



## Όραση Υπολογιστών

### 1η Εργαστηριακή Άσκηση: Εντοπισμός Σημείων Ενδιαφέροντος και Εξαγωγή Χαρακτηριστικών σε Εικόνες

8ο Εξάμηνο - Εαρινό εξάμηνο 2018-19 - Ροή Σ

Αντωνιάδης Παναγιώτης (03115009 - e115009@central.ntua.gr)  
Μπαζώτης Νικόλαος (03115739 - e115739@central.ntua.gr)

---

## Περιεχόμενα

<b>1 Ανίχνευση Ακμών σε Γκρίζες Εικόνες</b>	<b>2</b>
1.1 Δημιουργία Εικόνων Εισόδου . . . . .	2
1.2 Υλοποίηση Αλγορίθμων Ανίχνευσης Ακμών . . . . .	3
1.3 Αξιολόγηση των Αποτελεσμάτων Ανίχνευσης Ακμών . . . . .	5
1.4 Εφαρμογή των Αλγορίθμων Ανίχνευσης Ακμών σε Πραγματικές εικόνες . . . . .	11
<b>2 Ανίχνευση Σημείων Ενδιαφέροντος (Interest Point Detection)</b>	<b>15</b>
2.1 Ανίχνευση Γωνιών . . . . .	16
2.2 Πολυχλιμακωτή Ανίχνευση Γωνιών . . . . .	21
2.3 Ανίχνευση Blobs . . . . .	24
2.4 Πολυχλιμακωτή Ανίχνευση Blobs . . . . .	26
2.5 Επιτάχυνση με την χρήση Box Filters και Ολοκληρωτικών Εικόνων (Integral Images) . . . . .	26
<b>3 Εφαρμογές σε Ταίριασμα και Κατηγοριοποίηση Εικόνων με Χρήση Τοπικών Περιγραφητών στα Σημεία Ενδιαφέροντος</b>	<b>30</b>
3.1 Ταίριασμα Εικόνων υπό Περιστροφή και Αλλαγή Κλίμακας . . . . .	30
3.2 Κατηγοριοποίηση Εικόνων . . . . .	31
<b>Βιβλιογραφία</b>	<b>34</b>

# 1 Ανίχνευση Ακμών σε Γκρίζες Εικόνες

Στο πρώτο μέρος, αφού επεξεργαστούμε τις εικόνες προσθέτοντάς τους θόρυβο, προσπαθούμε να εκτιμήσουμε τις ακμές τους με αλγόριθμους οι οποίοι χρησιμοποιούν γραμμικές και μορφολογικές μεθόδους. Τέλος, έχουμε την αξιολόγηση των αλγορίθμων αυτών ως προς την επιτυχή εκτίμηση των ακμών στις εικόνες. Για τις διαδικασίες αυτές χρησιμοποιήθηκαν οι παρακάτω δύο εικόνες:



Εικόνα 1: Αρχικές εικόνες χωρίς θόρυβο

## 1.1 Δημιουργία Εικόνων Εισόδου

Σε αυτό το κομμάτι, γίνεται το διάβασμα των εικόνων με την χρήση της συνάρτησης `imread` ώστε να υποστούν την κατάλληλη επεξεργασία και να μελετηθούν στην συνέχεια. Πιο συγκεκριμένα, γίνεται χρήση της πρώτης εικόνας που παρατίθεται παραπάνω. Στο ερώτημα 1.1.2 εισάγουμε στην εικόνα Gaussian λευκό θόρυβο μηδενικής μέσης τιμής και τυπικής απόκλισης  $\sigma_n$ , η οποία προκύπτει από την τιμή του PSNR.

$$I(x, y) = I_0(x, y) + n(x, y) \quad (1)$$

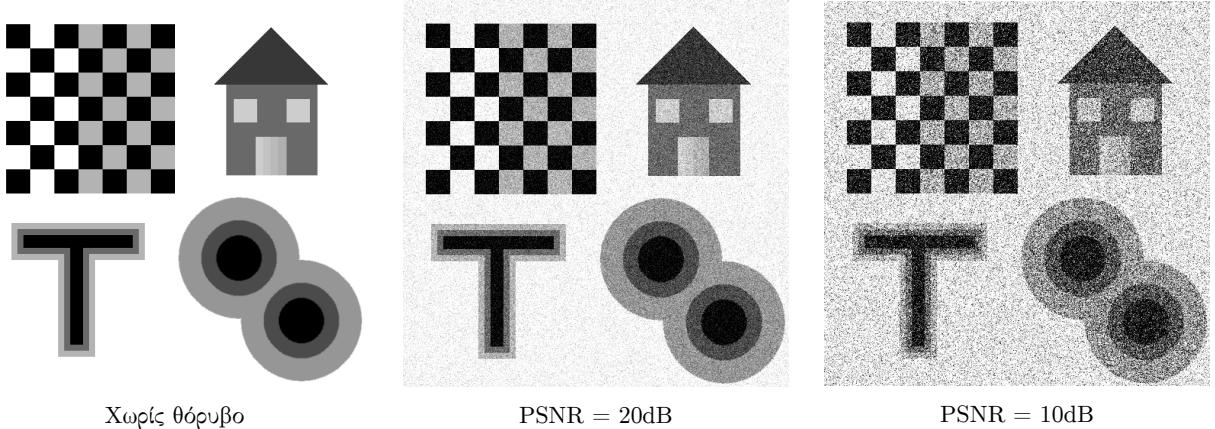
όπου  $n(x, y)$  ο Gaussian θόρυβος.

$$PSNR = 20 \log_{10} \frac{I_{max} - I_{min}}{\sigma_n} \implies \sigma_n = \frac{I_{max} - I_{min}}{10^{\frac{PSNR}{20}}} \quad (2)$$

Έτσι, προκύπτουν οι παρακάτω τιμές:

- Για  $PSNR=20$ dB προκύπτει Gaussian λευκός θόρυβος τυπικής απόκλισης  $\sigma_n = 0.1$ .
- Για  $PSNR=10$ dB προκύπτει Gaussian λευκός θόρυβος τυπικής απόκλισης  $\sigma_n = 0.3162$ .

Τέλος, παραθέτουμε το αποτέλεσμα του θορύβου πάνω στην εικόνα μας για τις διάφορες τιμές του PSNR.



Εικόνα 2: Η εικόνα με διαφορετικές τιμές PSNR

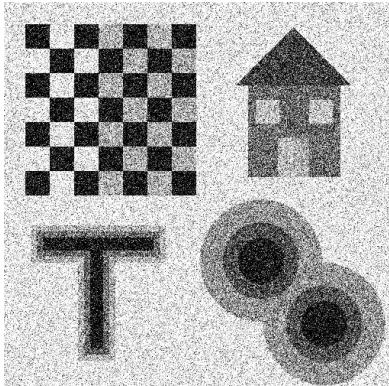
Για την εφαρμογή του θορύβου πάνω στην εικόνα έγινε χρήση της συνάρτησης imnoise, της οποίας παράμετροι είναι η εικόνα, ο τύπος θορύβου (Gaussian), η μέση τιμή της κατανομής του θορύβου και η τυπική απόκλισή της ( $\sigma_n$ ) όπως την υπολογίσαμε για τις διάφορες τιμές του PSNR. Όλη η διαδικασία της θορυβοποίησης υλοποιήθηκε στην συνάρτηση AddNoise της οποίας παράμετροι είναι η εικόνα και το PSNR με την σειρά που αναφέρθηκαν.

## 1.2 Υλοποίηση Αλγορίθμων Ανίχνευσης Ακμών

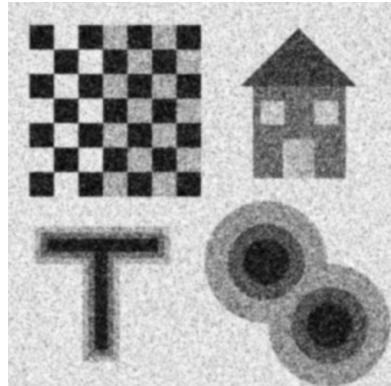
Για τις ανάγκες της υλοποίησης του αλγορίθμου κατασκευάστηκε η συνάρτηση EdgeDetect, η οποία αρχικά δημιουργεί 2 φίλτρα, το 1o με απόκριση δισδιάστατη Gaussian  $G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$  και το 2o με απόκριση την Laplacian of Gaussian  $L_\sigma(x, y) = \nabla^2 G_\sigma = \frac{x^2+y^2-2\sigma^2}{2\pi\sigma^6} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$ .

Για τα 2 παραπάνω φίλτρα, η παράμετρος σ είναι ίδια και δίνεται στις παραμέτρους της συνάρτησης όταν αυτή καλείται. Ο πυρήνας των φίλτρων αυτών είναι διαστάσεων  $n \times n$  και δίνεται από την εξίσωση  $n(\sigma) = 2ceil(3\sigma) + 1$  όπου ceil είναι η συνάρτηση του MATLAB που μετατρέπει το όρισμα της στον αμέσως μεγαλύτερο ακέραιο, ώστε το αποτέλεσμα των διαστάσεων να είναι ακέραιος αριθμός. Επίσης, χρησιμοποιήθηκε η συνάρτηση fspecial η οποία δημιούργησε τις αποχρίσεις των φίλτρων μας, παίρνοντας ως ορίσματα τον τύπο, τον πυρήνα και την απόκλιση σ.

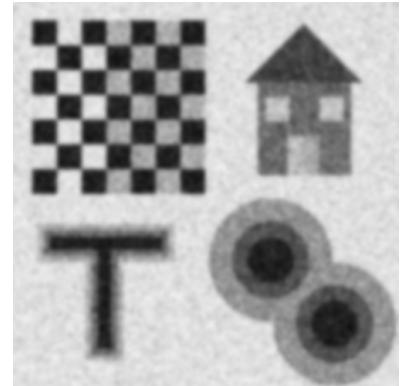
Στο επόμενο σκέλος, έχοντας τα φίλτρα θα τα χρησιμοποιήσουμε για να υλοποιήσουμε 2 διαφορετικές μεθόδους για την ανίχνευση ακμών, μια γραμμική και μια με μορφολογικά φίλτρα. Για αρχή θα κάνουμε συνέλιξη της εικόνας με την απόκριση του φίλτρου της Gaussian για να πάρουμε την εξομαλυμένη εικόνα, αλλά αυτό μπορούμε διαισθητικά να το αντιληφθούμε από τις παρακάτω εικόνες για τις οποίες βλέπουμε ότι η παράμετρος σ παίζει σημαντικό ρόλο, καθώς αύξηση του σ συνεπάγεται ακόμα πιο θολή (blur) εξομαλυμένη εικόνα.



$$\sigma = 0.1$$



$$\sigma = 1.5$$



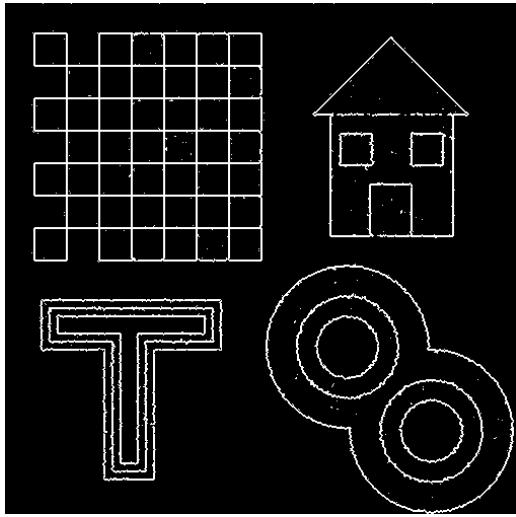
$$\sigma = 3$$

- Για την γραμμική μέθοδο (1η) δεν θα χρειαστούμε την εξομαλυμένη εικόνα γιατί θα κάνουμε συνέλιξη της εικόνας  $I$  με την απόκριση του φίλτρου της Laplacian of Gaussian που φτιάξαμε πριν, η οποία αντικατοπτρίζει αυτήν την σχέση  $L_1 = \nabla^2(G_\sigma * I) = \nabla^2(G_\sigma) * I = h_{LoG} * I$ .
- Για την μέθοδο με τα μορφολογικά φίλτρα (2η) θα κάνουμε dilation και erosion της εξομαλυμένης εικόνας με το  $B$  το οποίο γεωμετρικά παριστάνεται έτσι  $\begin{array}{c} \blacksquare \\ \blacksquare \quad \blacksquare \\ \blacksquare \end{array}$  αλλά για τους σκοπούς μας είναι  $\begin{array}{ccc} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{array}$ .

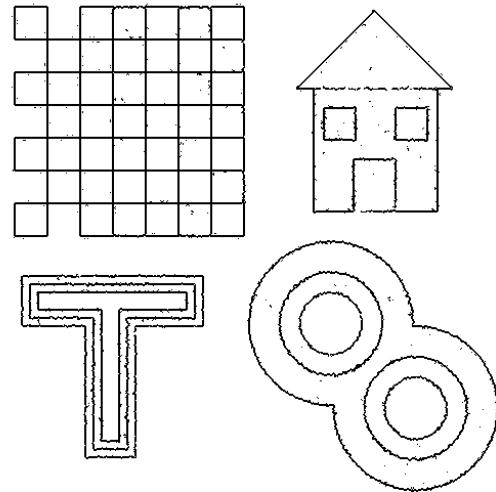
$$H \text{ συνολική σχέση} \text{ θα είναι } L_2 = (I_\sigma \oplus B) + (I_\sigma \ominus B) - 2I_\sigma.$$

Η έξοδος  $L$  ωστόσο έχει και αρνητικές τιμές τις οποίες δεν θέλουμε. Για τον λόγο αυτό, παίρνουμε όλες τις μη αρνητικές σε έναν πίνακα  $X$  και βρίσκουμε το περίγραμμα της εικόνας που προκύπτει από τις τιμές που επιλέξαμε σε έναν δυαδικό πίνακα  $Y$ . Με αυτόν τον τρόπο, παίρνοντας ουσιαστικά την παράγωγο του  $X$ , προσεγγίζουμε τα zero-crossings έχοντας  $Y[i, j] = 1$  εκεί που υπάρχουν οι μηδενισμοί. Αυτό έχει ως αποτέλεσμα να ανιχνευτούν αρκετές ακμές οι οποίες δεν αντιστοιχούν σε πραγματικές ακμές της εικόνας, αλλά προκύπτουν εξαιτίας του θορύβου ο οποίος αλλοιώνει την φυσική υφή της εικόνας. Στο σημείο αυτό, πρέπει να επιλέξουμε ποιες από τις ακμές αυτές που βρήκαμε θέλουμε να κρατήσουμε και, έτσι, ορίζουμε μια παράμετρο κατωφλιού  $\theta_{edge}$ , της οποίας η φυσική σημασία είναι να ορίζει ένα κατώφλι πάνω από το οποίο αναγνωρίζουμε το  $Y=1$  ως ακμή και το κρατάμε ενώ αν δεν περνάει το κατώφλι αυτό απορρίπτεται ως λανθάνουσα ακμή. Με αυτό τον τρόπο, θέλουμε να απορρίψουμε τα zero-crossings τα οποία βρίσκουμε σε σχετικά ομαλές περιοχές, κάτι που είναι παρενέργεια του θορύβου που εισήχθη στην αρχή.

Το υλοποιούμε αυτό με την εξίσωση  $D = (Y = 1 \wedge |\nabla I_\sigma(i, j)| > \theta_{edge} \max_{x,y} |\nabla I_\sigma|)$ . Για τον τελεστή  $\nabla$  χρησιμοποιούμε την συνάρτηση gradient του MATLAB. Το αποτέλεσμα όλης αυτής της διαδικασίας μας δίνει:



Εξόδος D



Εξόδος  $\neg D$

Εικόνα 4: Εξόδος D για  $PSNR = 20dB$   $\sigma=1.5$   $\theta_{edge} = 0.2$

### 1.3 Αξιολόγηση των Αποτελεσμάτων Ανίχνευσης Ακμών

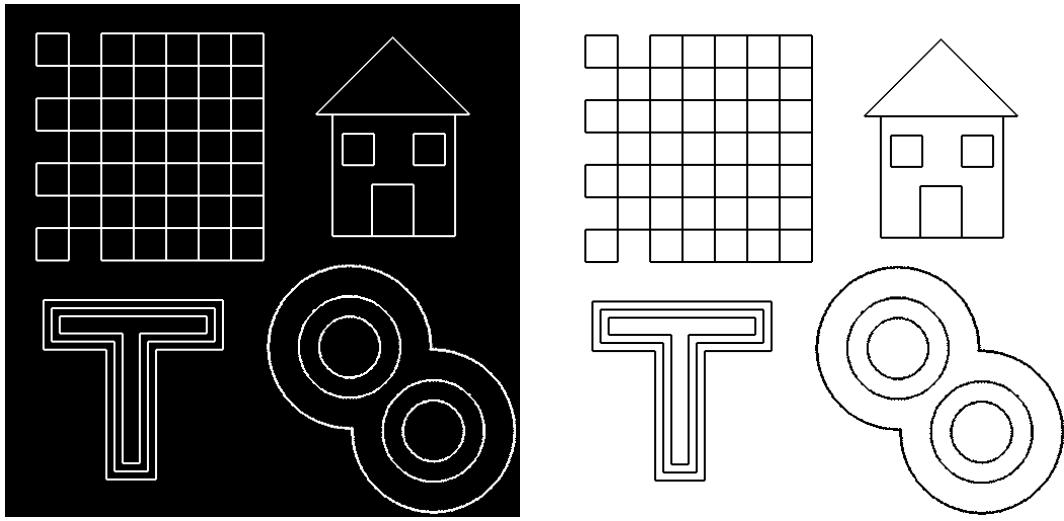
Έχει μεγάλη σημασία να δούμε κατά πόσο ο αλγόριθμος ανταποκρίθηκε στις απαιτήσεις του προβλήματος που του θέσαμε για αυτό κιόλας σε αυτό το μέρος θα προσπαθήσουμε να αξιολογήσουμε την επίδοση του αλγόριθμου και να εξετάσουμε την επίδραση των παραμέτρων. Αρχικά θα εκτιμήσουμε τις ακμές στην εικόνα την αρχική η οποία δεν έχει θόρυβο και στην συνέχεια θα συγχρίνουμε το αποτέλεσμα που θα βρούμε με αυτά του αλγόριθμου.

Για την εκτίμηση των ακμών αρχεί να βρούμε το περίγραμμα με παρόμοιο τρόπο όπως αυτός των μορφολογικών φίλτρων του παραπάνω αλγόριθμου συγκεκριμένα θα έχουμε

$$M = (I_\sigma \oplus B) + (I_\sigma \ominus B) \quad (3)$$

$$T = M > \theta_{realedge} \quad (4)$$

Με την εφαρμογή των παραπάνω πάνω στην αρχική εικόνα I, με  $B = \begin{matrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{matrix}$  όπως πριν και με παράμετρο κατωφλιού πάλι παίρνουμε την εικόνα με τις πραγματικές ακμές που φαίνεται παρακάτω:



Εικόνα 5: Αληθινές ακμές για  $\theta_{realedge} = 0.1$

Τώρα που βρήκαμε τις αληθινές ακμές μπορούμε να κατασκευάσουμε και μια μετρική η οποία να μας δείχνει το ποσοστό επιτυχίας του αλγορίθμου. Πιο συγκεκριμένα, θα εκτιμήσουμε το ποσοστό στο οποίο ο αλγόριθμος βρήκε αληθινές ακμές και το ποσοστό των ακμών που βρέθηκαν επιτυχώς στις συνολικές αληθινές ακμές (Precision and Recall). Άρα το μοντέλο της μετρικής που θα έχουμε είναι:

$$C = \frac{Pr(D|T) + Pr(T|D)}{2} \quad (5)$$

όπου D: έξοδος του αλγορίθμου και T: αληθινές ακμές. Το μοντέλο αυτό \*100% μας δίνει το ποσοστό επιτυχίας του αλγόριθμου μας πάνω στην εικόνα.

**Αξιολόγηση των παραμέτρων  $\sigma - \theta_{edge}$**

- Για PSNR = 20db

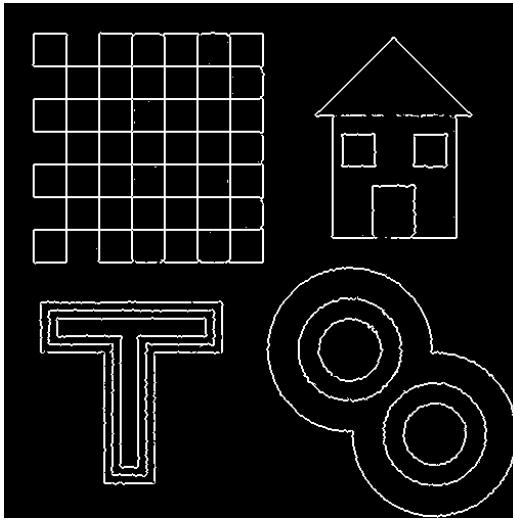
No	$\sigma$	$\theta_{edge}$	C
1	0.5	0.2	0.7901
2	0.5	0.3	0.7392
3	1.0	0.1	0.7579
4	1.0	0.2	0.8781
5	1.0	0.3	0.7722
6	1.5	0.1	0.8725
7	1.5	0.2	0.9253
8	1.5	0.25	0.8734
9	1.5	0.3	0.8150
Best	1.6	0.18	0.9324
11	2.0	0.1	0.8846
12	2.0	0.15	0.9231
13	2.0	0.2	0.9160
14	2.0	0.25	0.8666
15	2.0	0.3	0.8094

Table 1: Laplacian = 0 (Γραμμική μέθοδος).

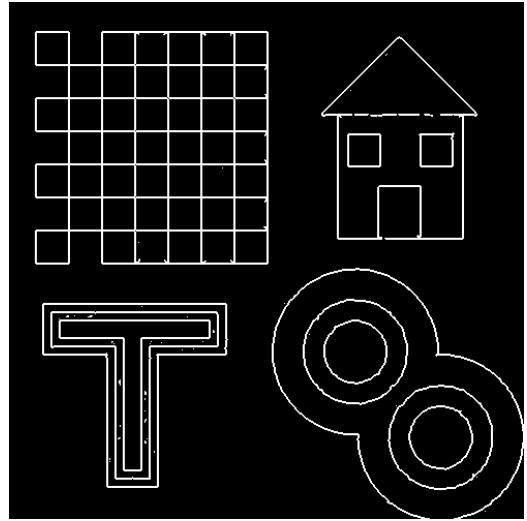
No	$\sigma$	$\theta_{edge}$	C
1	0.5	0.2	0.7783
2	0.5	0.3	0.7404
3	1.0	0.1	0.7407
4	1.0	0.2	0.8087
5	1.0	0.3	0.6470
6	1.5	0.1	0.8277
7	1.5	0.2	0.8384
8	1.5	0.25	0.7596
9	1.5	0.3	0.6877
10	1.6	0.18	0.8511
11	2.0	0.1	0.8300
Best	2.0	0.15	0.8552
13	2.0	0.2	0.8425
14	2.0	0.25	0.7624
15	2.0	0.3	0.6914

Table 2: Laplacian = 1 (Μορφολογικά φίλτρα).

**Παρατηρήσεις :** Βλέπουμε λοιπόν, σύμφωνα με τις παραπάνω μετρήσεις, ότι η γραμμική μέθοδος υπερέχει των μορφολογικών φίλτρων κατά ένα μικρό ποσοστό στην ακρίβεια των αποτελεσμάτων. Επίσης βλέπουμε ότι η παράμετρος κατωφλίου μεγιστοποιεί το αποτέλεσμα όταν βρίσκεται στην γειτονιά του  $0.2 \pm 0.05$ . Μεγαλύτερες τιμές της παραμέτρου αυτής αποκόπτουν αληθινές ακμές που βρήκαμε σωστά, ενώ μικρότερες τιμές περιλαμβάνουν ακμές τις οποίες λανθασμένα θεωρούμε ως ακμές. Η άλλη παράμετρος μας το  $\sigma$  (η απόκλιση του Gaussian φίλτρου) έχει να κάνει με τον βαθμό εξομάλυνσης της εικόνας. Σε μια εικόνα η οποία δεν έχει πολύ θόρυβο όσο ανεβαίνει η τιμή του σ χάνεται ουσιαστική πληροφορία και οι περιοχές γίνονται πιο ομαλές δυσκολεύοντας την εξεύρεση ακμών και τον διαχωρισμό τους από τις παρενέργειες του θορύβου. Αντιθέτως, τιμές οι οποίες είναι αρκετά χαμηλές δεν μας βοηθούν στο να ξεχωρίσουμε μετέπειτα τις ακμές, γιατί δεν υπάρχουν ομαλές περιοχές στις οποίες όταν βρούμε ακμή την απορρίπτουμε σύμφωνα πάντα με το κατώφλι που έχουμε θέσει. Οι τιμές, λοιπόν, που μας βοηθούν στο να λάβουμε καλύτερα αποτελέσματα για την παράμετρο  $\sigma$  είναι στην περιοχή του  $1.5 \pm 0.5$ , σύμφωνα με τις μετρήσεις.



(a) Γραμμική μέθοδος  $\sigma=1.6 \theta=0.18$



(b) Μορφολογικά φίλτρα  $\sigma=2 \theta=0.15$

- $\text{Για } \text{PSNR} = 10\text{dB}$

No	$\sigma$	$\theta_{edge}$	C
1	2	0.2	0.6894
<b>Best</b>	2	0.25	0.7151
3	2	0.3	0.6956
4	2	0.4	0.6426
5	2.5	0.2	0.7007
6	2.5	0.3	0.6803
7	2.5	0.4	0.6357
8	3	0.2	0.6832
9	3	0.3	0.6692
10	3	0.4	0.6162
11	3.5	0.2	0.6489
12	3.5	0.3	0.6353
13	3.5	0.4	0.6060
14	4.0	0.2	0.6353
15	4.0	0.3	0.6035

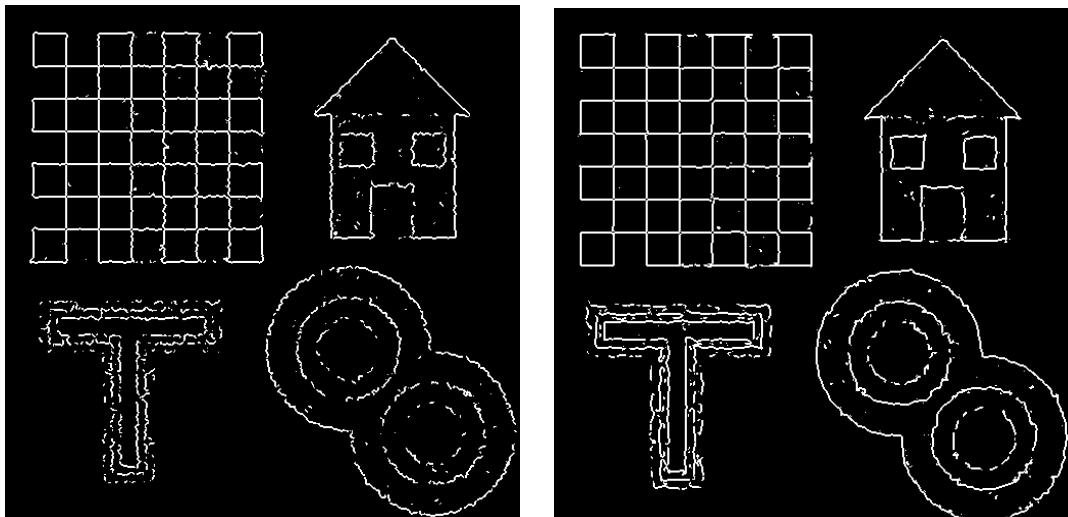
Table 3: Laplacian = 0 (Γραμμική μέθοδος).

No	$\sigma$	$\theta_{edge}$	C
1	2	0.2	0.6844
2	2	0.25	0.6683
3	2	0.3	0.6170
4	2	0.4	0.4978
<b>Best</b>	2.5	0.2	0.6964
6	2.5	0.3	0.6059
7	2.5	0.4	0.6010
8	3	0.2	0.6682
9	3	0.3	0.5843
10	3	0.4	0.4959
11	3.5	0.2	0.6464
12	3.5	0.25	0.6090
13	3.5	0.3	0.5838
14	4.0	0.2	0.6160
15	4.0	0.3	0.5575

Table 4: Laplacian = 1 (Μορφολογικά φίλτρα).

**Παρατηρήσεις:** Βλέπουμε, λοιπόν, σύμφωνα με τις παραπάνω μετρήσεις ξανά ότι η γραμμική μέθοδος υπερέχει των μορφολογικών φίλτρων κατά ένα μικρό ποσοστό στην ακρίβεια των αποτελεσμάτων. Παρατηρούμε, επίσης, ότι ενώ πρόκειται για την ίδια εικόνα αλλά για άλλα επίπεδα θορύβου ότι τα ποσοστά επιτυχίας του αλγορίθμου μας πέφτουν αρκετά κάτι που αποδίδεται εξ' ολοκλήρου στον ενισχυμένο θόρυβο που έχουμε στην συγκεκριμένη περίπτωση με  $\text{PSNR}=10\text{dB}$ . Εξαιτίας του θορύβου, έχουμε διαφορετικές τιμές παραμέτρων που βελτιστοποιούν το αποτέλεσμα και αυτό μας οδηγεί στο συμπέρασμα ότι οι παράμετροι που θα χρησιμοποιηθούν είναι πλήρως εξαρτώμενοι από τα επίπεδα θορύβου της εικόνας. Έτσι, χρειαστήκαμε μεγαλύτερη τυπική απόκλιση σ δηλαδή εξομάλυνση της εικόνας για να μπορέσουμε να λειάνουμε την εικόνα

και να διαχωρίσουμε καλύτερα τις ομαλές περιοχές της. Κατόπιν η συμπεριφορά της παραμέτρου κατωφλίου παρέμεινε σχετικά σταθερή βελτιστοποιώντας το αποτέλεσμα κοντά στην τιμή 0.2. Ωστόσο, χρειάστηκαν μεγαλύτερες τιμές σ για να μπορέσουμε να έχουμε καλύτερο αποτέλεσμα κοντά στην τιμή 2 αυτή την φορά. Παρατίθενται τα καλύτερα αποτελέσματα για τα 2 φίλτρα με τις παραπάνω τιμές:



(a) Γραμμική μέθοδος  $\sigma=2.5 \theta=0.2$

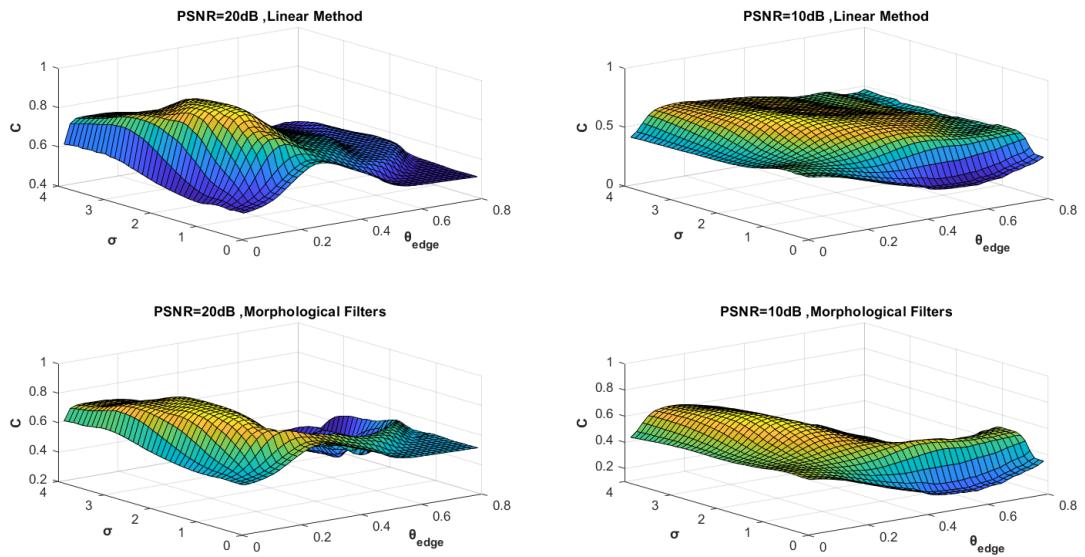
(b) Μορφολογικά φίλτρα  $\sigma=2 \theta=0.25$

#### Γενικά αποτελέσματα:

Εκτός των παραπάνω μετρήσεων που αναφέρθηκαν ενδεικτικά, υλοποιήθηκε για καλύτερη οπτικοποίηση όσων ειπώθηκαν η συνάρτηση  $C(\sigma, \theta_{edge})$  η οποία αναπαριστά μεταβολή της επίδοσης με πεδίο τιμών  $[0,1]$  του αλγορίθμου συναρτήσει των παραμέτρων  $\sigma$  και  $\theta_{edge}$ . Η συνάρτηση αυτή εφαρμόστηκε στις εξής περιπτώσεις:

1. Γραμμική μέθοδος (Laplacian=0), PSNR = 20dB.
2. Γραμμική μέθοδος (Laplacian=0), PSNR = 10dB.
3. Μορφολογικά φίλτρα (Laplacian=1) , PSNR = 20dB.
4. Μορφολογικά φίλτρα (Laplacian=1) , PSNR = 10dB.

Ακολουθούν οι γραφικές παραστάσεις των περιπτώσεων αυτών:



Για τις παραπάνω γραφικές παραστάσεις οι χαλύτερες τιμές του  $C$  είναι οι εξής:

	<b>Max value of <math>C_x (\times \%)</math></b>	$\sigma$	$\theta_{edge}$
Γραμμική μέθοδος PSNR = 20dB	93.26	1.5	0.18
Γραμμική μέθοδος PSNR = 10dB	70.48	2	0.26
Μορφολογικά φίλτρα PSNR = 20dB	85.63	1.9	0.16
Μορφολογικά φίλτρα PSNR = 10dB	69.19	2.4	0.2

## 1.4 Εφαρμογή των Αλγορίθμων Ανίχνευσης Ακμών σε Πραγματικές εικόνες

Στο ερώτημα αυτό θα ασχοληθούμε με μια πραγματική εικόνα και θα προσπαθήσουμε να ανιχνεύσουμε τις ακμές της με την βιοήθεια του αλγορίθμου που υλοποιήσαμε παραπάνω. Δεδομένου ότι μιλάμε για μια πραγματική εικόνα περιμένουμε ο όγκος πληροφορίας που θα δεχτούμε να είναι ιδιαίτερα μεγάλος και ως εκ τούτου δεν θα προσθέσουμε θόρυβο. Ωστόσο μπορούμε να θεωρήσουμε ότι υπάρχει θόρυβος ήδη.

**Πως θα αξιολογηθεί;** Στο σημείο αυτό θέλουμε να εξετάσουμε ποιοτικά τον αλγόριθμο. Στόχος μας δεν είναι να βρούμε όσο περισσότερες ακμές μπορούμε αλλά ακμές οι οποίες μας δίνουν όσο καλύτερη απεικόνιση και αντίληψη των αντικειμένων που παρουσιάζονται στην εικόνα. Οι 2 παράγοντες που θα επηρεάσουν το αποτέλεσμα μας είναι ο όγκος πληροφορία και ο θόρυβος που υπάρχει στην εικόνα. Επομένως η αξιολόγηση θα γίνει προσεγγιστικά δίχως κάποια μετρική που να υποδεικνύει τον «σωστό» συνδυασμό παραμέτρων.

**Γιατί χανά C;** Ο λόγος που δεν χρησιμοποιούμε ξανά τον δείκτη C είναι ότι πριν ο αλγόριθμος μας κλήθηκε να πάρει μια εικόνα με θόρυβο και να εκτιμήσει τις ακμές της και συγχρίθηκαν τα αποτελέσματα του με την εκτίμηση ακμών της εικόνας δίχως θόρυβο. Σε αυτήν την περίπτωση δεν έχουμε κάποιο μέτρο σύγχρισης για να τεστάρουμε τον αλγόριθμο μας με νούμερα διότι ότι θόρυβος υπάρχει είναι εσωτερικός.

Παρατίθεται η αρχική εικόνα:



Εικόνα 8: Αρχική εικόνα

- Αρχικά θα κάνουμε την προσομοίωση αλλάζοντας τις παραμέτρους στην γραμμική μέθοδο. Εδώ είναι μερικά τα αποτελέσματα για διάφορες τιμές παραμέτρων:



$\sigma=0.1$  και  $\theta_{edge} = 0.2$

$\sigma=0.1$  και  $\theta_{edge} = 0.3$

$\sigma=0.2$  και  $\theta_{edge} = 0.2$

Από τις πρώτες γραφικές συμπεραίνουμε ότι ο θόρυβος είναι ελάχιστος γιατί πάροντας μια καθαρή εικόνα ακμών οπτικά κοντά με την αρχική. Θέλουμε καλή ανάλυση των ακμών και λεπτομέρεια που προσφέρει η μικρή εξομάλυνση ωστόσο σε βαθμό που να μην αλλοιώνει την υφή των αντικειμένων. Επίσης, θέλουμε όση πληροφορία χρειάζεται για να μπορέσουμε να αντιληφθούμε το περιεχόμενο της εικόνας, καθώς περισσότερη πληροφορία αποτελεί οπτικό θόρυβο και δεν κάνει σαφές το περιεχόμενο. Οπότε θέλουμε ένα κατώφλι με τιμή στο 0.2 που αποδίδει επαρκώς το περιεχόμενο δίχως να προβληματίζει. Το στις παραπάνω εικόνες παρέμεινε σχετικά σταθερό. Ας δούμε σε μεγαλύτερες κλίμακες τι προσφέρει:



$\sigma=1$  και  $\theta_{edge} = 0.2$

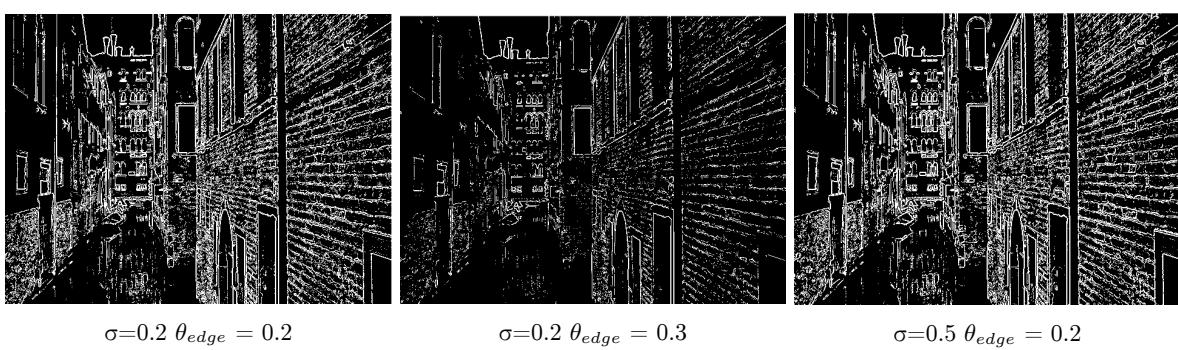
$\sigma=1$  και  $\theta_{edge} = 0.3$

$\sigma=2$  και  $\theta_{edge} = 0.2$

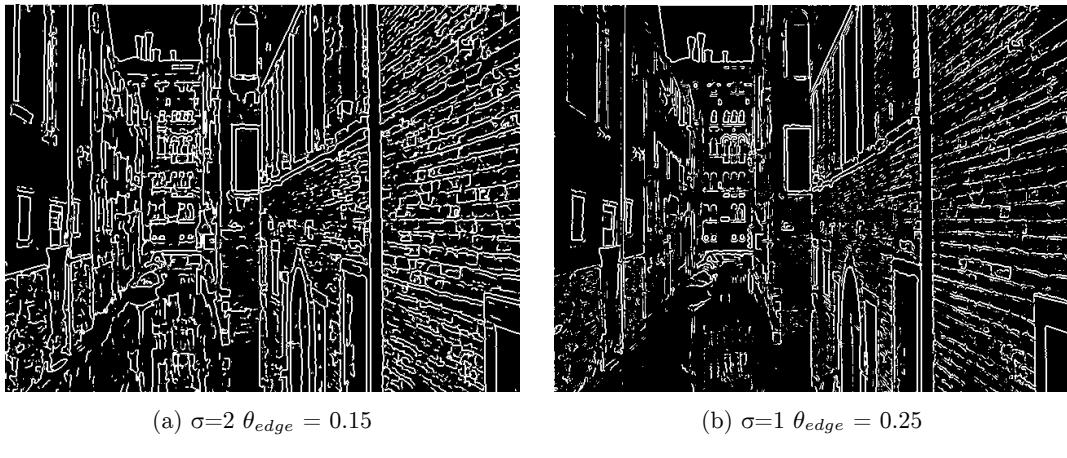
Στις μεγαλύτερες αυτές κλίμακες προσφέρει πληροφορία η οποία είναι λίγο αλλοιωμένη εξαιτίας της εξομάλυνσης που έχει υποστεί, επομένως αδύνατεί να διατηρήσει την πληροφορία ακέραια και την εμφανίζει με μια κάποια παραμόρφωση που επηρεάζει την αντίληψη των αντικειμένων σε αυτά. Η καλύτερη επιλογή παραμέτρων είναι για  $\theta_{edge} = 0.2$  και τυπική απόκλιση  $\sigma = 0.2$ .



- Στη συνέχεια θα επαναλάβουμε την διαδικασία αλλάζοντας τις παραμέτρους στην μέθοδο με τα μορφολογικά φίλτρα. Οι αλλαγές που περιμένουμε δεν είναι σημαντικές όπως είδαμε και σε προηγούμενα ερωτήματα.



Στις παραπάνω μετρήσεις δώσαμε έμφαση στις μικρές τιμές και για τις δύο παραμέτρους και τα αποτελέσματα είναι όλα ικανοποιητικά. Παρ' όλα αυτά αν κάποιο ξεχωρίζει μπορεί να πει κανείς ότι είναι αυτό με  $\sigma = 0.2$  και  $\theta_{edge} = 0.2$ . Οι τιμές με την μεγαλύτερη τυπική απόκλιση δεν έχουν τόσο λεπτές ακμές και η πληροφορία χάνεται όπως και με την αύξηση του  $\theta_{edge}$ .



Η καλύτερη επιλογή αυτών είναι για  $\sigma = 0.2$  και  $\theta_{edge} = 0.2$ , όπως και στο γραμμικό φίλτρο :



(a)  $\sigma=2$   $\theta_{edge} = 0.15$



(b)  $\sigma=1$   $\theta_{edge} = 0.2$

## 2 Ανίχνευση Σημείων Ενδιαφέροντος (Interest Point Detection)

Στο δεύτερο μέρος της εργαστηριακής άσκησης θα ασχοληθούμε με την ανίχνευση σημείων ενδιαφέροντος σε εικόνες. Αρχικά, θα παρουσιαστεί ένα σύνολο από ανιχνευτές και, στη συνέχεια, θα σχολιάσουμε τις διαφορές που έχουν μεταξύ τους. Οι εικόνες που χρησιμοποιήθηκαν για την ανάλυση φαίνονται παρακάτω:



Λουλούδια



Μπαλόνια με φόντο τον ουρανό

Εικόνα 16: Αρχικές εικόνες

Οι μέθοδοι που θα χρησιμοποιηθούν για την ανίχνευση των σημείων ενδιαφέροντος δεν χρειάζονται και τα τρία κανάλια RGB των έγχρωμων εικόνων. Για τον λόγο αυτό, αρχικά οι εικόνες μετατράπηκαν σε ασπρόμαυρες (grayscale) όπως φαίνεται παρακάτω:



Λουλούδια σε ασπρόμαυρη



Μπαλόνια με φόντο τον ουρανό σε ασπρόμαυρη

Εικόνα 17: Αρχικές εικόνες σε grayscale

## 2.1 Ανίχνευση Γωνιών

Μια διαισθητικά απλή κατηγορία σημείων ενδιαφέροντος είναι η επιλογή σημείων που αντιστοιχούν σε γωνίες της εικόνας, καθώς οι γωνίες είναι αυτές που σχηματίζουν τα διάφορα σχήματα σε αυτήν. Για την ανίχνευση των γωνιών χρησιμοποιήθηκε η κλασσική μέθοδο των Harris-Stephens, η οποία βασίζεται στην απλή ιδέα ότι σε ένα παράθυρο μίας εικόνας έχουμε γωνία όταν υπάρχει σημαντική διαφορά στην φωτεινότητα καθώς μετακινούμε το παράθυρο προς οποιαδήποτε κατεύθυνση.

Ο αλγόριθμος εντοπίζει τις διαφορές αυτές σχηματίζοντας έναν πίνακα  $M$  με τις μερικές παραγώγους και υπολογίζοντας τις ιδιοτιμές του. Όταν και οι δύο ιδιοτιμές είναι μεγάλες, αυτό δείχνει ότι η φωτεινότητα μεγαλώνει σε όλες τις κατευθύνσεις οπότε έχουμε γωνία. Πιο συγκεκριμένα, υπολογίστηκαν αρχικά τα στοιχεία  $J_1, J_2$  και  $J_3$  του δομικού τανυστή  $J$  σύμφωνα με τις παρακάτω εξισώσεις:

$$J_1(x, y) = G_p * \left( \frac{\partial I_\sigma}{\partial x} \frac{\partial I_\sigma}{\partial x} \right)(x, y) \quad (6)$$

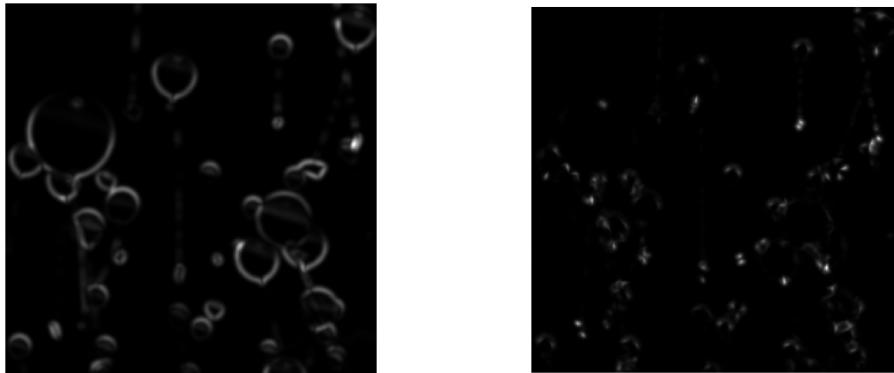
$$J_2(x, y) = G_p * \left( \frac{\partial I_\sigma}{\partial x} \frac{\partial I_\sigma}{\partial y} \right)(x, y) \quad (7)$$

$$J_3(x, y) = G_p * \left( \frac{\partial I_\sigma}{\partial y} \frac{\partial I_\sigma}{\partial y} \right)(x, y) \quad (8)$$

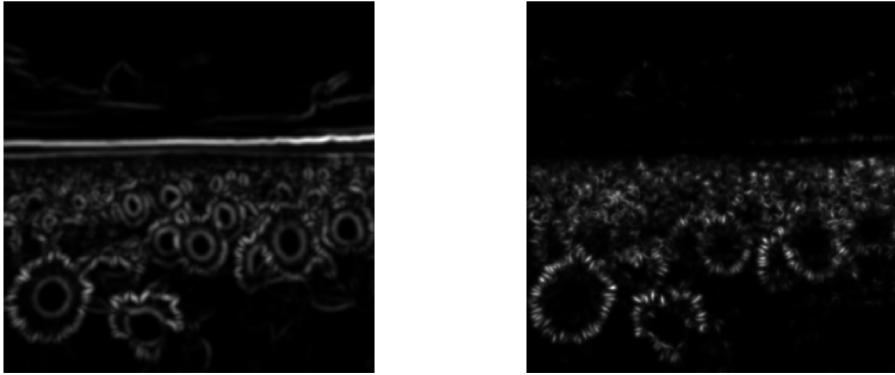
όπου όπου  $I_\sigma = G_\sigma * I$ . Στη συνέχεια, υπολογίζονται οι ιδιοτιμές  $\lambda_+, \lambda_-$  του τανυστή  $J$ , σύμφωνα με την παρακάτω σχέση:

$$\lambda_{\pm}(x, y) = \frac{1}{2} (J_1 + J_3 \pm \sqrt{(J_1 - J_3)^2 + 4J_2^2}) \quad (9)$$

Παρακάτω βλέπουμε τις τιμές  $\lambda_+$  και  $\lambda_-$  των δύο εικόνων σαν ασπρόμαυρες εικόνες:



Εικόνα 18: Ιδιοτιμές  $\lambda_+$  και  $\lambda_-$



Εικόνα 19: Ιδιοτιμές  $\lambda_+$  και  $\lambda_-$

Αυτό που παρατηρούμε είναι ότι η  $\lambda_+$  ιδιοτιμή έχει υψηλή τιμή τόσο στις ακμές όσο και στις γωνίες. Αντίθετα, η  $\lambda_-$  ιδιοτιμή έχει υψηλή τιμή κυρίως στις γωνίες.

Τέλος, έχουμε το ακόλουθο κριτήριο γωνιότητας:

$$R(x, y) = \lambda_- \lambda_+ - k(\lambda_- + \lambda_+)^2 \quad (10)$$

Οι παράμετροι που δέχεται είναι η κλίμακα διαφρίσης ( $\sigma$ ) και ολοκλήρωσης ( $\rho$ ), οι οποίες αποτελούν τις τυπικές αποκλίσεις των δισδιάστατων Gaussian πυρήνων ομαλοποίησης που χρησιμοποιούνται, η σταθερά γωνιότητας ( $k$ ) και το κατώφλι  $\theta_{corn}$ . Οι default τιμές για τις παραμέτρους ήταν οι παρακάτω:

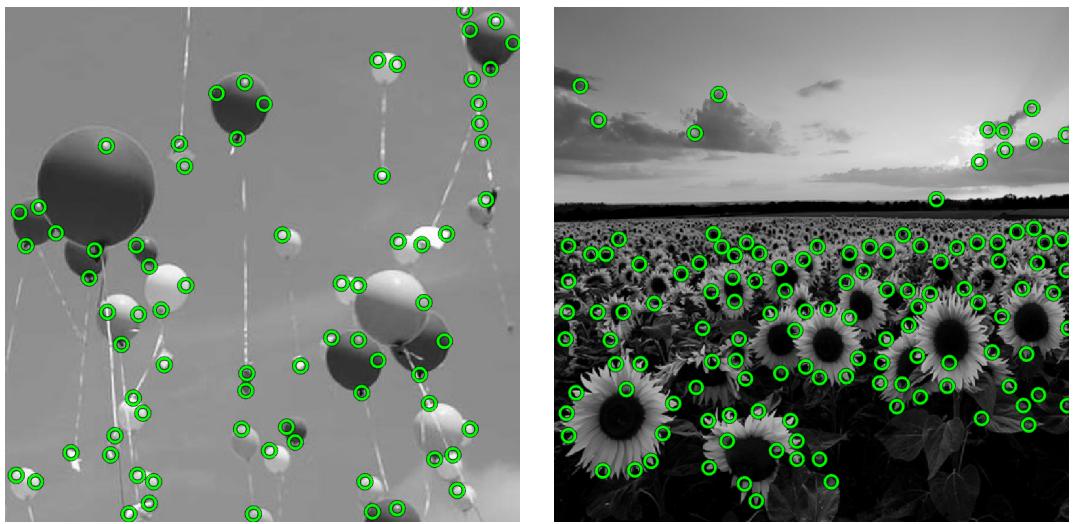
✓  $\sigma = 2$

✓  $k = 0.05$

✓  $\rho = 2.5$

✓  $\theta_{corn} = 0.005$

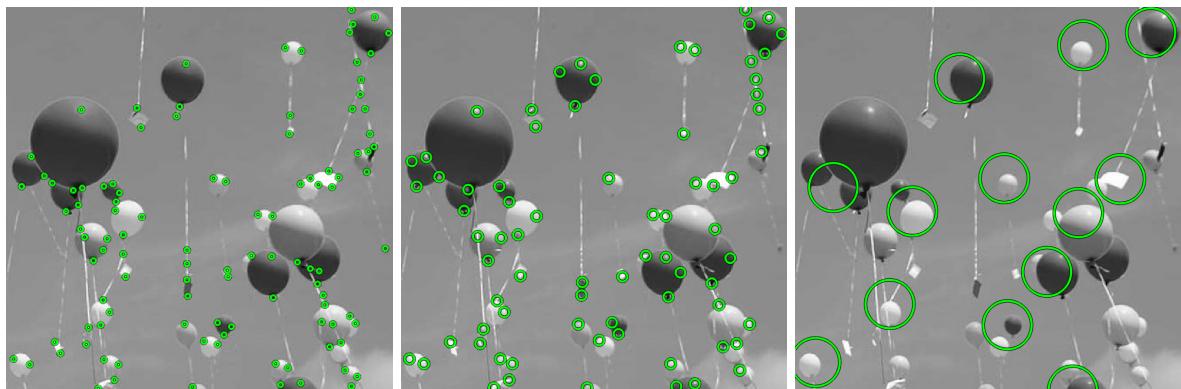
Με τις παραπάνω τιμές έχουμε τις γωνίες που φαίνονται με πράσινο κύκλο στην κάθε εικόνα (η ακτίνα του κύκλου σχετίζεται με το  $\sigma$ , δηλαδή με την κλίμακα στην οποία ανιχνεύτηκε η γωνία):



Εικόνα 20: Σημεία ενδιαφέροντος με την μέθοδο Harris-Stephens

Παρακάτω ακολουθούν τα αποτελέσματα μεταβάλλοντας κάθε φορά μία από τις παραμέτρους (οι υπόλοιπες έχουν τις default τιμές τους) και ένας σύντομος σχολιασμός.

- **Μεταβολή στην κλίμακα  $\sigma$ .**



$\Gamma\alpha \sigma = 1$

$\Gamma\alpha \sigma = 2$

$\Gamma\alpha \sigma = 10$



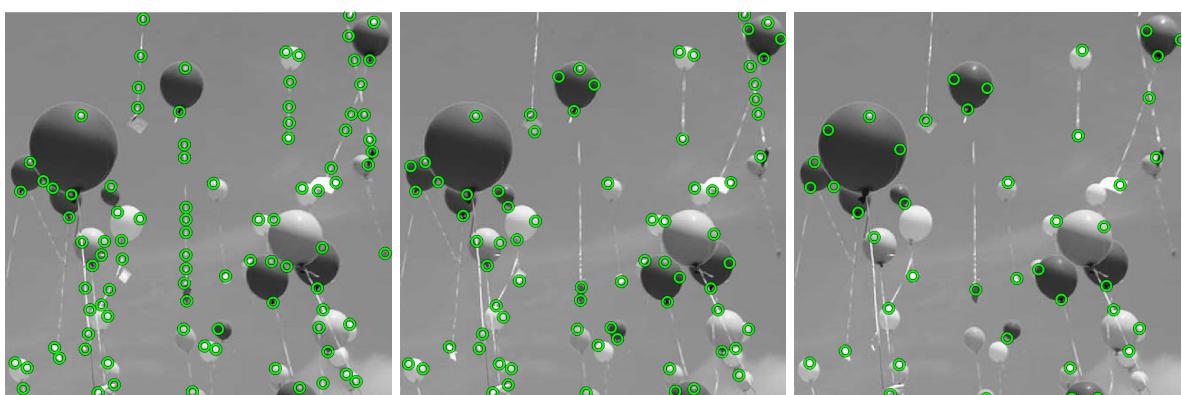
Για  $\sigma = 1$

Για  $\sigma = 2$

Για  $\sigma = 10$

Παρατηρούμε, λοιπόν, ότι το σ επηρεάζει την κλίμακα στην οποία ψάχνουμε γωνίες. Στις μικρές τιμές του, εντοπίζονται πολλά σημεία καθώς μία μικρή εναλλαγή στην εικόνα θεωρείται γωνία, ενώ αντίθετα στις πιο μεγάλες τιμές εντοπίζονται σημεία με μεγάλες γωνίες χωρίς να νοιαζόμαστε για την λεπτομέρεια.

- **Μεταβολή στην κλίμακα  $\rho$ .**



Για  $\rho = 0.5$

Για  $\rho = 2.5$

Για  $\rho = 10$



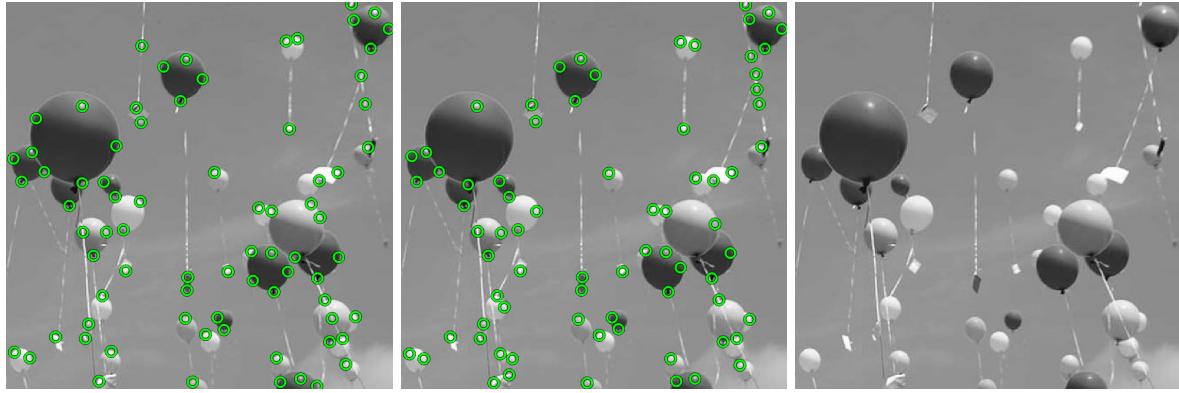
Για  $\rho = 0.5$

Για  $\rho = 2.5$

Για  $\rho = 10$

Δεν παρατηρείται κάποια συγκεκριμένη επιρροή της παραμέτρου, καθώς η αύξησή της ενώ από την μία αποκόπτει κάποιες γωνίες, από την άλλη εντοπίζει άλλες νέες.

- Μεταβολή στο χριτήριο γωνιότητας  $k$ .



Για  $k = 0.0001$

Για  $k = 0.05$

Για  $k = 0.5$



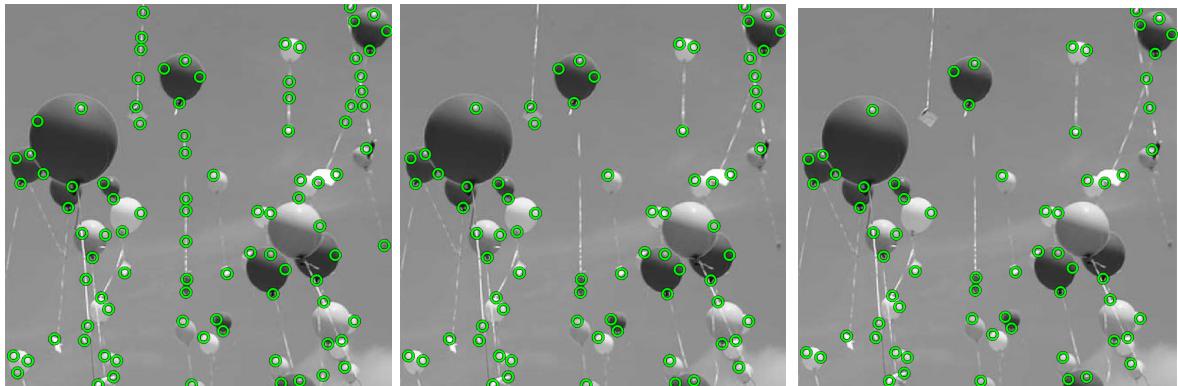
Για  $k = 0.0001$

Για  $k = 0.05$

Για  $k = 0.5$

Παρατηρούμε ότι καθώς αυξάνεται το  $k$  όλο και λιγότερες γωνίες εντοπίζονται. Αυτό είναι λογικό καθώς το χριτήριο γωνιότητας μειώνεται ενώ το κατώφλι παραμένει σταθερό. Συνεπώς, αυξάνοντας το  $k$  είναι σαν να δεχόμαστε σημεία που έχουν όλο και μεγαλύτερες μεταβολές στην φωτεινότητα.

- Μεταβολή στο κατώφλι  $\theta_{corn}$ .



Για  $\theta_{corn} = 0.001$

Για  $\theta_{corn} = 0.005$

Για  $\theta_{corn} = 0.01$



Για  $\theta_{corn} = 0.001$

Για  $\theta_{corn} = 0.005$

Για  $\theta_{corn} = 0.01$

Έχουμε παρόμοια συμπεριφορά με το  $k$ , μόνο που τώρα αυξάνουμε το κατώφλι για να δεχτούμε ένα σημείο ως γωνία, οπότε κάθε φορά λιγότερα σημεία θεωρούνται γωνίες.

## 2.2 Πολυκλιμακωτή Ανίχνευση Γωνιών

Στο προηγούμενο ερώτημα η ανίχνευση των γωνιών γινόταν σε μία μόνο κλίμακα η οποία οριζόταν από τις παραμέτρους  $s$  και  $r$  και, έτσι, χρειαζόταν κάθε φορά να ορίσουμε τις τιμές τους. Στο ερώτημα αυτό επεκτείνουμε την παραπάνω μέθοδο έτσι ώστε να ανιχνεύει γωνίες σε πολλαπλές κλίμακες και να μας επιστρέψει εκτός από τα σημεία που αντιστοιχούν σε γωνίες και την κλίμακα στην οποία ανιχνεύτηκαν. Η μέθοδος ονομάζεται Harris-Laplacian και αποτελείται από δύο στάδια, ένα για την επιλογή σημείων σε κάθε κλίμακα και ένα για την τελική επιλογή των σημείων που παρουσιάζουν μέγιστο στη κανονικοποιημένη Λαπλασιανή της Γκαουσιανής, που ορίζεται ως εξής:

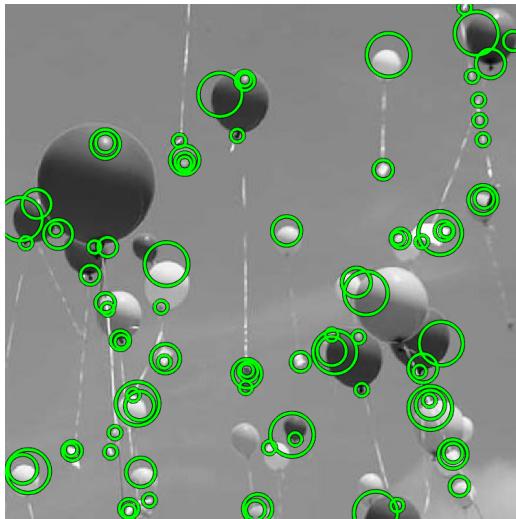
$$|LoG(x, \sigma_i)| = \sigma_i^2 |L_{xx}(x, \sigma_i) + L_{yy}(x, \sigma_i)|, i = 0, \dots, N - 1 \quad (11)$$

όπου  $N$  ο αριθμός των κλιμάκων.

Παρακάτω φαίνονται τα σημεία που εντοπίστηκαν για τις παρακάτω παραμέτρους:

- ✓  $\sigma = 2$
- ✓  $\rho = 2.5$
- ✓  $k = 0.05$

- ✓  $\theta_{corn} = 0.005$
- ✓  $s = 1.5$
- ✓  $N = 4$



Λουλούδια

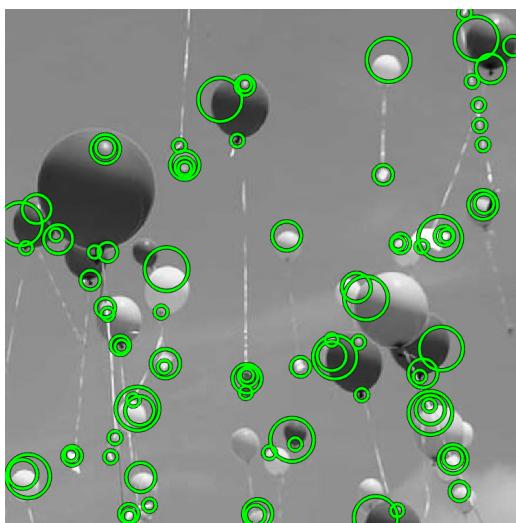


Μπαλόνια με φόντο τον ουρανό

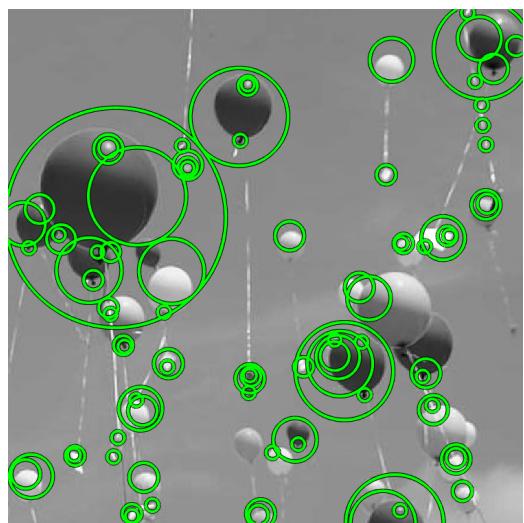
Εικόνα 29: Σημεία ενδιαφέροντος με την μέθοδο Harris-Laplacian

Παρατηρούμε ότι με την μέθοδο αυτή, εντοπίζουμε γωνίες ανεξάρτητες από την κλίμακα και τα αποτελέσματα είναι πιο ρεαλιστικά, καθώς είναι λογικό ως σημεία ενδιαφέροντος να είναι γωνίας και μικρής και μεγάλης κλίμακας. Παρακάτω, βλέπουμε την συμπεριφορά των σημείων καθώς αλλάζουμε τις παραμέτρους  $s$  και  $N$ .

- Μεταβολή  $N$



Για  $N = 4$



Για  $N = 8$

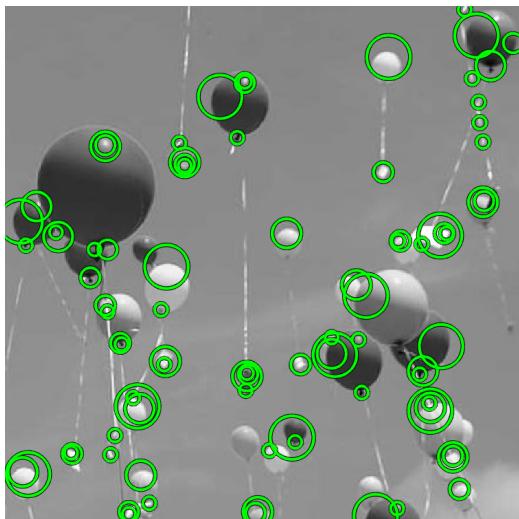


Γα N = 4

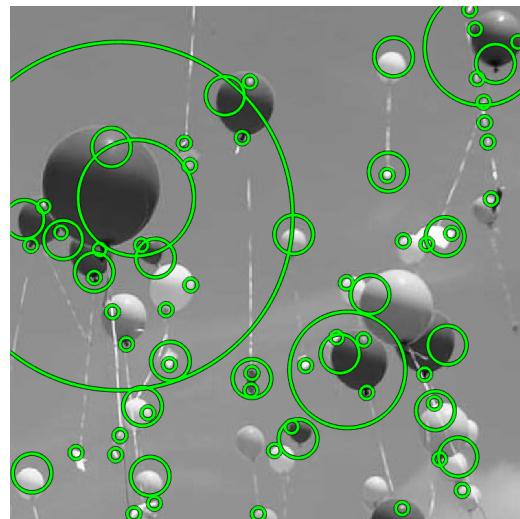


Γα N = 8

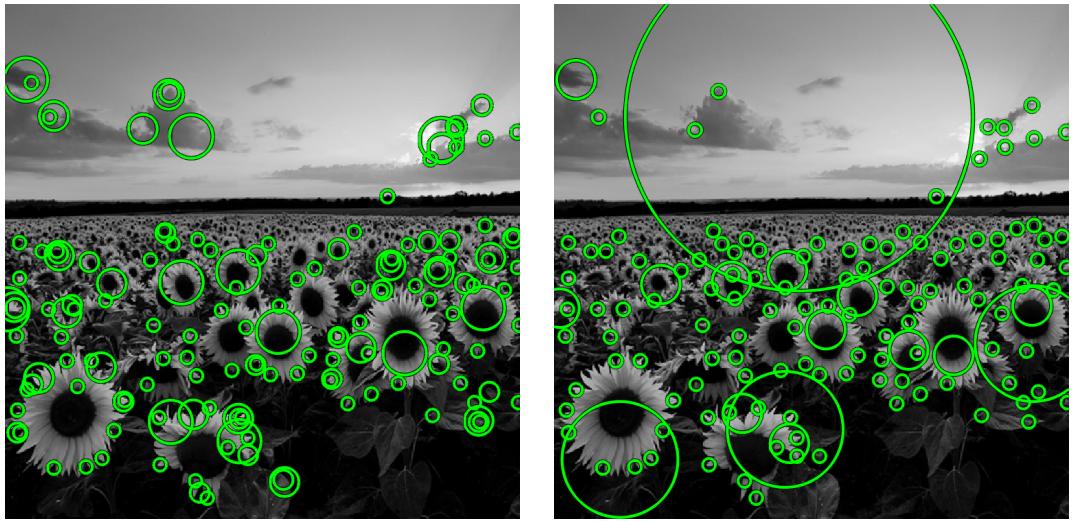
• Μεταβολή s



Γα s = 1.5



Γα s = 3



Για  $s = 1.5$

Για  $s = 3$

Παρατηρούμε ότι αυξάνοντας τον παράγοντα κλιμάκωσης  $s$ , τα αποτελέσματα είναι χειρότερα, καθώς αυξάνεται ο λόγος μεταξύ διαφορετικών κλιμάκων. Έτσι, πολλές γωνίες που θα μπορούσαν να αποτελέσουν σημεία ενδιαφέροντος αποκόπτονται.

### 2.3 Ανίχνευση Blobs

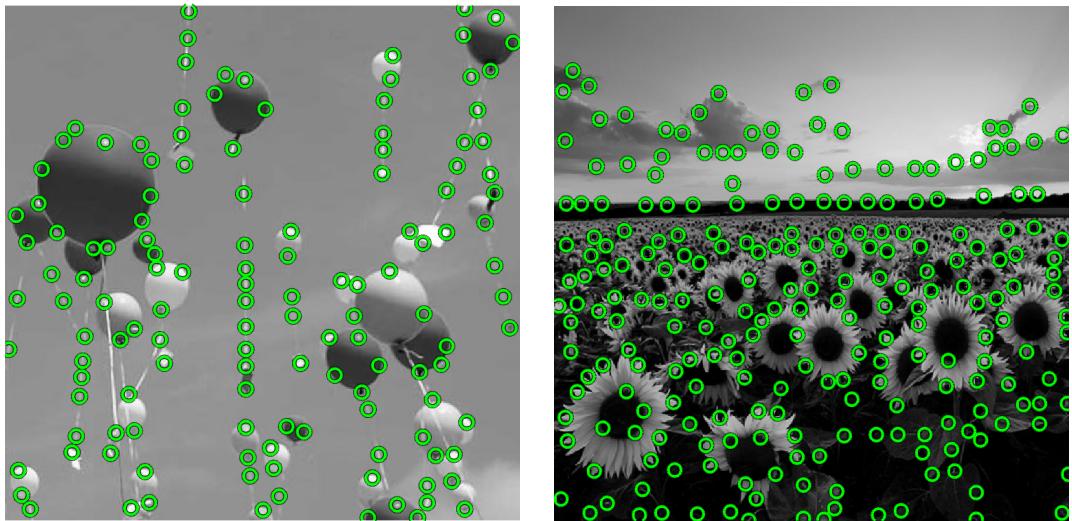
Μια από τις σημαντικότερες κατηγορίες σημείων ενδιαφέροντος βασίζονται στην ανίχνευση ‘blobs’, που ορίζονται ως περιοχές με κάποια ομοιογένεια που διαφέρουν σημαντικά από την γειτονιά τους. Για την εύρεση τέτοιων περιοχών, σε αντιστοιχία με το κριτήριο γωνιότητας της μεθόδου Harris, γίνεται χρήση των μερικών παραγώγων δεύτερης τάξης της εικόνας και συγκεκριμένα η ορίζουσα του πίνακα Hessian:

$$H(x, y) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix} \quad (12)$$

όπου  $L_{xx}(x, y, \sigma) = \frac{\partial^2}{\partial x^2}\{I_\sigma(x, y)\}$ ,  $L_{yy}(x, y, \sigma) = \frac{\partial^2}{\partial y^2}\{I_\sigma(x, y)\}$  και  $L_{xy}(x, y, \sigma) = \frac{\partial^2}{\partial x \partial y}\{I_\sigma(x, y)\}$ .

Στη συνέχεια, η ορίζουσα του πίνακα Hessian αποτελεί το κριτήριο γωνιότητας αντίστοιχα με την μέθοδο Harris. Πριν καν δούμε τα αποτελέσματα, μπορούμε να εντοπίσουμε ένα πλεονέκτημα της μεθόδου αυτής έναντι της Harris, καθώς εξαρτάται μόνο από 2 παραμέτρους σε σύγκριση με την Harris που εξαρτιόταν από 4.

Τα blobs που εντοπίστηκαν για  $\sigma = 2$  και  $\theta_{corn} = 0.005$  φαίνονται παρακάτω:

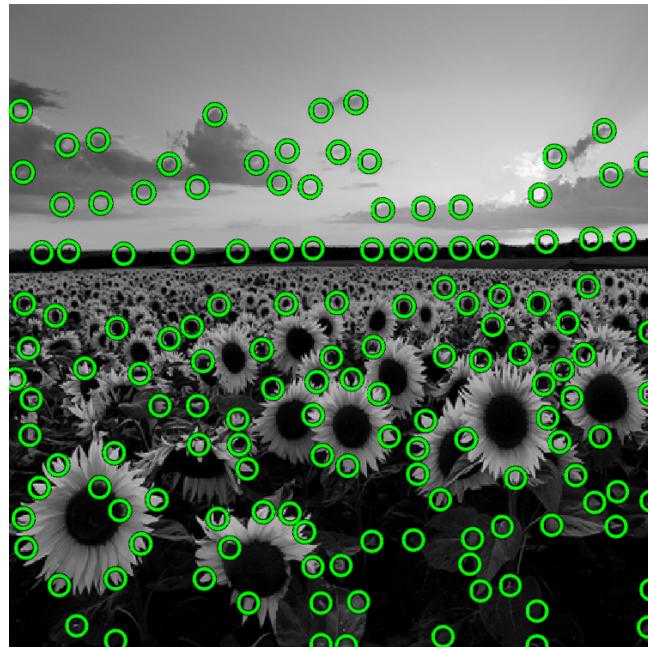


Λουλούδια

Μπαλόνια με φόντο τον ουρανό

Εικόνα 34: Σημεία ενδιαφέροντος μέσω της ανίχνευσης blobs

Παρατηρούμε ότι τα αποτελέσματα είναι καλύτερα από την μέθοδο Harris. Συγκεκριμένα, στην εικόνα με τα μπαλόνια εντοπίζονται όλα τα σημαντικά σημεία, που είναι το περίγραμμα των μπαλονιών. Όσων αφορά την εικόνα με τα λουλούδια, εντοπίζονται πάρα πολλά σημεία πράγμα που οφείλεται στο ότι η συγκεκριμένη εικόνα έχει πολλές μεταβολές. Έτσι, όπως είδαμε και στην μέθοδο Harris αν αυξήσουμε την παράμετρο  $\sigma$  θα αναζητήσουμε σημεία σε μεγαλύτερη κλίμακα. Πράγματι, όπως φαίνεται παρακάτω για  $\sigma = 2.5$  έχουμε πολύ καλά αποτελέσματα.

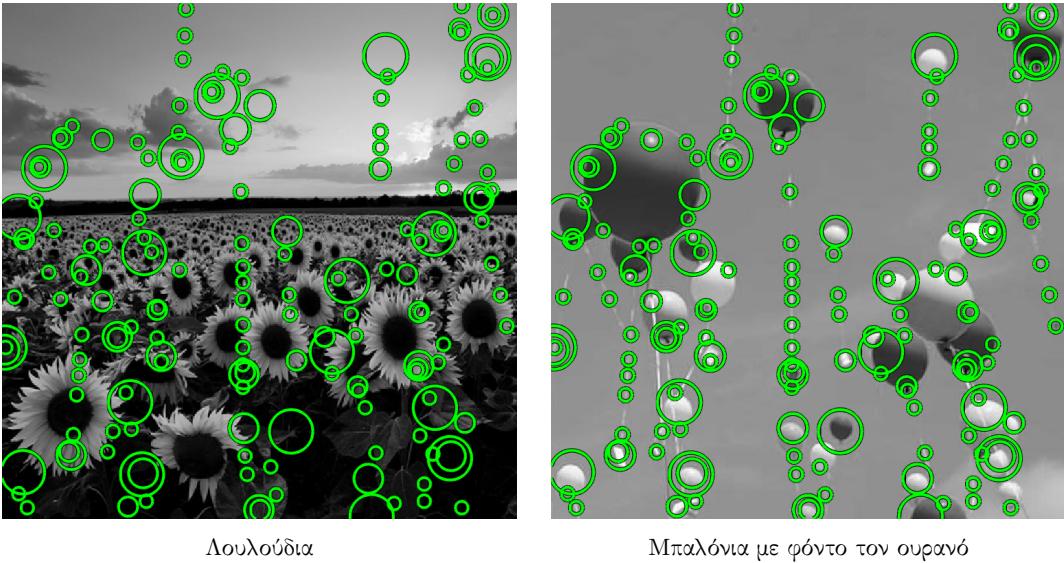


Εικόνα 35: Σημεία ενδιαφέροντος μέσω της ανίχνευσης Blobs για  $\sigma = 2.5$ .

Δεν χρειάζεται να αναλύσουμε και εδώ την επιρροή των παραμέτρων καθώς είναι αντίστοιχη με αυτή που είχαν στην μέθοδο Harris.

## 2.4 Πολυκλιμακωτή Ανίχνευση Blobs

Σε αυτό το σημείο επεκτείνουμε την παραπάνω μέθοδο για ανίχνευση Blobs σε πολλαπλές κλίμακες, αντίστοιχα με αυτό που κάναμε για τον Harris στο 2.2.

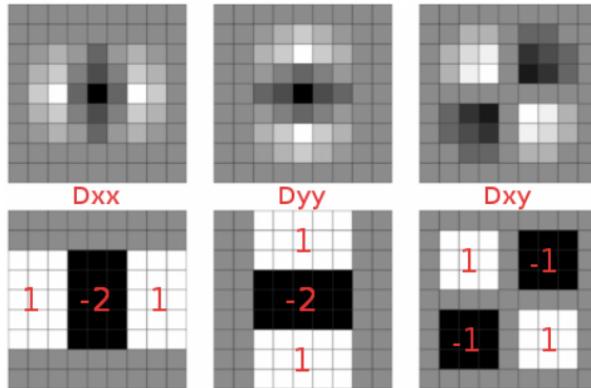


Εικόνα 36: Σημεία ενδιαφέροντος μέσω της πολυκλιμακωτής ανίχνευσης Blobs

Προφανώς, αυτός ο detector είναι και ο αποδοτικότερος από όλους, καθώς συνδυάζει τα πλεονεκτήματα των blobs και της πολυκλιμακωτής ανίσχευσης. Άμα παρατηρήσουμε τις εικόνες, εντοπίζονται σημεία με χρήσιμη πληροφορία χωρίς όμως να είναι πάρα πολλά (ακόμα και στην εικόνα των λουλουδιών όπου έχουμε πολλές μεταβολές).

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

Στην μέθοδο 2.3, ο υπολογισμός της Hessian για κάθε κλίμακα αντιστοιχεί στην συνέλιξη της εικόνας με αυξανόμενου μεγέθους φίλτρα, που είναι μια υπολογιστικά ακριβή διαδικασία. Για την επιτάχυνση της μεθόδου αυτής, προτάθηκε η προσέγγιση των φίλτρων 2ης παραγάγου με Box Filters, δηλαδή με φίλτρα που βασίζονται σε αθροίσματα ορθογώνιων περιοχών, όπως φαίνεται και στην παρακάτω εικόνα. Ο υπολογισμός τέτοιων αθροίσμάτων γίνεται πολύ αποτελεσματικά με χρήση Ολοκληρωτικών Εικόνων.



Εικόνα 37: Οπτικοποίηση της προσέγγισης των φίλτρων 2ης παραγάγου με Box Filters

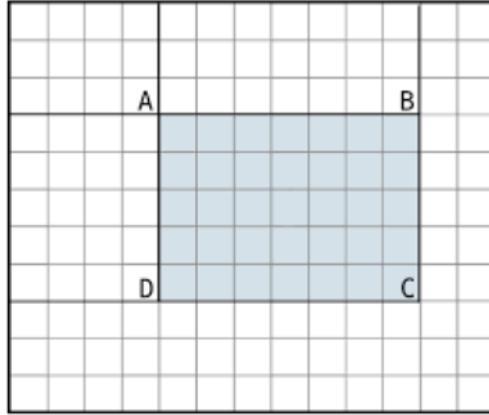
Για δεδομένο, το οποίο αντιστοιχεί σε φίλτρο μεγέθους  $n \times n$  με  $n = 2\text{ceil}(3\sigma) + 1$ , το μέγεθος των παραπάνω παραθύρων είναι ως εξής:

- $D_{xx}$  : Ύψος παραθύρου  $4 \times \text{floor}(n/6) + 1$  και πλάτος παραθύρου  $2 \times \text{floor}(n/6) + 1$ .
- $D_{yy}$  : Ύψος παραθύρου  $2 \times \text{floor}(n/6) + 1$  και πλάτος παραθύρου  $4 \times \text{floor}(n/6) + 1$
- $D_{xy}$  : Ύψος παραθύρου  $2 \times \text{floor}(n/6) + 1$  και πλάτος παραθύρου  $2 \times \text{floor}(n/6) + 1$

Αφού υπολογίσουμε τα  $L_{xx}$ ,  $L_{xy}$  και  $L_{yy}$  με την χρήση των box filters και των ολοκληρωτικών εικόνων (λεπτομέρειες παρακάτω), υπολογίζουμε τα τοπικά μέγιστα (όπως στα προηγούμενα μοντέλα) με το ακόλουθο χριτήριο:

$$R(x, y) = L_{xx}(x, y)L_{yy}(x, y) - (0.9L_{xy}(x, y))^2 \quad (13)$$

**Τυλοποίηση:** Στο μέρος αυτό, κρίνεται σκόπιμο να αναφερθούν ορισμένα σημεία σχετικά με την υλοποίησή σε Matlab καθώς παρουσιάζει ενδιαφέρον ο τρόπος χρήσης ολοκληρωτικών εικόνων για την επιτάχυνση της διαδικασίας. Συγκεκριμένα, όταν έχουμε μία ολοκληρωτική εικόνα  $S$ , οποιοδήποτε άθροισμα εντός ενός διθέντος παραθύρου πάνω στην αρχική εικόνα μπορεί να υπολογιστεί άμεσα.



Εικόνα 38: Παράθυρο Υπολογισμού Αθροίσματος

Για παράδειγμα, στην παραπάνω εικόνα αν θέλουμε να υπολογίσουμε το άθροισμα των στοιχείων της εντός του παραθύρου ABCD αρκεί να υπολογίζουμε το παρακάτω:

$$\sum_{ABCD} \sum I(i, j) = S_A + S_C - S_B - S_D \quad (14)$$

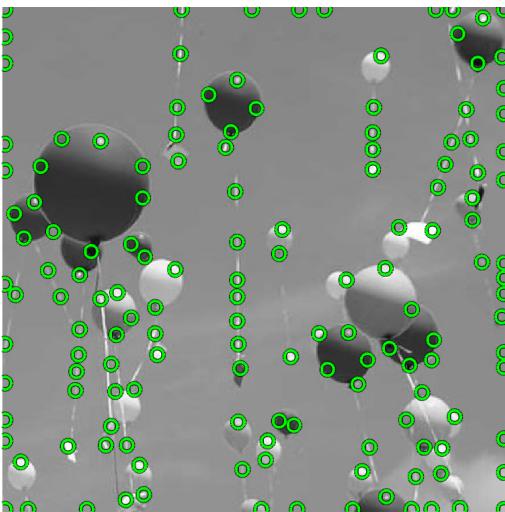
Παρατηρούμε, επίσης, ότι τα τρία φίλτρα της εικόνας 35 αποτελούν σταθμισμένο συνδυασμό αποκρίσεων ενός παραθύρου σταθερού μεγέθους. Έτσι, αν υπολογίσουμε με την παραπάνω μέθοδο για κάθε φίλτρο την συνέλιξη της εικόνας με ένα κεντρικό φίλτρο μεγέθους  $height * width$ , έστω  $S$ , τότε έχουμε:

- Το  $D_{xx}$  ισούται με  $S(-width) - 3 * S + S(width)$ .
- Το  $D_{yy}$  ισούται με  $S(-height) - 3 * S + S(height)$ .
- Το  $D_{xy}$  ισούται με  $S(1 + \frac{D_{xy}}{2}, 1 + \frac{D_{xy}}{2}) + S(-1 - \frac{D_{xy}}{2}, -1 - \frac{D_{xy}}{2}) - S(1 + \frac{D_{xy}}{2}, -1 - \frac{D_{xy}}{2}) - S(-1 - \frac{D_{xy}}{2}, 1 + \frac{D_{xy}}{2})$ .

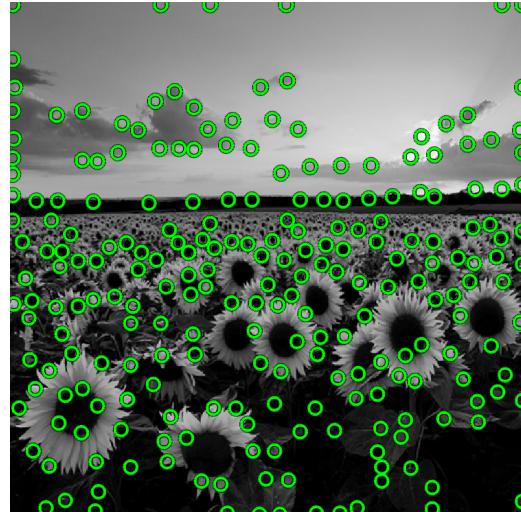
Συνεπώς, ενώ γενικά για εικόνα μεγέθους  $N \times M$  και φίλτρο μεγέθους  $h_1 \times h_2$  η συνέλιξή τους γίνεται σε  $\mathcal{O}(NMh_1h_2)$  χρόνο, με την μέθοδό που αναφέρθηκε έχουμε  $\mathcal{O}(1)$  χρόνο.

Τα αποτελέσματα για τις default τιμές που έγιναν και στα μέρη 1.3 και 1.4 φαίνονται παρακάτω:

- **Σε συγκεκριμένη κλίμακα:**



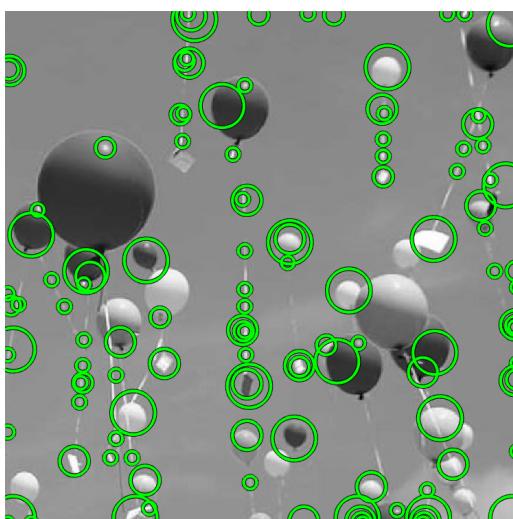
Λουλούδια



Μπαλόνια με φόντο τον ουρανό

Εικόνα 39: Σημεία ενδιαφέροντος μέσω της ανίχνευσης Blobs με χρήση Box Filters

- Πολυκλιμακωτή ανάλυση:



Λουλούδια



Μπαλόνια με φόντο τον ουρανό

Εικόνα 40: Σημεία ενδιαφέροντος μέσω της πολυκλιμακωτής ανίχνευσης Blobs με χρήση Box Filters

Παρατηρούμε ότι και στις δύο περιπτώσεις τα σημεία ενδιαφέροντος που εντοπίστηκαν είναι παρόμοια με αυτά της μεθόδου όπου η συνέλιξη υπολογίζοταν αναλυτικά. Αυτό που κερδίσαμε όμως τώρα είναι η ταχύτητα εκτέλεσης, όπως φαίνεται στον παρακάτω πίνακα (προφανώς για μεγαλύτερες εικόνες και φίλτρα θα είχαμε αντίστοιχα μεγαλύτερη βελτίωση):

Μέθοδος	Χρόνος εκτέλεσης
Συνέλιξη σε μία κλίμακα	0.102533 s
Box Filters σε μία κλίμακα	0.058712 s
Συνέλιξη σε πολλές κλίμακες	0.362171 s
Box Filters σε πολλές κλίμακες	0.329298 s

Table 5: Σύγκριση στην χρονική επίδοση των 2 μεθόδων

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

Τα σημεία ενδιαφέροντος δίνουν μια εκτίμηση περιοχών, οι οποίες περιέχουν σημαντικά χαρακτηριστικά της εικόνας. Για τον λόγο αυτό από τις περιοχές αυτές εξάγουμε τοπικούς περιγραφητές (local descriptors), που χωδικοποιούν μια γειτονιά (με ακτίνα που εξαρτάται από την κλίμακα) γύρω από τα σημεία ενδιαφέροντος. Ως τοπικούς περιγραφητές θα χρησιμοποιήσουμε τους εξής:

- **SURF** (Speed Up Robust Features)

Αρχικά, υπολογίζουμε τις κατεύθυντικές παραγώγους χρησιμοποιώντας Haar Wavelets. Στη συνέχεια, βρίσκουμε την κατεύθυνση για μια περιοχή γύρω από το σημείο ενδιαφέροντος και, τελικά, υπολογίζουμε ένα 64-διάστατο διάνυσμα χαρακτηριστικών για ένα τετραγωνικό παράθυρο με την παραπάνω κατεύθυνση και με μέγεθος εξαρτώμενο από την κλίμακα.

- **HOG** (Histogram of Oriented Gradients)

Αρχικά, υπολογίζουμε για κάθε pixel της εικόνας το μέτρο και η γωνία των κατεύθυντικών παραγώγων ως εξής:

$$m(x, y) = \sqrt{I_x^2(x, y) + I_y^2(x, y)} \quad (15)$$

$$\theta(x, y) = \arctan \frac{I_y(x, y)}{I_x(x, y)} \quad (16)$$

Στη συνέχεια, χωρίζουμε την εικόνα σε ορθογώνια κελιά (επικαλυπτόμενα ή μη) και για κάθε ένα από αυτά υπολογίζουμε ένα ιστογράμμα κατεύθυνσης. Για την δημιουργία του κάθε ιστογράμματος, χωρίζουμε το εύρος  $0\text{-}180^\circ$  σε  $n$  τμήματα και για κάθε pixel προσθέτουμε το μέτρο της παραγώγου του  $m(x, y)$  στο κομμάτι στο οποίο ανήκει με βάση το  $\theta(x, y)$ . Τέλος, μετά από μία κανονικοποίηση κατά μπλοκ, όλα τα ιστογράμματα συνενώνονται δίνοντας τον τελικό περιγραφητή της εικόνας.

#### 3.1 Ταίριασμα Εικόνων υπό Περιστροφή και Αλλαγή Κλίμακας

Στο μέρος αυτό θα εξετάσουμε την ικανότητα εύρεσης της περιστροφής και της κλίμακας με τη χρήση των ανιχνευτών σημείων ενδιαφέροντος που υλοποιήθηκαν στο Μέρος 2 και τους παραπάνω τοπικούς περιγραφητές. Συγχεκριμένα για το πέραμα αυτό έχουν χρησιμοποιηθεί 3 εικόνες, τις οποίες έχουμε παραμορφώσει περιστρέφοντάς τες (5 περιστροφές:  $-20^\circ, -10^\circ, 0^\circ, 10^\circ, 20^\circ$ ) και αλλάζοντας το μέγεθός τους (4 κλίμακες: 0.6, 0.8, 1.0, 1.2). Συνεπώς έχουμε συνολικά 20 παραμορφώσεις (μετασχηματισμοί ομοιότητας) και χρησιμοποιώντας μια από τις εικόνες ως εικόνα αναφοράς, θα προσπαθήσουμε να αντιστοιχίσουμε τους τοπικούς περιγραφητές της με αυτούς των υπόλοιπων παραμορφωμένων εικόνων. Η διαδικασία αυτή ονομάζεται ταίριασμα (matching).

Στη συνέχεια, ακολουθεί ένας πίνακας με το μέσο σφάλμα εκτίμησης κλίμακας και γωνίας περιστροφής για κάθε εικόνας και για κάθε συνδυασμό detector-descriptor.

Detector	Descriptor	Scale Error			Theta Error		
		Image 1	Image 2	Image 3	Image 1	Image 2	Image 3
Harris	SURF	0.0438	0.0319	0.0566	1.5907	1.5338	4.1411
Multiscale Harris	SURF	0.0014	0.0067	0.0085	0.0759	0.5857	0.3817
Blobs	SURF	0.1069	0.0226	0.1605	3.2753	0.5772	4.5209
Multiscale Blobs	SURF	0.0013	0.0043	0.0033	0.1411	0.2069	0.2406
Multiscale Box Filters	SURF	0.0074	0.0296	0.0087	0.3874	1.3637	0.4057
Harris	HOG	0.2042	0.2618	0.2955	24.9528	18.5592	19.6358
Multiscale Harris	HOG	0.2299	0.2020	0.2420	17.5603	24.2281	14.5359
Blobs	HOG	0.2882	0.3340	0.2852	28.5189	30.0686	23.9066
Multiscale Blobs	HOG	0.1670	0.1901	0.1393	22.0846	13.4198	13.3652
Multiscale Box Filters	HOG	0.2350	0.3510	0.1999	22.2157	30.7645	13.0144

### Ανάλυση ως προς τον περιγραφητή:

Τα SURF βασίζονται στην προσέγγιση των σημείων ενδιαφέροντος και βρίσκουν τις κατευθύνσεις για την περιοχή γύρω από αυτά για την εξαγωγή περιστροφικά ανεξάρτητων περιγραφητών. Δεδομένου ότι οι περιοχές γύρω από τα σημεία ενδιαφέροντος είναι ανάλογες της κλίμακας, είναι λογικό να μην αντιμετωπιστεί πρόβλημα στο matching με διαφορετική κλίμακα. Η εξαγωγή περιστροφικά ανεξάρτητων περιγραφητών μας δίνει την ανεκτικότητα στην περιστροφή και την διατήρηση των γεωμετρικών ιδιοτήτων.

Αντίθετα, τα Ιστογράμματα Προσανατολισμένης Κλίσης (HOGs) αποτελούν ένα σύνολο χαρακτηριστικών για την περιγραφή της δομής του σχήματος σε μια εικόνα και έχουν χρησιμοποιηθεί με μεγάλη επιτυχία σε tasks όπως το classification. Ουσιαστικά, δίνουν μια πυκνή επικαλυπτόμενη περιγραφή των περιοχών της εικόνας και υπολογίζονται σε ένα πυκνό πλέγμα ομοιόμορφα κατανευμημένων κελιών. Αυτός ο τρόπος όμως αποδειχνύεται μη ανεκτικός σε αλλαγή της κλίμακας και σε περιστροφή καθώς γίνεται σε όλα τα κελιά.

Συγκρίνοντας τα δύο, οι αδυναμίες των HOG στο matching και τα πλεονεκτήματα των SURF αντίστοιχα αποτυπώνονται στα αποτελέσματα που παρατέθηκαν πιο πάνω δίνοντας σημαντικό προβάδισμα στους τοπικούς περιγραφητές SURF στο task αυτό.

## 3.2 Κατηγοριοποίηση Εικόνων

Στο μέρος αυτό θα αξιολογηθεί η επίδοση και η καταλληλότητα των διαφόρων ανιχνευτών και περιγραφητών σε ένα τυπικό πρόβλημα κατηγοριοποίησης εικόνων. Συγκεχριμένα, θα χρησιμοποιηθεί ένα σύνολο εικόνων από τη βάση Pascal VOC2005. Κάθε εικόνα ανήκει σε μια από τις τρεις ακόλουθες κλάσεις: αυτοκίνητο, άνθρωπος και ποδήλατο. Σκοπός είναι η κατηγοριοποίηση της κάθε εικόνας στη σωστή κλάση χρησιμοποιώντας σαν χαρακτηριστικά αναγνώρισης τους περιγραφητές που θα υπολογίσουμε.

**Η διαδικασία της κατηγοριοποίησης αποτελείται από τα παρακάτω στάδια:**

1. Εξαγωγή των σημείων ενδιαφέροντος για κάθε εικόνα με την χρήση ενός από τους detectors του 2ου μέρους.
2. Εξαγωγή των τοπικών περιγραφητών για κάθε εικόνα (SURF ή HOG) χρησιμοποιώντας τα σημεία ενδιαφέροντος. Η συνάρτηση FeatureExtraction.p αναλαμβάνει τα δύο πρώτα βήματα.
3. Διαχωρισμός των εικόνων σε train και test set μέσω της συνάρτησης createTrainTest.p .
4. Δημιουργία της Bag of Visual Words αναπαράστασης και για τα δύο set. Για να εφαρμόσουμε αλγορίθμους κατηγοριοποίησης, πρέπει να έχουμε για κάθε εικόνα ένα διάνυσμα σταθερών χαρακτηριστικών και, προφανώς, η συνένωση των τοπικών περιγραφητών της δεν μας το προσφέρει αυτό. Έτσι, κατσκευάζουμε BoVW αναπαραστάσεις, οι οποίες υλοποιούνται στην συνάρτηση BagOfWords.p . Μία λεπτομερής περιγραφή της

διαδικασίας αυτής βρίσκεται παρακάτω, όπου υλοποιήσαμε μόνοι μας ένα BoVW μοντέλο στη συνάρτηση OurBagOfWords.m .

5. Εκπαίδευση ενός SVM classifier στα δεδομένα εκπαίδευσης.
6. Αξιολόγηση του μοντέλου στα δεδομένα αξιολόγησης. Τα δύο τελευταία βήματα βρίσκονται στην συνάρτηση svm.p .

### Κατασκευή αναπαράστασης Bag of Visual Words:

Στο σημείο αυτό έχουμε ως είσοδο για κάθε εικόνα τους τοπικούς περιγραφητές γύρω από τα σημεία ενδιαφέροντός της. Στόχος μας είναι να καταλήξουμε με ένα συνολικό (global) διάνυσμα χαρακτηριστικών με την ίδια διάσταση για κάθε εικόνα, ώστε να εφαρμόσουμε αλγόριθμους κατηγοριοποίησης με βάση αυτό. Η διαδικασία αποτελείται από τα παρακάτω βήματα ενώ ο αντίστοιχος κώδικας βρίσκεται στο αρχείο **OurBagOfWords.m** .

- **Δημιουργία οπτικού λεξικού:** Αρχικά, συνενώνουμε όλους τους περιγραφητές όλων των εικόνων εκπαίδευσης σε ένα ενιαίο διάνυσμα. Στη συνέχεια, εφαρμόζουμε τον αλγόριθμο kmeans για ένα τυχαίο υποσύνολο του παραπάνω διανύσματος ορίζοντας και το πλήθος των κέντρων που επιθυμούμε (500 - 2000).
- **Κατασκευή ιστογράμματος για κάθε εικόνα:** Για κάθε εικόνα του test και του train set υπολογίζουμε την ελάχιστη ευκλείδια απόσταση του κάθε τοπικού περιγραφητή της από τα κέντρα που υπολογίστηκαν προηγουμένως. Στη συνέχεια, κατασκευάζουμε το ιστόγραμμά της που προκύπτει από τη συχνότητα εμφάνισης των οπτικών λέξεων του λεξικού με βάση την παραπάνω απόσταση κανονικοποιημένο με την L2 νόρμα (συνάρτηση image\_histogram.m).

Ακολουθεί ένας πίνακας που συνοψίζει το accuracy που έχουμε για κάθε συνδυασμό detector-descriptor:

Detector	Descriptor	Average Classification Accuracy (%)
Multiscale Harris	SURF	63.586207
Multiscale Blobs	SURF	67.310345
Multiscale Box Filters	SURF	60.827586
Multiscale Harris	HOG	67.862069
Multiscale Blobs	HOG	<b>69.241379</b>
Multiscale Box Filters	HOG	61.103448

### Σχολιασμός αποτελεσμάτων:

Παρατηρούμε ότι δεν έχουμε πολύ μεγάλες αποκλίσεις μεταξύ των διάφορων μοντέλων και σε όλες τις περιπτώσεις το accuracy κυμαίνεται μεταξύ 60-70%, ένα ποσοστό σχετικά καλό.

Από την μεριά των detectors, τα συμπεράσματα στα οποία καταλήξαμε στο 2ο μέρος επιβεβαιώνονται για άλλη μία φορά, καθώς ο πολυχλιμακωτός Blobs detector πετυχαίνει τα καλύτερα αποτελέσματα. Αυτό είναι λογικό, γιατί από την στιγμή που τα σημεία ενδιαφέροντος που επιλέχθηκαν είναι πιο αντιπροσωπευτικά, η απόδοση ολόκληρης της κατηγοριοποίησης θα βελτιωθεί. Σημαντικό, επίσης, είναι να αναφερθεί ότι η υλοποίηση με τα Box Filters, αν και προσεγγιστική, πετυχαίνει ικανοποιητικά αποτελέσματα και θα ήταν πιο κατάλληλη σε περιπτώσεις όπου θέλαμε να κάνουμε classification με μεγάλη ταχύτητα, όπως για παράδειγμα σε βίντεο.

Όσον αφορά τους descriptors, σε όλες τις περιπτώσεις, ο HOG έχει ένα προβάδισμα έναντι των SURF. Αυτό οφείλεται, κυρίως, στα εξής:

- Τα HOGs παρέχουν μια πυκνή επικαλυπτόμενη περιγραφή των περιοχών μια εικόνας και υπολογίζονται σε ένα πυκνό πλέγμα ομοιόμορφα κατανεμημένων κελιών. Αυτό τα κάνει κατάλληλα για classification και object detection, σε αντίθεση με τα SURF που υπολογίζουν το gradient μόνο γύρω από τα σημεία ενδιαφέροντος

και καταλήγουν να κρατάνε πιο τοπική πληροφορία και να χρησιμοποιούνται με μεγαλύτερη επιτυχία στο matching (όπως είδαμε στο 3-1).

- Το αρνητικό των HOG features, όπως αναφέρθηκε και παραπάνω, είναι το γεγονός ότι δεν είναι αμετάβλητα στην περιστροφή. Στο dataset μας, ωστόσο, όπου θέλουμε να αναγνωρίσουμε ανθρώπους, αυτοκίνητα και ποδήλατα δεν υπάρχει, προφανώς, κάποια εικόνα σε περιστροφή, με αποτέλεσμα να μην χρειάζεται αυτή η ιδιότητα που του λείπει.

## Βιβλιογραφία

- [1] P. Maragos, *Σημειώσεις Όρασης Υπολογιστών*. 2015.
- [2] Open-Cv, “Introduction to surf (speeded-up robust features).” [https://opencv-python-tutroals.readthedocs.io/en/latest/py\\_tutorials/py\\_feature2d/py\\_surf\\_intro/py\\_surf\\_intro.html?fbclid=IwAR0WsroWPVmHLB6hfaFoSM8ND7YkEPblc66NIgvy\\_UVnrce992JKNnZhafk](https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_surf_intro/py_surf_intro.html?fbclid=IwAR0WsroWPVmHLB6hfaFoSM8ND7YkEPblc66NIgvy_UVnrce992JKNnZhafk), 2013.
- [3] Intel, “Histogram of oriented gradients (hog) descriptor.” [https://software.intel.com/en-us/ipp-dev-reference-histogram-of-oriented-gradients-hog-descriptor?fbclid=IwAR2M70u7MKqNZas0u57Q\\_1hYcF2Zv5\\_iPTx612z0R20zXSyMjpo5iAd2Kri](https://software.intel.com/en-us/ipp-dev-reference-histogram-of-oriented-gradients-hog-descriptor?fbclid=IwAR2M70u7MKqNZas0u57Q_1hYcF2Zv5_iPTx612z0R20zXSyMjpo5iAd2Kri), 2018.
- [4] S. P. Ebrahim Karami and M. Shehata, “Image matching using sift, surf, brief and orb: Performance comparison for distorted images,”
- [5] H. of Oriented Gradients for Human Detection, “Navneet dalal and bill triggs,”