

Όραση Υπολογιστών

1^η Εργαστηριακή Άσκηση

Ημερομηνία: 03/04/2021

Ονοματεπώνυμο	A.M.
Σιαφαρίκας Χρήστος	03117123
Τσιγγέλης Κωνσταντίνος	03117149

Θέμα: Εντοπισμός Σημείων Ενδιαφέροντος και Εξαγωγή Χαρακτηριστικών σε Εικόνες

Μέρος 1: Ανίχνευση Ακμών σε Γκρίζες Εικόνες

Άσκηση 1.1. Δημιουργία Εικόνων Εισόδου

(1.1.1)

Δίνεται η γκρίζα εικόνα $I_0 = \text{"edgeset_10.png"}$, η οποία διαβάζεται με τη βοήθεια συναρτήσεων της `opencv` δίνοντας το ακόλουθο αποτέλεσμα:

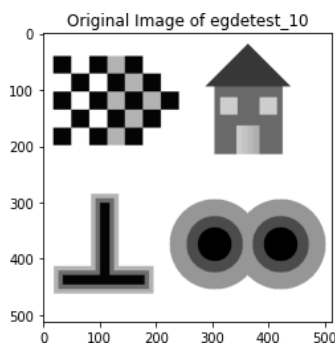


Figure 1. Αρχική εικόνα `edgetest_10.png`

Για να είναι πιο διακριτά τα σχήματα, η εικόνα μετατράπηκε σε binary. Στην αρχική εικόνα όταν το pixel είχε τιμή 255 έδινε λευκό χρώμα, όταν είχε τιμή 0 έδινε μαύρο, ενώ όταν είχε κάποια τιμή στο διάστημα $[0,255]$ έδινε την αντίστοιχη απόχρωση γκριζου χρώματος. Κανονικοποιώντας την εικόνα σε float image με τιμές από 0.0 μέχρι 1.0, γίνεται η μετατροπή της εικόνας σε binary δίνοντας έτσι μαύρο όταν η τιμή του pixel είναι κάτω από 0.5 και λευκό αντιθέτως, όπως φαίνεται στην Εικόνα 2.

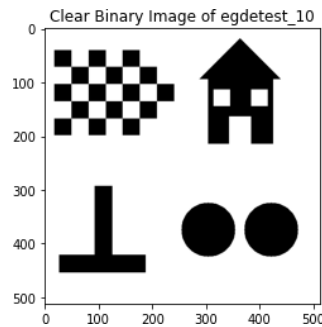


Figure 2. Binary εικόνα edgetest_10.png

(1.1.2)

Για την ανίχνευση ακμών είναι απαραίτητο η εικόνα να έχει θόρυβο, για αυτό το λόγο θα προστεθεί λευκός Gaussian θόρυβος στην edgetest_10.png, σύμφωνα με την σχέση 1.1 :

$$I(x, y) = I_0(x, y) + n(x, y) \quad 1.1$$

Ο λευκός Gaussian θόρυβος θα πρέπει να έχει μέση τιμή 0 και τυπική απόκλιση σ_n τέτοια ώστε ο peak-to-peak Signal to Noise Ratio (PSNR) να ισούται με 10dB την μία περίπτωση και 20dB την άλλη. Ο τύπος για το PSNR δίνεται από την ακόλουθη σχέση 1.2 :

$$PSNR = 20 \log_{10} \left(\frac{I_{max} - I_{min}}{\sigma_n} \right), \text{ σε dB} \quad 1.2$$

όπου

- $I_{max} = \max \{ I(x, y) \}$
- $I_{min} = \min \{ I(x, y) \}$

Από την εξίσωση 1.2, λύνοντας ως προς σ_n προκύπτει η ακόλουθη σχέση 1.3 για το σ_n , η οποία χρησιμοποιήθηκε για την υλοποίηση μιας συνάρτησης, η οποία προσθέτει θόρυβο σε κάθε pixel της εικόνας.

$$\sigma_n = \frac{Imax - Imin}{10^{PSNR/20}} \quad 1.3$$

Επομένως, με χρήση της συνάρτησης `getTheScale` (η οποία δημιουργήθηκε στον κώδικα) γυρίζει την τυπική απόκλιση σ_n , η οποία χρησιμοποιήθηκε για την προσθήκη θορύβου στην εικόνα, δίνοντας τις ακόλουθες θορυβώδεις εικόνες 3 και 4.

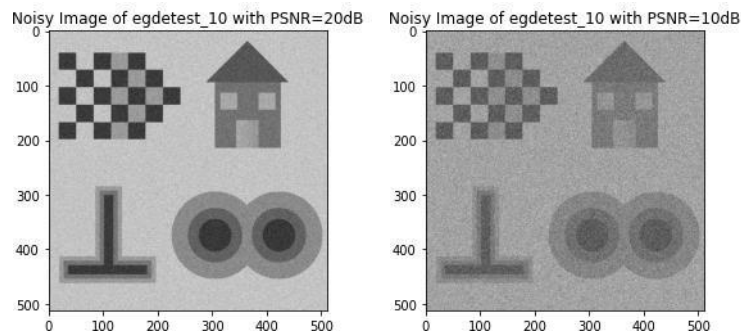


Figure 3. Εικόνα "edgetest_10.png" με θόρυβο PSNR=20dB αριστερά & PSNR=10dB δεξιά

Άσκηση 1.2. Υλοποίηση Αλγορίθμων Ανίχνευσης Ακμών

Σε αυτό το ερώτημα γίνεται η υλοποίηση της συνάρτησης `EdgeDetect`, η οποία υλοποιεί τον αλγόριθμο ανίχνευσης ακμών σε γκριζες εικόνες. Η διαδικασία κατασκευής της περιγράφεται στα ακόλουθα βήματα.

Σημείωση: Στα παρακάτω βήματα περιέχονται εικόνες, που προέκυψαν ως αποτέλεσμα υλοποίησης κάθε σταδίου υλοποίησης του αλγορίθμου. Επομένως, καθώς η αναφορά προχωρούσε, ο αλγόριθμος αναβαθμιζόταν δίνοντας διαφορετικές εξόδους, με τα στάδια συγχωνευμένα. Ο πιο ακριβής αντιπροσωπευτικός κώδικας κάθε σταδίου παρατίθεται σε ένα επιπλέον αρχείο, όπου αν τρέξει φαίνεται η έξοδος κάθε βήματος.

(1.2.1) Δημιουργία φίλτρων

Σε αυτό το βήμα γίνεται η υλοποίηση του μέρους της συνάρτησης , στο οποίο γίνεται η δημιουργία δύο διακριτών γραμμικών φίλτρων, που προσεγγίζουν τα συνεχή φίλτρα, και εφαρμόζονται διαδοχικά στην εικόνα. Συγκεκριμένα τα φίλτρα, αυτά, έχουν τις ακόλουθες κρουστικές αποκρίσεις:

i. Δισδιάστατη Gaussian

$$G_{\sigma}(x, y)$$

1.4

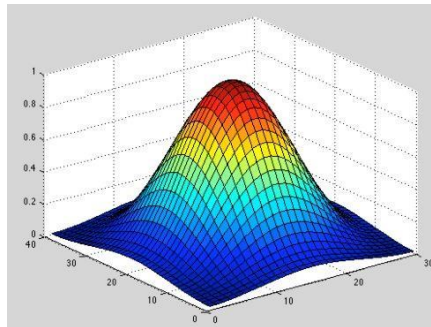


Figure 4. Οπτικοποίηση Gaussian 2D φίλτρου

δίνοντας το ακόλουθο φιλτράρισμα στις εικόνες:

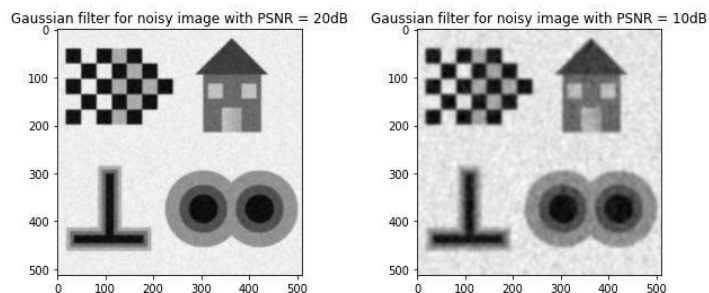


Figure 5. Εικόνα "edgetest_10.png" μετά από φιλτράρισμα με Gaussian φίλτρο για θόρυβο PSNR=20dB αριστερά & PSNR=10dB δεξιά

ii. Laplacian of Gaussian

$$(\text{LoG}) h(x, y) = \nabla^2 G_{\sigma}(x, y) = \frac{e^{-r^2/2\sigma^2}}{2\pi\sigma^4} \left(2 \frac{r^2}{\sigma^2} \right), \text{ με } r = \sqrt{x^2 + y^2} \quad 1.5$$

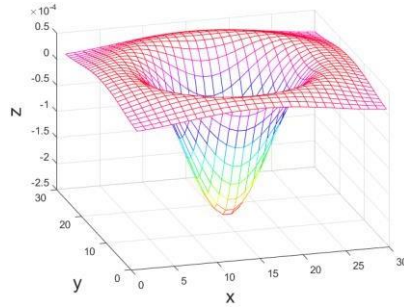


Figure 6. Οπτικοποίηση LoG φίλτρου γνωστού και ως "Inverse Mexican Hat"

δίνοντας το ακόλουθο φιλτράρισμα στις εικόνες:

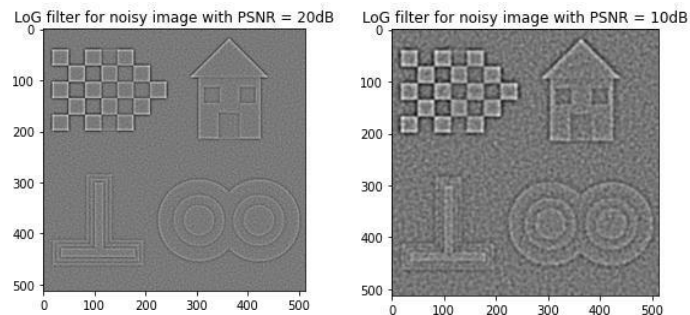


Figure 7. Εικόνα "edgetest_10.png" μετά από φιλτράρισμα με Laplacian of Gaussian φίλτρο για θόρυβο PSNR=20dB αριστερά & PSNR=10dB δεξιά

Στις παραπάνω εικόνες έχουν γίνει πλέον διακριτά τα περιγράμματα των σχημάτων, των οποίων θα σχηματιστούν οι ακμές τους.

(1.2.2) Χρήση τελεστή 2^{ης} παραγώγου (Laplacian)

Σε αυτό το στάδιο θα γίνει η προσέγγιση της Laplacian της εξομαλυμένης εικόνας με μια γραμμική και μια μη γραμμική μέθοδο. Πρώτα, χρειάζεται να αναλυθούν οι ακόλουθες θεωρητικές σχέσεις:

1. Δίσκος B: δομικό στοιχείο στο οποίο ανήκουν τα σημεία στα οποία αν τους προσθέσουμε μια τιμή z (πχ μια ακτίνα) το $x+z$ παραμένει στοιχείο του δίσκου, όπως εκφράζει και μαθηματικά ο ακόλουθος τύπος.

$$X \mapsto X_{+z} \triangleq \{x + z : x \in X\} \quad 1.6$$

2. Dilation: η ένωση ενός χώρου X και ενός δίσκου B , όταν αυτός εφαρμόζεται πάνω σε όλα τα σημεία του χώρου X , όπως εκφράζει και μαθηματικά ο ακόλουθος τύπος.

$$\delta_B(X) = X \oplus B \triangleq \{z : (B^s)_{+z} \cap X \neq \emptyset\} = \cup_{b \in B} X_{+b} \quad 1.7$$

3. Erosion: η τομή ενός χώρου X και ενός δίσκου B , όταν αυτός εφαρμόζεται πάνω σε όλα τα σημεία του χώρου X , όπως εκφράζει και μαθηματικά ο ακόλουθος τύπος.

$$\varepsilon_B(X) = X \ominus B \triangleq \{z : B_{+z} \subseteq X\} = \cap_{b \in B} X_{-b} \quad 1.8$$

Συνεπώς, με γνώση των παραπάνω σχέσεων μπορούν να προκύψουν οι ακόλουθες εκτιμήσεις της Laplacian.

- i. Γραμμική προσέγγιση L_1 : συνέλιξη της I με την LoG h , όπως εκφράζει και μαθηματικά ο ακόλουθος τύπος

$$L_1 = \nabla^2 (G_\sigma * I) = (\nabla^2 G_\sigma) * I = h * I \quad 1.9$$

δηλαδή διαδοχική εφαρμογή των φίλτρων που δημιουργήθηκαν στο παραπάνω ερώτημα, δίνοντας με αυτό τον τρόπο τον ακόλουθο φιλτράρισμα στις θορυβώδεις εικόνες.

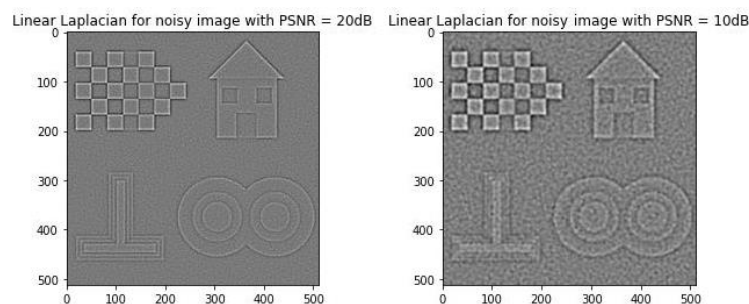


Figure 8. Γραμμική προσέγγιση της Laplacian για τις θορυβώδεις εικόνες με PSNR=20dB αριστερά και PSNR=10dB δεξιά

- ii. Μη Γραμμική προσέγγιση L_2 : μη γραμμική εκτίμηση της Laplacian της I_σ , δηλαδή της εικόνας που έχει περάσει από ένα Gaussian filter 2D, σύμφωνα με την ακόλουθη σχέση.

$$L_2 = I_{\sigma} \oplus B + I_{\sigma} \ominus B - 2 * I_{\sigma} \quad 1.10$$

δηλαδή εφαρμογή πρώτα ενός dilation και ύστερα ενός erosion, δίνοντας με αυτό τον τρόπο τον ακόλουθο φιλτράρισμα στις θορυβώδεις εικόνες.

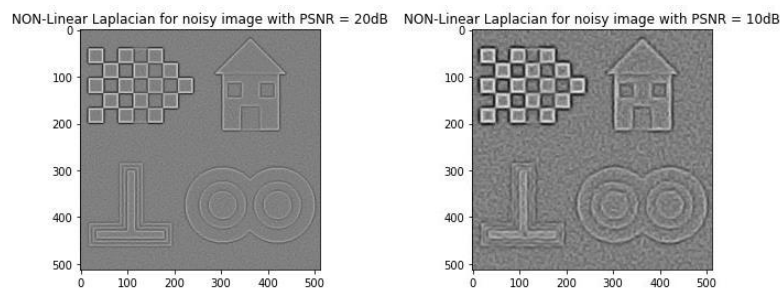


Figure 9. Μη γραμμική προσέγγιση της Laplacian για τις θορυβώδεις εικόνες με PSNR=20dB αριστερά και PSNR=10dB δεξιά

Συγκρίνοντας τις παραπάνω εκτιμήσεις της Laplacian μεταξύ τους παρατηρείται ότι είτε επιλεγεί η γραμμική είτε η μη γραμμική μέθοδος και οι δύο καταλήγουν στην ίδια επεξεργασμένη εικόνα, σε διαφορετική γκρι απόχρωση. Επιπλέον, αυτοί οι δύο τρόποι προσέγγισης καταλήγουν σε σωστό αποτέλεσμα, μιας και οι εικόνες των Figure.8 , Figure.9 είναι ίδιες με αυτές της Figure.7 .

Σημείωση: Στα επόμενα βήματα παρατίθενται οι εικόνες που αντιστοιχούν στη γραμμική προσέγγιση της Laplacian. Δε χρησιμοποιήθηκαν και οι δύο για να μην πλατειάσει η αναφορά, καθώς όπως δείχθηκε στο ερώτημα (1.2.2) κατέληξαν και οι δύο στην ίδια επεξεργασμένη εικόνα.

(1.2.3) Εντοπισμός σημείων μηδενισμού (τοπικά ακρότατα-zero crossing)

Σε αυτό το στάδιο θα γίνει η προσέγγιση των σημείων μηδενισμού (zerocrossings) της εικόνας L , που προέκυψε από τις προσεγγίσεις της Laplacian στο ερώτημα (1.2.2).

Για την προσέγγιση των zerocrossings, λοιπόν, δημιουργήθηκε πρώτα μια binary εικόνα στην οποία μονάδα έχουν μόνο οι τιμές της εικόνας L , που ήταν θετικές, όπως φαίνεται ακολούθως.

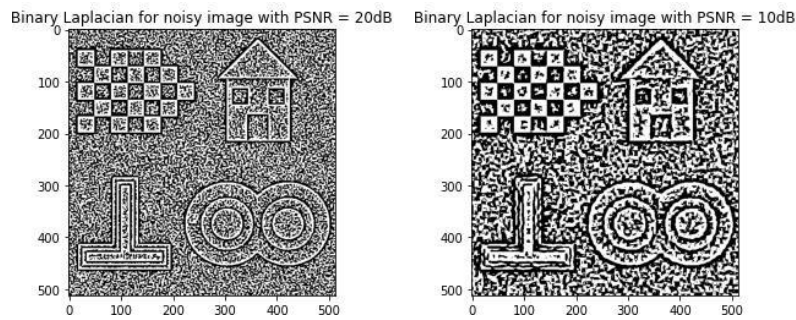


Figure 10. Δυναδική εικόνα L για τις θορυβώδεις εικόνες με PSNR=20dB αριστερά και PSNR=10dB δεξιά

Έπειτα, χρειάζεται η εύρεση του περιγράμματος της δυναδικής εικόνας που φτιάχτηκε στο προηγούμενο βήμα, σύμφωνα με την ακόλουθη σχέση.

$$Y = X \oplus B - X \ominus B \approx \partial X \quad 1.11$$

Με υλοποίηση αυτής της σχέσης σε κώδικα python προκύπτουν οι ακόλουθες επεξεργασμένες εικόνες, στις οποίες τα zerocrossings αντιστοιχούν στα σημεία στα οποία η δυναδική εικόνα Y έχει την τιμή 1.

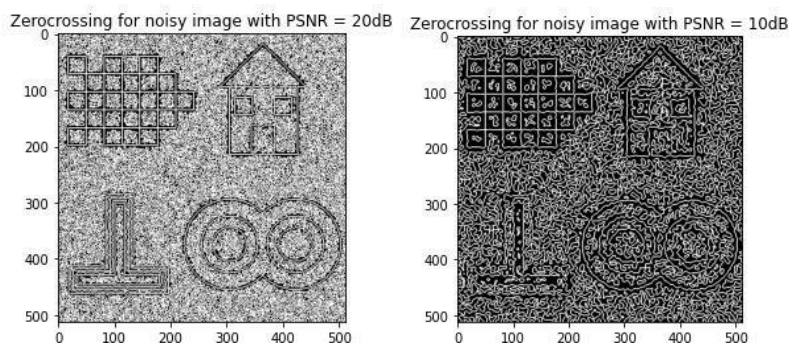


Figure 11. Επεξεργασμένες εικόνες για προσέγγιση zerocrossing των θορυβωδών εικόνων με PSNR=20dB αριστερά και PSNR=10dB δεξιά

(1.2.4) Κατωφλιοποίηση (απόρριψη zerocrossing σε ομαλές περιοχές)

Σε αυτό το στάδιο της συνάρτησης EdgeDetect θα χρησιμοποιηθεί ένα κριτήριο απόρριψης των zerocrossing σε ομαλές περιοχές, καθώς οι ακμές διακρίνονται από απότομη μεταβολή 1.0 σε 0.0. Για αυτό το λόγο, διατρέχοντας λογικά κάθε pixel, απορρίπτονται εκείνα τα οποία μεν έχουν τιμή 1, όμως η παράγωγος της

γκαουσιανής εικόνας στο αντίστοιχο pixel δεν ξεπερνάει ένα κατώφλι, το οποίο εκφράζεται από την ακόλουθη σχέση.

$$Y[i, j] = 1 \ \&\& \ ||\nabla I_{\sigma}[i, j]|| > \theta_{\text{edge}} * \max ||\nabla I_{\sigma}|| \quad 1.12$$

Συνεπώς, με εφαρμογή της συνάρτησης EdgeDetect, τα βήματα της οποίας αναλύθηκαν στα προηγούμενα υποερωτήματα προκύπτουν οι ακόλουθες ακμές με γραμμική και μη γραμμική προσέγγιση της LoG αντίστοιχα.

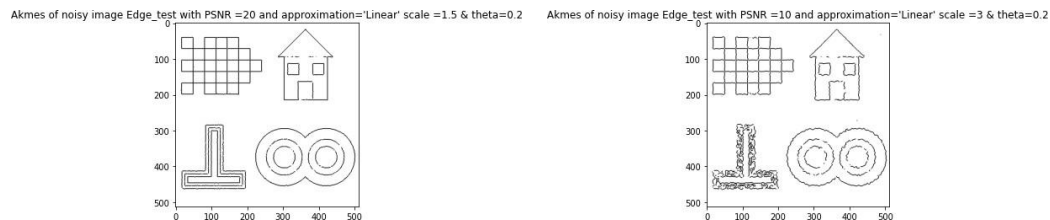


Figure 12. Οι ακμές που προέκυψαν με χρήση της EdgeDetect στην εικόνα Edge_test για γραμμική προσέγγιση για PSNR=20dB αριστερά και PSNR=10dB δεξιά

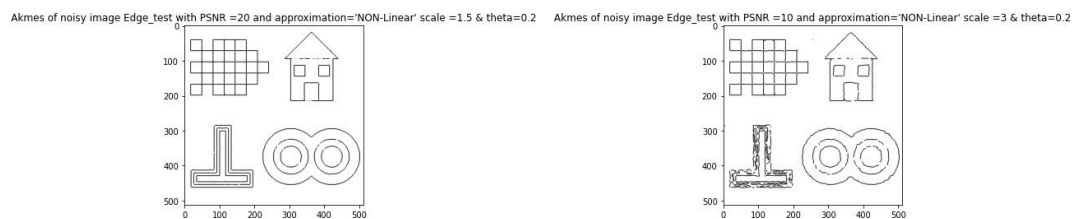


Figure 13. Οι ακμές που προέκυψαν με χρήση της EdgeDetect στην εικόνα Edge_test για μη γραμμική προσέγγιση για PSNR=20dB αριστερά και PSNR=10dB δεξιά

Συγκρίνοντας, τις παραπάνω εικόνες παρατηρείται ότι η γραμμική και η μη-γραμμική προσέγγιση, προσομοιάζουν αρκετά καλά τη Laplacian, ανιχνεύοντας με αρκετά καλό ποσοστό επιτυχίας τις ακμές στην εικόνα, καταλήγοντας στην ίδια επεξεργασμένη εικόνα περίπου.

Επιπλέον, φαίνεται ότι όταν χρησιμοποιήθηκε θορύβου PSNR=20dB οι ακμές είναι πιο ευκρινείς σε σχέση με χρήση θορύβου PSNR=10dB.

Άσκηση 1.3. Αξιολόγηση των Αποτελεσμάτων Ανίχνευσης Ακμών

(1.3.1)

Σε αυτό το σημείο θα γίνει η δημιουργία μιας εικόνας περιέχει μόνο τις ακμές των σχημάτων με χρήση της αρχικής εικόνας I_0 , που δεν έχει θόρυβο. Για να γίνει αυτό υπολογίζεται πρώτα μια εικόνα M , σύμφωνα με την ακόλουθη σχέση.

$$M = I_0 \oplus B - I_0 \ominus B \quad 1.13$$

δηλαδή η εικόνα M προκύπτει από εφαρμογή ενός dilation και με αφαίρεση ενός erosion από την αρχική εικόνα I_0 .

Ύστερα, η εικόνα αυτή μετατρέπεται σε δυαδική με κατωφλιοποίηση, χρησιμοποιώντας ως κατώφλι $\theta_{\text{edgereal}} = 0.2$, δίνοντας με αυτό τον τρόπο την ακόλουθη εικόνα T .

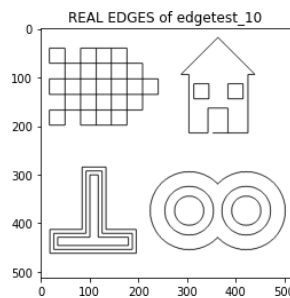


Figure 14. Ακμές σχημάτων όπως προέκυψαν από επεξεργασία της αρχικής εικόνα I_0

(1.3.2)

Παραπάνω υλοποιηθήκαν δύο αλγόριθμοι υπολογισμού ακμών για εικόνες με θόρυβο και χωρίς. Η εικόνα χωρίς θόρυβο δίνει τις ακριβείς ακμές των σχημάτων μιας εικόνας, καθώς είναι η ιδανική περίπτωση, ενώ ο αλγόριθμος υπολογισμού ακμών μιας θορυβώδους εικόνας δίνει λιγότερο ακριβοί αποτελέσματα, όμως πιο κοντά στην πραγματικότητα. Για τον έλεγχο της αποδοτικότητας του αλγορίθμου υπολογισμού ακμών υλοποιήθηκε μια συνάρτηση, η οποία υπολογίζει προγραμματιστικά το ακόλουθο ποσοστό.

$$C = [\text{Pr}(D/T) + \text{Pr}(T/D)]/2$$

1.14

όπου D: εικόνα ακμών που προέκυψε από θορυβώδη εικόνα με την EdgeDetect

T: εικόνα ακμών που προέκυψε από εικόνα χωρίς θόρυβο

$\text{Pr}(D/T)$: ποσοστό των ανιχνευθεισών ακμών που είναι αληθινές

$\text{Pr}(T/D)$: ποσοστό των αληθινών ακμών που ανιχνεύθηκαν

Η συνάρτηση αυτή χρησιμοποιήθηκε στις θορυβώδεις εικόνες του Edge_test, από τις οποίες είχαν σχηματιστεί οι ακμές των σχημάτων με χρήση της EdgeDetect, δίνοντας τα ακόλουθα αποτελέσματα:

Για γραμμική προσέγγιση της Edge test:

Για PSNR = 20dB: $C \approx 0.939$ ή 93,9%

Για PSNR = 10dB: $C \approx 0.645$ ή 64,5%

Για μη γραμμική προσέγγιση της Edge test::

Για PSNR = 20dB: $C \approx 0.968$ ή 96,8%

Για PSNR = 10dB: $C \approx 0.739$ ή 73,9%

Συγκρίνοντας τα παραπάνω ποσοστά ως προς την γραμμική ή μη προσέγγιση της LoG κατά την υλοποίηση της EdgeDetect φαίνεται ότι για PSNR = 20dB η απόκλιση του ποσοστού είναι μικρή, σε αντίθεση με το PSNR = 10dB όπου η απόκλιση είναι μεγάλη.

Συγκρίνοντας τα παραπάνω ποσοστά ως προς το PSNR θορύβου φαίνεται ότι για υψηλότερο PSNR έχουμε καλύτερη απόδοση του αλγορίθμου.

(1.3.3)

Για PSNR = 20dB, κρατώντας σταθερή την $\theta_{edge} = 0.2$ και μεταβάλλοντας την τυπική απόκλιση παρατηρήθηκε ότι όσο αυξάνεται το σ μειώνεται και το ποσοστό απόδοσης του αλγορίθμου. Όμως, δεν γίνεται να μειώνεται συνεχώς το σ και η απόδοση να βελτιώνεται, καθώς παρατηρήθηκε ότι από ένα κατώφλι

και μετά μειώνεται πάλι η απόδοση. Αυτό το κατώφλι πειραματικά βρέθηκε ότι είναι το $\sigma=1.5$.

σ (τυπική απόκλιση)	C (ποσοστό)
0.5	0.753
1	0.93
1.5	0.94
2	0.932
3	0.803

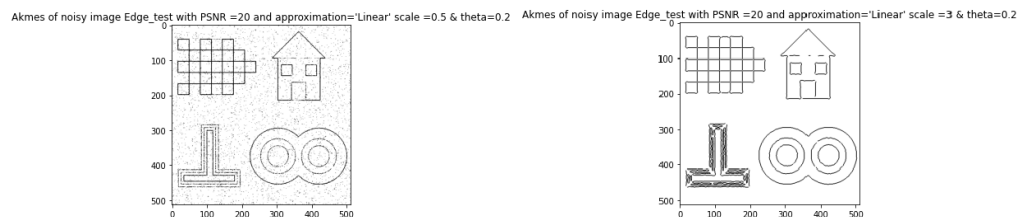


Figure 15. Οι ακμές της edge_test για $\sigma=0.5$ αριστερά και για $\sigma=3$ δεξιά

Για PSNR = 20dB, κρατώντας σταθερή την τυπική απόκλιση $\sigma=1.5$ και μεταβάλλοντας την θ_{edge} παρατηρήθηκε ότι όπως και στην μεταβολή της τυπικής απόκλισης υπάρχει μια τιμή πέρα από την οποία, είτε αυξηθεί είτε μειωθεί το θ , η απόδοση του αλγορίθμου θα πέφτει. Επιπλέον, παρατηρείται και οπτικά ότι με χρήση μεγαλύτερου θ_{edge} κόβονται κάποιες χρήσιμες ακμές, ενώ με χρήση μικρότερου θ_{edge} προστίθενται κάποιες ακμές, σαν ανεπιθύμητος θόρυβος.

θ_{edge}	C (ποσοστό)
0.1	0.89
0.2	0.93
0.3	0.81

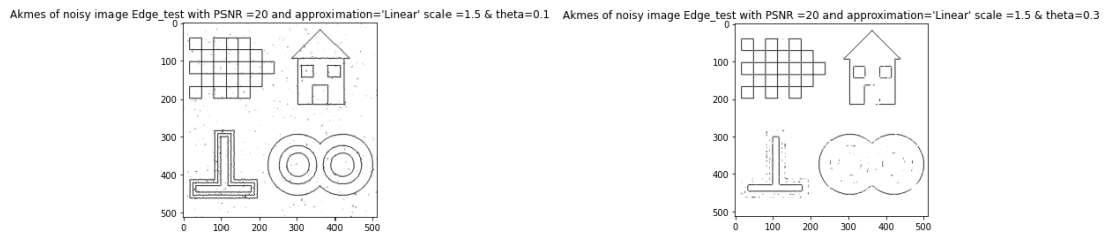


Figure 16. Οι ακμές της edge_test για $\theta=0.1$ αριστερά και για $\theta=0.3$ δεξιά

Για την εύρεση της βέλτιστης εικόνας για PSNR=20dB χρησιμοποιήθηκε η γραμμική προσέγγιση, ενώ παρατηρήθηκε ότι έδινε παρόμοια συμπεριφορά και η μη γραμμική προσέγγιση. Επομένως, η βέλτιστη εικόνα για PSNR=20dB είναι η ακόλουθη:

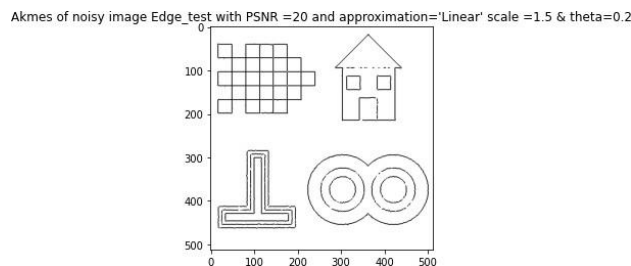


Figure 17. Βέλτιστη εικόνα ακμών για PSNR=20dB

Για PSNR = 10dB, κρατώντας σταθερή την $\theta_{edge} = 0.2$ και μεταβάλλοντας την τυπική απόκλιση παρατηρήθηκε ότι παρόμοια συμπεριφορά με τις μεταβολές για εικόνα θορύβου PSNR = 20dB. Σε αυτή την περίπτωση, όμως, το κατώφλι βρέθηκε ίσο με $\sigma=3$. Επιπλέον, πρέπει να τονιστεί ότι με μείωση του σ από 3 σε 2 παρατηρήθηκε το ποσοστό απόδοσης του αλγόριθμου αυξήθηκε όμως η ποιότητα της εικόνας μειώθηκε. Αυτό οφείλεται στην προσθήκη ακμών τύπου ανεπιθύμητου θορύβου σε επιθυμητά σημεία, τα οποία συνυπολογίζονται στην εύρεση του ποσοστού.

σ (τυπική απόκλιση)	C (ποσοστό)
1	0.461
2	0.682
3	0.639
4	0.591
5	0.544

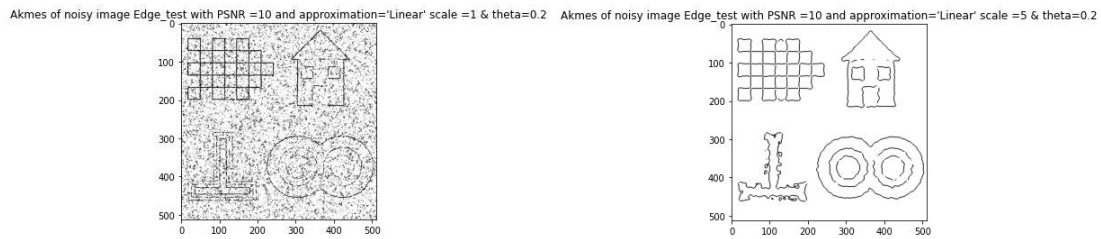


Figure 18.Οι ακμές της edge_test για $\sigma=1$ αριστερά και για $\sigma=5$ δεξιά

Για PSNR = 10dB, κρατώντας σταθερή την τυπική απόκλιση $\sigma=3$ και μεταβάλλοντας την θ_{edge} παρατηρήθηκε ότι παρόμοια συμπεριφορά με τις μεταβολές για εικόνα θορύβου PSNR = 20dB. Σε αυτή την περίπτωση, όμως, το κατώφλι βρέθηκε ίσο με $\theta_{edge}=0.2$.

θ_{edge}	C (ποσοστό)
0.1	0.516
0.2	0.649
0.3	0.646

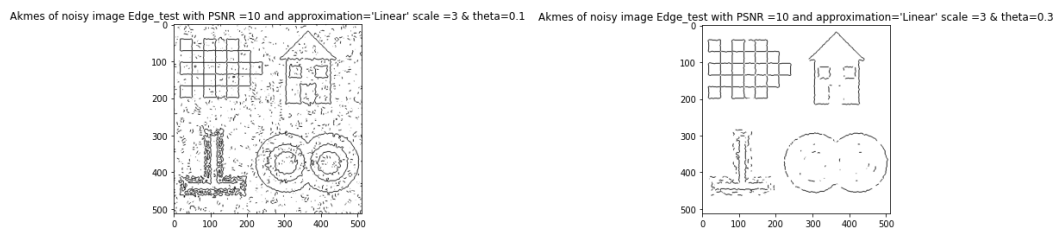


Figure 19.Οι ακμές της edge_test για $\theta=0.1$ αριστερά και για $\theta=0.3$ δεξιά

Για την εύρεση της βέλτιστης εικόνας για PSNR=10dB χρησιμοποιήθηκε η γραμμική προσέγγιση, ενώ παρατηρήθηκε ότι έδινε παρόμοια συμπεριφορά και η μη γραμμική προσέγγιση. Επομένως, η βέλτιστη εικόνα για PSNR=10dB είναι η ακόλουθη:

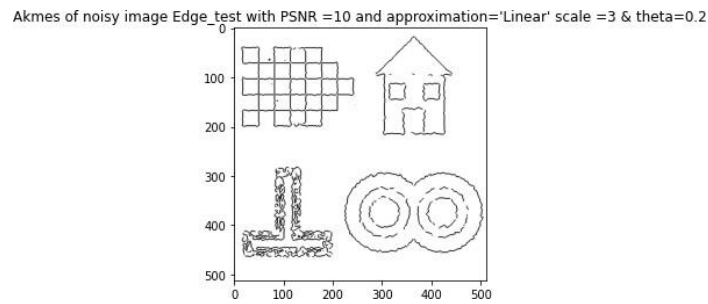


Figure 20. Βέλτιστη εικόνα ακμών για PSNR=10dB

Άσκηση 1.4. Εφαρμογή των Αλγορίθμων Ανίχνευσης Ακμών σε Πραγματικές εικόνες

(1.4.1)

Στην αρχή φορτώθηκε η εικόνα urban_edges και βρέθηκαν οι πραγματικές ακμές της με συνάρτησης που υλοποιήθηκε στο ερώτημα (1.3.1), όπως φαίνεται στην ακόλουθη εικόνα.

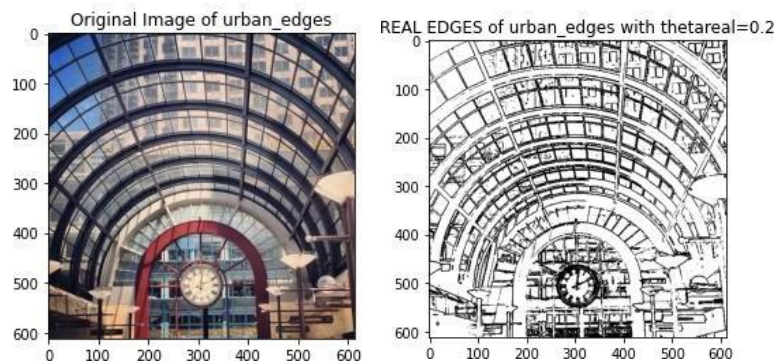


Figure 21. Η αρχική εικόνα urban_edges αριστερά και οι ακμές της δεξιά

Με εφαρμογή των αλγορίθμων ανίχνευσης ακμών EdgeDetect, που υλοποιήθηκαν παραπάνω, προέκυψαν οι ακόλουθες εικόνες.

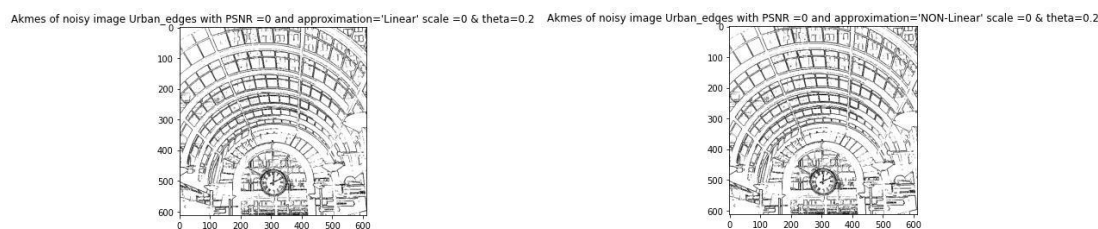


Figure 22. Ακμές που προέκυψαν με εφαρμογή της EdgeDetect για Linear προσέγγιση αριστερά και NON-Linear προσέγγιση δεξιά

Για τον έλεγχο της απόδοσης του αλγορίθμου υπολογισμού ακμών χρησιμοποιήθηκε η συνάρτηση που υλοποιήθηκε στο ερώτημα (1.3.2) δίνοντας τα ακόλουθα ποσοστά.

Για γραμμική προσέγγιση της Urban_edges: $C \approx 0.8187320$

Για μη γραμμική προσέγγιση της Urban_edges: $C \approx 0.8187370$

Επομένως, βγαίνει εύκολα το συμπέρασμα ότι η γραμμική και μη γραμμική μέθοδος για τον υπολογισμό μιας αθόρυβης εικόνας βγάζει παραπλήσια αποτελέσματα και με μεγάλο ποσοστό επιτυχίας.

(1.4.2)

Στην εικόνα Urban_edges, κρατώντας σταθερή την $\theta_{edge} = 0.2$ και μεταβάλλοντας την τυπική απόκλιση παρατηρήθηκε ότι παρόμοια συμπεριφορά με τη μεταβολή της τυπικής απόκλισης για την εικόνα Edge_test, με τη διαφορά ότι σε αυτή την περίπτωση το κατώφλι βρίσκεται στην τιμή $\sigma=0.5$, επιτυγχάνοντας με αυτό τη μέγιστη απόδοση του αλγορίθμου εύρεσης ακμών.

Σ (τυπική απόκλιση)	C (ποσοστό)
0	0.818
0.5	0.874
1	0.782
2	0.615
3	0.471

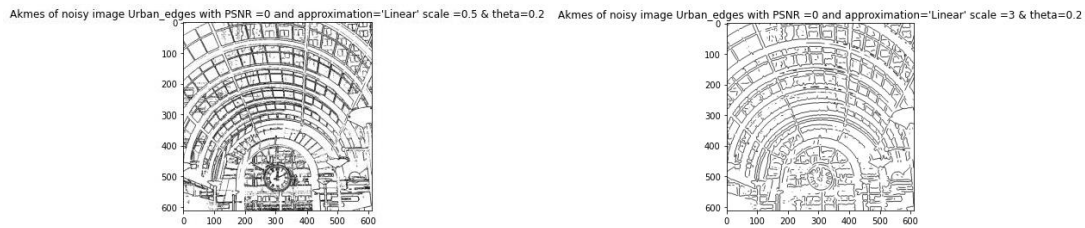


Figure 23. Οι ακμές τις urban_edges για $\sigma=0.5$ αριστερά & $\sigma=5$ δεξιά

Κρατώντας σταθερή την τυπική απόκλιση $\sigma=0.5$ και μεταβάλλοντας την θ_{edge} παρατηρήθηκε παρόμοια συμπεριφορά με τις προηγούμενες μεταβολές, δηλαδή σε αυτή την περίπτωση το peak της συνάρτησης βρίσκεται στην τιμή $\theta_{edge}=0.2$.

Επιπλέον, παρατηρήθηκε ότι για $\theta_{edge}=0.1$ και $\theta_{edge}=0.3$ τα ποσοστά ανίχνευσης ακμών είναι παραπλήσια όμως οι εικόνες διαφέρουν αισθητά. Συγκεκριμένα, για $\theta_{edge}=0.1$ έχουν «περάσει» περισσότερες ακμές στην εικόνα σε σημεία που η real image δεν έχει ακμές, για αυτό και φαίνεται να έχει πιο έντονο χρώμα. Από την άλλη, για $\theta_{edge}=0.3$ «περάσει» λιγότερες ακμές από αυτές που έχει η real image.

θ_{edge}	C (ποσοστό)
0.1	0.793
0.2	0.874
0.3	0.771

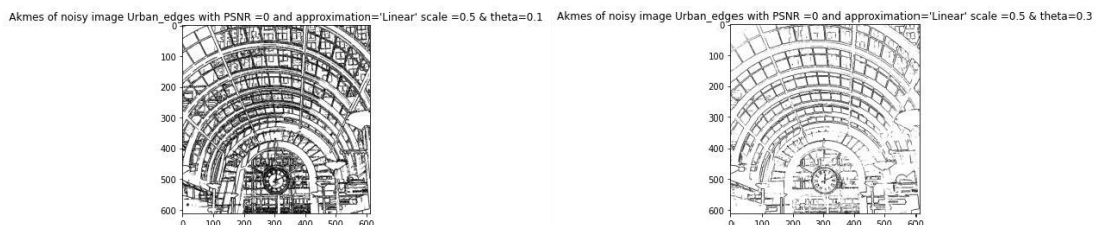


Figure 24. Οι ακμές τις urban_edges για $\theta=0.1$ αριστερά & $\theta=0.3$ δεξιά

Για την εύρεση της βέλτιστης εικόνας χρησιμοποιήθηκε η γραμμική προσέγγιση, ενώ παρατηρήθηκε ότι έδινε παρόμοια συμπεριφορά και η μη γραμμική προσέγγιση. Επομένως, η βέλτιστη εικόνα urban_edges είναι η ακόλουθη:

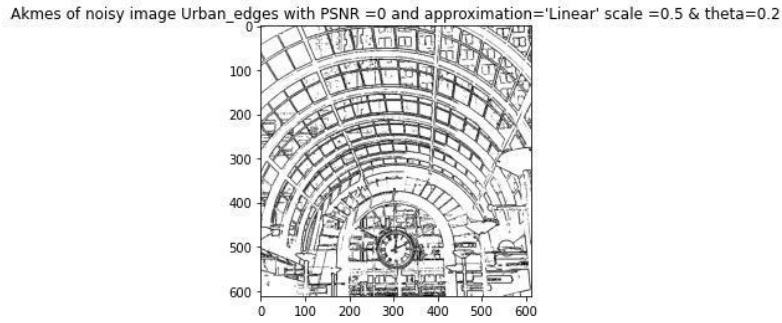


Figure 25. Βέλτιστη εικόνα ακμών urban_edges

Μέρος 2: Ανίχνευση Σημείων Ενδιαφέροντος (Interest Point Detection)

Άσκηση 2.1. Ανίχνευση Γωνιών

Για την ανίχνευση των γωνιών εικόνας θα χρησιμοποιηθεί η μέθοδος Harris-Stephens , η οποία βασίστηκε στη χρήση ενός μικρού χωρικού παραθύρου, το οποίο όταν εντοπίζει μεγάλη μεταβολή στη φωτεινότητα θα έχει εντοπίσει και μια μετατόπιση στην αντίστοιχη κατεύθυνση. Σύμφωνα με αυτή τη λογική η φωτεινότητα μοντελοποιείται από την ακόλουθη σχέση.

$$E(u, v) \approx [u \quad v] * A * \begin{bmatrix} u \\ v \end{bmatrix} \quad 2.1$$

Όπου u, v : μικρές μετατοπίσεις πάνω στην εικόνα

A : 2x2 πίνακας Harris

(2.1.1)

Συγκεκριμένα, ο πίνακας Harris A εκφράζεται από την ακόλουθη σχέση:

$$A(x, y, \sigma) = G_{\rho} * \begin{bmatrix} \left(\frac{\partial I}{\partial x} * \frac{\partial I}{\partial x} \right) & \left(\frac{\partial I}{\partial x} * \frac{\partial I}{\partial y} \right) \\ \left(\frac{\partial I}{\partial x} * \frac{\partial I}{\partial y} \right) & \left(\frac{\partial I}{\partial y} * \frac{\partial I}{\partial y} \right) \end{bmatrix} \quad 2.2$$

Όπου I_σ : εικόνα εξομαλυμένη με Γκαουσιανό πυρήνα G_σ .

G_σ : Γκαουσιανό πυρήνας με τυπική απόκλιση σ (κλίμακα διαφόρισης)

G_ρ : Γκαουσιανό πυρήνας με τυπική απόκλιση ρ (κλίμακα ολοκλήρωσης)

Σε αυτό το στάδιο γίνεται η υλοποίηση αυτών των δομικών τανυστών J σε κάθε pixel της εικόνας. Η μοντελοποίησή τους υλοποιείται με χρήση των στοιχείων του πίνακα στην σχέση 2.2, δίνοντας τις ακόλουθες γκρίζες εικόνες.

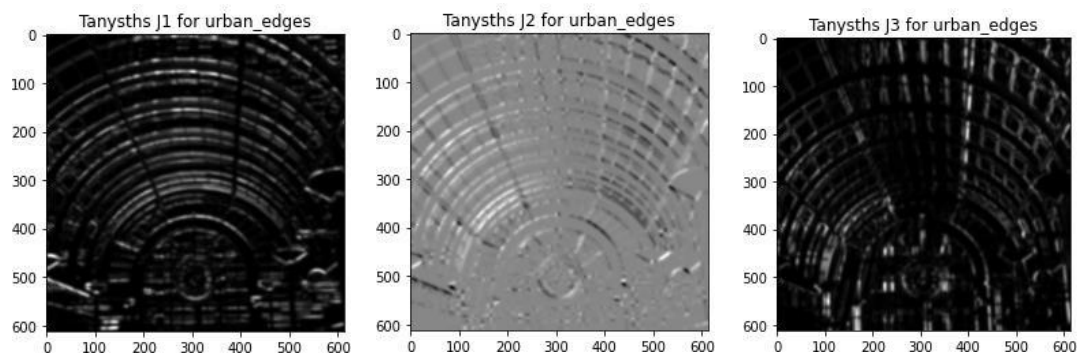


Figure 26. Δομικοί Τανυστές J1, J2, J3 αντιστοίχως για την εικόνα urban_edges

(2.1.2)

Ο πίνακας Harris, που υλοποιήθηκε στο προηγούμενο ερώτημα, αποτελεί ένα επιπλέον κριτήριο ανίχνευσης γωνιών, καθώς μεγάλες ιδιοτιμές σε ένα pixel υποδεικνύουν την ύπαρξη γωνίας. Συγκεκριμένα, διακρίνονται οι ακόλουθες περιπτώσεις:

- $\lambda_1 \approx 0$ & $\lambda_2 \approx 0$: το pixel ανήκει σε ομαλή περιοχή
- $\lambda_1 \gg 0$ & $\lambda_2 \approx 0$ ή $\lambda_1 \approx 0$ & $\lambda_2 \gg 0$: το pixel ανήκει σε ακμή
- $\lambda_1 \gg 0$ & $\lambda_2 \gg 0$: το pixel ανήκει σε γωνία

Σε αυτό το στάδιο γίνεται η υλοποίηση αυτών των ιδιοτιμών με χρήση της σχέσης 2.3:

$$\lambda_{\pm}(x, y) = \frac{1}{2} (J_1 + J_2 \pm \sqrt{(J_1 - J_2)^2 + 4J_3^2}) \quad 2.3$$

δίνοντας τις ακόλουθες γκρίζες εικόνες για τις ιδιοτιμές λ_+ και λ_- αντιστοίχως.

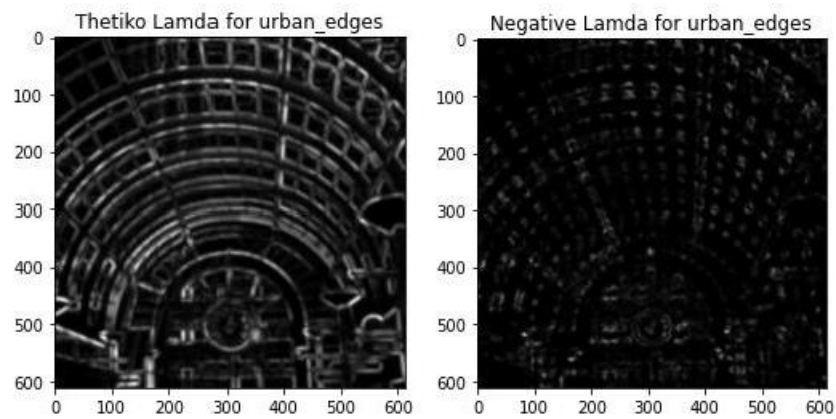


Figure 27. Οι ιδιοτιμές του πίνακα Harris για την εικόνα urban_edges

(2.1.3)

Ο αλγόριθμος Harris-Stephens ολοκληρώνεται με χρήση του κριτηρίου τους 2.4,

$$R = \det(\mathbf{A}) - k \cdot \text{trace}^2(\mathbf{A}) \quad 2.4$$

το οποίο προγραμματιστικά μοντελοποιήθηκε με τη σχέση 2.5

$$R = \lambda_- \cdot \lambda_+ - k \cdot (\lambda_- + \lambda_+)^2 \quad 2.5$$

Αυτό το κριτήριο παίρνει μεγάλες τιμές σε σημεία που αντιστοιχούν σε γωνίες. Από αυτά κρατιούνται μόνο τα τοπικά μέγιστα και απορρίπτονται τα σημεία κάτω από κάποιο προκαθορισμένο κατώφλι. Υλοποιώντας τη σχέση 2.5 προγραμματιστικά προκύπτουν οι ακόλουθες εικόνες για τον αλγόριθμο Harris-Stephens.

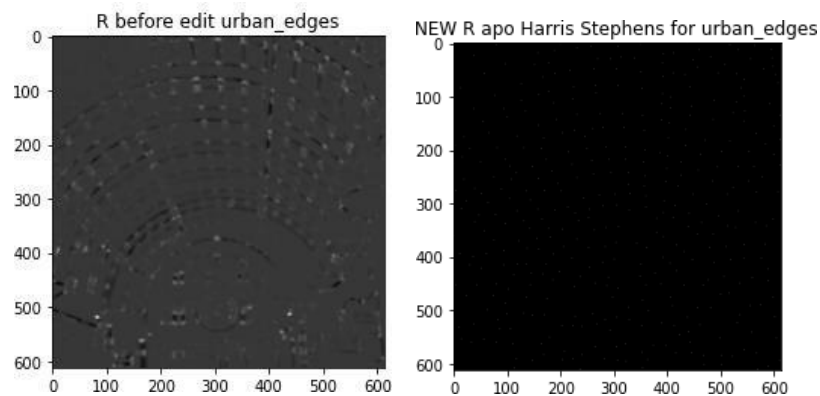


Figure 28. Οι δυαδικές εικόνες που προέκυψαν από τον αλγόριθμο Harris-Stephens, χωρίς το κατώφλι αποκοπής αριστερά, με το κατώφλι αποκοπής δεξιά

Ο αλγόριθμος, που υλοποιήθηκε, βρήκε για την εικόνα urban_edges 398, για την εικόνα mars 160 γωνίες και για την εικόνα blood_smear 171.

Visualization of gwnies for urban edges image for $s = 2$

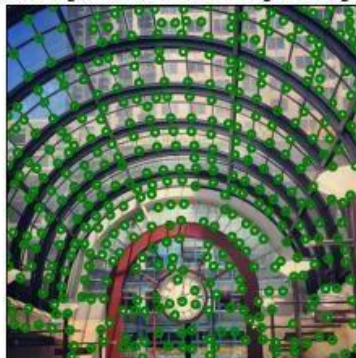


Figure 29. Οπτικοποίηση των γωνιών, που ανιχνεύθηκαν στην εικόνα urban_edges για τυπική απόκλιση $\sigma=2$

Άσκηση 2.2. Πολυκλιμακωτή Ανίχνευση Γωνιών

Ο ανιχνευτής Harris-Stephens, που υλοποιήθηκε στο ερώτημα (2.1), έχει ένα βασικό μειονέκτημα, ότι δεν είναι ανεξάρτητος ως προς την κλίμακα, μιας και χρησιμοποιείται ένα προκαθορισμένου μεγέθους Γκαουσιανό φίλτρο. Για αυτό το λόγο, είναι επιθυμητή η πολυκλιμακωτή ανίχνευση γωνιών για διάφορες τιμές του σ .

(2.2.1)

Το πρώτο στάδιο της πολυκλιμακωτής ανίχνευσης γωνιών αποτελείται από την εφαρμογή του αλγορίθμου εύρεσης γωνιών μονής κλίμακας για διαφορετικές κλίμακες ολοκλήρωσης και διαφόρισης. Για αυτό το λόγο κατασκευάζεται μια Γκαουσιανή πυραμίδα με τιμές που δίνονται από τη σχέση 2.6

$$\begin{aligned}\sigma_0, \sigma_1, \dots, \sigma_{N-1} &= s^0\sigma_0, s^1\sigma_0, \dots, s^{N-1}\sigma_0 \\ \rho_0, \rho_1, \dots, \rho_{N-1} &= s^0\rho_0, s^1\rho_0, \dots, s^{N-1}\rho_0\end{aligned}\quad 2.6$$

Όπου σ_0 : αρχική κλίμακα διαφόρισης

ρ_0 : αρχική κλίμακα ολοκλήρωσης

(2.2.2)

Η πολυκλιμακωτή ανίχνευση γωνιών με τον αλγόριθμο Harris-Stephens θα επεκταθεί με χρήση της Λαπλασιανής ώστε να ανιχνεύει γωνίες σε πολλαπλές κλίμακες. Συγκεκριμένα, κανονικοποιημένη LoG ως προς κάποια κλίμακα σ για $I = 0, \dots, N-1$ επίπεδα ανάλυσης, δίνεται από την σχέση 2.7

$$|LoG(x, y, \sigma_i)| = \sigma_i^2 |L_{xx}(x, y, \sigma_i) + L_{yy}(x, y, \sigma_i)| \quad 2.7$$

Ο αλγόριθμος, που προέκυψε, ονομάζεται Harris-Laplace. Αυτός, αρχικά, ανιχνεύει τις γωνίες που προκύπτουν από το κριτήριο Harris-Stephens για κάθε κλίμακα ξεχωριστά. Έπειτα, απορρίπτει τις γωνίες, για τις οποίες η κλίμακα που ανιχνεύθηκαν δεν μεγιστοποιεί τη LoG μετρική σε μια γειτονιά διαδοχικών γειτονικών κλιμάκων.

Τα αποτελέσματα αυτής της συνάρτησης εισήχθησαν σαν ορίσματα στη δοθείσα συνάρτηση **interest_points_visualization** δίνοντας τις ακόλουθες εικόνες, στις οποίες φαίνονται οι γωνίες, που εντοπίστηκαν, μέσα σε κύκλους διαφορετικών μεγεθών, αντίστοιχης κλίμακας της τυπικής απόκλισης που ανιχνεύθηκαν.

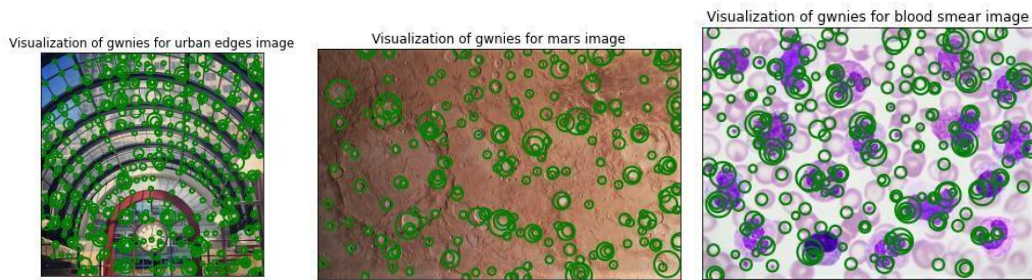


Figure 30. Οι γωνίες που ανιχνεύθηκαν για την εικόνα mars.png αριστερά & blood_smear.jpg δεξιά

Ο αλγόριθμος, που υλοποιήθηκε, βρήκε για την εικόνα urban_edges 509 , για την εικόνα mars 221 γωνίες και για την εικόνα blood_smear 233.

Άσκηση 2.3. Ανίχνευση Blobs

Με τον όρο blobs ορίζονται περιοχές με κάποια ομογένεια, οι οποίες διαφέρουν σημαντικά από τη γειτονιά τους.

(2.3.1)

Η ανίχνευση των blobs ακολουθεί παρόμοια λογική με αυτή της ανίχνευσης ακμών με τη διαφορά ότι βασίζεται στον πίνακα Hessian. Ο πίνακας Hessian ορίζεται σύμφωνα με τη σχέση 2.8.

$$H(x, y) = \begin{bmatrix} Lxx(x, y, \sigma) & Lxy(x, y, \sigma) \\ Lxy(x, y, \sigma) & Lyy(x, y, \sigma) \end{bmatrix} \quad 2.8$$

$$\text{Όπου } Lxx(x, y, \sigma) = \frac{\partial^2}{\partial x^2} \{I\sigma(x, y)\}$$

$$Lyy(x, y, \sigma) = \frac{\partial^2}{\partial y^2} \{I\sigma(x, y)\}$$

$$Lxy(x, y, \sigma) = \frac{\partial^2}{\partial x \partial y} \{I\sigma(x, y)\}$$

(2.3.2)

Όπως και στον ανιχνευτή Harris, έτσι και στον Hessian τα ιδιοδιανύσματα δίνουν τις κατευθύνσεις της μέγιστης και της ελάχιστης καμπυλότητας και οι ιδιοτιμές το μέγεθος της καμπυλότητας.

Σε αυτή την περίπτωση το κριτήριο σημαντικότητας προκύπτει από την ορίζουσα του πίνακα Hessian ως μετρική (Baudet Operator), σύμφωνα με την ακόλουθη σχέση.

$$R = \det(H(x, y)) = L_{xx} * L_{yy} - L_{xy}^2 \quad 2.9$$

Επομένως, με χρήση του αλγορίθμου Hessian ανιχνεύθηκαν τα blobs που φαίνονται στις ακόλουθες δυαδικές εικόνες.

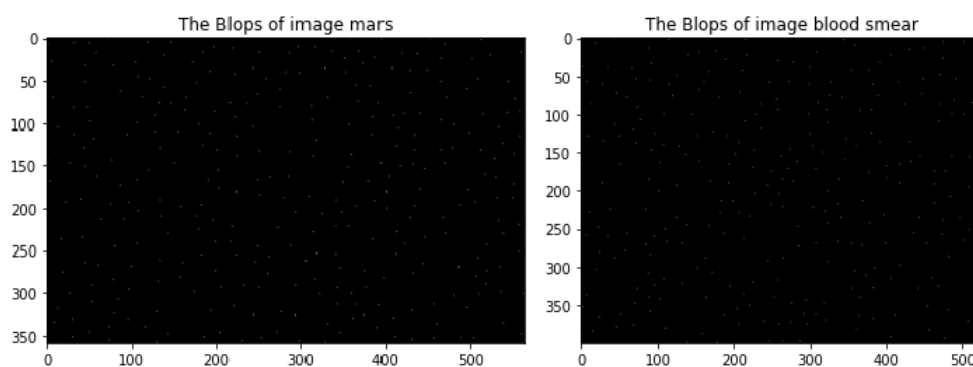


Figure 31. Τα blobs που ανιχνεύθηκαν για την δυαδική εικόνα mars.png αριστερά & blood_smear.jpg δεξιά

Ο αλγόριθμος, που υλοποιήθηκε, βρήκε για την εικόνα mars 312 blobs και για την εικόνα blood_smear 278.

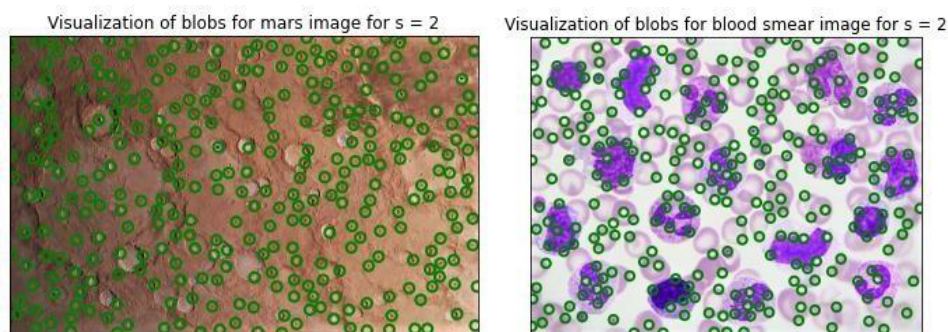


Figure 32. Οπτικοποίηση των blobs που ανιχνεύθηκαν για την εικόνα mars.png αριστερά & blood_smear.jpg δεξιά

Άσκηση 2.4. Πολυκλιμακωτή Ανίχνευση Blobs

(2.4.1)

Ακολουθώντας παρόμοια λογική με τον ανιχνευτή Harris-Stephens, θα γίνει επέκταση του αλγορίθμου Hessian με χρήση της Λαπλασιανής. Τα αποτελέσματα αυτής της συνάρτησης εισήχθησαν σαν ορίσματα στη δοθείσα συνάρτηση `interest_points_visualization` δίνοντας τις ακόλουθες εικόνες, στις οποίες φαίνονται τα blobs, που εντοπίστηκαν, μέσα σε κύκλους διαφορετικών μεγεθών, αντίστοιχης κλίμακας της τυπικής απόκλισης που ανιχνεύθηκαν.

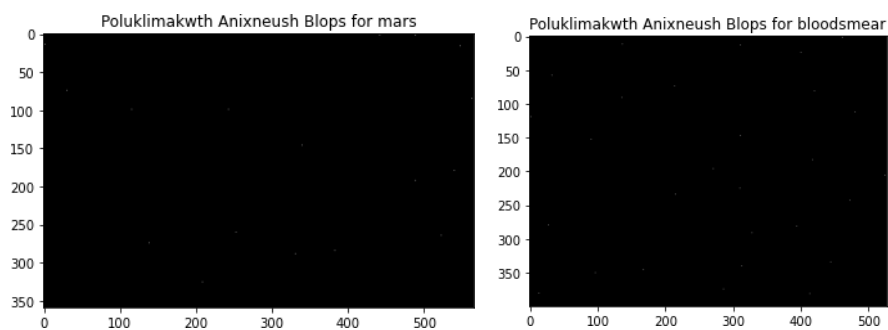


Figure 33. Τα πολυκλιμακωτά blobs που ανιχνεύθηκαν για την δυαδική εικόνα `mars.png` αριστερά & `blood_smear.jpg` δεξιά

Ο αλγόριθμος, που υλοποιήθηκε, βρήκε για την εικόνα `mars` 406 blobs και για την εικόνα `blood_smear` 324.

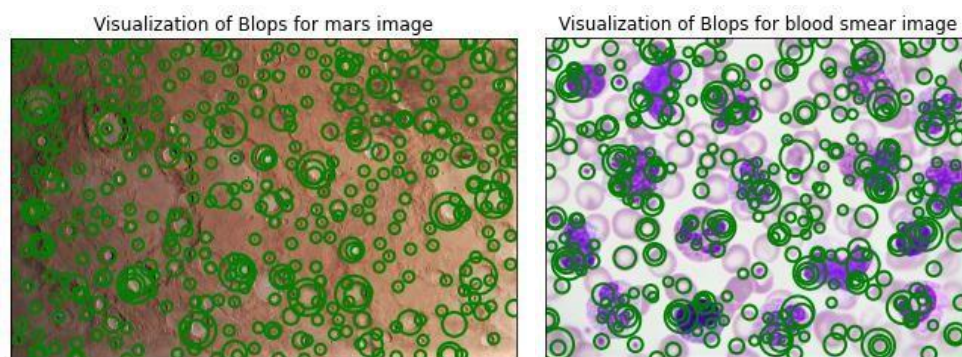


Figure 34. Τα πολυκλιμακωτά blobs που ανιχνεύθηκαν για την εικόνα `mars.png` αριστερά & `blood_smear.jpg` δεξιά

Άσκηση 2.5. Επιτάχυνση με την χρήση Box Filters και Ολοκληρωτικών Εικόνων (Integral Images)

Με τον όρο Box Filters ορίζονται φίλτρα που βασίζονται σε αθροίσματα ορθογώνιων περιοχών, ενώ με τον όρο ολοκληρωτική εικόνα ορίζεται μια εικόνα, στην οποία κάθε στοιχείο προκύπτει από το άθροισμα των πάνω αριστερά στοιχείων του.

(2.5.1)

Για την κατασκευή της ολοκληρωτικής εικόνας, για τις εικόνες mars και blood_smear, χρησιμοποιήθηκε η συνάρτηση `numpy.cumsum` αντίστοιχως, δίνοντας τα ακόλουθα αποτελέσματα.

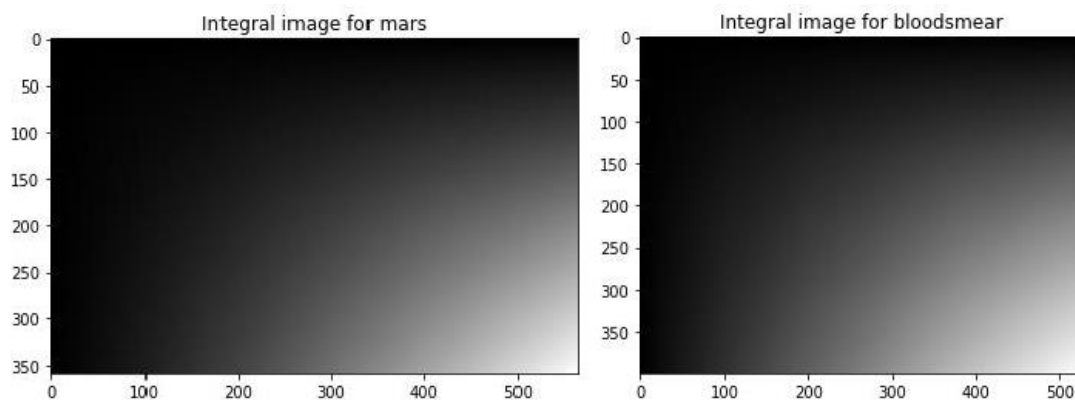


Figure 35. Ολοκληρωτική εικόνα για την εικόνα mars αριστερά και την blood_smear δεξιά

Στις παραπάνω εικόνες φαίνεται ότι το κάθε pixel έχει προέλθει από το άθροισμα όλων το στοιχείων που βρίσκονται πάνω και αριστερά του, για τις εικόνες mars και blood_smear αντίστοιχα.

(2.5.2)

Στο στάδιο αυτό θα υλοποιηθούν τρία box filters με τα οποία επιταχύνεται η διαδικασία εύρεσης σημείων ενδιαφέροντος. Συγκεκριμένα, τα box filters αποτελούν ένα παράθυρο με το οποίο συνελίσσεται η προς επεξεργασία εικόνα, δηλαδή κάθε στοιχείο της εικόνας θα πρέπει να πολλαπλασιαστεί με τον

αντίστοιχο συντελεστή παραθύρου. Τα φίλτρα αυτά έχουν την ακόλουθη μορφή:

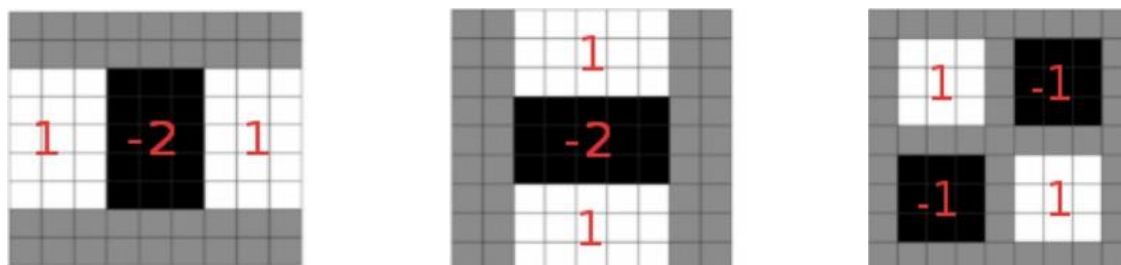


Figure 36. Οπτικοποίηση box filters του Dxx αριστερά , του Dyy στο κέντρο και του Dxy δεξιά

Στις παραπάνω εικόνες των box filters, το μαύρο χρώμα αντιστοιχεί σε περιοχή συντελεστή πολλαπλασιασμού ίσο με -2, το λευκό χρώμα αντιστοιχεί σε περιοχή συντελεστή πολλαπλασιασμού ίσο με 1 και το γκριζό χρώμα αντιστοιχεί σε περιοχή συντελεστή πολλαπλασιασμού ίσο με 0.

Για να γίνει αυτή η συνέλιξη μεταξύ box filter και εικόνας χρησιμοποιήθηκε η σχέση ότι το άθροισμα των στοιχείων εντός του παράθυρου της εικόνας 33, που οριοθετείται από τα σημεία A,B,C και D, υπολογίζεται με χρήση της ολοκληρωτικής εικόνας σύμφωνα με την σχέση 2. και ισοδυναμεί με τον υπολογισμό του εμβαδού του σκιασμένου κομματιού της εικόνας 33.

$$\sum \sum_{ABCD} I(i, j) = S_A + S_C - S_B - S_D \quad 2.10$$

Όπου S_A : η ολοκληρωτική εικόνα μέχρι το σημείο A , ομοίως για τα υπόλοιπα σημεία

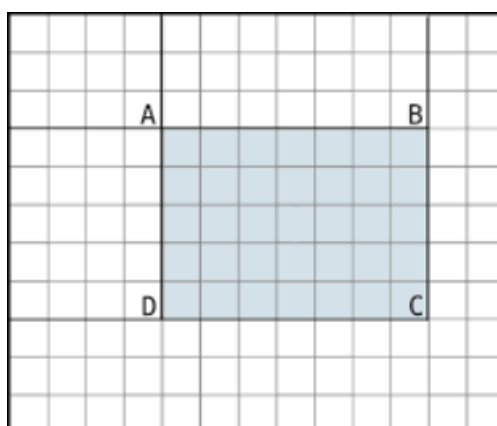


Figure 37. Παράθυρο υπολογισμού αθροίσματος

Σύμφωνα με αυτή τη λογική, υλοποιήθηκαν συναρτήσεις που να πολλαπλασιάζουν κατάλληλα κάθε pixel της ολοκληρωτικής εικόνας με αντίστοιχο συντελεστή ανά περιοχή, δίνοντας τις ακόλουθες μερικές παραγώγους δεύτερης τάξης της εικόνας L_{xx} , L_{xy} , L_{yy} .

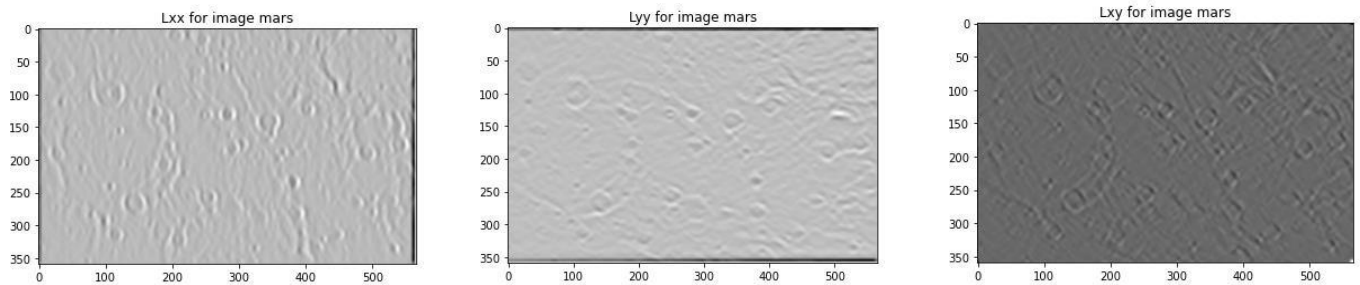


Figure 38. Οπτικοποίηση του L_{xx} αριστερά, του L_{yy} στο κέντρο και του L_{xy} δεξιά

Συγκρίνοντας, τα παραπάνω αποτελέσματα με τις μερικές παραγώγους δεύτερης τάξης της εικόνας που προέκυψαν κατά το ερώτημα (2.3) φαίνεται ότι τα box filters λειτούργησαν σωστά, επιταχύνοντας τη διαδικασία υπολογισμού.

Παρατήρηση: Όπως παρατηρούμε στις οπτικοποιήσεις των μερικών παραγώγων με τη χρήση Box Filters προκύπτουν περιφερειακά(πάνω και κάτω) κάποιες μαύρες στήλες (στο L_{xx}) και γραμμές (στο L_{yy}) όπου διαπιστώσαμε πως μεταφράζονται σε σημεία ενδιαφέροντος (blobs) στο visualization των blobs της εικόνας mars -την οποία και δοκιμάσαμε- με αυτόν τον αλγόριθμο(παρατίθενται παρακάτω). Για να αποφύγουμε τον εντοπισμό ψευδών σημείων ενδιαφέροντος και με τη σύμβαση πως η πληροφορία που μας ενδιαφέρει –τουλάχιστον στα πλαίσια του εργαστηρίου- βρίσκεται κατά κόρον στο κέντρο της εικόνας και όχι στην περιφέρεια αφαιρέσαμε τα blobs των 5 πάνω και κάτω γραμμών και αριστερών και δεξιών στηλών της τελικής εικόνας.

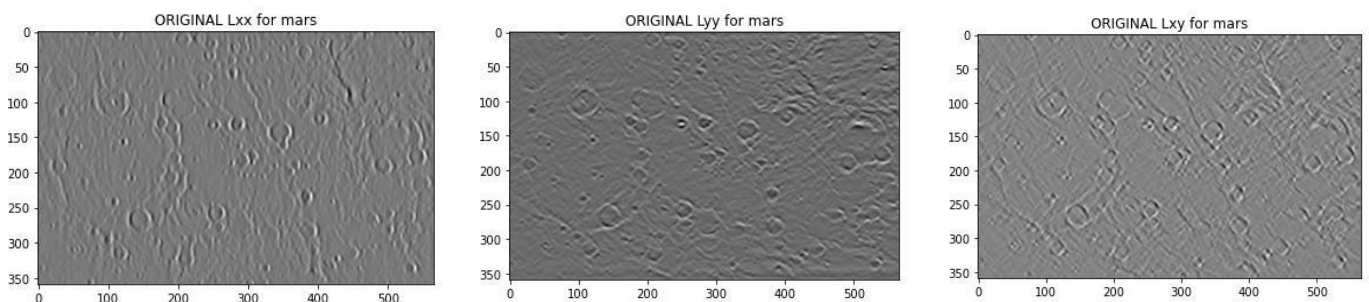


Figure 39. Μερικές παραγώγους δεύτερης τάξης της εικόνας mars όπου είναι L_{xx} αριστερά, L_{xy} στη μέση & L_{yy} δεξιά

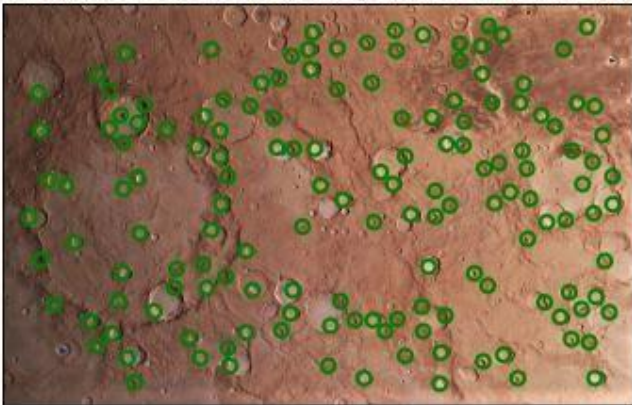
(2.5.3)

Από αυτά τα σημεία που βρέθηκαν από την εφαρμογή των box filters στη εικόνα απορρίπτονται όσα δεν πληρούν τις προϋποθέσεις του προσεγγιστικού κριτηρίου Hessian, όπως περιγράφεται από την ακόλουθη σχέση.

$$R = \det(H(x, y)) = L_{xx} * L_{yy} - (0.9 * L_{xy})^2 \quad 2.12$$

Με εφαρμογή του παραπάνω κριτηρίου αποκοπής και οπτικοποίηση των αποτελεσμάτων με χρήση της συνάρτησης `interest_points_visualization`, δίνονται τα ακόλουθα blops για μία κλίμακα $\sigma=2$.

Visualization of Blops for mars image with BOX FILTERS for $s=2$



Visualization of Blops for blood smear image with BOX FILTERS for $s=2$

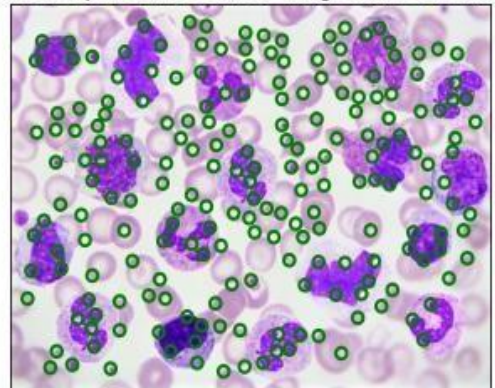
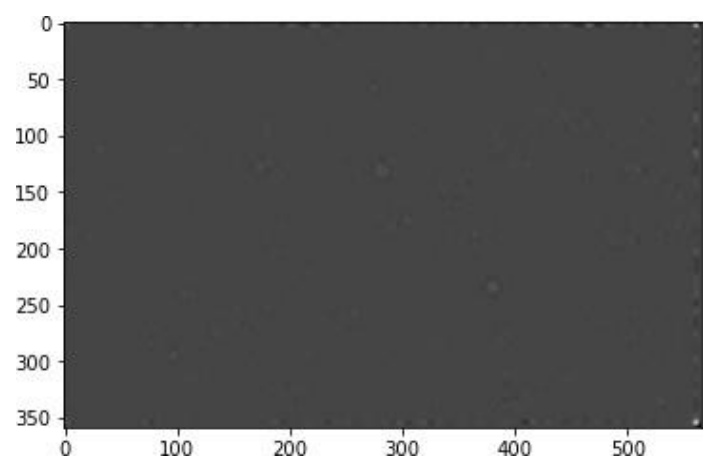
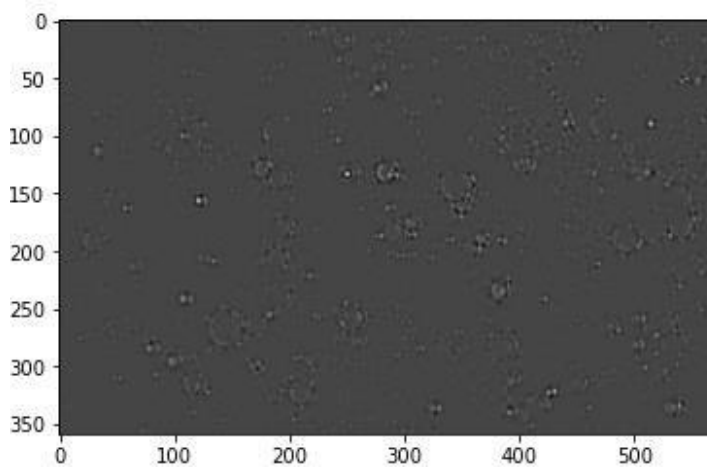


Figure 38. Οπτικοποίηση blops για τυπική απόκλιση $\sigma=2$ για την εικόνα mars αριστερά και blood smear δεξιά

Στη συνέχεια οπτικοποιούμε το κριτήριο R της απλής Hessian μεθόδου και το παραπάνω προσεγγιστικό κριτήριο R σε 4 κλίμακες ($s = 2, 4, 6, 8$). Παρατίθενται παρακάτω οι προκύπτουσες οπτικοποιήσεις αλλά και ο αριθμός των blops που ανιχνεύθηκαν:

Visualisation of R with Harris-Hessian and Box Filters for $s=2$,respectively:

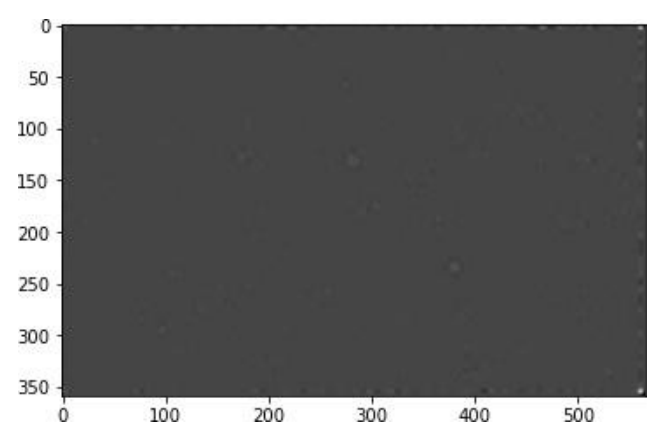
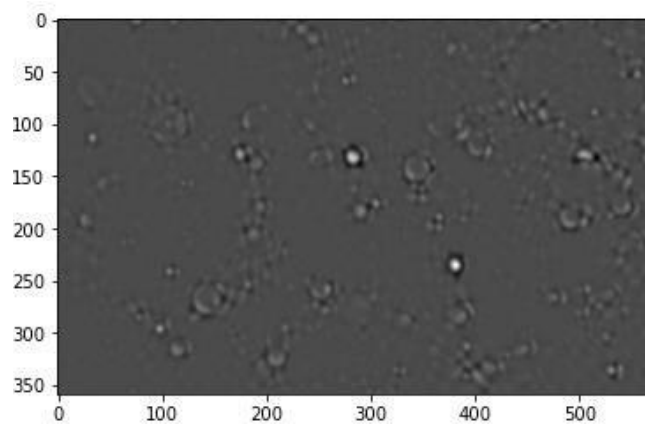


Approach's quality comparison for $s=2$:

The number of Blops of Rmars with Harris-Hessian for $s=2$ is: 312

The number of Blops of Rmars with Box Filters for $s=2$ is: 149

Visualisation of R with Harris-Hessian and Box Filters for $s=4$,respectively:

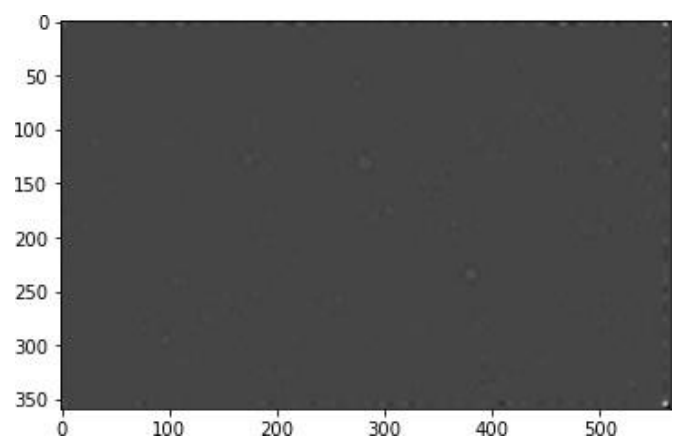
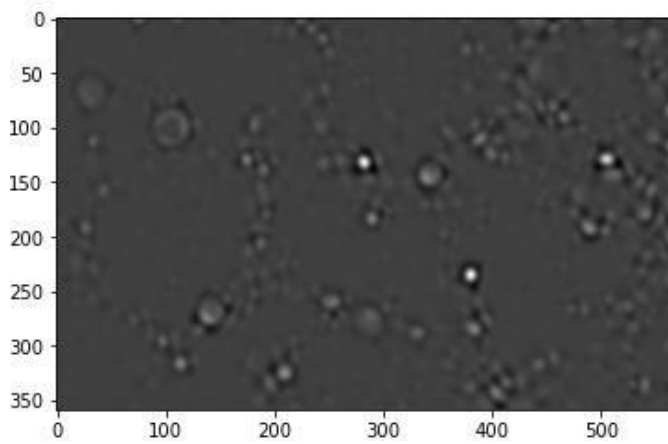


Approach's quality comparison for $s=4$:

The number of Blops of Rmars with Harris-Hessian for $s=4$ is: 85

The number of Blops of Rmars with Box Filters for $s=4$ is: 53

Visualisation of R with Harris-Hessian and Box Filters for $s=6$,respectively:

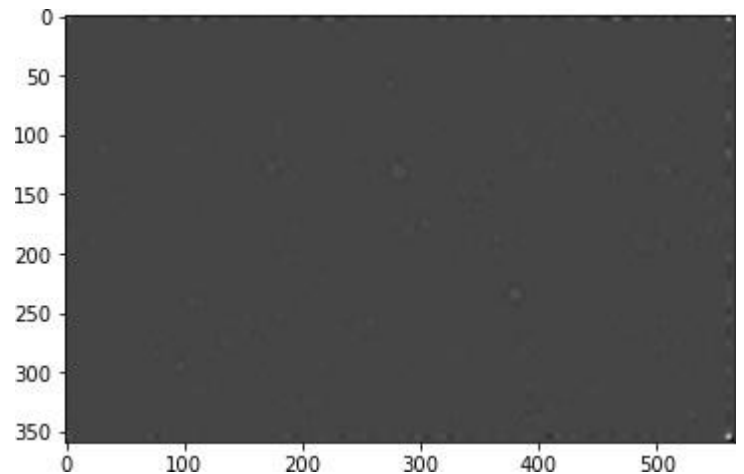
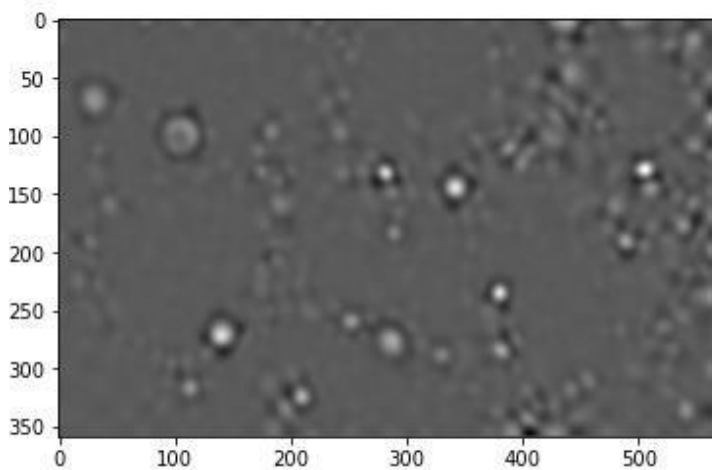


Approach's quality comparison for $s=6$:

The number of Blops of Rmars with Harris-Hessian for $s=6$ is: 34

The number of Blops of Rmars with Box Filters for $s=6$ is: 24

Visualisation of R with Harris-Hessian and Box Filters for $s=8$,respectively:



Approach's quality comparison for $s=8$:

The number of Blops of Rmars with Harris-Hessian for $s=8$ is: 25

The number of Blops of Rmars with Box Filters for $s=8$ is: 16

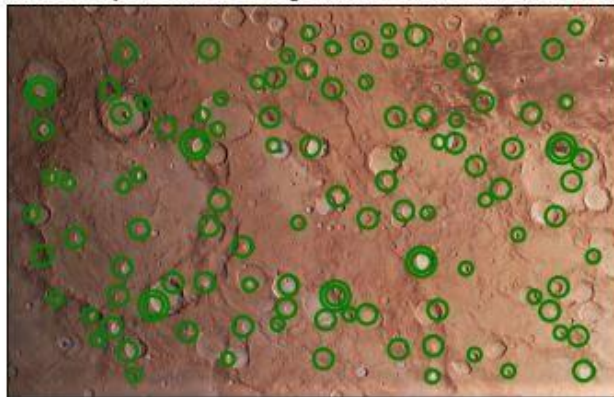
Από τα παραπάνω συνάγεται το συμπέρασμα πως η ανίχνευση blobs με τη χρήση Box Filters είναι λιγότερο ακριβής από αυτή με τον απλό Hessian. Μάλιστα, λογικό είναι πως όσο η κλίμακα μεγαλώνει τόσο θα μικραίνει και η διαφορά των blobs. Ωστόσο, είναι αρκετά ταχύτερη εφόσον η πολυπλοκότητά της εξαρτάται μόνο απ' τη διάσταση της εικόνας σε αντίθεση με τον απλό Hessian όπου εξαρτάται και απ' τη διάσταση του πυρήνα που τη συνελίσσουμε. Επιγραμματικά: χάνουμε ακρίβεια , κερδίζουμε ταχύτητα.

(2.5.4)

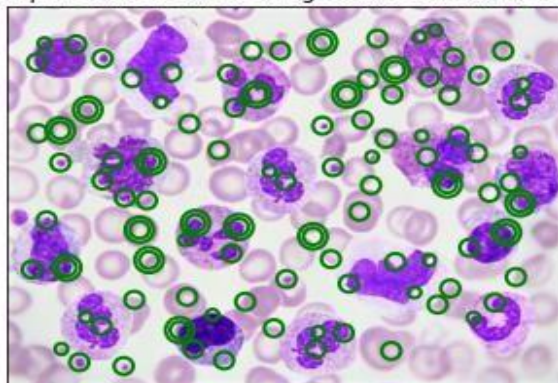
Σε αυτό το ερώτημα θα βρεθούν τα blobs ενδιαφέροντος σε διάφορες κλίμακες σ και θα κρατηθούν εκείνα τα σημεία τα οποία μεγιστοποιούν τη LoG μετρική γειτονικών κλιμάκων, όπως ακριβώς έγινε και στα ερωτήματα (2.2) και (2.4) .

Με εφαρμογή, επομένως, μιας παρόμοιας συνάρτησης με τα ερωτήματα αυτά ανιχνευθήκαν τα ακόλουθα blobs για πολλαπλές κλίμακες.

Visualization of Blops for mars image with BOX FILTERS for multiple scales



Visualization of Blops for blood smear image with BOX FILTERS for multiple scales



Μέρος 3: Εφαρμογές σε Ταίριασμα και Κατηγοριοποίηση Εικόνων με Χρήση Τοπικών Περιγραφητών στα Σημεία Ενδιαφέροντος

Με τον όρο τοπικός περιγραφητής ορίζεται ένας τρόπος κωδικοποίησης της εικόνας σε μια γειτονιά γύρω από ένα σημείο ενδιαφέροντος, για την εξαγωγή χρήσιμων χαρακτηριστικών της εικόνας. Οι περιγραφητές, που θα χρησιμοποιηθούν σε αυτό το μέρος, είναι ο SURF και ο HOG, όπου και οι δύο περιγραφητές βασίζονται στη κωδικοποίηση της πληροφορίας υποσημάτων της γειτονιάς αυτής με χρήση της πρώτης κατευθυντικής παραγώγου.

Συγκεκριμένα, μια συνοπτική ανάλυση των δύο αλγορίθμων φαίνεται ακολούθως.

SURF (Speed Up Robust Features):

Οι τοπικοί περιγραφητές SURF προκύπτουν από τα ακόλουθα βήματα:

1. Εύρεση των κατευθυντικών παραγώγων με χρήση των Haar Wavelets
2. Εύρεση της κατεύθυνσης για μια περιοχή γύρω από το σημείο για εξαγωγή περιστροφικά ανεξάρτητων περιγραφητών
3. Υπολογισμός ενός 64-διάστατου διανύσματος χαρακτηριστικών για ένα τετραγωνικό παράθυρο με μέγεθος που εξαρτάται από την κλίμακα και την κατεύθυνση, που βρέθηκε προηγουμένως.

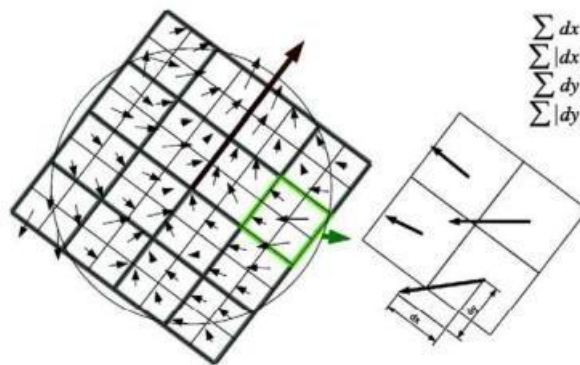


Figure 39. Οπτικοποίηση για την εξαγωγή περιγραφητών με τη μέθοδο SURF

HOG (Histogram of Oriented Gradients):

Τα ιστογράμματα προσανατολισμένης κλίσης αποτελούν ένα σύνολο χαρακτηριστικών για την περιγραφή της δομής του σχήματος σε μια εικόνα και

παρέχουν μια πυκνή επικαλυπτόμενη περιγραφή των περιοχών μιας εικόνας, υπολογισμένες σε ένα πυκνό πλέγμα ομοιόμορφα κατανεμημένων κελιών. Ο αλγόριθμος αυτός συνοψίζεται με τα ακόλουθα βήματα του διαγράμματος ροής.

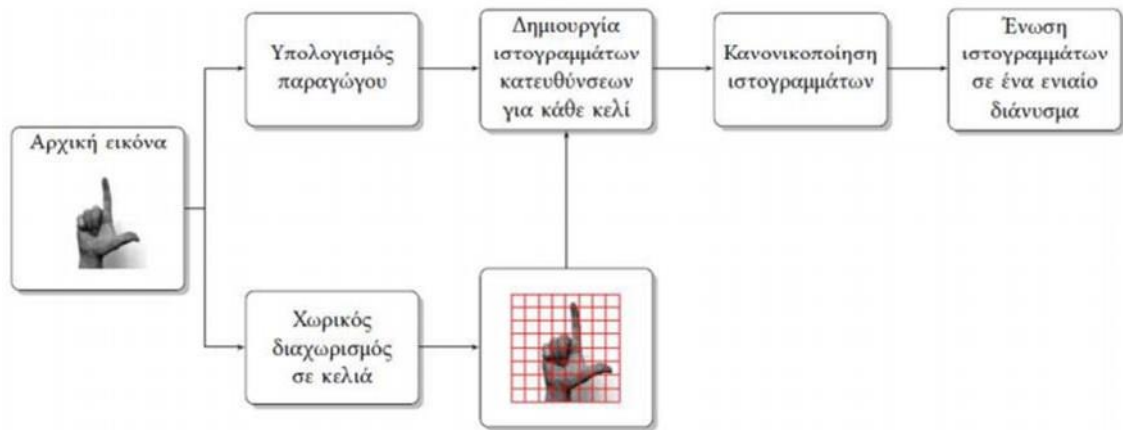


Figure 40. Διάγραμμα ροής για την εξαγωγή χαρακτηριστικών HOG

Άσκηση 3.1. Ταίριασμα Εικόνων υπό Περιστροφή και Αλλαγή Κλίμακας

Στο μέρος αυτό, εξετάζεται η ικανότητα εύρεσης της περιστροφής και της κλίμακας με τη χρήση των ανιχνευτών σημείων ενδιαφέροντος, που υλοποιήθηκαν από τους παραπάνω τοπικούς περιγραφητές. Συγκεκριμένα, η διαδικασία που θα υλοποιηθεί ονομάζεται *matching* και σύμφωνα με αυτή αντιστοιχίζονται οι τοπικοί περιγραφητές μιας εικόνας αναφοράς με τους τοπικούς περιγραφητές των υπόλοιπων παραμορφωμένων (γυρισμένες κατά μια γωνία ή αλλαγμένες ως προς το μέγεθός τους) εικόνων.

Στόχος, λοιπόν, είναι η εκτίμηση αυτής της αστοχίας αντιστοίχισης των τοπικών περιγραφητών ως προς τη περιστροφή (θ) και κλίμακα (s) των εικόνων. Για την επίτευξη αυτού του στόχου χρησιμοποιήθηκε η δοθείσα συνάρτηση *matching_evaluation*, προσαρμοσμένη στους ανιχνευτές, που υλοποιήθηκαν, δίνοντας τα ακόλουθα αποτελέσματα.

Για τον Harris:

```
FOR THE SURF WITH HARRIS
For The SURF with Harris Avg. Scale Error for Image 1: 0.003
For The SURF with Harris Avg. Theta Error for Image 1: 1.968
For The SURF with Harris Avg. Scale Error for Image 2: 0.002
For The SURF with Harris Avg. Theta Error for Image 2: 0.318
For The SURF with Harris Avg. Scale Error for Image 3: 0.097
For The SURF with Harris Avg. Theta Error for Image 3: 12.909
FOR THE HOG WITH HARRIS
For The HOG with Harris Avg. Scale Error for Image 1: 0.186
For The HOG with Harris Avg. Theta Error for Image 1: 22.619
For The HOG with Harris Avg. Scale Error for Image 2: 0.351
For The HOG with Harris Avg. Theta Error for Image 2: 19.199
For The HOG with Harris Avg. Scale Error for Image 3: 0.285
For The HOG with Harris Avg. Theta Error for Image 3: 23.699
```

Για τον Harris-Laplace:

```
FOR THE SURF WITH HARRIS POLYKLIMAKWTA
For The SURF with Harris Laplace Avg. Scale Error for Image 1: 0.001
For The SURF with Harris Laplace Avg. Theta Error for Image 1: 0.075
For The SURF with Harris Laplace Avg. Scale Error for Image 2: 0.001
For The SURF with Harris Laplace Avg. Theta Error for Image 2: 0.106
For The SURF with Harris Laplace Avg. Scale Error for Image 3: 0.002
For The SURF with Harris Laplace Avg. Theta Error for Image 3: 0.078
FOR THE HOG WITH HARRIS POLYKLIMAKWTA
For The HOG with Harris Laplace Avg. Scale Error for Image 1: 0.259
For The HOG with Harris Laplace Avg. Theta Error for Image 1: 19.465
For The HOG with Harris Laplace Avg. Scale Error for Image 2: 0.788
For The HOG with Harris Laplace Avg. Theta Error for Image 2: 15.713
For The HOG with Harris Laplace Avg. Scale Error for Image 3: 0.325
For The HOG with Harris Laplace Avg. Theta Error for Image 3: 19.923
```

Για τον Hessian:

```
FOR THE SURF WITH HESIAN
For The SURF with Hessian Avg. Scale Error for Image 1: 0.027
For The SURF with Hessian Avg. Theta Error for Image 1: 7.759
For The SURF with Hessian Avg. Scale Error for Image 2: 0.010
For The SURF with Hessian Avg. Theta Error for Image 2: 0.229
For The SURF with Hessian Avg. Scale Error for Image 3: 0.001
For The SURF with Hessian Avg. Theta Error for Image 3: 0.054
FOR THE HOG WITH HESIAN
For The HOG with Hessian Avg. Scale Error for Image 1: 0.186
For The HOG with Hessian Avg. Theta Error for Image 1: 7.231
For The HOG with Hessian Avg. Scale Error for Image 2: 0.100
For The HOG with Hessian Avg. Theta Error for Image 2: 13.674
For The HOG with Hessian Avg. Scale Error for Image 3: 0.154
For The HOG with Hessian Avg. Theta Error for Image 3: 27.219
```

Για τον Hessian-Laplace:

```
FOR THE SURF WITH HESIAN POLYKLIMAKWTA
For The SURF with Hessian Laplace Avg. Scale Error for Image 1: 0.002
For The SURF with Hessian Laplace Avg. Theta Error for Image 1: 0.159
For The SURF with Hessian Laplace Avg. Scale Error for Image 2: 0.002
For The SURF with Hessian Laplace Avg. Theta Error for Image 2: 0.192
For The SURF with Hessian Laplace Avg. Scale Error for Image 3: 0.001
For The SURF with Hessian Laplace Avg. Theta Error for Image 3: 0.107
FOR THE HOG WITH HESIAN POLYKLIMAKWTA
For The HOG with Hessian Laplace Avg. Scale Error for Image 1: 0.110
For The HOG with Hessian Laplace Avg. Theta Error for Image 1: 10.406
For The HOG with Hessian Laplace Avg. Scale Error for Image 2: 0.131
For The HOG with Hessian Laplace Avg. Theta Error for Image 2: 6.393
For The HOG with Hessian Laplace Avg. Scale Error for Image 3: 0.190
For The HOG with Hessian Laplace Avg. Theta Error for Image 3: 25.758
```

Για τον Box-Filters-Laplace:

```
FOR THE SURF WITH BOXES
For The SURF with Boxes polyklimakwta Avg. Scale Error for Image 1: 0.003
For The SURF with Boxes polyklimakwta Avg. Theta Error for Image 1: 0.227
For The SURF with Boxes polyklimakwta Avg. Scale Error for Image 2: 0.003
For The SURF with Boxes polyklimakwta Avg. Theta Error for Image 2: 0.219
For The SURF with Boxes polyklimakwta Avg. Scale Error for Image 3: 0.003
For The SURF with Boxes polyklimakwta Avg. Theta Error for Image 3: 0.130
FOR THE HOG WITH BOXES
For The HOG with Boxes polyklimakwta Avg. Scale Error for Image 1: 0.336
For The HOG with Boxes polyklimakwta Avg. Theta Error for Image 1: 13.824
For The HOG with Boxes polyklimakwta Avg. Scale Error for Image 2: 0.183
For The HOG with Boxes polyklimakwta Avg. Theta Error for Image 2: 26.801
For The HOG with Boxes polyklimakwta Avg. Scale Error for Image 3: 0.235
For The HOG with Boxes polyklimakwta Avg. Theta Error for Image 3: 19.497
```

Τα παραπάνω αποτελέσματα έχουν μπει στους 2 ακόλουθους πίνακες για να είναι πιο εύκολη η σύγκρισή τους.

Σύγκριση ως προς τους αλγορίθμους:

Χρησιμοποιήθηκε ενδεικτικά η εικόνα 1 καθώς παρατηρείται παρόμοια συμπεριφορά ως προς τους αλγορίθμους και για τις υπόλοιπες.

Detector	SURF (scale-theta)		HOG (scale-theta)	
Harris	0.003	1.96	0.186	22.619
Harris-Laplace	0.001	0.075	0.259	19.465
Hessian	0.027	7.759	0.186	7.231
Hessian-Laplace	0.002	0.159	0.110	10.406
Box Filters-L	0.003	0.227	0.336	13.824

Από τα παραπάνω δεδομένα φαίνεται ότι ο αλγόριθμος SURF βγάζει μικρότερο σφάλμα τόσο για scale-error όσο και για theta-error για όλους τους ανιχνευτές σημείων ενδιαφέροντος. Συγκεκριμένα, και τα δύο errors έχουν ένα σφάλμα της κλίμακας 100x παραπάνω στον αλγόριθμο HOG, σε όλους τους ανιχνευτές εκτός από τον Hessian, όπου το σφάλμα γωνίας είναι σχεδόν ίδιο.

Σύγκριση ως προς τις εικόνες:

Χρησιμοποιήθηκε ενδεικτικά ο αλγόριθμος SURF καθώς παρατηρείται παρόμοια συμπεριφορά ως προς τις εικόνες και για τον HOG.

Detector	Image 1 (scale-theta)		Image 2 (scale-theta)		Image 3 (scale-theta)	
Harris	0.003	1.96	0.002	0.318	0.097	12.909
Harris-Laplace	0.001	0.075	0.001	0.106	0.002	0.078
Hessian	0.027	7.759	0.01	0.229	0.001	0.054
Hessian-Laplace	0.002	0.159	0.002	0.192	0.001	0.107
Box Filters-L	0.003	0.227	0.003	0.219	0.003	0.13

Από τα παραπάνω δεδομένα φαίνεται ότι το σφάλμα στην τυπική απόκλιση ανά τις εικόνες είναι περίπου ίδιο σε αντίθεση με το σφάλμα στη γωνία που φαίνεται να διαφοροποιείται σημαντικά για τις τρεις εικόνες.

Άσκηση 3.2. Κατηγοριοποίηση Εικόνων

Στο μέρος αυτό αξιολογείται η επίδοση των ανιχνευτών και περιγραφητών, που υλοποιήθηκαν, με χρήση της βάσης δεδομένων-εικόνων Pascal VOC2005. Αυτή η βάση αποτελείται από εικόνες που ανήκουν σε τρεις κλάσεις: αυτοκίνητο, άνθρωπος και ποδήλατο.

Στόχος αυτού του μέρους είναι η υλοποίηση ενός αλγορίθμου αναγνώρισης του είδους της εικόνας και κατηγοριοποίησής της στην σωστή κλάση. Για την υλοποίηση αυτού του αλγορίθμου χρησιμοποιήθηκαν οι δοθείσες συναρτήσεις: FeatureExtraction, createTrainTest και svm , οι οποίες προσομοιάζουν ένα νευρωνικό δίκτυο το οποίο εκτιμά την πιθανότητα εύστοχης εκτίμησης της κατηγοριοποίησης των εικόνων σε κλάσεις. Τα αποτελέσματα εκτίμησης αυτού του αλγορίθμου φαίνονται παρακάτω.

Για το SURF:

$\sigma = 2$		$\sigma = 4$		$\sigma = 8$	
Type of Detector	Mean Accuracy(%)	Type of Detector	Mean Accuracy(%)	Type of Detector	Mean Accuracy(%)
Harris-Laplace	62.621	Harris-Laplace	59.724	Harris-Laplace	61.103
Hessian-Laplace	58.759	Hessian-Laplace	54.483	Hessian-Laplace	47.862
Box Filters-Laplace	59.310	Box Filters-Laplace	58.897	Box Filters-Laplace	56.138

Για το HOG:

$\sigma = 2$		$\sigma = 4$		$\sigma = 8$	
Type of Detector	Mean Accuracy(%)	Type of Detector	Mean Accuracy(%)	Type of Detector	Mean Accuracy(%)
Harris-Laplace	67.310	Harris-Laplace	65.793	Harris-Laplace	65.655
Hessian-Laplace	67.034	Hessian-Laplace	61.517	Hessian-Laplace	56.828
Box Filters-Laplace	66.207	Box Filters-Laplace	64.414	Box Filters-Laplace	63.034

Από τα παραπάνω αποτελέσματα φαίνεται ότι καλύτερος αλγόριθμος ανίχνευσης είναι ο HOG, αυξάνοντας κατά πολύ το ποσοστό ανίχνευσης για όλους τους ανιχνευτές, σε σχέση με τον αλγόριθμο SURF. Επιπλέον, παρατηρήθηκε ότι όσο αυξανόταν η τιμή της τυπικής απόκλισης τόσο μειωνόταν το ποσοστό επιτυχούς εκτίμησης του νευρωνικού.

Συνεπώς, από τις διάφορες δοκιμές επάνω σε παραμέτρους και αλγορίθμους για την αποδοτικότερη λειτουργία του νευρωνικού, προκύπτει το συμπέρασμα ότι αποδοτικότερος ανιχνευτής είναι ο Harris-Laplace. Ο Harris-Laplace είναι ο ανιχνευτής που χρησιμοποιήθηκε για την ανίχνευση γωνιών σε διάφορες κλίμακες. Επομένως, φαίνεται ότι σε αυτή την κατηγοριοποίηση των εικόνων στις κλάσεις αυτοκίνητο, ποδήλατο και άνθρωπος, παίζουν σημαντικό ρόλο οι γωνίες για το διαχωρισμό. Αυτό στέκει και από λογικής απόψεως αφού το ποδήλατο και το αυτοκίνητο ως αντικείμενο έχουν περισσότερες γωνίες από ότι ο άνθρωπος, που φαίνεται να είναι πιο συμμετρικός και να εμφανίζει κυρτότητες. Επίσης, ένα ποδήλατο σε σύγκριση με ένα αυτοκίνητο διαφέρουν επίσης αρκετά ως προς τις γωνίες μας και το ένα έχει τις λεπτές ακτίνες της ρόδας να ξεχωρίζουν, καθώς και άλλα σημεία που δεν εμφανίζουν μεγάλη κυρτότητα, όπως το τιμόνι και τα πετάλια. Η έλλειψη της κυρτότητας αυτής στο ποδήλατο και το αυτοκίνητο είναι που ρίχνει την απόδοση των αλγορίθμων Hessian-Laplace και των Box Filters-Laplace, που βασίζονται στην ανίχνευση blobs.

Βιβλιογραφία-Πηγές-Ιστοσελίδες:

(1) Maragos_CV_Book2018_chapter10_FeatureDetection