



## ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ  
ΥΠΟΛΟΓΙΣΤΩΝ

---

### ΠΡΩΤΗ ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΝΑΦΟΡΑ ΟΡΑΣΗ ΥΠΟΛΟΓΙΣΤΩΝ

---

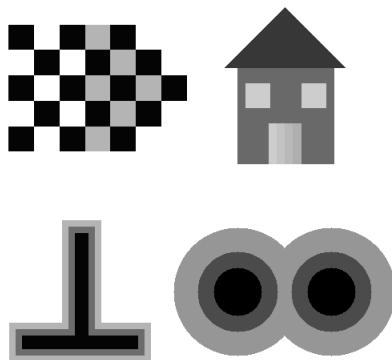
Γεώργιος Καούκης  
03119006  
geoka01@gmail.com  
Αναστασία Χριστίνα Λίβα  
03119029  
anachriliva@gmail.com

## Ανίχνευση Ακμών σε Γκρίζες Εικόνες

Σκοπός του πρώτου μέρους της εργαστηριακής ασκησης είναι η υλοποίηση αλγορίθμων ανίχνευσης ακμών βάσει των δοθέντων βημάτων.

### Δημιουργία Εικόνων Εισόδου

Φορτώνουμε την εικόνα που δίνεται στο notebook μας.



Σχήμα 1: Εικόνα Edgetest

Έπειτα προσθέτουμε λευκό gaussian θόρυβο μέσης τιμής 0 και τυπική απόκλιση έτσι ώστε

1. PSRN = 20 dB
2. PSNR = 10 dB

Όπου:

$$PSNR = 20 \log_{10} \frac{I_{max} - I_{min}}{\sigma_n}$$

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

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

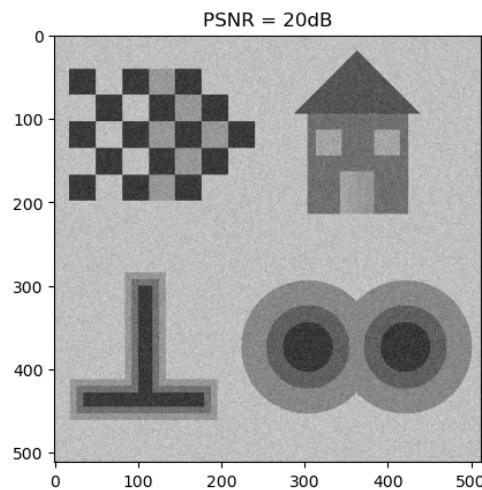
Βάσει αυτού προκύπτει:

$$\sigma_n = \frac{I_{max} - I_{min}}{10^{\frac{PSNR}{20}}}$$

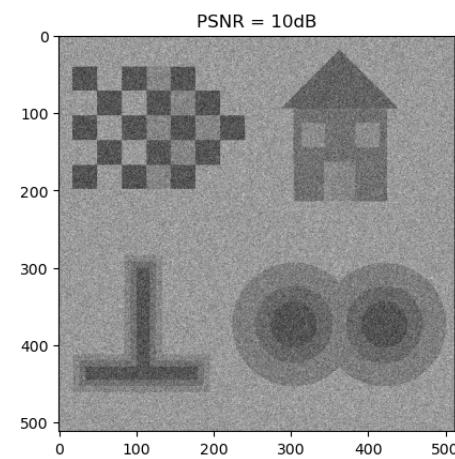
Οπότε:

1. PSNR = 20 dB:  $\sigma_n = 0.1$
2. PSNR = 10 dB:  $\sigma_n = 0.316$

Μετά από εφαρμογή του θορύβου προκύπτουν οι ακόλουθες εικόνες:



Σχήμα 2: Edgetest: PSNR = 20 dB



Σχήμα 3: Edgetest: PSNR = 10 dB

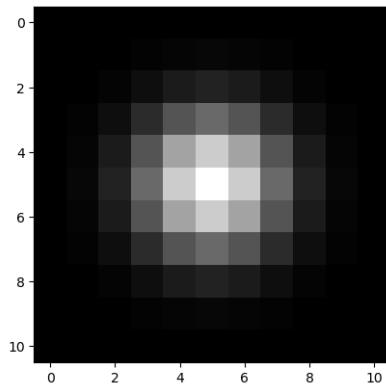
## Υλοποίηση Αλγορίθμων Ανίχνευσης Ακμών

Βάσει των οδηγιών που δίνονται στην εκφώνηση κατασκευάζουμε τη συνάρτηση edgedetect. Γνωρίζουμε πως εάν εξωμαλύνουμε μία εικόνα με υόρυβο μέσω κατάλληλου φίλτρου οι ακμές της προκύπτουν από τα σημεία εκείνα με μεγάλη κλίση όπου μηδενίζεται η Λαπλασιανή.

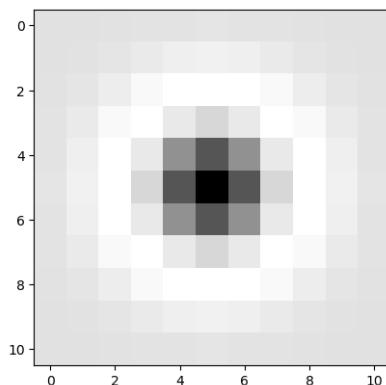
Δημιουργούμε λοιπόν δύο φίλτρα με αποχρίσεις:

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

$$L_\sigma(x, y) = \nabla^2 G_\sigma$$



Σχήμα 4: Gauss kernel



Σχήμα 5: Laplacian of Gaussian

Ο πυρήνας των δύο φίλτρων έχει  $n \times n$  διαστάσεις, όπου  $n = 2ceil(3\sigma) + 1$ , ενώ η παράμετρος σ είναι κοινή για τα δύο φίλτρα.

**Σημείωση 1:** Η συνάρτηση `ceil` ανήκει στη βιβλιοθήκη `numpy` της Python και έξιδος της είναι ο αμέσως μεγαλύερος ακέραιος του ορίσματός της.

**Σημείωση 2:** Η LoG δημιουργήθηκε και με τους δύο τρόπους που αναφέρονται στην εκφώνηση.

Για την προσέγγιση της Λαπλασιανής της εικόνας έχουμε δύο εναλλακτικές:

- **Γραμμική Μέθοδος:**  $L_1 = (\nabla^2 G_\sigma) * I = h * I$ . Χρησιμοποιούμε την κανονική εικόνα διότι όταν πραγματοποιήσουμε συνέλιξη της εικόνας  $I$  με την απόχριση του Laplacian of Gaussian φίλτρου που ισοδυναμεί με τη σχέση  $L_1 = \nabla(G_\sigma * I) = \nabla^2(G_\sigma * I) = h_{LoG} * I$
- **Μη Γραμμική Μέθοδος:** Πραγματοποιούμε Dilation και Erosion της εξομαλυμένης εικόνας με το  $B$ :  $L_2 = I_\sigma \oplus B + I_\sigma \ominus B - 2I_\sigma$

$$B = \begin{array}{|c|c|c|} \hline & \bullet & \\ \hline \bullet & \bullet & \bullet \\ \hline & \bullet & \\ \hline \end{array}$$

Στη συνέχεια εντοπίζουμε τα σημεία μηδενισμού της Laplacian σύμφωνα με τον αλγόριθμο της εκφώνησης. Ο αλγόριθμος περιγράφεται ως εξής:

1. Δημιουργία Δυαδικής Εικόνας Προσήμου  $X$  για την  $L$ , δηλαδή  $X = (L \geq 0)$
2. Εύρεση του περιγράμματος της  $X$  (συμετρικό ως προς  $X$  και  $X^c$ ):  $Y = (X \oplus B) - (X \ominus B) \approx \partial X$

Τα σημεία αυτά αντιστοιχούν σε πιθανές ακμές.

Τώρα πρέπει να ανίχνεύσουμε "ακμές" που δεν αντιστοιχούν σε πραγματικές ακμές αλλα προέκυψαν λόγω του υορύβου που προσθέσαμε. Για να επιλέξουμε ποιες ακμές όταν κρατήσουμε ορίζουμε την παράμετρο κατωφλίου  $\theta_{edge}$ . Εάν δεν περαστεί το κατώφλι αυτό για κάποιο  $Y$ , σημαίνει ότι στο  $Y$  αυτό η εικόνα έχει μικρή κλίση, άρα δεν υπάρχει ακμή εκεί. Σημειώνεται πως τα zero-crossings έχουν τιμή  $Y = 1$ .

Μαθηματικά ό,τι ειπώθηκε προηγουμένως περιγράφεται βάσει των επόμενων σχέσεων:

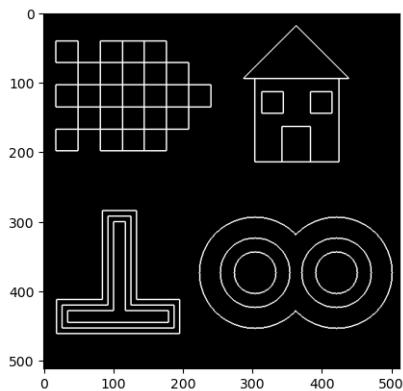
$$Y_{i,j} = 1 \text{ και } \|\nabla I_\sigma(i, j)\| > \theta_{edge} \cdot \max_{x,y} \|\nabla I_\sigma\|$$

### Αξιολόγηση των Αποτελεσμάτων Ανίχνευσης Ακμών

Για να αξιολογήσουμε τα αποτελέσματα της συνάρτησης που υλοποιήσαμε νωρίτερα βρίσκουμε την πραγματική δυαδική εικόνα ακμών βάσει του ακόλουθου τύπου:

$$M = (I_0 \oplus B) - (I_0 \ominus B)$$

Χρησιμοποιούμε επίσης ένα μικρό κατώφλι  $\vartheta_{real\ edge} = 0.1$



Σχήμα 6: Binary Image

Έχοντας βρει τις πραγματικές ακμές μπορούμε να φτιάξουμε συνάρτηση που επιστρέφει το ποσοστό επιτυχίας του αλγορίθμου σύμφωνα με τον ακόλουθο τύπο:

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

όπου  $Pr(D|T)$  είναι το ποσοστό των ανιχνευθεισών ακμών που είναι αληθινές και  $Pr(T|D)$  το ποσοστό των αληθινών ακμών που ανιχνεύθησαν. Αν τα  $D$  και  $T$  ειδωθούν σαν διακριτά σύνολα (με στοιχεία τα πιξελς στα οποία η δυαδική εικόνα έχει τιμή 1, τότε ισχύει:  $Pr(T|D) = \frac{|D \cap T|}{\text{card}(T)}$ , όπου το  $\text{card}(.)$  δίνει τον αριθμό των στοιχείων ενός συνόλου).

Βάσει του κριτηρίου αυτού προκύπτουν:

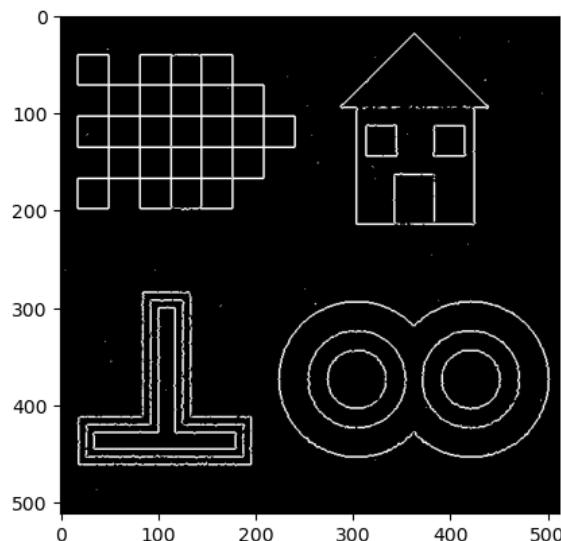
- Precision: 1.0
- Recall: 0.999848
- Quality criterion C: 0.99992

Επομένως καθίσταται προφανές ότι στην εικόνα χωρίς θόρυβο η συνάρτησή μας χάνει σχεδόν μηδενικό αριθμό pixels.

Για τις θορυβώδεις εικόνες τρέχουμε ένα for loop μέσω του οποίου εξετάζουμε τα αποτελέσματα για διάφορες τιμές των  $\sigma$  και  $\theta$  οπότε και καταλήγουμε στα εξής συμπεράσματα:

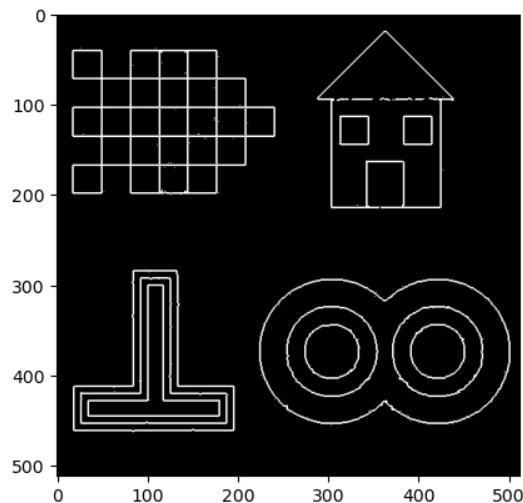
**Εικόνα με θόρυβο PSNR = 20 dB:**

- Με τη γραμμική προσέγγιση της Λαπλασιανής έχουμε  $C \approx 0.9449$  για  $\sigma = 1$  και  $\theta = 0.18$  και η εικόνα είναι η εξής:



Σχήμα 7: Linear PSNR = 20 dB

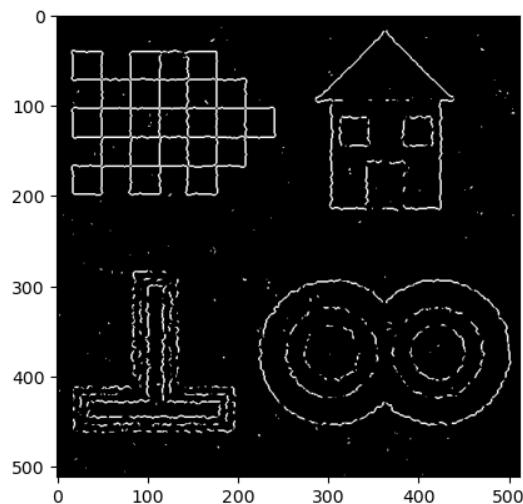
- Με τη μη γραμμική προσέγγιση της Λαπλασιανής έχουμε  $C \approx 0.9716$  για  $\sigma = 1.8$  και  $\theta = 0.15$  και η εικόνα είναι η εξής:



Σχήμα 8: Non-Linear PSNR = 20 dB

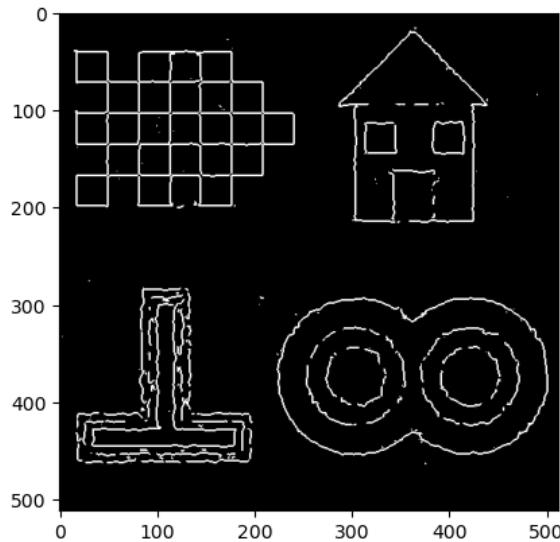
**Εικόνα με θόρυβο PSNR = 10 dB:**

- Με τη γραμμική προσέγγιση της Λαπλασιανής έχουμε  $C \approx 0.674$  για  $\sigma = 1.8$  και  $\theta = 0.25$  και η εικόνα είναι η εξής:



Σχήμα 9: Linear PSNR = 10 dB

- Με τη μη γραμμική προσέγγιση της Λαπλασιανής έχουμε  $C \approx 0.7679$  για  $\sigma = 2.2$  και  $\theta = 0.2$  και η εικόνα είναι η εξής:



Σχήμα 10: Non-Linear PSNR = 10 dB

**Παρατηρήσεις:** Είναι εμφανές πως σε κάθε περίπτωση η μη γραμμική μέθοδος δίνει καλύτερα αποτελέσματα από τη γραμμική. Παρατηρούμε ότι για την τυπική απόκλιση  $\sigma$ , σε εικόνες με λιγότερο θόρυβο πάρινουμε καλύτερα αποτελέσματα για μικρότερες τιμές, καθώς μεγαλύτερες τιμές συνεπάγονται περισσότερη ομαλοποίηση και άρα απώλεια πληροφορίας. Οι μεγαλύτερες τιμές απαιτούνται στην εικόνα με τον περισσότερο θόρυβο, για να τον εξομαλύνουν. Σχετικά με τη δεύτερη μας παράμετρο, την τιμή κατωφλίου, παρατηρούμε ότι η αύξησή της είναι πιο χρήσιμη σε εικόνες με θόρυβο, ώστε να μην επισημάνουμε λανθασμένα σημεία με θόρυβο ως ακμές, Ωστόσο, υπέρμετρη αύξησή της οδηγεί σε απόρριψη περισσότερων zero crossings, με κίνδυνο να απορρίψουμε κάποιες πραγματικές ακμές της εικόνας. Για τον λόγο αυτό, όπως παρατηρούμε, τα βέλτιστα αποτελέσματα βρίσκονται στην τιμή περιοχή  $\theta = 0.2 \pm 0.05$ .

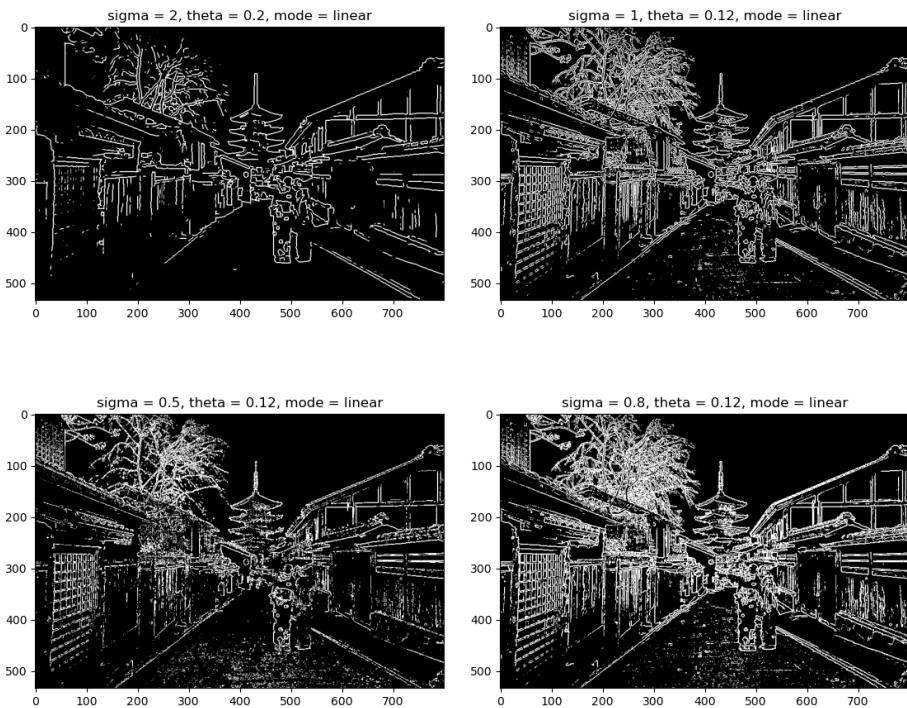
## Εφαρμογή των Αλγορίθμων Ανίχνευσης Ακμών σε Πραγματικές Εικόνες

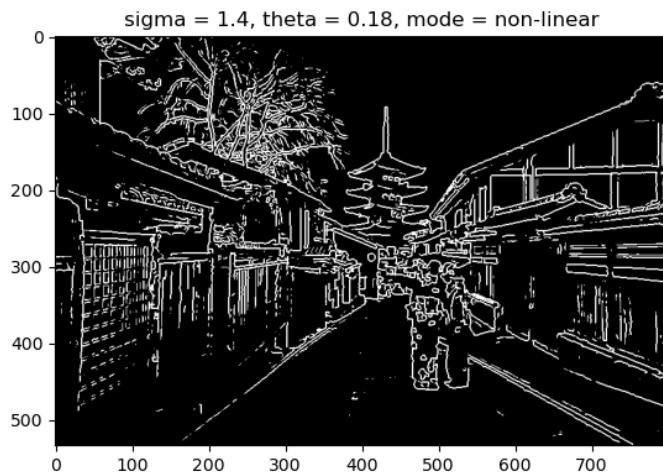
Τώρα θα εφαρμόσουμε τον αλγόριθμο που φτιάξαμε νωρίτερα σε μια πραγματική εικόνα χωρίς να προσθέσουμε θόρυβο, αφού εφόσον πρόκειται για πραγματική εικόνα ο όγκος πληροφορίας που δεχόμαστε είναι αρκετά μεγάλος.



Σχήμα 11: Αρχική Εικόνα

Δοκιμάζοντας δαφορετικές παραμέτρους βλέπουμε πως η συνάρτηση εντοπίζει ικανοποιητικά τις ακμές τις εικόνας, οπότε και επιβεβαιώνονται οι παρατηρήσεις μας. Παρακάτω, παρατίθενται οι εικόνες των ακμών που εντοπίστηκαν για τις βέλτιστες παραμέτρους:





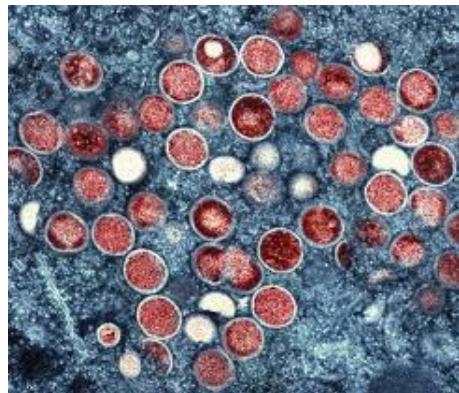
Όπως βλέπουμε, τα βέλτιστα αποτελέσματα προκύπτουν για τη μη γραμμική προσέγγιση με μικρά σ και θ. Καθώς η εικόνα δεν έχει πρόσθετο όγκο, η εξομάλυνση με μεγαλύτερα σ ή η απόρριψη ακμών με μεγαλύτερα θ οδηγούν σε απώλεια λεπτομέρειας.

## Ανίχνευση Σημείων Ενδιαφέροντος

Στόχος του δεύτερου μέρους της εργαστηριακής άσκησης είναι ο εντοπισμός σημείων ενδιαφέροντος, χρησιμοποιώντας διαφορετικά είδη ανιχνευτών.

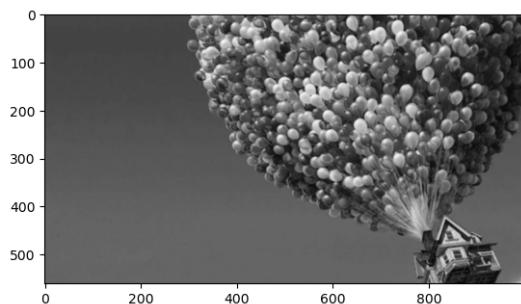


Σχήμα 12: Up

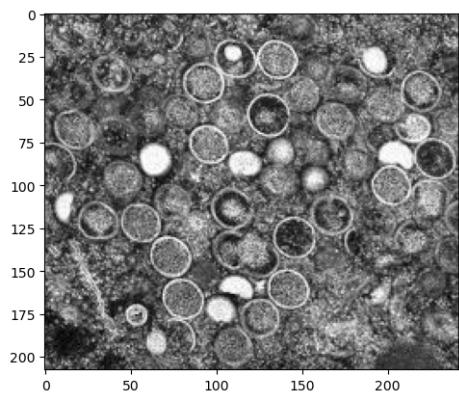


Σχήμα 13: cells

Για να χρησιμοποιήσουμε τον αλγόριθμο που δημιουργήσαμε νωρίτερα ως χρησιμοποιήσουμε την απόσύμαυρη εκδοχή των εικόνων αυτών.



Σχήμα 14: Up in grayscale



Σχήμα 15: cells in grayscale

## Ανίχνευση Γωνιών

Μία απλή κατηγορία σημείων ενδιαφέροντος είναι τα σημεία στα οποία βρίσκονται οι γωνίες της εικόνας. Η μέθοδος Harris-Stephens ανίχνευσης που χρησιμοποιείται στο συγκεκριμένο ερώτημα βασίζεται στην ιδέα πως σε ένα παράθυρο μίας εικόνας υπάρχει γωνία, όταν υπάρχει σημαντική διαφορά στη φωτεινότητα, καθώς αυτό μετακινείται.

Ο αλγόριθμος εντοπίζει τις διαφορές αυτές σχηματίζοντας έναν πίνακα  $M$  με τις μερικές παραγώγους και υπολογίζοντας τις ιδιοτιμές του. Εάν και οι δύο ιδιοτιμές είναι αρκετά μεγάλες γνωρίζουμε πως η φωτεινότητα μεγαλώνει σε όλες τις κατευθύνσεις, συνεπώς υπάρχει γωνία.

Για την υλοποίηση του αλγορίθμου λοιπόν υπολογίζουμε τα στοιχεία  $J_1$ ,  $J_2$ ,  $J_3$  του δομικού τανυστή  $J$  σε κάθε pixel ( $x, y$ ) της εικόνας, σύμφωνα με τις ακόλουθες σχέσεις:

$$J_1(x, y) = G_\rho * \left( \frac{\partial I_\sigma}{\partial x} \cdot \frac{\partial I_\sigma}{\partial x} \right)(x, y)$$

$$J_2(x, y) = G_\rho * \left( \frac{\partial I_\sigma}{\partial x} \cdot \frac{\partial I_\sigma}{\partial y} \right)(x, y)$$

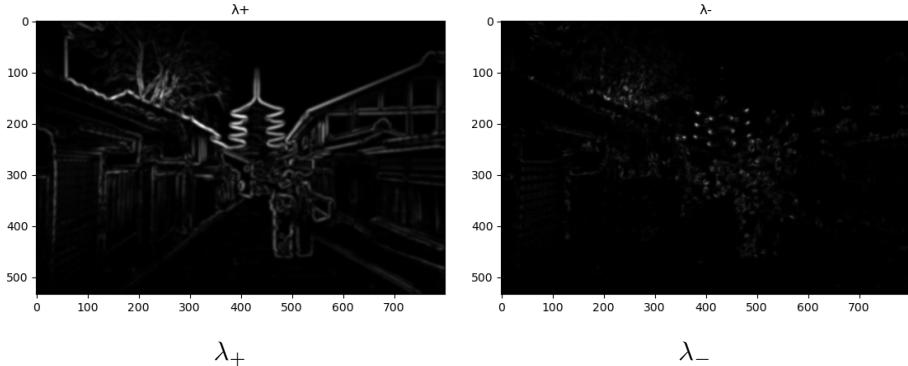
$$J_3(x, y) = G_\rho * \left( \frac{\partial I_\sigma}{\partial y} \cdot \frac{\partial I_\sigma}{\partial y} \right)(x, y)$$

όπου  $I_\sigma = G_\sigma * I$  και  $G_\sigma$ ,  $G_\rho$  δισδιάστατοι Gaussian πυρήνες ομαλοποίησης με τυπικές αποκλίσεις  $\sigma$

Συνεχίζουμε υπολογίζοντας τις ιδιοτιμές  $\lambda_+$  και  $\lambda_-$  για τον τανυστή  $J$  σύμφωνα με την ακόλουθη σχέση:

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

Απεικονίζοντας τις δύο ιδιοτιμές ως ασπρόμαυρες εικόνες προκύπτει:



Σχήμα 16: Απεικόνιση των Ιδιοτιμών

Είναι εμφανές πως η  $\lambda_+$  ιδιοτιμή έχει υψηλή τιμή και για τις ακμές και για τις γωνίες, ενώ η  $\lambda_-$  κυρίως στις γωνίες.

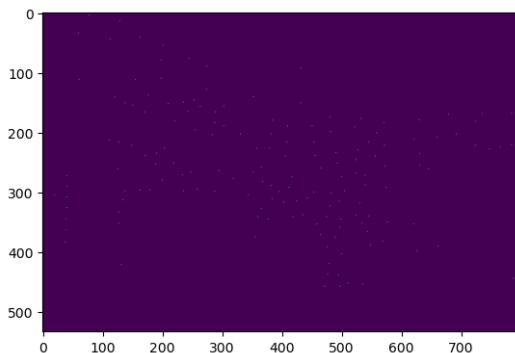
Στη συνέχεια υπολογίζουμε το κριτήριο γωνιότητας  $R$  βάσει της ακόλουθης σχέσης:

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

όπου  $k$  μικρή, θετική σταθερά γωνιότητας.

Τελικά επιλέγω ως γωνίες τα pixels (x,y) που:

- Είναι μέγιστα του  $R$  εντός τετραγωνικών παραθύρων που τα περιβάλλουν το μέγεθος των οποίων εξαρτάται από την κλίμακα σ
- Αντιστοιχούν σε τιμή του  $R$  μεγαλύτερη από ένα ποσοστό του ολικού μεγίστου του  $P$ , δηλαδή  $R(x, y) > \theta_{corn} \cdot R_{max}$ , όπου  $\theta_{corn}$  ένα κατάλληλα επιλεγμένο κατώφλι.



Σχήμα 17: Pixels που επιλέχθηκαν βάσει του κριτηρίου γωνιότητας

**Kyoto:** Τρέχοντας τη συνάρτηση που φτιάξαμε λοιπόν με παραμέτρους:

- $\sigma = 2$
- $\theta_{corn} = 0.005$
- $k = 0.05$
- $\rho = 2$

λαμβάνουμε την εξής εικόνα:



Σχήμα 18: Kyoto Interest Points in grayscale



Σχήμα 19: Kyoto Interest Points

Πειράζοντας την παράμετρο σ' έχουμε:



$\sigma = 1$



$\sigma = 2$



$\sigma = 4$



$\sigma = 6$



$\sigma = 10$

Παρατηρούμε, λοιπόν, ότι το σ επηρεάζει την κλίμακα στην οποία φάχνουμε 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.

Πειράζοντας την παράμετρο  $\rho$  έχουμε:



$\rho = 0.5$



$\rho = 2$



$\rho = 3.5$

$\rho = 5$

$\rho = 7$

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



$k = 0.008$

$k = 0.02$



$k = 0.07$

$k = 0.1$

$k = 0.21$

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.



$\theta = 0.01$

$\theta = 0.05$



$\theta = 0.1$

$\theta = 0.15$

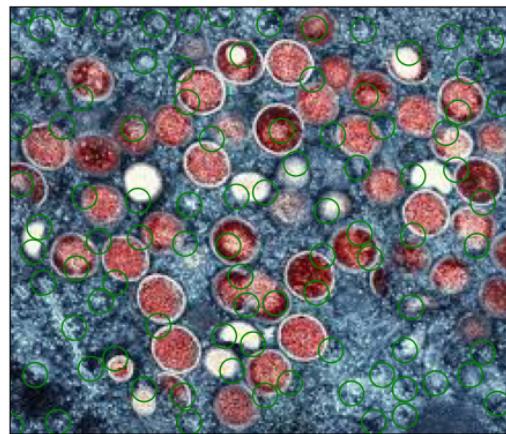
$\theta = 0.2$

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

**Cells:** Τρέχοντας τη συνάρτηση που φτιάζαμε λοιπόν με παραμέτρους:

- $\sigma = 1.5$
- $\theta_{corn} = 0.01$
- $k = 0.05$
- $\rho = 2$

λαμβάνουμε την εξής εικόνα:



Σχήμα 20: Cells Interest Points

**Up:** Τρέχοντας τη συνάρτηση που φτιάζαμε λοιπόν με παραμέτρους:

- $\sigma = 1.5$
  - $\theta_{corn} = 0.01$
  - $k = 0.05$
  - $\rho = 2$
- λαμβάνουμε την εξής εικόνα:



Σχήμα 21: Up Interest Points

## Πολυκλιμακωτή Ανίχνευση Γωνιών

Πριν η ανίχνευση των γωνιών γινόταν σε μία μόνο κλίμακα η οποία οριζόταν από τις παραμέτρους  $\sigma$  και  $\rho$  και, έτσι, χρειαζόταν κάθε φορά να ορίσουμε τις τιμές τους. Τώρα 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:

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

Οι κλίμακες ορίζονται ως εξής:

$$\sigma_0, \sigma_1, \dots, \sigma_{N-1} = s_0\sigma_0, s_1\sigma_0, \dots, s_{N-1}\sigma_0$$

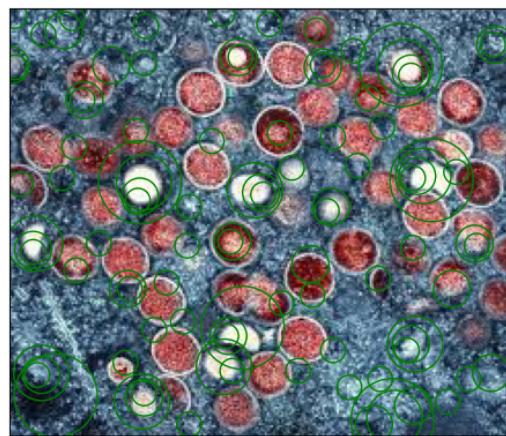
$$\rho_0, \rho_1, \dots, \rho_{N-1} = s_0\rho_0, s_1\rho_0, \dots, s_{N-1}\rho_0$$

Τρέχοντας τη συνάρτηση που φτιάξαμε λοιπόν με παραμέτρους:

- $N = 4$
- $\sigma = 2$
- $\theta_{corn} = 0.03$
- $k = 0.05$
- $\rho = 2$
- $s_0 = 2$

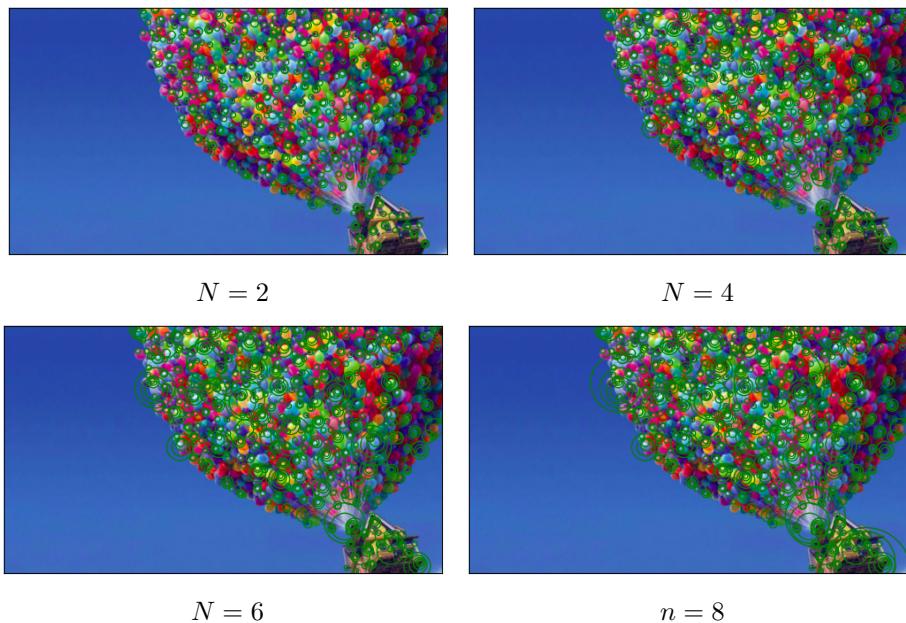


Σχήμα 22: Up Harris Detector



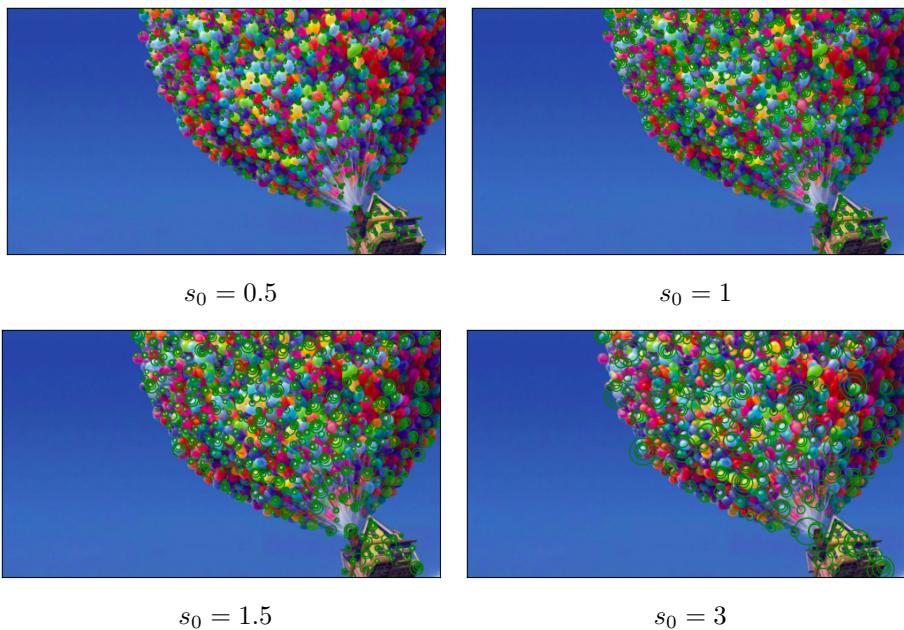
Σχήμα 23: Cells Harris Detector

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.



Τυάρχουν πολλά πλεονεκτήματα στη χρήση ενός ανιχνευτή Harris πολλαπλής κλίμακας, μερικά από τα οποία είναι τα εξής:

- Βελτιωμένη στιβαρότητα στις παραλλαγές κλίμακας, καθώς ο ανιχνευτής πολλαπλής κλίμακας μας επιτρέπει να καταγράψουμε γωνίες σε πολλαπλές κλίμακες, καθιστώντας τον πιο ανθεκτικό στις παραλλαγές κλίμακας στην εικόνα.
- Πιο ακριβής εντοπισμός, διότι πολλαπλών κλιμάκων συμβάλλει επίσης στη βελτίωση της ακρίβειας του εντοπισμού γωνιών, επιτρέποντας στον ανιχνευτή να βελτιώσει τη θέση των γωνιών σε διαφορετικές κλίμακες.
- Μειωμένη ευαισθησία στο ύδρυβο, καθώς εφαρμόζοντας τον ανιχνευτή σε προοδευτικά θολές εικόνες, η προσέγγιση πολλαπλών κλιμάκων μπορεί να μειώσει την ευαισθησία του ανιχνευτή στο ύδρυβο στην εικόνα.



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.

## 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. Ουσιαστικά εξετάζουμε τις δεύτερες παραγώγους της εικόνας ως προς τους ορθογώνιους άξονες. Με το Dilatation βλέπουμε τα γειτονικά pixels σε ένα παράθυρο δίσκου και τα αθροίζουμε. Εάν το άθροισμα είναι μεγαλύτερο από κάποια ορισμένη τιμή θεωρούμε πως είναι blob για το σ αυτο.

Για την εύρεση τέτοιων περιοχών, σε αντιστοιχία με το χριτήριο γωνιότητας της μεθόδου Harris, γίνεται χρήση των μερικών παραγώγων δεύτερης τάξης της εικόνας και συγκεκριμένα η ορίζουσα του πίνακα Hessian:

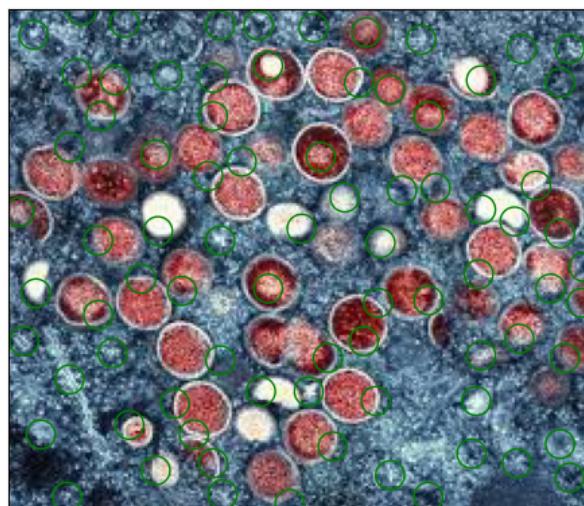
$$H = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix}$$

Για τη μέθοδο αυτή χριτήριο γωνιότητας αποτελεί η ορίζουσα του πίνακα  $H(x, y)$ .

Τρέχοντας τη συνάρτηση λοιπόν για  $\sigma = 2.5$  και  $\theta_{corn} = 0.01$  έχουμε:



Σχήμα 24: Blobs Detect



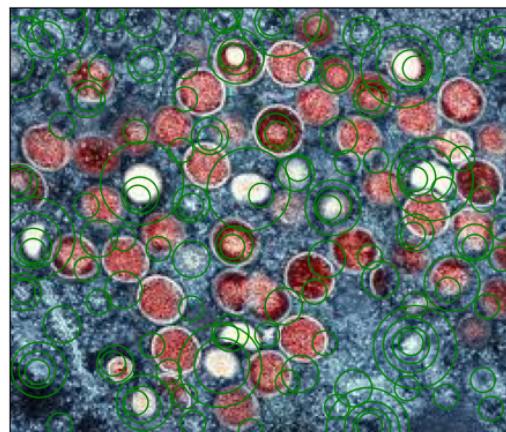
Σχήμα 25: Blobs Detect

## Πολυκλιμακωτή Ανίχνευση Blobs

Σε αυτό το σημείο επεκτείνουμε την παραπάνω μέθοδο για ανίχνευση σε πολλαπλές κλίμακες, αντίστοιχα με το ερώτημα 2.2. Είναι φανερό ότι ο συγκεκριμένος ανιχνευτής θα έχει την καλύτερη απόδοση, καθώς συνδυάζει τα πλεονεκτήματα της χρήσης της ορίζουσας του πίνακα Hessian με αυτά της χρήσης πολλών διαφορετικών κλιμάκων. Η έξοδος που παίρνουμε είναι η εξής:



Σχήμα 26: UP: Multiscale Blobs Detect

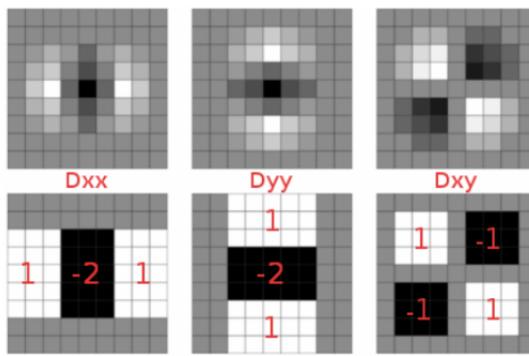


Σχήμα 27: CELLS: multiscale Blobs Detect

Προφανώς, αυτός ο ανιχνευτής είναι αποδοτικότερος από όλους, καθώς συνδυάζει τα πλεονεκτήματα των blobs και της πολυχλιμακωτής ανίσχευσης. Εάν παρατηρήσουμε τις εικόνες, θα διαπιστώσουμε ότι εντοπίζονται αρκετά περισσότερα σημεία, βελτιώνοντας τα αποτελέσματά μας, καθώς διαθέτουν χρήσιμη πληροφορία.

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

Ο υπολογισμός της Hessian που εκτελέσαμε νωρίτερα για κάθε κλίμακα είναι συνέλιξη της εικόνας με αυξανόμενου μεγέθους φίλτρα, που είναι μια υπολογιστικά ακριβή διαδικασία. Για την επιτάχυνση της μεθόδου αυτής, μπορούμε να προσεγγίσουμε τα φίλτρα 2ης παραγώγου με Box Filters, δηλαδή με φίλτρα που βασίζονται σε αθροίσματα ορθογώνιων περιοχών, όπως φαίνεται και στην παρακάτω εικόνα. Ο υπολογισμός τέτοιων αθροισμάτων γίνεται πολύ αποτελεσματικά με χρήση Ολοκληρωτικών Εικόνων.



Σχήμα 28: προσέγγιση φίλτρων 2ης παραγώγου

- $D_{xx}$  : Ύψος παραθύρου  $4 \times \lfloor n/6 \rfloor + 1$  και πλάτος παραθύρου  $2 \times \lfloor n/6 \rfloor + 1$
- $D_{yy}$  : Ύψος παραθύρου  $2 \times \lfloor n/6 \rfloor + 1$  και πλάτος παραθύρου  $4 \times \lfloor n/6 \rfloor + 1$
- $D_{xy}$  : Ύψος παραθύρου  $2 \times \lfloor n/6 \rfloor + 1$  και πλάτος παραθύρου  $2 \times \lfloor n/6 \rfloor + 1$

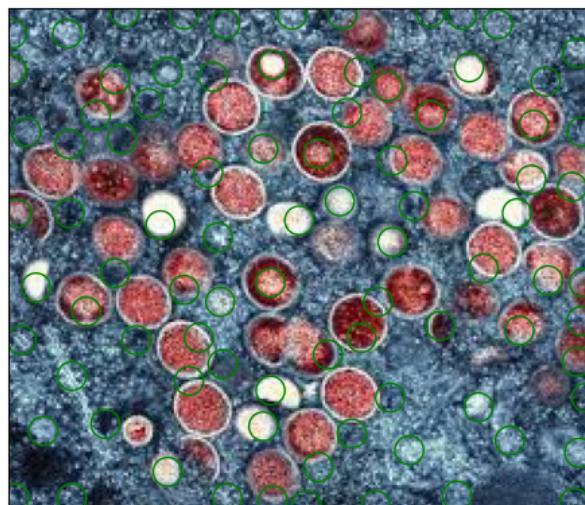
Αφού υπολογίσουμε τις μερικές παραγώγους χρησιμοποιούμε το ακόλουθο κριτήριο:

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

Τελικά προκύπτει:



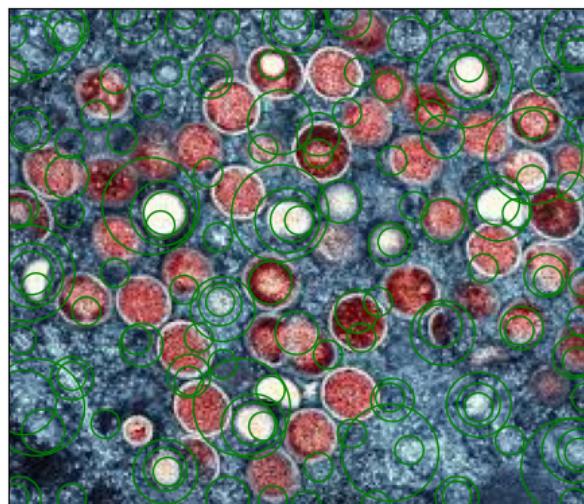
$\Sigma\chi\mu\alpha$  29: UP: box filter



$\Sigma\chi\mu\alpha$  30: CELLS: box filter



Σχήμα 31: UP: multi box filter



Σχήμα 32: CELLS: multi box filter

## Εφαρμογές σε Ταιριασμα 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 κωδικοποιούν 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 παραγώγους κατευθύνσεων μέσω 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):** Ψηφολογίζουμε για κάθε pixel της εικόνας το μέτρο και η γωνία των κατευθυντικών παραγώγων ως εξής:

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

$$\theta(x, y) = \arctan\left(\frac{I_y(x, y)}{I_x(x, y)}\right)$$

Στη συνέχεια, χωρίζουμε την εικόνα σε ορθογώνια κελιά (επικαλυπτόμενα ή μη) και για κάθε ένα από αυτά υπολογίζουμε ένα ιστόγραμμα κατεύθυνσης. Τα ιστογράμματα όταν συνενώνονται δίνουν τον τελικό περιγραφητή της εικόνας.

## 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. Συγκεκριμένα για το πέραμα αυτό έχουν χρησιμοποιηθεί 3 εικόνες, τις οποίες έχουμε παραμορφώσει περιστρέφοντάς τες και αλλάζοντας το μέγεθός τους. 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.

Image	Avg. Scale Error (1)	Avg. Theta Error (1)
Image 1	0.011	1.517
Image 2	0.001	0.158
Image 3	0.008	1.852

Πίνακας 1: SURF: Corner Detection

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.113	17.484
<b>Image 2</b>	0.848	13.249
<b>Image 3</b>	0.205	15.226

Πίνακας 2: HOG corner detection

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.001	0.121
<b>Image 2</b>	0.002	0.118
<b>Image 3</b>	0.002	0.162

Πίνακας 3: SURF Harris Laplacian

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.120	14.770
<b>Image 2</b>	0.435	8.100
<b>Image 3</b>	0.161	15.000

Πίνακας 4: HOG Harris Laplacian

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.036	2.659
<b>Image 2</b>	0.006	0.682
<b>Image 3</b>	0.001	0.056

Πίνακας 5: SURF Blob Detection

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.106	7.630
<b>Image 2</b>	0.136	12.299
<b>Image 3</b>	0.144	22.726

Πίνακας 6: HOG Blob Detection

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.002	0.082
<b>Image 2</b>	0.002	0.094
<b>Image 3</b>	0.001	0.065

Πίνακας 7: SURF:Multiscale blob detection

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.112	14.423
<b>Image 2</b>	0.032	3.542
<b>Image 3</b>	0.203	23.761

Πίνακας 8: HOG Multiscale Blob Detection

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.003	0.139
<b>Image 2</b>	0.007	0.320
<b>Image 3</b>	0.002	0.132

Πίνακας 9: SURF multiscale box filter

<b>Image</b>	<b>Avg. Scale Error (1)</b>	<b>Avg. Theta Error (1)</b>
<b>Image 1</b>	0.180	15.175
<b>Image 2</b>	0.594	31.584
<b>Image 3</b>	0.351	37.698

Πίνακας 10: HOG Multiscale Box Filter

Τα SURF βασίζονται στην προσέγγιση των σημείων ενδιαφέροντος και βρίσκουν τις χατευθύνσεις 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 τασκς όπως το classification. Ουσιαστικά, δίνουν μια πυκνή επικαλυπτόμενη περιγραφή των περιοχών της εικόνας και υπολογίζονται σε ένα πυκνό πλέγμα ομοιόμορφα κατανεμημένων κελιών. Αυτός ο τρόπος όμως αποδεικνύεται μη ανεκτικός σε αλλαγή της κλίμακας και σε περιστροφή καθώς γίνεται σε όλα τα κελιά. Συγχρίνοντας 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.

## 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. Για την κατασκευή ακολουθούμε τα βήματα της εκφώνησης.

Χρησιμοποιούμε την αναπαράσταση Bag of Visual Words (BoVW) για ταξινόμηση εικόνων, επειδή παρέχει έναν τρόπο να συνοψίσουμε τις πληροφορίες που υπάρχουν σε ένα μεγάλο σύνολο τοπικών περιγραφητών σε ένα ενιαίο καυθολικό διάνυσμα χαρακτηριστικών, το οποίο μπορεί να χρησιμοποιηθεί για ταξινόμηση. Η προσέγγιση BoVW βασίζεται στην ιδέα της ομαδοποίησης τοπικών περιγραφητών σε ένα σύνολο από visual words, οι οποίες χρησιμοποιούνται για την αναπαράσταση της εικόνας. Αυτό μειώνει τη διάσταση των δεδομένων και τα καυθιστά πιο υπολογιστικά προσπελάσιμα.

Στο τελευταίο ερώτημα της άσκησης, χρησιμοποιούμε την αναπαράσταση Bag of Visual Words, για να εκπαιδεύσουμε τον ταξινομητή SVM που θα χρησιμοποιήσουμε. Παρακάτω παρουσιάζονται τα αποτελέσματα, σχετικά με την ακρίβεια της ταξινόμησης για κάθε μέθοδο:

Algorithm	Feature Extraction Time	Accuracy
corner_detection2	SURF	52.552%
	HOG	62.759%
Harris_Laplacian2	SURF	62.345%
	HOG	66.483%
blob_detection2	HOG	67.310%
	SURF	65.793%
blob_detection_scales2	SURF	58.897%
	HOG	69.793%
BoxFilter_scales2	SURF	61.241%
	HOG	65.103%

Πίνακας 11: Feature extraction time and mean accuracy for various algorithms with different descriptors

Παρατηρούμε πως τα αποτελέσματά μας συμφωνούν σε έναν βαθμό με τα πλεονεκτήματα και μειονεκτήματα που αναφέρθηκαν για τον κάθε ανιχνευτή στο μέρος 2. Όπως είναι φανερό, ο ανιχνευτής γωνιών παρουσιάζει βελτιωμένα αποτελέσματα, όταν λειτουργεί σε πολλές κλίμακες. Ωστόσο, καθώς τα blobs διατηρούν περισσότερη χρήσιμη πληροφορία, ο ανιχνευτής τους εμφανίζει σημαντικά καλύτερη απόδοση, η οποία αυξάνεται ακόμα περισσότερο στη χρήση πολλών κλιμάκων με τη χρήση του HOG, αλλά αντίθετα με αυτό που περιμέναμε μειώνεται κατά τη χρήση του SURF, πιθανώς λόγω της αφαίρεσης πολλών σημείων που δεν αποτελούν τοπικά μέγιστα σε διαδοχικές κλίμακες. Τέλος, η υλοποίηση με Box Filters, αν και προσεγγιστική, παρουσιάζει ικανοποιητικά αποτελέσματα και θα μπορούσε να χρησιμοποιηθεί για επιτάχυνση του classification (παρόλο που κατά την υλοποίησή μας δεν παρουσιάστηκε επιτάχυνση, διότι δεν ήταν βέλτιστη).

Σχετικά με τους descriptors, σε όλες τις περιπτώσεις, ο HOG έχει καλύτερα αποτελέσματα. Αυτό οφείλεται, κυρίως, στο ότι τα HOGs παρέχουν μια πυκνή επικαλυπτόμενη περιγραφή των περιοχών μια εικόνας και υπολογίζονται σε ένα πυκνό πλέγμα ομοιόμορφα κατανεμημένων κελιών, γεγονός που τα καθιστά κατάλληλα για classification και object detection, σε αντίθεση με τα SURF που υπολογίζουν το γραδιεντ μόνο γύρω από τα σημεία ενδιαφέροντος, οπότε κρατάνε τοπική πληροφορία.