



## Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

Όραση Υπολογιστών

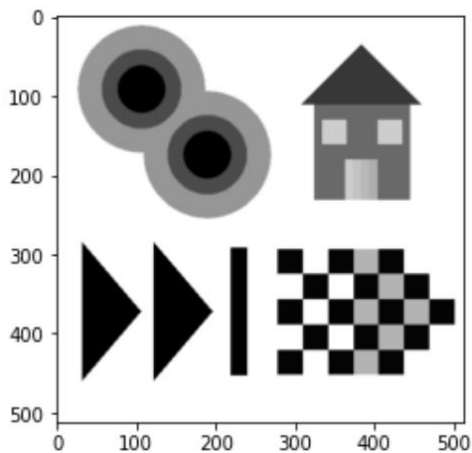
Εξάμηνο 8<sup>ο</sup> (Εαρινό Εξάμηνο 2021-2022)

1<sup>η</sup> Εργαστηριακή Άσκηση ,

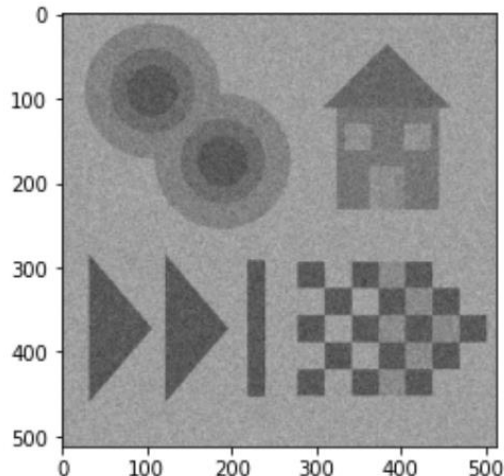
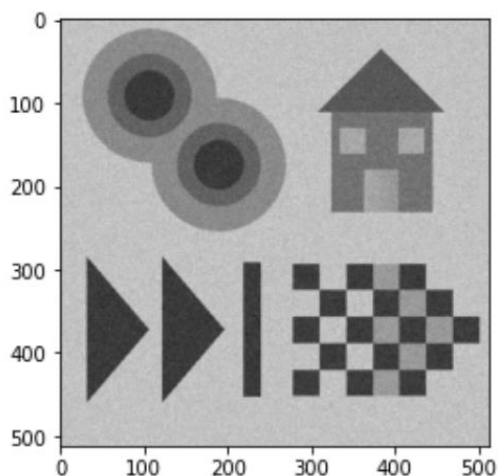
Χάιδος Νικόλαος ei18096, Σπανός Νικόλαος ei18822

### Μέρος 1

1.1.1) Η αρχική εικόνα, σε grayscale, είναι η εξής:



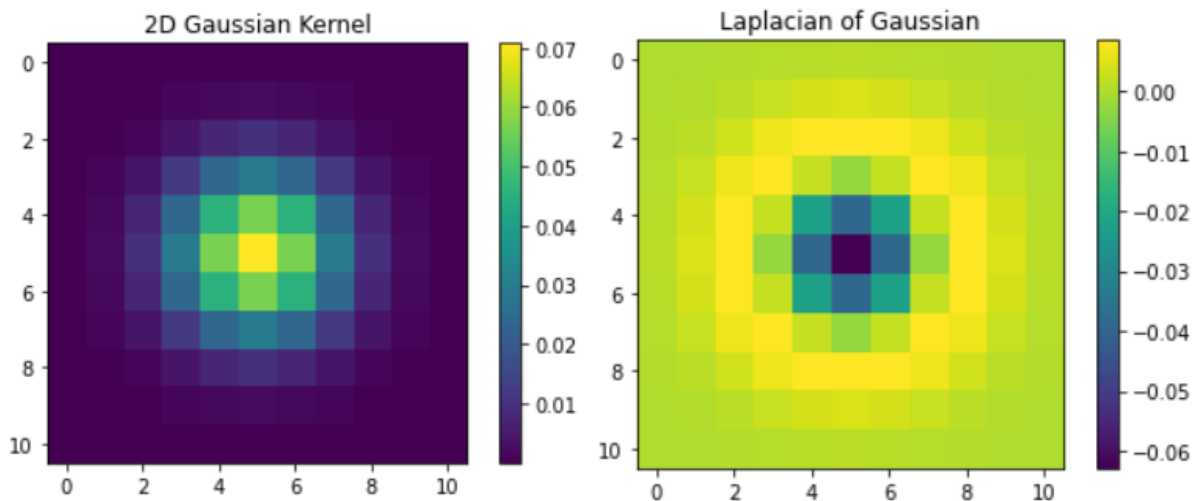
1.1.2) Μετά από την πρόσθεση θορύβου, οι εικόνες για  $PSNR = 20dB$  και  $PSNR = 10dB$ , αντίστοιχα, είναι οι εξής:



Μπορούμε να δούμε ότι στην δεύτερη εικόνα, ο θόρυβος είναι αρκετά πιο εμφανής.

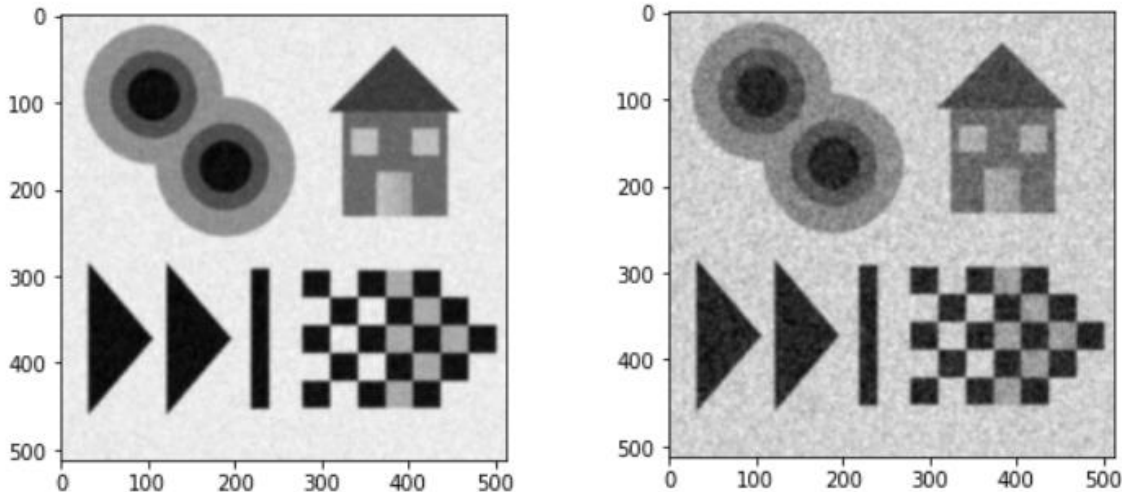
**1.2.1)** Σχεδιάσαμε την δισδιάστατη Gaussian χρησιμοποιώντας την *getGaussianKernel* της *OpenCV*, και για την Λαπλασιανή της Gaussian, την υπολογίσαμε μέσω του *meshgrid* και τον γνωστό τύπο της LoG.

Το δισδιάστατο Gaussian Kernel και το LoG είναι:



Η τυπική απόκλιση που χρησιμοποιήσαμε για τα δύο εικονιζόμενα kernels είναι  $\sigma = 1.5$ . Μπορούμε να το επιβεβαιώσουμε και οπτικά ότι τα φίλτρα απεικονίζουν τις σωστές συναρτήσεις, αφού το Gaussian Kernel έχει ένα peak στο κέντρο, ενώ το LoG έχει ένα ελάχιστο στο κέντρο (αντίστροφο «μεξικάνικο καπέλο»).

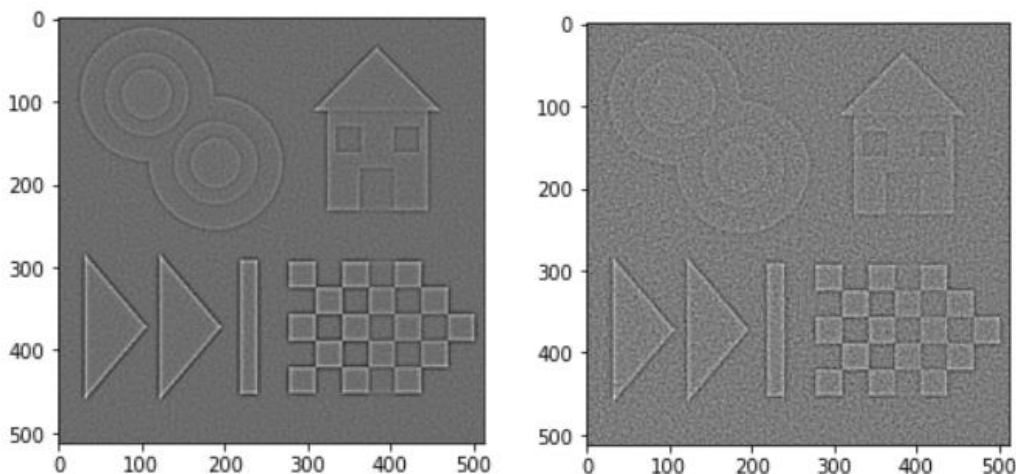
**1.2.2)** Για αυτό το βήμα, αρχικά κάναμε smoothing στις εικόνες με το Gaussian Kernel, όπως το ορίσαμε παραπάνω. Οι εξομαλυμένες εικόνες είναι οι εξής: (αριστερά για PSNR 20dB, και δεξιά για 10dB)



Ύστερα, υλοποιήσαμε τις δύο διαφορετικές μεθόδους εύρεσης ακμών, γραμμική και μη-γραμμική:

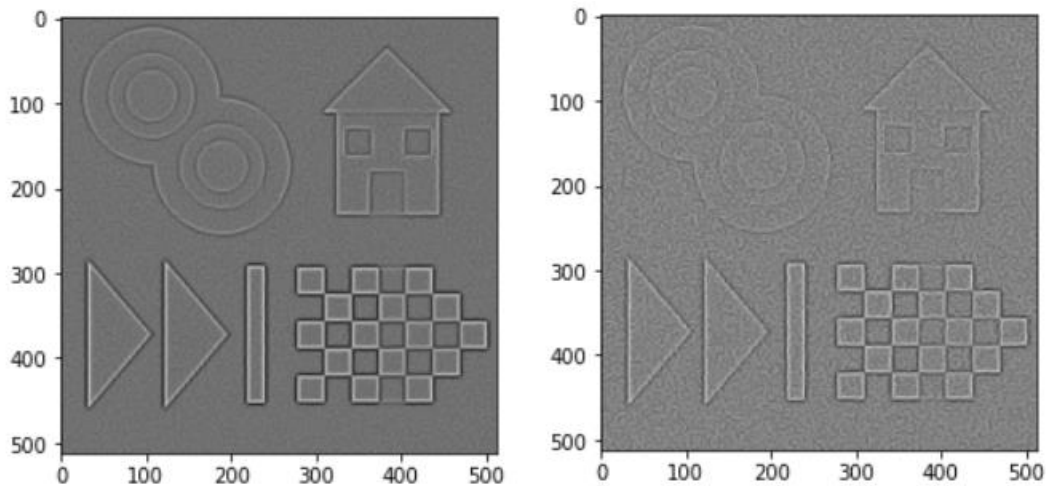
### 1. Γραμμική

Για την γραμμική προσέγγιση, απλά εφαρμόζουμε το LoG φίλτρο που σχεδιάσαμε προηγουμένως, στην αρχική εικόνα (δεν χρειάζεται εξομάλυνση, αφού είναι ενσωματωμένη στο φίλτρο LoG). Οι Laplacian των δύο εικόνων είναι: (αριστερά για PSNR 20dB, και δεξιά για 10dB)



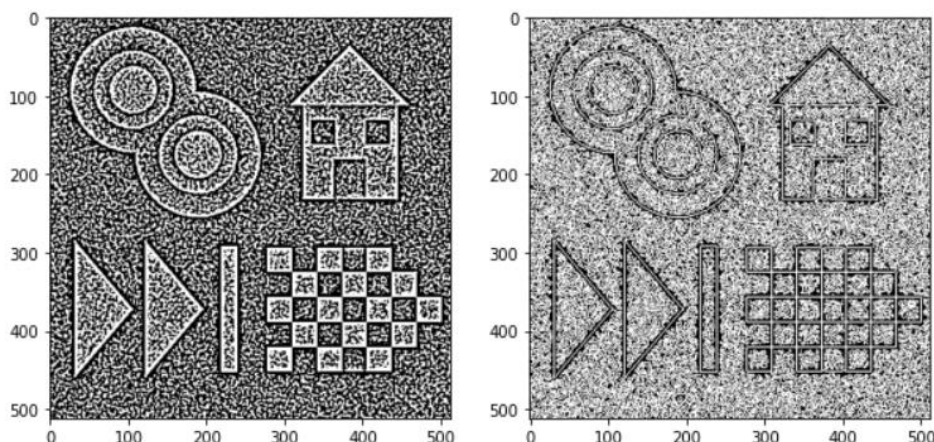
## 2. Μη-Γραμμική

Με την μη-γραμμική προσέγγιση, εφαρμόζουμε στην εξομαλυμένη εικόνα ένα dilation, ένα erosion, και αφού τα προσθέσουμε, αφαιρούμε την εξομαλυμένη εικόνα (με παράγοντα 2). Αυτή η πράξη είναι η αντίστοιχη προσέγγιση της λαπλασιανής της εικόνας, αλλά χρησιμοποιώντας μη-γραμμικά φίλτρα (erosion, dilation). Οι Laplacian των δύο εικόνων είναι: (αριστερά για PSNR 20dB, και δεξιά για 10dB)

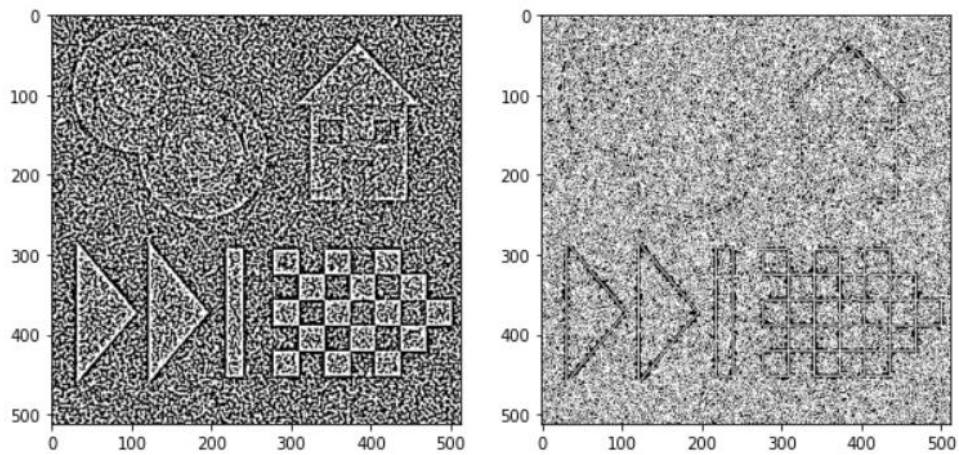


**1.2.3)** Για κάθε μία προσέγγιση, και για κάθε μία από τις δύο τιμές *PSNR*, υπολογίσαμε την Δυναμική Εικόνα Προσέγγισης  $X$ , και το περίγραμμα της  $X$ . Συγκεκριμένα, οι δύο αυτές εικόνες για κάθε περίπτωση είναι οι εξής: (αριστερά η Δυναμική Εικόνα Προσέγγισης  $X$ , δεξιά το περίγραμμα της  $X$ )

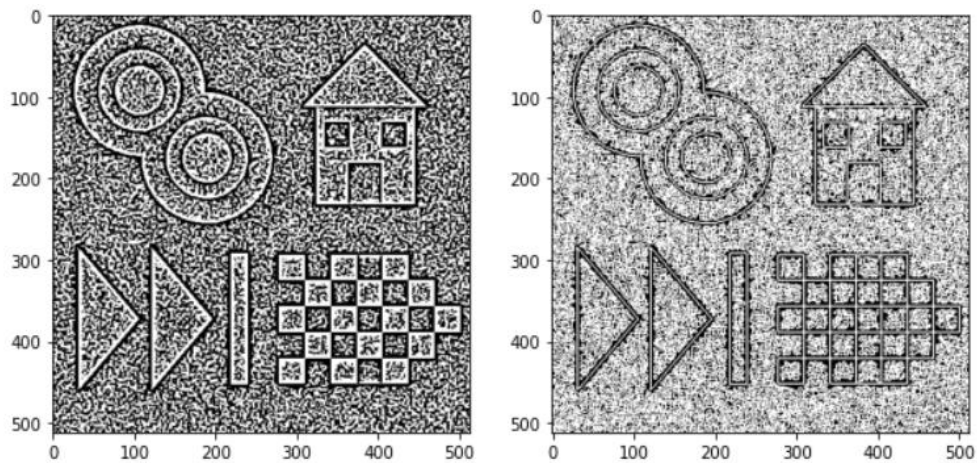
### 1. *PSNR* = 20dB, Γραμμική Προσέγγιση



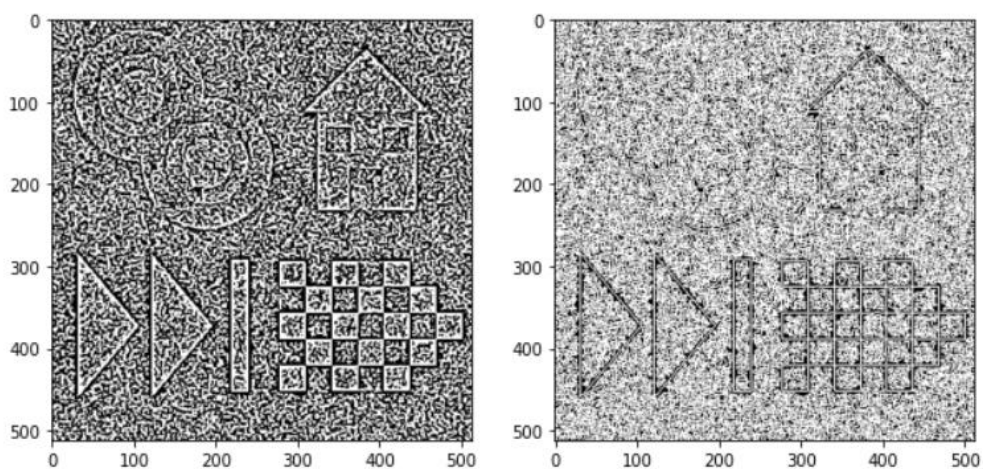
2.  $PSNR = 10dB$ , Γραμμική Προσέγγιση



3.  $PSNR = 20dB$ , Μη-Γραμμική Προσέγγιση



4.  $PSNR = 10dB$ , Μη-Γραμμική Προσέγγιση

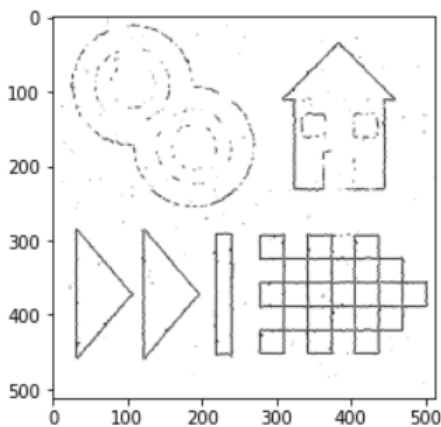


Μπορούμε ήδη να παρατηρήσουμε ότι η μη-γραμμική προσέγγιση δίνει καλύτερα αποτελέσματα, ειδικά στην περίπτωση  $PSNR = 10\text{ dB}$ . Στο σχήμα τις εικόνες που βρίσκεται κάτω δεξιά, και έχει αρκετές γωνίες και ακμές, η μη-γραμμική προσέγγιση τις απεικονίζει πιο καθαρά.

**1.2.4)** Τέλος, εφαρμόζουμε και ένα threshold  $\theta_{edge}$  στο περίγραμμα του  $X$ , για να αποκλείουμε μερικές τιμές που αντιστοιχούσαν σε θόρυβο.

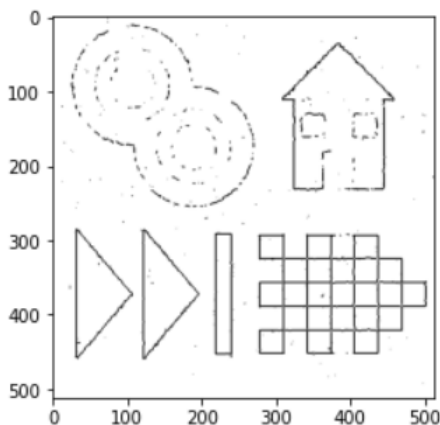
Τα καλύτερα αποτελέσματα για κάθε προσέγγιση, σε κάθε εικόνα, τα πήραμε με τις εξής παραμέτρους:

- $PSNR = 10\text{ dB}$ , Γραμμική Προσέγγιση,  $\theta_{edge} = 0.3$



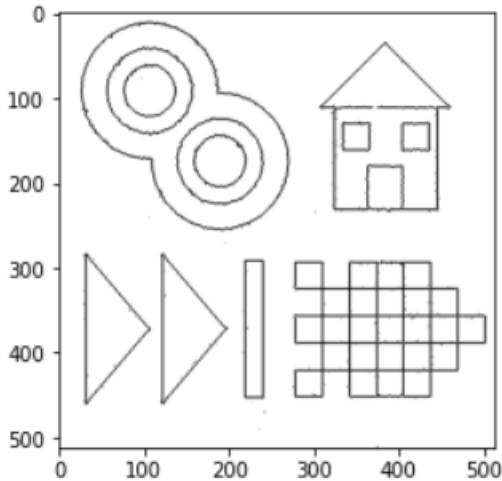
Είχαμε ένα trade-off στο  $\theta_{edge}$ . Αν το αυξήσουμε περισσότερο, τότε θα ξεκινήσουν να αποκλείονται και οι κανονικές ακμές των σχημάτων, ενώ αν το μειώσουμε θα έχει μεγάλη επίδραση ο θόρυβος.

- $PSNR = 10\text{ dB}$ , Μη-Γραμμική Προσέγγιση,  $\theta_{edge} = 0.3$



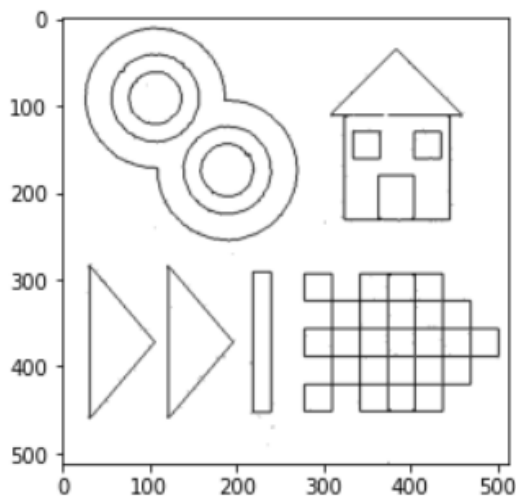
Με τις ίδιες παραμέτρους μπορούμε να δούμε ότι η Μη-Γραμμική προσέγγιση είναι πιο εύρωστη στο θόρυβο (πολύ λιγότερες κουκίδες, ειδικά στην περιοχή του κάτω-δεξιά σχήματος).

- $PSNR = 20dB$ , Γραμμική Προσέγγιση,  $\theta_{edge} = 0.13$



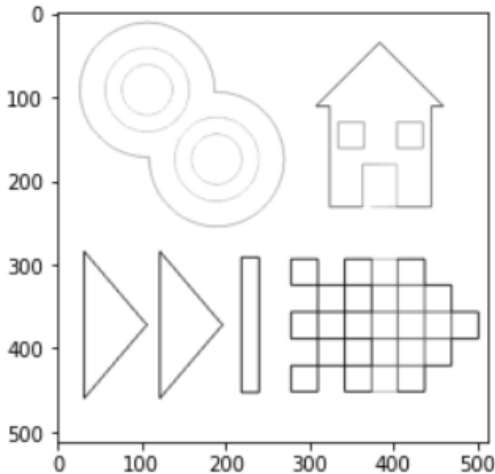
Στην περίπτωση του  $PSNR = 20dB$ , μπορούμε να χαμηλώσουμε ακόμα περισσότερο το  $\theta_{edge}$  για να πάρουμε μεγαλύτερη λεπτομέρεια στις πραγματικές ακμές (αφού έχουμε λιγότερη επιρροή από τον θόρυβο).

- $PSNR = 20dB$ , Μη-Γραμμική Προσέγγιση,  $\theta_{edge} = 0.13$



Σε αυτήν την περίπτωση, δεν είναι τόσο εμφανής η διαφορά μεταξύ Γραμμικού και Μη-Γραμμικού, κυρίως λόγω του μικρού θορύβου. Αλλά και πάλι έχουμε ένα καλό αποτέλεσμα για  $\theta_{edge} = 0.13$ .

**1.3.1)** Υπολογίζοντας T στην αρχική εικόνα, για να βρούμε τις “αληθινές” ακμές της εικόνας *edgetest<sub>22</sub>.png* (χωρίς θόρυβο), παίρνουμε τις εξής “αληθινές” ακμές: (για  $\theta_{realedge} = 0.2$ )



**1.3.2)** Για την αξιολόγηση της ανίχνευσης ακμών, υλοποιήσαμε την συνάρτηση *evaluation*, η οποία παίρνει σαν arguments, την αρχική καθαρή εικόνα, τις ακμές που έχουμε υπολογίσει, και την παράμετρο  $\theta_{realedge}$ . Ουσιαστικά, υπολογίζει το *precision* και το *recall* από τους τύπους που δόθηκαν, και επιστρέφει τον μέσο όρο τους.

**1.3.3)** Χρησιμοποιήσαμε την συνάρτηση *Edge\_Detect*, και *Evaluation* που υλοποιήσαμε, για να υπολογίσουμε την μετρική C σε διάφορες περιπτώσεις θορύβου της εικόνας “*edgetest\_22.png*”. Τα τελικά αποτελέσματα, φαίνονται στον εξής πίνακα:

Parameters	C-evaluation
PSNR=20dB, $\sigma=1.3$ , $\theta=0.35$ , linear approach	0.853121
PSNR=20dB, $\sigma=1.3$ , $\theta=0.35$ , non-linear approach	0.854772
PSNR=15dB, $\sigma=1.6$ , $\theta=0.4$ , linear approach	0.837746
PSNR=15dB, $\sigma=1.6$ , $\theta=0.4$ , non-linear approach	0.846461
PSNR=10dB, $\sigma=1.7$ , $\theta=0.45$ , linear approach	0.774321
PSNR=10dB, $\sigma=1.7$ , $\theta=0.45$ , non-linear approach	0.807682



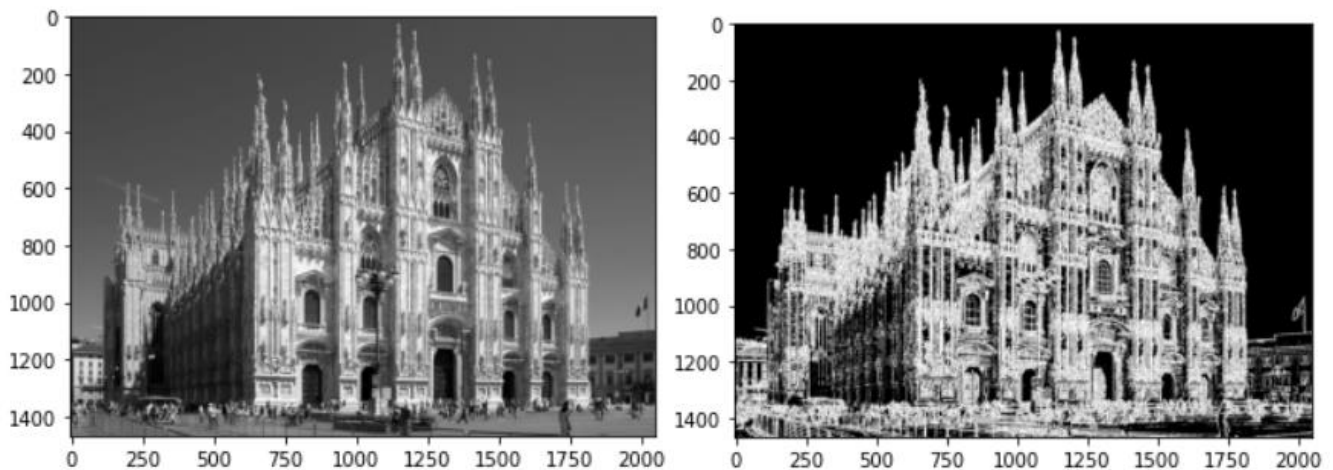
Αλλάξαμε τις παραμέτρους, και τα αποτελέσματα που φαίνονται παραπάνω είναι τα καλύτερα που πετύχαμε.

Αρχικά παρατηρούμε το λογικό αποτέλεσμα, ότι πετυχαίνουμε καλύτερο σκορ, στις εικόνες που είναι λιγότερες αλλοιωμένες από θόρυβο.

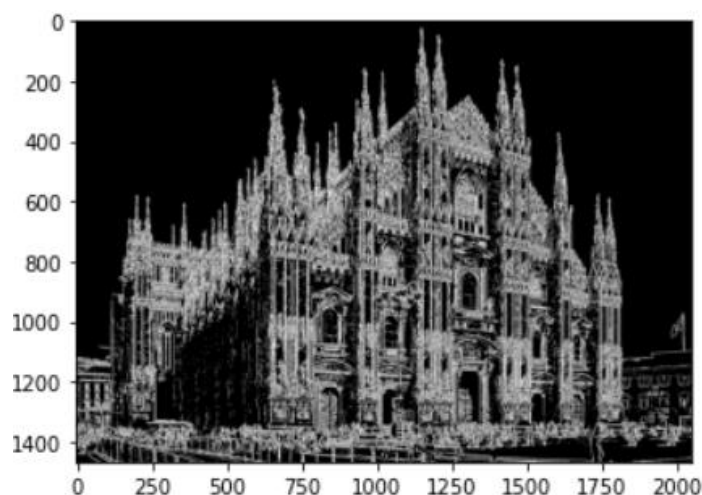
Επίσης, βλέπουμε ότι σε όλες τις περιπτώσεις, η μη-γραμμική προσέγγιση δίνει καλύτερα αποτελέσματα από την γραμμική. Μάλιστα, η διαφορά στο σκορ μεταξύ τους, αυξάνεται καθώς μεγαλώνει ο θόρυβος σε μία εικόνα. Αυτό συμβαίνει επειδή η μη-γραμμική μέθοδος παρουσιάζει περισσότερη αντοχή στον θόρυβο από ότι η γραμμική.

**1.4.1)** Για την εικόνα "*duomo\_edges.jpg*", επειδή έχει μερικές πολύ “λεπτές” ακμές (στα πλάγια του κτηρίου) και μαζεμένες σε μικρό χώρο, χρησιμοποιήσαμε πολύ μικρό σίγμα και  $\theta_{edge}$ , με σκοπό να κρατήσουμε όσες περισσότερες ακμές γίνεται.

Συγκεκριμένα, για  $\sigma = 0.03$  και  $\theta_{edge} = 0.05$  με μη-γραμμική προσέγγιση, πήραμε το παρακάτω ικανοποιητικό αποτέλεσμα (αριστερά, η πραγματική εικόνα για σύγκριση):

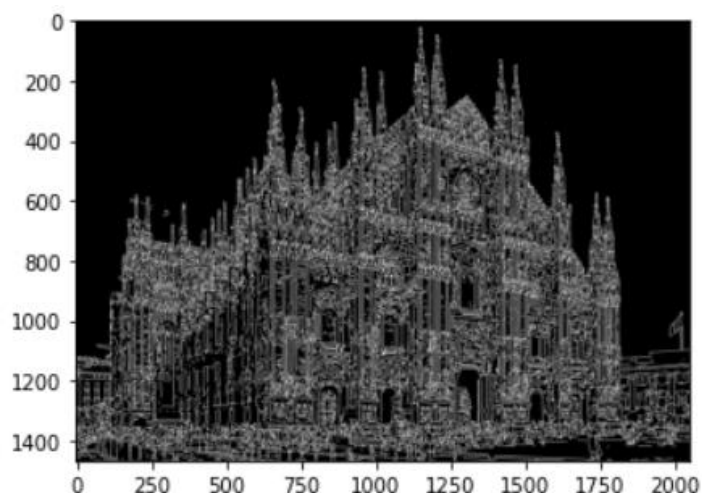


Με την γραμμική προσέγγιση, το καλύτερο αποτέλεσμα που καταφέραμε να πάρουμε, ήταν με παραμέτρους  $\sigma = 1$  και  $\theta_{edge} = 0.1$ , όπου έχουμε το εξής:

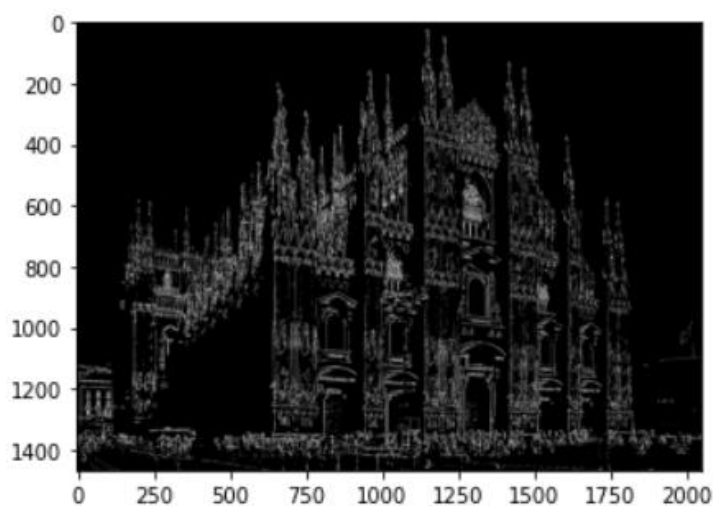


Γενικά, η μη-γραμμική προσέγγιση έδινε πιο έντονα τις ακμές στο πρόσωψη του κτηρίου, αλλά και έδινε αρκετές ακμές στην πλάγια πλευρά του κτηρίου (βλέπουμε ότι η γραμμική μέθοδος παραλείπει αρκετές ακμές σε αυτήν την περιοχή).

Εν γένει, αν αυξήσουμε το  $\sigma$ , το smoothing θα εξαφανίσει πολλές μικρές ακμές της αρχικής εικόνας, π.χ. για  $\sigma = 3$  και  $\theta_{edge} = 0.05$  με μη-γραμμική προσέγγιση:



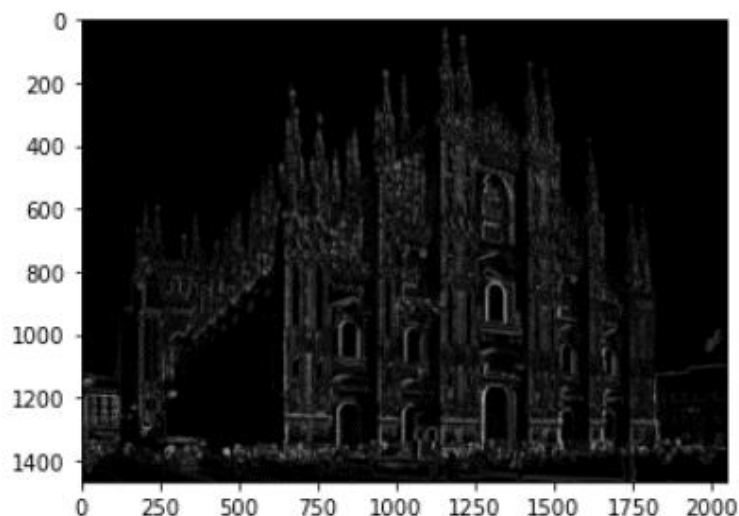
Όμοιο αποτέλεσμα θα έχουμε αν αυξήσουμε το  $\theta_{edge}$ , π.χ. για  $\theta_{edge} = 0.3$  και  $\sigma = 0.03$  με μη-γραμμική προσέγγιση:

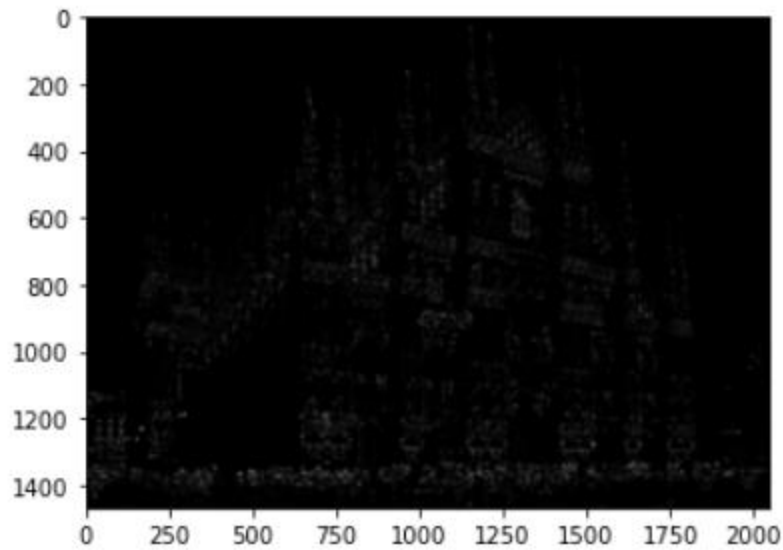


Μπορούμε να δούμε ότι κράτησε μόνο τις πιο έντονες ακμές, και απέρριψε τις πιο “λεπτές” (π.χ. στα πλάγια του κτηρίου, αλλά και το μικρό κτίσμα που βρίσκεται κάτω-δεξιά στην αρχική εικόνα)

## Μέρος 2

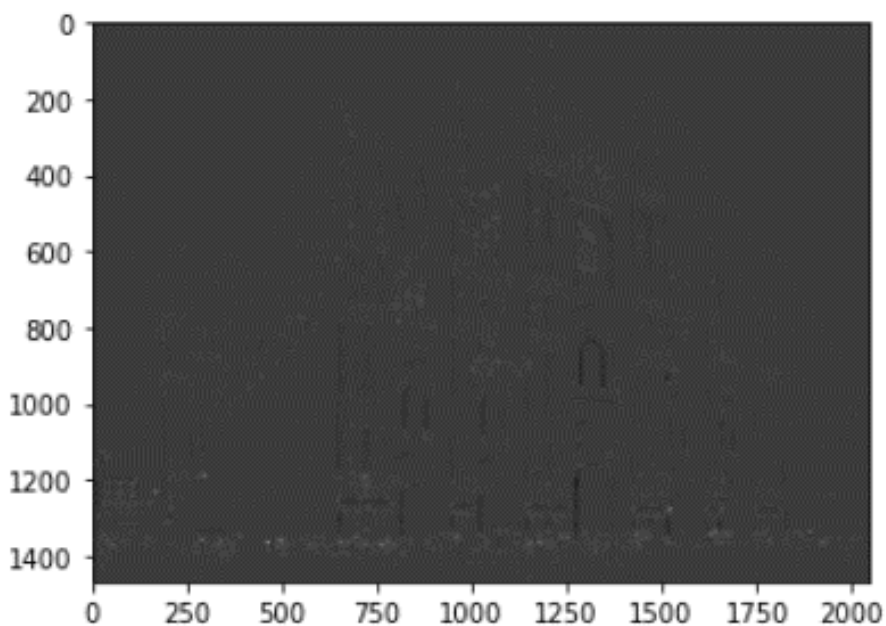
**2.1.2)** Χρησιμοποιώντας τα στοιχεία του δομικού τανυστή  $J$  που φτιάξαμε στο ερώτημα 1, παράγουμε τις δύο ιδιοτιμές για κάθε pixel. Παρατίθενται οι γκριζες εικόνες του  $\lambda_+$  και  $\lambda_-$  αντίστοιχα:





Στην εικόνα του  $I_+$  μπορούμε με ευκολία να διακρίνουμε τις ακμές των αντικειμένων που βρίσκονται στην εικόνα, ενώ οι ιδιοτιμές  $I_-$  μας δείχνουν τις γωνίες των αντικειμένων, δηλαδή τα σημεία τομής των ακμών.

**2.1.3)** Υπολογίζοντας το κριτήριο γωνιότητας, παίρνουμε το εξής αποτέλεσμα:



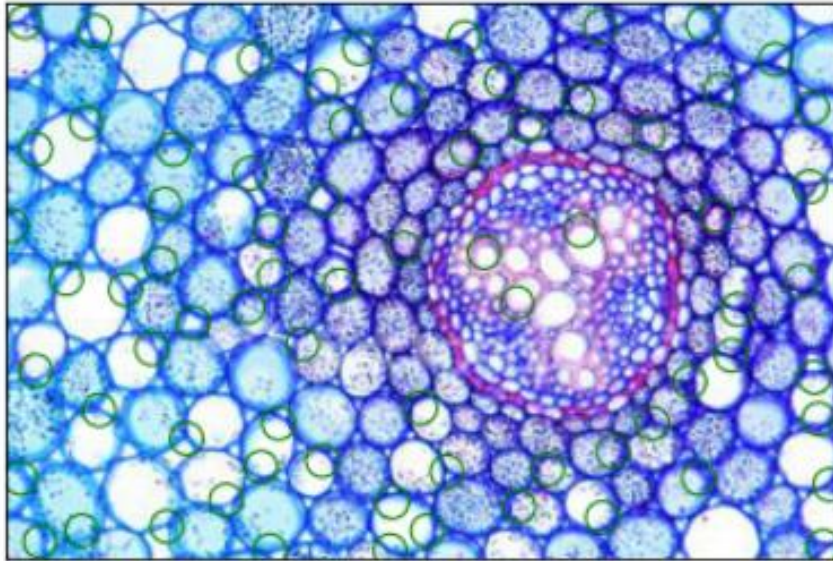
Παρατηρούμε στην εικόνα πως τα άσπρα σημεία υποδεικνύουν τις γωνίες των αντικειμένων, ενώ τα μαύρα τις ακμές.

Εφαρμόζοντας τις δύο συνθήκες στο κριτήριο γωνιότητας και τοποθετώντας τον πίνακα στην δοσμένη συνάρτηση `interest_points_visualization` παίρνουμε τελικά το εξής αποτέλεσμα για τις γωνίες των αντικειμένων της εικόνας:



Παρατηρούμε ότι η διαδικασία έχει σωστά ανιχνεύσει τις περισσότερες γωνίες που βρίσκονται στην εικόνα. Έχοντας υλοποιήσει την συνάρτηση, την εφαρμόζουμε και στις φωτογραφίες `donuts.jpg` και `cells.jpg`:

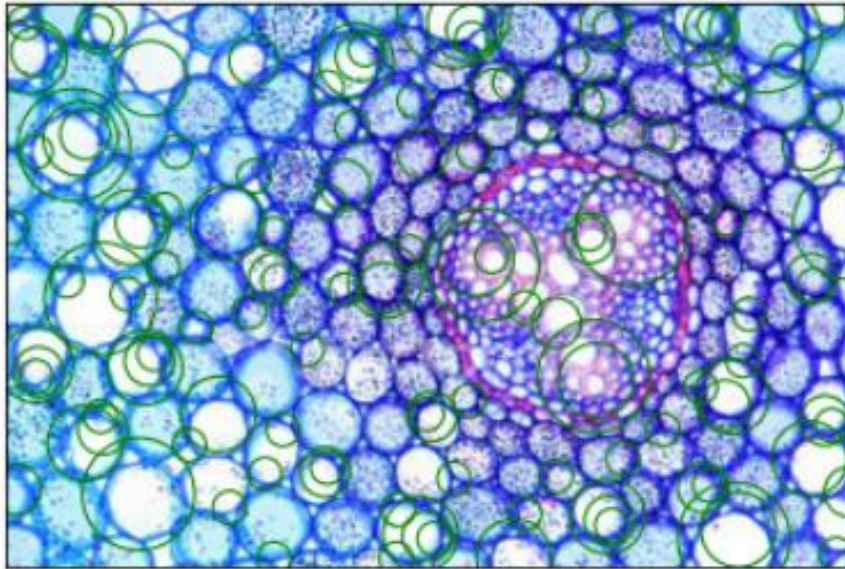




**2.2.2)** Χρησιμοποιώντας ως αρχική κλίμακα διαφόρισης και ολοκλήρωσης το 2.5, παίρνουμε κλίμακες 3.75, 5.625 και 8.4375 για την πολυκλιμακωτή ανίχνευση. Χρησιμοποιώντας το δοθέν κριτήριο για την Λαπλασιανή για κάθε κλίμακα διαφόρισης και εφαρμόζοντας τις ίδιες δύο συνθήκες που εφαρμόσαμε και στο προηγούμενο ερώτημα, παίρνουμε τα εξής αποτελέσματα για τις δύο εικόνες:

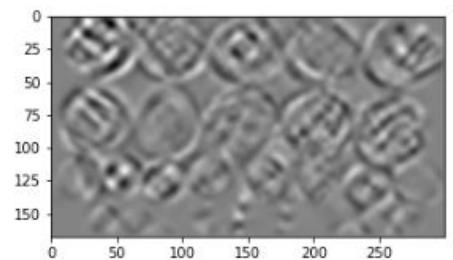
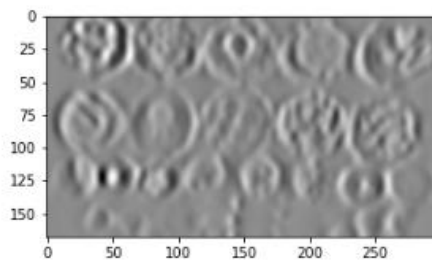
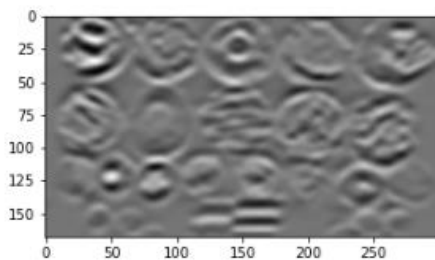




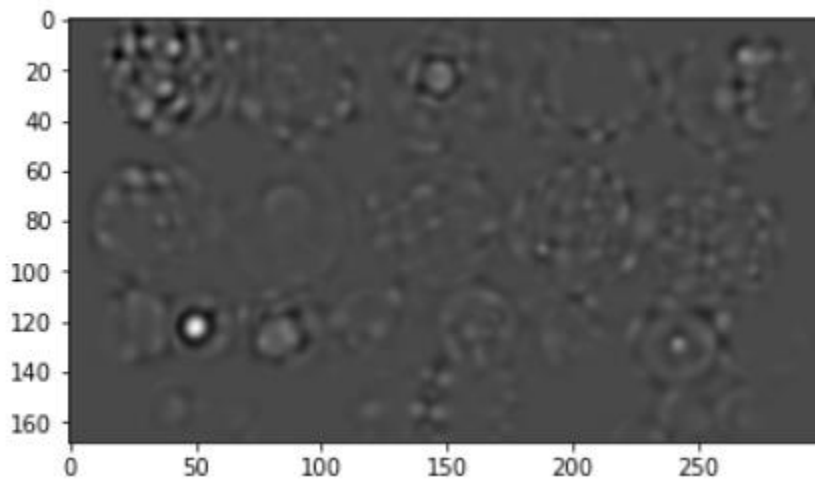


Παρατηρούμε σε ορισμένα σημεία των εικόνων πως οι μεγαλύτερες κλίμακες έχουν ανιχνεύσει γωνίες που οι μικρότερες δεν κατάφεραν να ανιχνεύσουν. Παράλληλα, μπορούμε να παρατηρήσουμε ότι ορισμένες γωνίες ανιχνεύονται και από πολλαπλές κλίμακες.

**2.3.1)** Εφαρμόζοντας τον αλγόριθμο , παίρνουμε τα εξής αποτελέσματα για τα  $L_{xx}$ ,  $L_{yy}$  και  $L_{xy}$  αντίστοιχα:



Το κριτήριο R βγαίνει ως εξής:

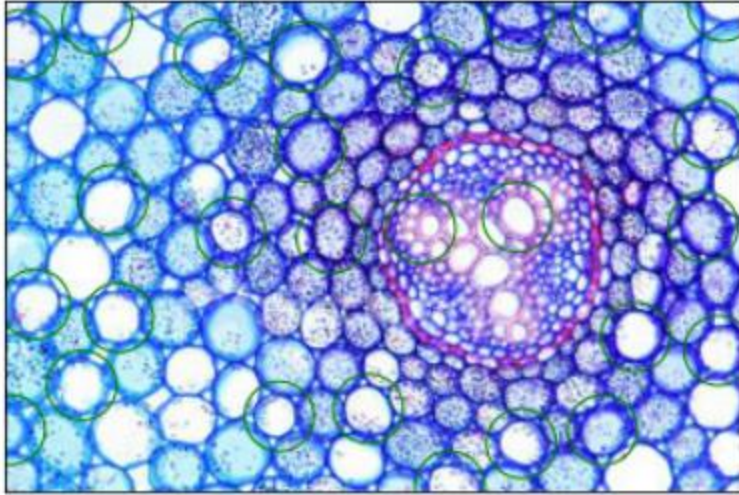


**2.3.2)** Εφαρμόζοντας τις συνθήκες για την επιλογή των σημείων blobs, παίρνουμε το εξής αποτέλεσμα:



Μπορούμε να παρατηρήσουμε ότι έχουν ανιχνευτεί περιοχές οι οποίες διαφέρουν αρκετά με τις γειτονικές, όπως το κεράσι κάτω αριστερά, τα μούρα πάνω δεξιά και οι λουκουμάδες κάτω αριστερά. Για την εικόνα cells, αυξάνοντας το  $\sigma$  για καλύτερη απόδοση, παίρνουμε το εξής αποτέλεσμα:

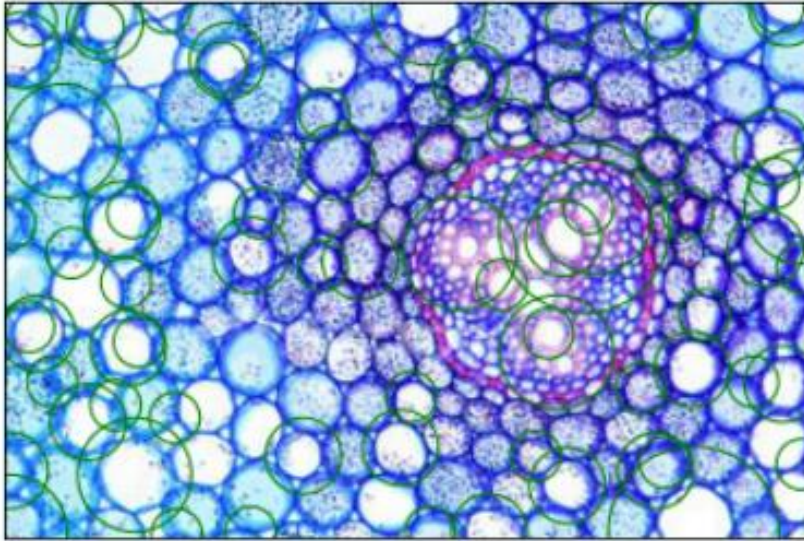




Εδώ παρατηρούμε ότι ανιχνεύονται κύτταρα τα οποία διαφέρουν χαρακτηριστικά από τα γειτονικά τους.

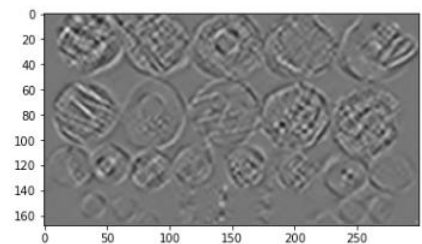
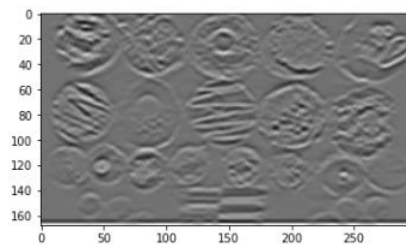
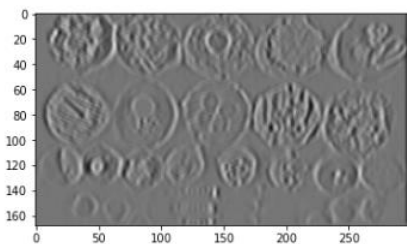
**2.4)** Για την πολυκλιμακωτή ανίχνευση blobs, ακολουθούμε ακριβώς την ίδια διαδικασία με την ανίχνευση γωνιών. Παίρνουμε τα εξής αποτελέσματα:





Σε κάθε εικόνα, ανιχνεύονται περισσότερα blobs από ότι με την μία κλίμακα, με αποτέλεσμα να ανιχνεύονται περισσότερες περιοχές ενδιαφέροντος , ανάλογα με τα αντικείμενα της εικόνας.

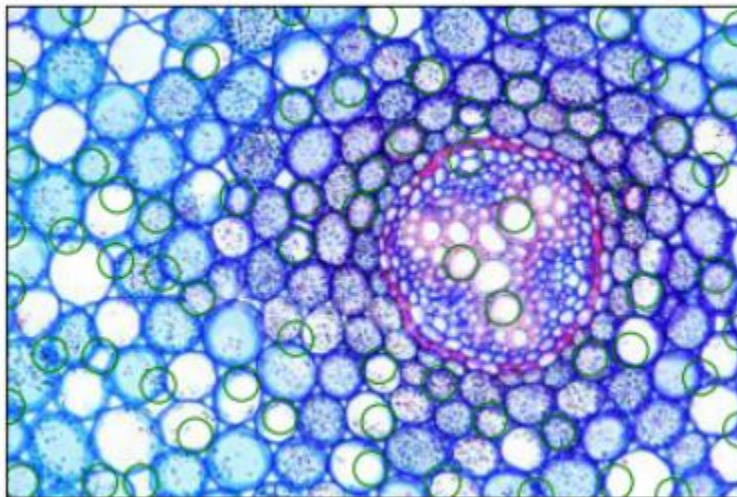
**2.5.2)** Υλοποιούμε την συνάρτηση για την παραγωγή των δεύτερων παραγώγων με προσέγγιση με Box Filters. Για τις δεύτερες παραγώγους παίρνουμε τα εξής αποτελέσματα για τα  $L_{xx}$ ,  $L_{yy}$  και  $L_{xy}$  αντίστοιχα για donuts και cells:



### 2.5.3)

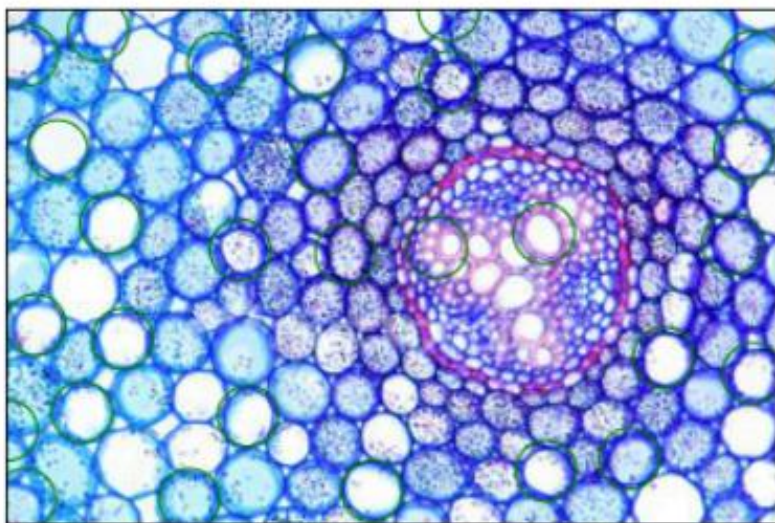
Για την εικόνα donuts παίρνουμε τα εξής αποτελέσματα για τα αντίστοιχα  $\sigma$ :

$\sigma = 3$ :

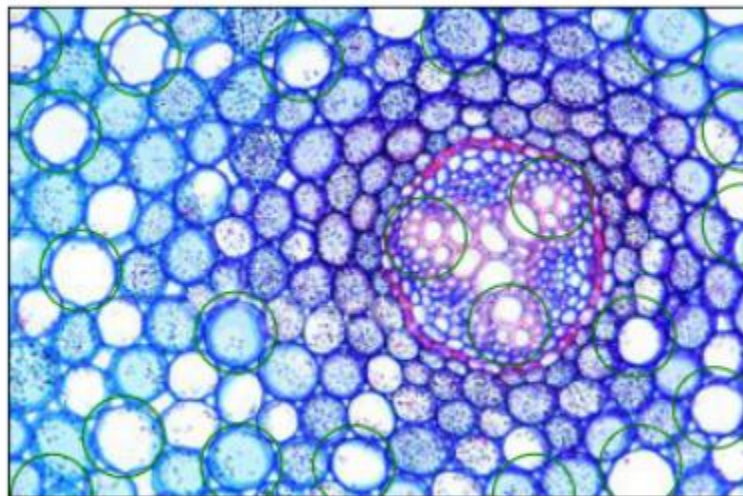




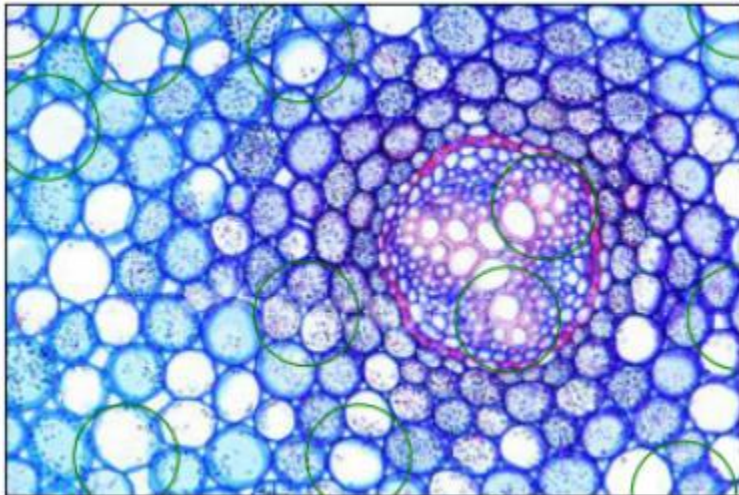
$\sigma = 5$ :



$\sigma = 7$ :



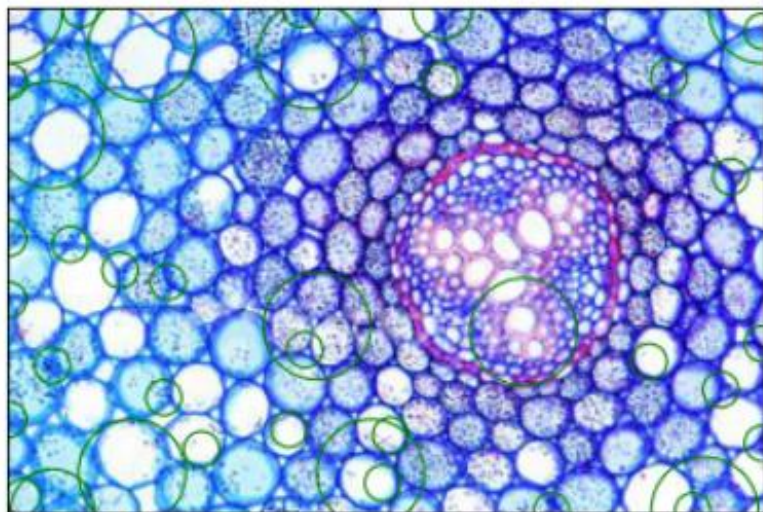
$\sigma = 9$ :



Για την εικόνα donuts, όσο ανεβαίνουμε κλίμακα, ανιχνεύονται μεγαλύτερα αντικείμενα με διαφορές σε σχέση με τα γειτονικά τους (blobs), αλλά είναι σημαντικά λιγότερα και οι λεπτομέρειες δεν είναι τόσο εύκολα διακριτές. Για την εικόνα cells, σε μικρές κλίμακες δεν αναγνωρίζονται καλά οι περιοχές ενδιαφέροντος, διότι τα περισσότερα κύτταρα είναι μεγάλου μεγέθους, ενώ για μεγαλύτερες κλίμακες έχουμε καλύτερη διάκριση.



**2.5.4)** Στην συνέχεια εκτελούμε πολυκλιμακωτή ανίχνευση με Box Filters. Παίρνουμε τα εξής αποτελέσματα:



Όπως και για την προσέγγιση χωρίς Box Filters, με χρήση πολυκλιμακωτής ανίχνευσης μπορούμε να διακρίνουμε αρκετές περιοχές ενδιαφέροντος με διαφορετικά χαρακτηριστικά και στις δύο εικόνες.

### Μέρος 3

#### 3.1.2

Χρησιμοποιώντας τις δύο συναρτήσεις περιγραφητών που μας δόθηκαν, θα συγκρίνουμε τα λάθη που παρουσιάζονται αν συνδυάσουμε τον κάθε έναν από τους 5 ανιχνευτές σημείων ενδιαφέροντος, με τους δύο περιγραφητές.

Τα αποτελέσματα συνοψίζονται στα παρακάτω πίνακάκια:

Για την μέθοδο SURF:

	Corner Detection	Corner Multiscale	Blob Detection	Blob Multiscale	Blob Integral Image Multiscale
Λάθος Περιστροφής Εικόνας 1	1.968	0.102	7.759	0.073	6.954
Λάθος Κλιμάκωσης Εικόνας 1	0.003	0.002	0.027	0.001	0.059
Λάθος Περιστροφής Εικόνας 2	0.318	0.239	0.229	0.064	4.813
Λάθος Κλιμάκωσης Εικόνας 2	0.002	0.003	0.01	0.001	0.043
Λάθος Περιστροφής Εικόνας 3	12.909	0.138	0.054	0.070	17.993
Λάθος Κλιμάκωσης Εικόνας 3	0.097	0.002	0.001	0.001	0.189

Για την μέθοδο SURF , καλύτερη απόδοση είχαν όλες οι μέθοδοι εκτός της μεθόδου ανίχνευσης γωνίας (μονής κλίμακας) και της μεθόδου ολοκληρωτικής εικόνας. Γενικά,



όμως, όλες οι μέθοδοι ήταν αρκετά αποτελεσματικές στο να ανιχνεύουν την σωστή κλιμάκωση, αλλά κάποιες παρουσίαζαν πρόβλημα στην ανίχνευση περιστροφής (κυρίως, η πολυκλιμακωτή Blob με Integral Image, και η Corner Detect μονής κλίμακας).

Για την μέθοδο HOG:

	Corner Detection	Corner Multiscale	Blob Detection	Blob Multiscale	Blob Integral Image Multiscale
Λάθος Περιστροφής Εικόνας 1	22.619	18.012	7.231	16.226	27.268
Λάθος Κλιμάκωσης Εικόνας 1	0.186	0.143	0.186	0.256	0.176
Λάθος Περιστροφής Εικόνας 2	19.199	19.554	13.674	13.866	22.343
Λάθος Κλιμάκωσης Εικόνας 2	0.351	0.539	0.1	0.134	0.136
Λάθος Περιστροφής Εικόνας 3	23.699	16.641	27.219	17.463	29.556
Λάθος Κλιμάκωσης Εικόνας 3	0.285	0.609	0.154	0.348	0.207

Για την μέθοδο HOG , τα αποτελέσματα είναι σημαντικά χειρότερα, ειδικότερα στην ανίχνευση της περιστροφής , όπου η απόκλιση είναι της τάξης των  $10^{\omega\text{v}}$  μοιρών σε όλες τις μεθόδους. Η ανωτερότητα της μεθόδου SURF ως προς αυτόν τον τομέα οφείλεται στο γεγονός ότι πρώτα υπολογίζεται η γενική κατεύθυνση της γειτονιάς , ώστε να

εξάγονται περιστροφικά ανεξάρτητοι περιγραφητές, ενώ για την HOG υπάρχει μεγάλη εξάρτηση από την περιστροφή.

### 3.2.4

Για την έτοιμη συνάρτηση σχηματισμού Bag of Visual Words, εκπαιδεύσαμε το νευρωνικό δίκτυο χρησιμοποιώντας τον SVM Classifier που δίνεται στα δεδομένα.

Τα αποτελέσματα για την ακρίβεια της κάθε μεθόδου συνοψίζονται στο παρακάτω πίνακάκι:

	Corner Multiscale	Blob Multiscale	Blob Integral Image Multiscale
<b>SURF</b>	59.172%	58.621%	55.172%
<b>HOG</b>	62.069%	64.000%	61.241%

Παρατηρούμε ότι για την κατηγοριοποίηση των εικόνων, η μέθοδος HOG παρουσιάζει καλύτερα αποτελέσματα από την μέθοδο SURF. Οπότε, τελικά, όσον αφορά το ταίριασμα των εικόνων, η καλύτερη μέθοδος αποδείχθηκε ότι ήταν η Πολυκλιμακωτή Ανίχνευση Blob με περιγραφητή SURF, ενώ για την κατηγοριοποίηση η Πολυκλιμακωτή Ανίχνευση Blob με περιγραφητή HOG.

Αντίστοιχα, με την δική μας υλοποίηση του **Bag of Visual Words**, τα αποτελέσματα που παίρνουμε για το τελικό accuracy του SVM Classifier, είναι τα εξής:

	Corner Multiscale	Blob Multiscale	Blob Integral Image Multiscale
<b>SURF</b>	59.034%	59.034%	53.379%
<b>HOG</b>	63.586%	65.793%	59.724%