΙΣΟΣΤΑΘΜΙΣΙ');
figure,imhist(ph);
title('ΙΣΤΟΓΡΑΜΜΑ ΜΕΤΑ ΤΙΝ ΙΣΟΣΤΑΘΜΙΣΙ');
axis tight;
```

!!! ΤΡΕΧΟΥΜΕΤΗΝ ΣΥΝΑΡΤΗΣΗ ΜΕ ΤΗΝ ΕΝΤΟΛΗ myHeist('onoma_eikonas') !!!

SCREENSHOT



ΕΡΩΤΗΜΑ 1.1

Μετά την ισοστάθμιση μιας φωτογραφίας τα επίπεδα του γκρι εξαπλώνονται, από το κέντρο που ήταν, σε όλο το μήκος του ιστογράμματος με αποτέλεσμα η εικόνα να «καθαρίζει» και να φαίνονται περισσότερο οι λεπτομέρειες της, λόγω της αύξησης της αντίθεσης της φωτογραφίας.

ΕΡΩΤΗΜΑ 1.2

`J = histeq(I,hgram)`

- Μετατρέπει την εικόνα grayscale “I” με τέτοιο τρόπο ώστε το ιστόγραμμα της εξαγόμενης εικόνας “J”, με μήκος “hgram”, να αποθηκεύει προσεγγιστικά τις τιμές που ταιριάζουν στο ιστόγραμμα στόχου “hgram”.

`J = histeq(I,n)`

- Μετατρέπει την εικόνα grayscale “I”, επιστρέφοντας στην “J” μια grayscale εικόνα με “n” διακριτά επίπεδα του γκρι. Περίπου ίσος αριθμός πίξελ αντιστοιχίζει σε κάθε ένα από τα “n” επίπεδα του γκρι της “J”, με αποτέλεσμα το ιστόγραμμα της “J” να είναι σχεδόν επίπεδο. Αν το “n” είναι πολύ μικρότερο από τον αριθμό των διακριτών επιπέδων του “I”, τότε το ιστόγραμμα θα είναι ακόμα πιο επίπεδο.

`[J,T] = histeq(I)`

- Επιστρέφει την grayscale μετατροπή “T”, η οποία αντιστοιχεί στα επίπεδα του γκρι της εικόνας “I” με τα επίπεδα του γκρι της εικόνας “J”.

ΕΡΩΤΗΜΑ 2

ΑΡΧΙΚΟΣ ΚΩΔΙΚΑΣ ΥΠΕΡΘΕΣΗΣ.

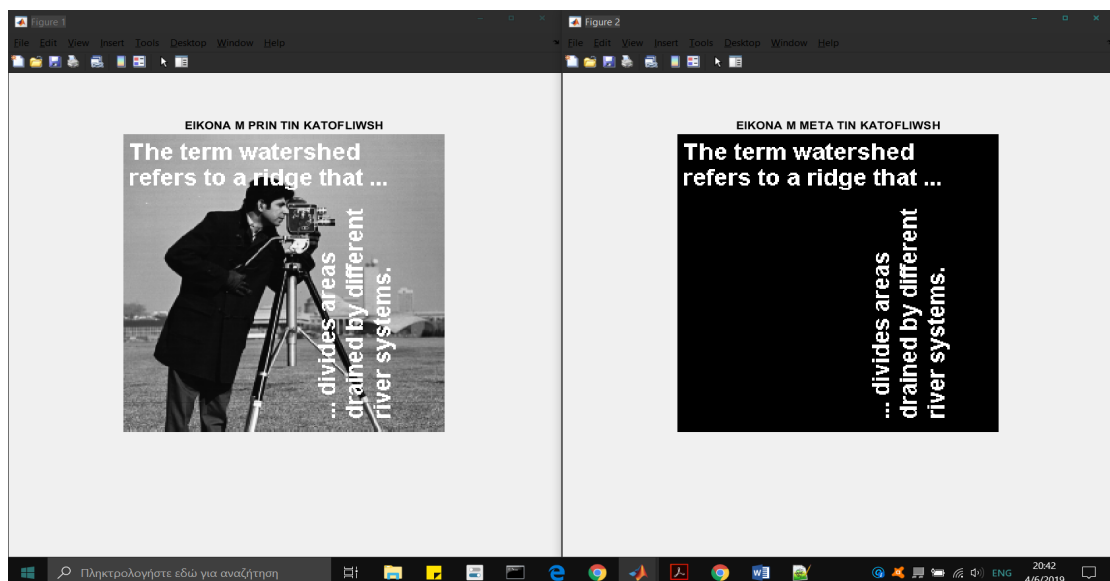
```
t=imread('text.png');  
c=imread('cameraman.tif');  
m=uint8(double(c)+255*double(t));  
figure,imshow(m);
```

ΕΡΩΤΗΜΑ 2.1

ΚΩΔΙΚΑΣ

```
t=imread('text.png');  
c=imread('cameraman.tif');  
m=uint8(double(c)+255*double(t));  
figure,imshow(m);  
title('EIKONA M PRIN TIN KATOFLIWSH');  
ma=im2bw(m,0.999);  
figure,imshow(ma);  
title('EIKONA M META TIN KATOFLIWSH');
```

SCREENSHOT



ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

BW = `im2bw(I,level)`

- Αυτή η συνάρτηση μετατρέπει την εικόνα “I” σε δυαδική εικόνα “BW”, μετατρέποντας όλα τα pixel που είναι μεγαλύτερα του “level” σε 1, δηλαδή άσπρα και όλα τα υπόλοιπα σε 0, δηλαδή μαύρα.

Στην περίπτωση μας, 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 εικόνας “cameraman.tif” και της εικόνας “text.png”. Η κατωφλίωση είχε ως αποτέλεσμα την απομόνωση του κειμένου από την υπόλοιπη εικόνα.

ΕΡΩΤΗΜΑ 2.2

ΚΩΔΙΚΑΣ

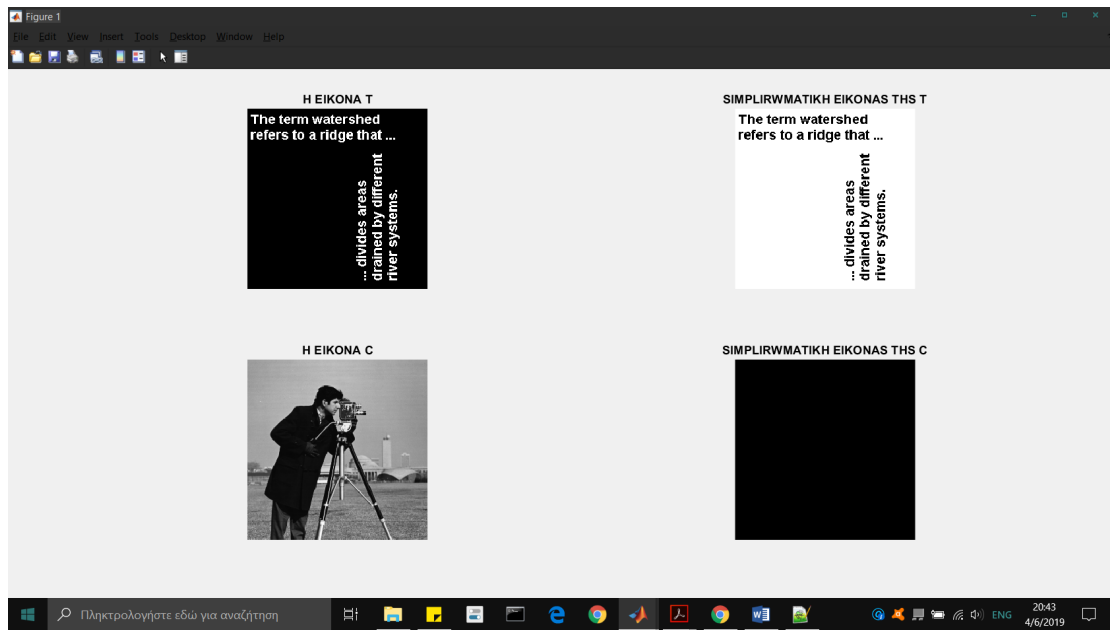
```
t=imread('text.png');
c=imread('cameraman.tif');
t1 = 1-t;
c1 = 1-c;
subplot(2,2,1)
imshow(t);
title('Η ΕΙΚΟΝΑ T');
subplot(2,2,3)
imshow(c);
title('Η ΕΙΚΟΝΑ C');
subplot(2,2,2)
imshow(t1);
title('SIMPLIRWMATIKH EIKONAS THS T');
subplot(2,2,4)
imshow(c1);
title('SIMPLIRWMATIKH EIKONAS THS C');
figure;
```

```

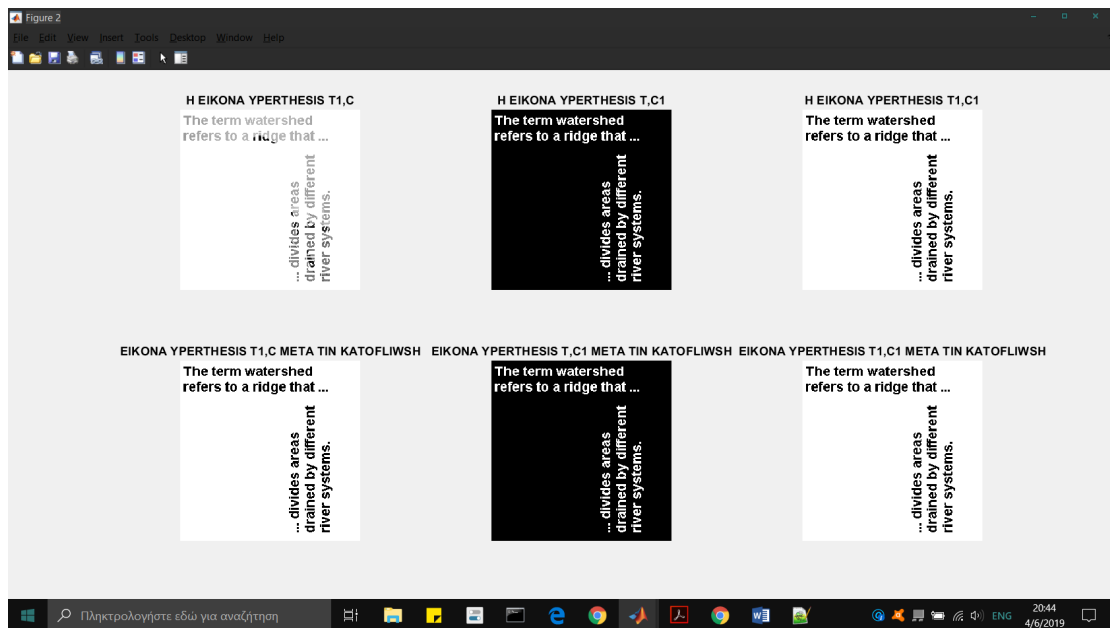
t2=uint8(double(c)+255*double(t1));
subplot(2,3,1)
imshow(t2);
title('H EIKONA YPERTHESES T1,C')
subplot(2,3,4)
t2a=im2bw(t2,0.999);
imshow(t2a);
title('EIKONA YPERTHESES T1,C META TIN KATOFLIWSH');
c2=uint8(double(c1)+255*double(t));
subplot(2,3,2)
imshow(c2);
title('H EIKONA YPERTHESES T,C1')
subplot(2,3,5)
c2a=im2bw(c2,0.999);
imshow(c2a);
title('EIKONA YPERTHESES T,C1 META TIN KATOFLIWSH');
ct=uint8(double(c1)+255*double(t1));
subplot(2,3,3)
imshow(ct);
title('H EIKONA YPERTHESES T1,C1')
subplot(2,3,6)
cta=im2bw(ct,0.0001);
imshow(cta);
title('EIKONA YPERTHESES T1,C1 META TIN KATOFLIWSH');

```

SCREENSHOT



2.1) ΕΠΕΞΗΓΗΣΗ ΤΩΝ ΕΙΚΟΝΩΝ T,T1 & C,C1



ΠΑΡΑΤΗΡΗΣΕΙΣ

Οι εικόνες $c2 = (t, c1)$, $ct = (t1, c1)$ δεν χρειαζόταν 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 κατωφλίωσης μπορούν να κινηθούν στα όρια του 0 και του 1. Δηλαδή ανάμεσα στο $0 < level < 1$, όποια τιμή και αν επιλεγθεί, πάντα το κείμενο θα είναι ευδιάκριτο.

ΕΡΩΤΗΜΑ 3

ΚΩΔΙΚΑΣ ΣΥΝΑΡΤΗΣΗΣ myPsmed

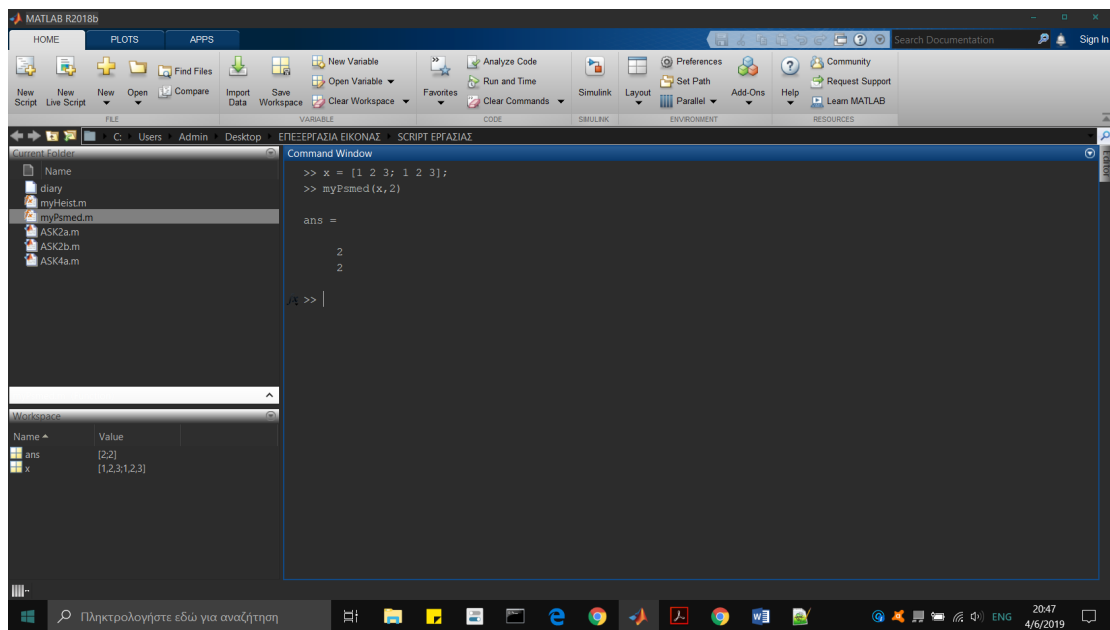
```
function [p] = myPsmed (x, dim)
    if nargin<1 || nargin>2
        error('Invalid number of input arguments')
    end
    if ndims(x)>2
        error('arrays of more than 2 dimensions are not supported')
    end
    if nargin<2
        if size(x,2)==1
            dim = 1;
        elseif size(x,1)==1
            dim = 2;
        else
            dim = 1;
        end
    end
    if dim<1 || dim>2
        error('dim must be a valid dimension');
```

```

end
% If applicable, switch dimension
if dim == 2
    x = x.';
end
% Obtain row dimensions
s = size(x);
m = s(1);
n = s(2);
% Create xi and xj values with the i <= j restriction enforced
q = logical(triu(ones(m,m),0));
i = uint32((1:m)'*ones(1,m));
xi = x(i(q),:);
j = uint32(ones(m,1)*(1:m));
xj = x(j(q),:);
% Calculate pairwise means (Walsh averages)
W = (xi+xj)./2;
% Calculate ordinary median of Walsh averages
p = median(W);
% If applicable, switch dimension
if dim == 2
    p = p.';
end
!!! TPEXOYME THN SYNAPTHZH ME THN ENTOAH myPsmed(pinakas,diastaeis)
!!!

```

SCREENSHOT



ΕΡΩΤΗΜΑ 4

ΑΡΧΙΚΟΣ ΚΩΔΙΚΑΣ ΜΕΤΑΤΡΟΠΗΣ ΕΙΚΟΝΑΣ RGB ΣΕ GRAYSCALE

```
f=imread('C:\Users\Admin\Desktop\epexsergasia\flowers.tif');  
fg=rgb2gray(f);  
fg=im2uint8(fg(30:285,60:315)));  
imshow(fg);
```

ΕΡΩΤΗΜΑ 4.1

ΚΩΔΙΚΑΣ

```
f=imread('C:\Users\Admin\Desktop\epexsergasia\flowers.tif');  
fg=rgb2gray(f);  
fg=im2uint8(fg(30:285,60:315)));  
%ΕΙΚΟΝΑ ΧΩΡΙΣ SALT AND PEPPER  
subplot(2,3,1)
```

```

imshow(fg);
title('EIKONA XWRIS SALT AND PEPPER');
%EIKONA ME SALT AND PEPPER 5%
fg_sp = imnoise(fg,'salt & pepper',0.05);
subplot(2,3,4)
imshow(fg_sp);
title('EIKONA ME SALT AND PEPPER 5%');
a = fspecial('average',[7,7]);
a1 = imfilter(fg_sp,a);
subplot(2,3,2)
imshow(a1);
title('EIKONA ME FILTRO AVERAGE');
a2 = medfilt2(fg_sp,2);
subplot(2,3,5)
imshow(a2);
title('EIKONA ME MEDIAN FILTRO');
a3 = myPsmmed(fg_sp);
subplot(2,3,3)
imshow(a3);
title('EIKONA ME PSEUDO-MEDIAN FILTRO');

```

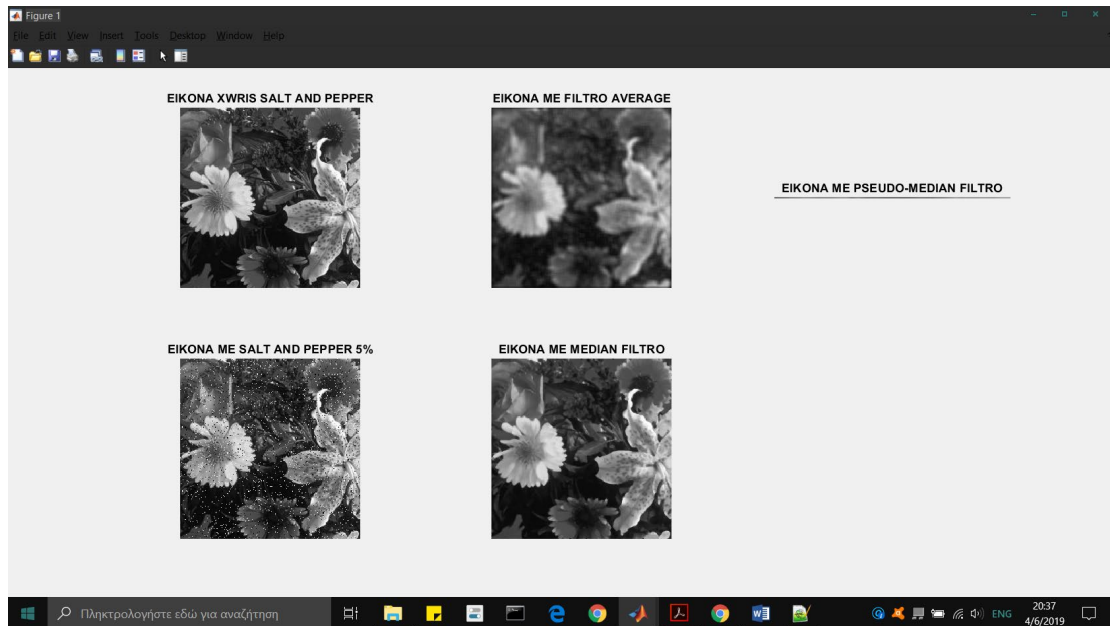
ΠΑΡΑΤΗΡΗΣΕΙΣ

Η αλλαγή της εικόνας, ακόμα και με 5% salt and pepper, είναι εμφανής. Το φίλτρο average, καθαρίζει την εικόνα από τον θόρυβο αλλά επίσης την κάνει πολύ πιο θολή, με αποτέλεσμα να αλλοιώνεται το αποτέλεσμα και η ίδια η εικόνα.

Το median filter κάνει σαφώς καλύτερη δουλειά από το average. Η εικόνα φαίνεται όπως η αρχική, με λίγο χαμηλότερη ανάλυση.

Το pseudo-median filter μου δυστυχώς δεν δουλεύει σωστά. Ουσιαστικά αυτό που κάνει είναι να βρίσκει την μέση τιμή κάθε στήλης και να βάζει το αποτέλεσμα στην 1^η γραμμή σε κάθε στήλη. Οπότε δεν είναι σε θέση να επεξεργαστεί σωστά μια εικόνα. Πάραυτα ο κώδικας με τον οποίο το φίλτρο θα καθάριζε η εικόνα είναι σωστός, δηλαδή έτσι θα περνούσαμε ως όρισμα την εικόνα στο φίλτρο και έτσι θα εμφανιζόταν η εικόνα.

SCREENSHOT



ΕΡΩΤΗΜΑ 4.2

ΚΩΔΙΚΑΣ SALT AND PEPPER 10%

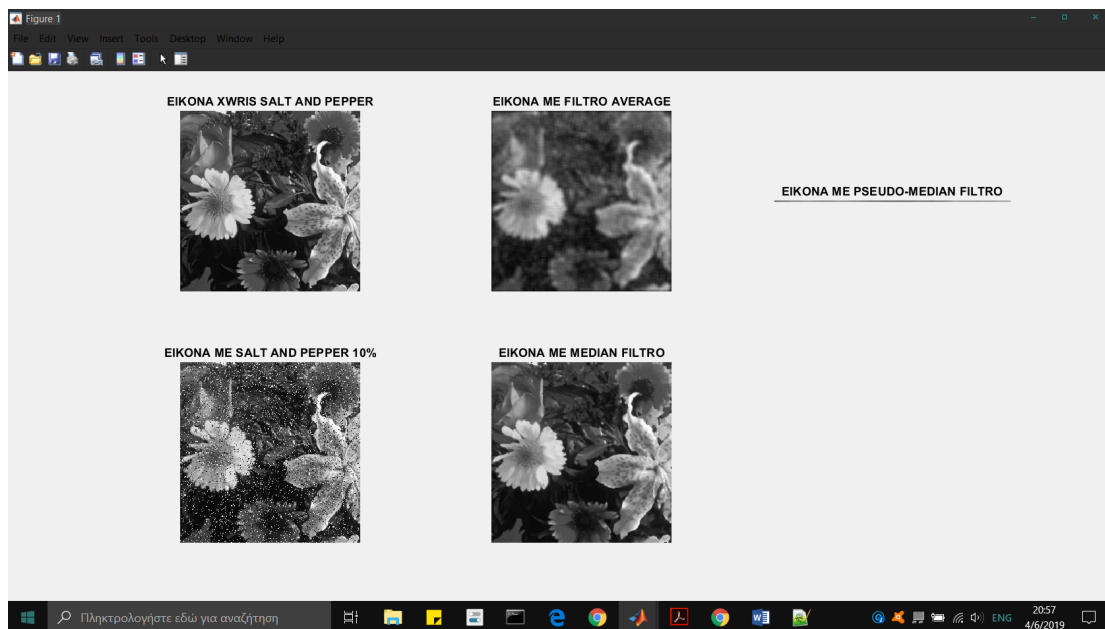
```
f=imread('C:\Users\Admin\Desktop\epeksergia\flowers.tif');  
fg=rgb2gray(f);  
fg=im2uint8(fg(30:285,60:315));  
%EIKONA XWRIS SALT AND PEPPER  
subplot(2,3,1)  
imshow(fg);  
title('EIKONA XWRIS SALT AND PEPPER');  
%EIKONA ME SALT AND PEPPER 10%  
fg_sp = imnoise(fg,'salt & pepper',0.1);
```

```

subplot(2,3,4)
imshow(fg_sp);
title('EIKONA ME SALT AND PEPPER 10%');
a = fspecial('average',[7,7]);
a1 = imfilter(fg_sp,a);
subplot(2,3,2)
imshow(a1);
title('EIKONA ME FILTRO AVERAGE');
a2 = medfilt2(fg_sp);
subplot(2,3,5)
imshow(a2);
title('EIKONA ME MEDIAN FILTRO');
a3 = myPsmmed(fg_sp);
subplot(2,3,3)
imshow(a3);
title('EIKONA ME PSEUDO-MEDIAN FILTRO');

```

SCREENSHOT



ΠΑΡΑΤΗΡΗΣΕΙΣ

Η διαφορά μεταξύ των εικόνων με θόρυβο salt and pepper 5% και με θόρυβο salt and pepper 10%, δεν είναι μεγάλη. Η κύρια διαφορά είναι πως η εικόνα με 10% θόρυβο είναι αρκετά πιο κατεστραμμένη από αυτήν με 5% θόρυβο. Ότι είπαμε και στις παρατήσεις του κώδικα του ερωτήματος 4.1 ισχύουν και εδώ.

ΚΩΔΙΚΑΣ SALT AND PEPPER 20%

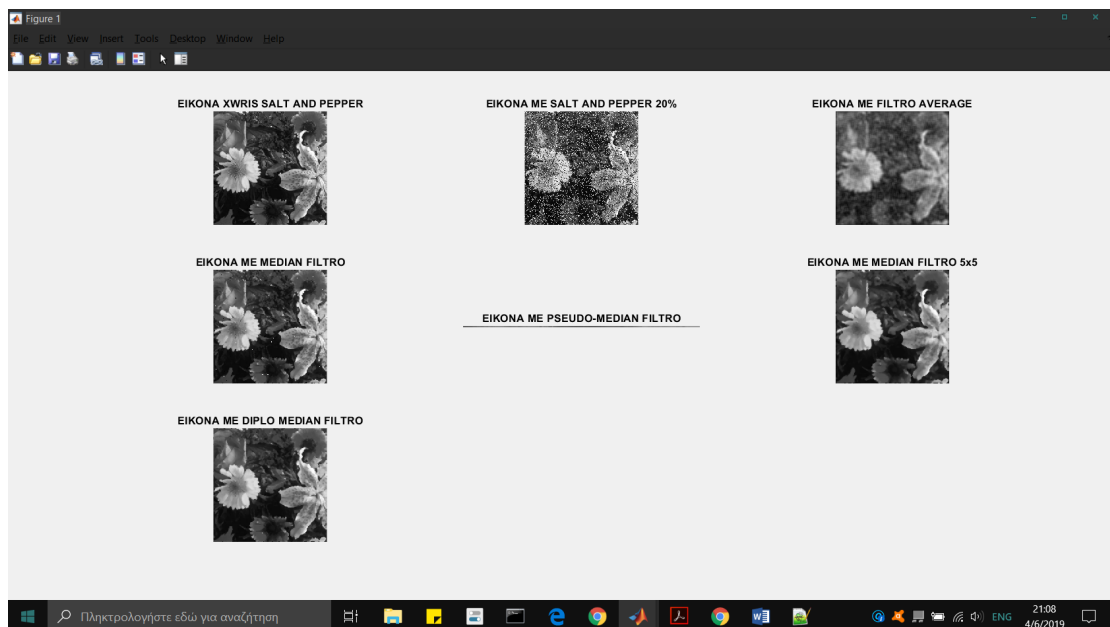
```
f=imread('C:\Users\Admin\Desktop\epexsergasia\flowers.tif');
fg=rgb2gray(f);
fg=im2uint8(fg(30:285,60:315));
%ΕΙΚΟΝΑ ΧWRIS SALT AND PEPPER
subplot(3,3,1)
imshow(fg);
title('ΕΙΚΟΝΑ ΧWRIS SALT AND PEPPER');
%ΕΙΚΟΝΑ ΜΕ SALT AND PEPPER 10%
fg_sp = imnoise(fg,'salt & pepper',0.2);
subplot(3,3,2)
imshow(fg_sp);
title('ΕΙΚΟΝΑ ΜΕ SALT AND PEPPER 20%');
a = fspecial('average',[7,7]);
a1 = imfilter(fg_sp,a);
subplot(3,3,3)
imshow(a1);
title('ΕΙΚΟΝΑ ΜΕ FILTRO AVERAGE');
a2 = medfilt2(fg_sp);
subplot(3,3,4)
imshow(a2);
title('ΕΙΚΟΝΑ ΜΕ MEDIAN FILTRO');
a3 = myPsmmed(fg_sp);
subplot(3,3,5)
imshow(a3);
title('ΕΙΚΟΝΑ ΜΕ PSEUDO-MEDIAN FILTRO');
```

```

a4 = medfilt2(fg_sp,[5,5]);
subplot(3,3,6)
imshow(a4);
title('EIKONA ME MEDIAN FILTRO 5x5');
a5 = medfilt2(fg_sp);
a6 = medfilt2(a5);
subplot(3,3,7)
imshow(a6);
title('EIKONA ME DIPLO MEDIAN FILTRO');

```

SCREENSHOT



ΠΑΡΑΤΗΡΗΣΕΙΣ

Η διαφορά μεταξύ των εικόνων με θόρυβο salt and pepper 20% σε σχέση με αυτές που έχουν θόρυβο salt and pepper 5% και salt and pepper 10% είναι αρκετά μεγάλη. Η κύρια διαφορά είναι πως η εικόνα με 20% θόρυβο είναι αρκετά πιο κατεστραμμένη από αυτήν με 10% θόρυβο και με 5% θόρυβο.

Το φίλτρο average, καθαρίζει την εικόνα από τον θόρυβο αλλά επίσης την κάνει πολύ πιο θολή, με αποτέλεσμα να αλλοιώνεται το αποτέλεσμα και η ίδια η εικόνα. Η εικόνα είναι ακόμα πιο θολή σε σχέση με τις προηγούμενες.

Το median filter κάνει σαφώς καλύτερη δουλειά από το average. Η εικόνα φαίνεται παρόμοια με την αρχική, χωρίς όμως να έχει καθαρίσει πλήρως η εικόνα.

Το 5x5 median filter κάνει σαφώς καλύτερη δουλειά από το απλό median filter. Η εικόνα φαίνεται σαν την αρχική, με χαμηλότερη ανάλυση, ενώ έχει καθαρίσει πλήρως η εικόνα.

Το double median filter κάνει ακόμα καλύτερη δουλειά από το απλό median filter και από το 5x5 median filter. Η εικόνα φαίνεται σαν την αρχική, με χαμηλότερη ανάλυση, ενώ έχει καθαρίσει πλήρως η εικόνα.

Το pseudo-median filter μου δυστυχώς δεν δουλεύει σωστά. Ουσιαστικά αυτό που κάνει είναι να βρίσκει την μέση τιμή κάθε στήλης και να βάζει το αποτέλεσμα στην 1^η γραμμή σε κάθε στήλη. Οπότε δεν είναι σε θέση να επεξεργαστεί σωστά μια εικόνα. Πάραυτα ο κώδικας με τον οποίο το φίλτρο θα καθάριζε η εικόνα είναι σωστός, δηλαδή έτσι θα περνούσαμε ως όρισμα την εικόνα στο φίλτρο και έτσι θα εμφανιζόταν η εικόνα.

ΕΡΩΤΗΜΑ 5

ΚΩΔΙΚΑΣ

```
f=imread('C:\Users\Admin\Desktop\epeksergasia\flowers.tif');
fg=rgb2gray(f);
fg=im2uint8(fg(30:285,60:315));
%EIKONA XWRIS GAUSSIAN NOISE
subplot(2,3,1)
imshow(fg);
title('EIKONA XWRIS GAUSSIAN NOISE');
a = imnoise(fg,'gaussian',0,0.01);
subplot(2,3,2)
imshow(a);
title('EIKONA ME GAUSSIAN NOISE 0.01');
b = imnoise(fg,'gaussian',0,0.02);
subplot(2,3,3)
```

```

imshow(b);
title('EIKONA ME GAUSSIAN NOISE 0.02');
c = imnoise(fg,'gaussian',0,0.05);
subplot(2,3,4)
imshow(c);
title('EIKONA ME GAUSSIAN NOISE 0.05');
d = imnoise(fg,'gaussian',0,0.1);
subplot(2,3,5)
imshow(d);
title('EIKONA ME GAUSSIAN NOISE 0.1');
e = imnoise(fg,'gaussian',0,0.2);
subplot(2,3,6)
imshow(e);
title('EIKONA ME GAUSSIAN NOISE 0.2');
figure;
avg = fspecial('average',[7,7]);
a1 = filter2(avg,a);
subplot(2,2,1)
imshow(a1);
title('EIKONA a ME FILTRO AVERAGE');
a2 = filter2(avg,b);
subplot(2,2,2)
imshow(a2);
title('EIKONA b ME FILTRO AVERAGE');
a3 = filter2(avg,c);
subplot(2,2,3)
imshow(a3);
title('EIKONA c ME FILTRO AVERAGE');
a4 = filter2(avg,d);
subplot(2,2,4)
imshow(a4);
title('EIKONA d ME FILTRO AVERAGE');

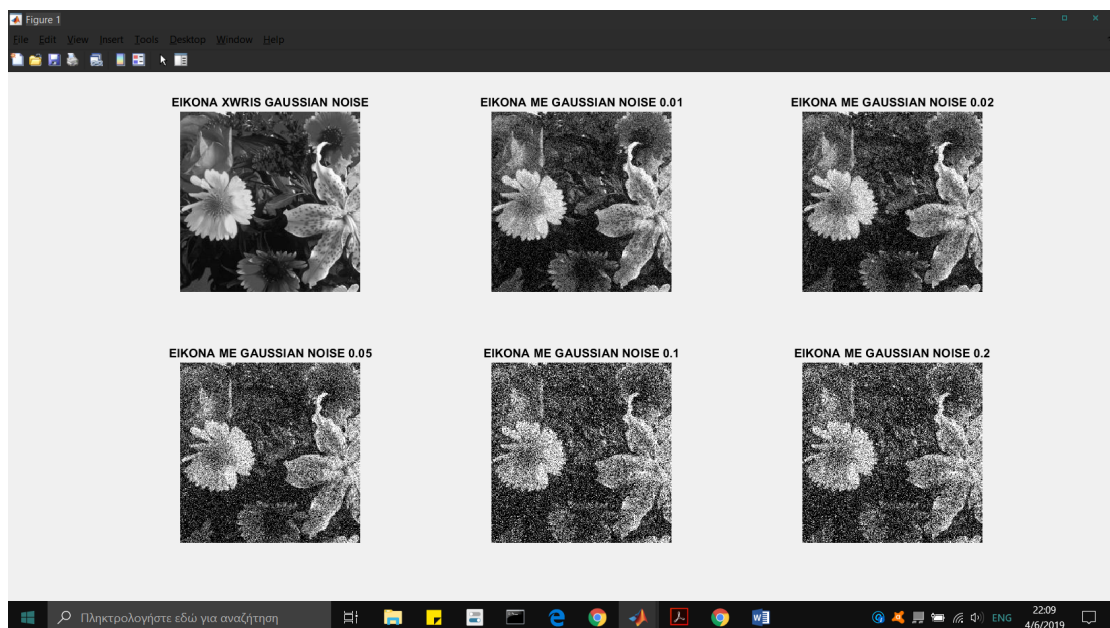
```

```

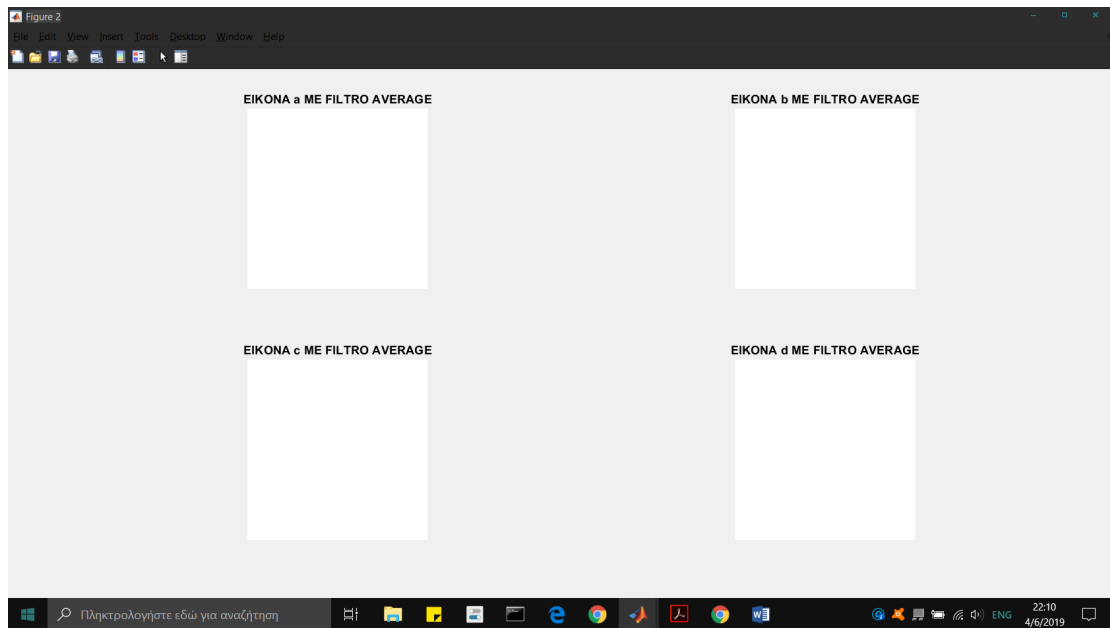
figure;
b1 = wiener2(a);
subplot(2,2,1);
imshow(b1);
title('EIKONA a ME FILTRO WIENER');
b2 = wiener2(b);
subplot(2,2,2);
imshow(b2);
title('EIKONA b ME FILTRO WIENER');
b3 = wiener2(c);
subplot(2,2,3);
imshow(b3);
title('EIKONA c ME FILTRO WIENER');
b4 = wiener2(d);
subplot(2,2,4);
imshow(b4);
title('EIKONA d ME FILTRO WIENER');

```

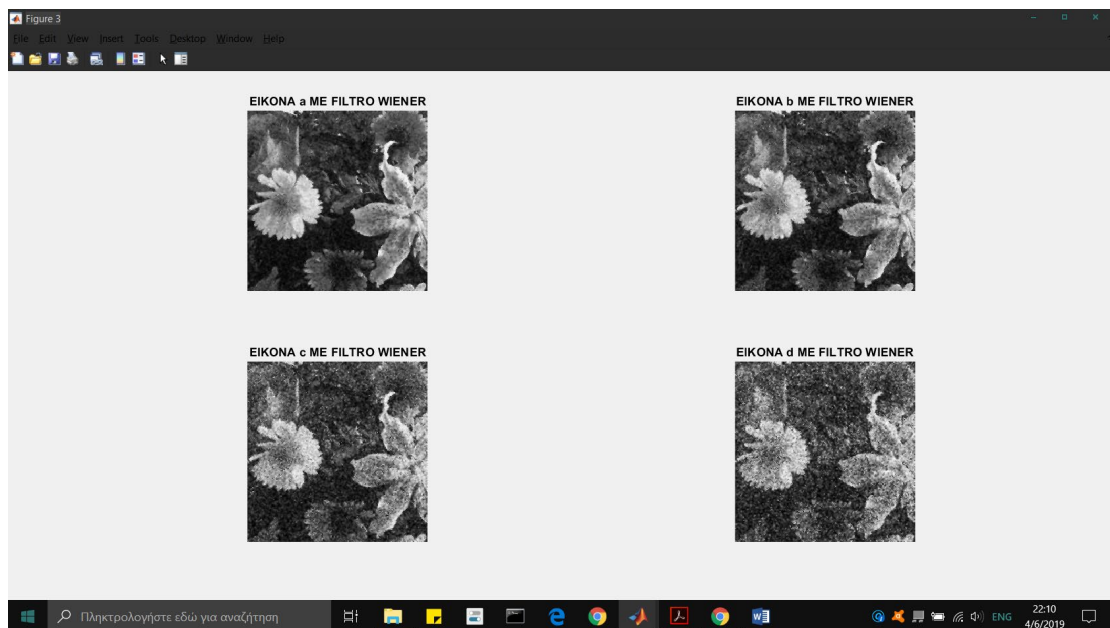
SCREENSHOT



5.1) ΑΝΑΠΑΡΑΣΤΑΣΗ ΕΙΚΟΝΩΝ ΜΕ GAUSSIAN NOISE



5.2) ΑΝΑΠΑΡΑΣΤΑΣΗ ΕΙΚΟΝΩΝ ΜΕ AVERAGE FILTER



5.3) ΑΝΑΠΑΡΑΣΤΑΣΗ ΕΙΚΟΝΩΝ ΜΕ WIENER FILTER

ΠΑΡΑΤΗΡΗΣΕΙΣ

Αρχικά γίνεται αναπαράσταση των εικόνων με gaussian noise, εκτός από την πρώτη που είναι χωρίς noise για λόγους σύγκρισης, με τις τιμές που υπάρχουν στην εκφώνηση συν μια ακόμα εικόνα με gaussian noise, μέσης τιμής 0 και διακύμανσης 0.2 για λόγους σύγκρισης και καλύτερης παρουσίασης των εικόνων.

Έπειτα εφαρμόζεται average filter στις τέσσερις εικόνες και το συμπέρασμα είναι πως το average filter δεν καθαρίζει την εικόνα, αλλά αντιθέτως την ασπρίζει.

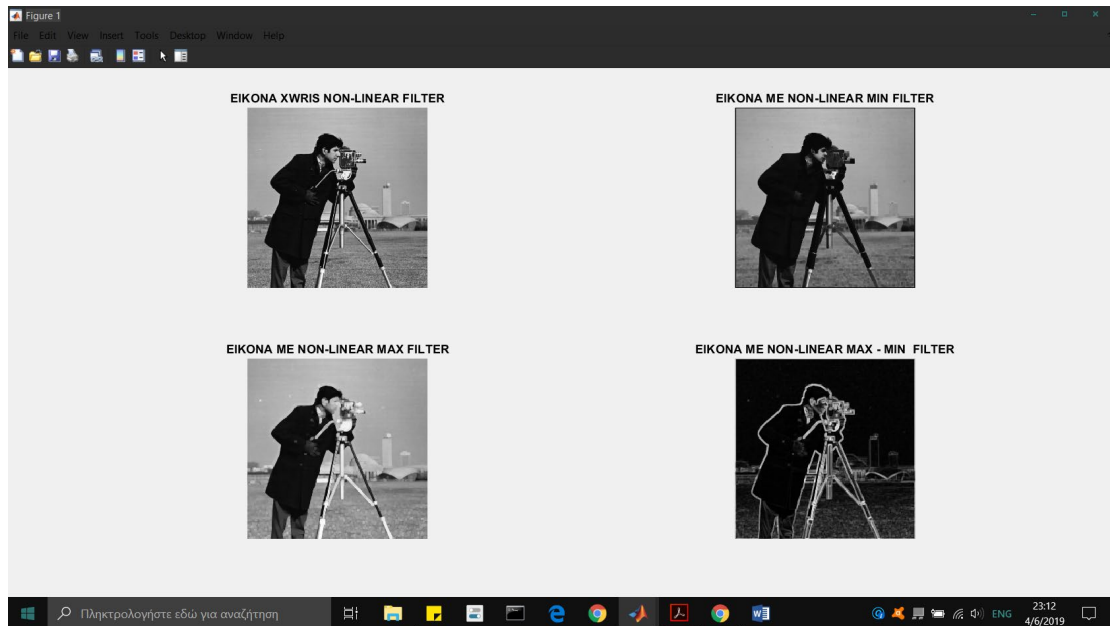
Τέλος εφαρμόζεται wiener filter σε κάθε εικόνα. Το wiener filter είναι πολύ πιο αποδοτικό από το average filter και καθαρίζει την εικόνα καλύτερα. Βέβαια όσο μεγαλύτερος ο θόρυβος τόσο λιγότερο καθαρή είναι η εικόνα μετά την εφαρμογή του φίλτρου.

ΕΡΩΤΗΜΑ 6

ΚΩΔΙΚΑΣ

```
function [p,el] = diffmaxmin (I, n)
p = imread(I);
el = n;
subplot(2,2,1);
imshow(p);
title('ΕΙΚΟΝΑ ΧΩΡΙΣ NON-LINEAR FILTER');
cmin=nlfilter(p,[n,n],'min(x(:))');
subplot(2,2,2);
imshow(cmin);
title('ΕΙΚΟΝΑ ΜΕ NON-LINEAR MIN FILTER');
cmax=nlfilter(p,[n,n],'max(x(:))');
subplot(2,2,3);
imshow(cmax);
title('ΕΙΚΟΝΑ ΜΕ NON-LINEAR MAX FILTER');
c = cmax-cmin;
subplot(2,2,4);
imshow(c);
title('ΕΙΚΟΝΑ ΜΕ NON-LINEAR MAX - MIN FILTER');
!!! ΤΡΕΧΟΥΜΕ ΤΗΝ ΣΥΝΑΡΤΗΣΗ ΜΕ ΤΗΝ ΕΝΤΟΛΗ
diffmaxmin('onoma_eikonas',arithmos); !!!
```

SCREENSHOT



ΠΑΡΑΤΗΡΗΣΕΙΣ

Με \min non-linear filter η εικόνα γίνεται πιο σκοτεινή, λιγότερο ευκρινής. Επίσης οι γκρι λεπτομέρειες της εικόνας έχουν γίνει πιο σκούρο γκρι, τείνουν προς το μαύρο.

Με \max non-linear filter η εικόνα γίνεται λιγότερη ευκρινής, πίο πολύ από το \min non-linear filter, και χρησιμοποιεί λιγότερες αποχρώσεις του γκρι. Επίσης οι γκρι λεπτομέρειες της εικόνας έχουν γίνει άσπρες.

Με \max non-linear filter - \min non-linear filter η εικόνα γίνεται πολύ πιο σκοτεινή με τις λεπτομέρειες να έχουν μαύρο περίγραμμα.