



Επεξεργασία Εικόνας και Βίντεο

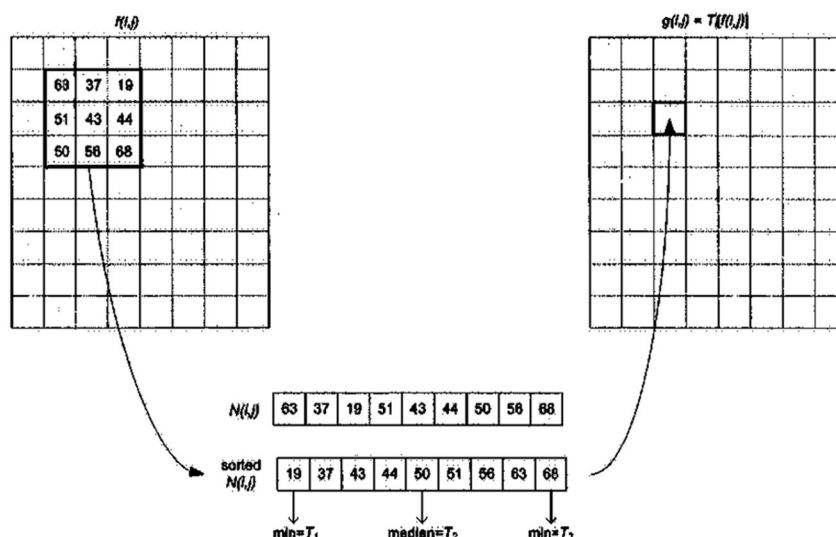
3^η Εργαστηριακή Άσκηση Φίλτρα εικόνων με MATLAB

Περιεχόμενα

1. Εισαγωγή στα φίλτρα Εικόνας
2. Εντολές Φίλτρων στην εργαλειοθήκη (Image Processing Toolbox)
3. Ασκήσεις
4. Λύσεις Ασκήσεων

ΕΙΣΑΓΩΓΗ

Η εφαρμογή φίλτρων είναι ίσως η πιο βασική μέθοδος προεπεξεργασίας (preprocessing) ή βελτίωσης (enhancement) μιας ψηφιακής εικόνας. Ανάλογα με την φύση του φίλτρου οι μέθοδοι μπορούν να τονίσουν ακμές, ή αντικείμενα, να «λειάνουν» περιοχές, ή και τα δύο ταυτόχρονα. Το φιλτράρισμα βασίζεται στην κίνηση ενός κυλιόμενου παραθύρου, το οποίο σαρώνει ολόκληρη την επιφάνεια της εικόνας και συνήθως αντικαθιστά την τιμή της φωτεινότητας του κεντρικού εικονοστοιχείου με τιμή η οποία υπολογίζεται ανάλογα με το φίλτρο. Το μέγεθος του παραθύρου είναι μια παράμετρος, η οποία μπορεί να διαφέρει ανάλογα με την εφαρμογή. Στο παρακάτω σχήμα φαίνεται για παράδειγμα η εφαρμογή ενός φίλτρου διαμέσου (median filter) με μέγεθος 3 επί 3. Ένα τυχαίο τρέχον παράθυρο φαίνεται στο αριστερό μέρος την εικόνας, καθώς αυτή σαρώνεται. Από το παράθυρο αυτό λαμβάνονται όλες οι φωτεινότητες των εικονοστοιχείων, ταξινομούνται με αύξουσα σειρά και επιλέγεται η διάμεσος τιμή δηλαδή η κεντρική τιμή τις κατάταξης. Στην τελική φιλτραρισμένη εικόνα η τιμή της φωτεινότητας του εικονοστοιχείου που βρίσκεται στο κέντρο αυτού του παραθύρου αντικαθίσταται με την διάμεσο τιμή.



Σχήμα 1: Εφαρμογή Φίλτρου διάμεσου

Με την ίδια ακριβώς λογική μπορεί να υλοποιηθεί το φίλτρο μέσης τιμής, το οποίο έχει ως αποτέλεσμα την λείανση περιοχών. Κάποια βασικά φίλτρα επεξεργασίας εικόνας φαίνονται στον παρακάτω πίνακα.

1. Εντολές Εισαγωγής-Εξαγωγής

Φίλτρο	Εντολή MATLAB	Περιγραφή
1 Γκαουσιανό	imgaussfilt	2-D Gaussian filtering of images
2 Wiener	wiener2	2-D adaptive noise-removal filtering
3 Διάμεσου	medfilt2	2-D median filtering
4 Τυπικής Απόκλισης	stdfilt	Local standard deviation of image
5 Εύρους	rangefilt	Local range of image
6 Εντροπίας	entropyfilt	Local entropy of grayscale image



ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ

Άσκηση 1

Εισάγεται μια εικόνα από τον φάκελο datasets και υλοποιήστε ένα φίλτρο μέσης τιμής (mean filter). Ορίστε σε παράμετρο το μέγεθος του φίλτρου. Μετά την εφαρμογή του φίλτρου εμφανίστε και την αρχική εικόνα αλλά και την φιλτραρισμένη με σκοπό να φανούν οι διαφορές τους.

Υπόδειξη: Ακολουθείστε τα βήματα της υπόδειξης για να βοηθηθείτε στην συγγραφή του κώδικά

1. Διαβάστε την Εικόνα από το αρχείο τις με την εντολή **imread()**.
2. Αρχικοποιήστε μια εικόνα με μηδενικές τιμές ίδιου μεγέθους με την αρχική, ώστε αυτή να είναι η τελική φιλτραρισμένη
3. Ορίστε σε μεταβλητή το μέγεθος του παραθύρου. Π.χ. αν το φίλτρο είναι 3x3 ορίστε του ίσο με 3.
4. Υπολογίστε από το μέγεθος του φίλτρου πόσο είναι το offset από το κεντρικό Pixel σε κάθε πλευρά.
5. Αρχικοποιήστε ένα μηδενικό παράθυρο διάστασης ίσης με το παράθυρο που θα σαρώσουμε την εικόνα.
6. Υλοποιήστε διπλό φωλιασμένο βρόχο **for** ώστε να τρέχει στις γραμμές και τις στήλες.
7. Για κάθε τρέχον παράθυρο μέσα στον δεύτερο φωλιασμένο βρόχο να υπολογίζεται η μέση τιμή της φωτεινότητας του παραθύρου.
8. Βάλτε την μέση τιμή που υπολογίσατε στην κεντρική θέση που «τρέχει» το παράθυρο, στην τελική φιλτραρισμένη εικόνα.
9. Τερματίστε τους δύο βρόχους
10. Εμφανίστε και την αρχική εικόνα και την τελική σε ξεχωριστά σχήματα με την εντολή **figure, imshow()**

Τι παρατηρείτε στην φιλτραρισμένη εικόνα σε σχέση με την αρχική;

Άσκηση 2

Υλοποιήστε κατά αντιστοιχία ένα φίλτρο διάμεσου. Εφαρμόστε το σε εικόνα με salt and peper θόρυβο και δείτε τα αποτελέσματα. Τι παρατηρείτε;

Υπόδειξη: Χρησιμοποιήστε την ίδια ακριβώς δομή με την πρώτη άσκηση αλλάζοντας μόνο τον κώδικα μέσα στους βρόχους ώστε να υπολογίζεται η διάμεσος τιμή αντί της μέσης.



ΛΥΣΕΙΣ

ΑΣΚΗΣΗ 1

```
% FILTERS
clc, clear, close all % Καθαρίζει το command line, το workspace,
                      % και κλείνει όλα τα figures

cd('datasets') % Για να μπει στον φάκελο με τις εικόνες
I = imread('Noise_salt_and_pepper.png'); % Διαβάζει την εικόνα
filtered = uint8(zeros(size(I,1),size(I,2))); % Αρχικοποιεί την φιλτραρισμένη εικόνα με
                                             % μηδενικά

megethos_parathyrou = 3; % Το μέγεθος του κυλιόμενου παραθύρου
offset = megethos_parathyrou/2 - 0.5; % Η "ακτίνα" του παραθύρου
parathyro = zeros(megethos_parathyrou,megethos_parathyrou); % Αρχικοποίηση παραθύρου

for i=1+offset:size(I,1)-offset % τρέχει στις γραμμές
    for j=1+offset:size(I,2)-offset % τρέχει στις στήλες
        parathyro(:, :) = I(i-offset:i+offset, j-offset:j+offset); % Το τρέχον παράθυρο
        filtered(i, j) = uint8(round(mean2(parathyro))); % Η μέση τιμή όλου το παραθύρου
    end
end

figure, imshow(I) % Για να εμφανίσει την αρχική εικόνα
figure, imshow(filtered) % Εμφανίζει την φιλτραρισμένη εικόνα

cd('..')
```

ΑΣΚΗΣΗ 2

```
clc, clear, close all % Καθαρίζει το command line, το workspace,
                      % και κλείνει όλα τα figures

cd('datasets') % Για να μπει στον φάκελο με τις εικόνες
I = imread('Noise_salt_and_pepper.png'); % Διαβάζει την εικόνα
filtered = uint8(zeros(size(I,1),size(I,2))); % Αρχικοποιεί την φιλτραρισμένη εικόνα με
                                             % μηδενικά

megethos_parathyrou = 3; % Το μέγεθος του κυλιόμενου παραθύρου
offset = megethos_parathyrou/2 - 0.5; % Η "ακτίνα" του παραθύρου
parathyro = zeros(megethos_parathyrou,megethos_parathyrou); % Αρχικοποίηση παραθύρου

for i=1+offset:size(I,1)-offset % τρέχει στις γραμμές
    for j=1+offset:size(I,2)-offset % τρέχει στις στήλες

        parathyro(:, :) = I(i-offset:i+offset, j-offset:j+offset); % Το τρέχον παράθυρο
        line_parathyro = parathyro(:);
        median_parathyroy = median(line_parathyro);
        filtered(i, j) = median_parathyroy;

    end
end

figure, imshow(I) % Για να εμφανίσει την αρχική εικόνα
figure, imshow(filtered) % Εμφανίζει την φιλτραρισμένη εικόνα

cd('..')
```