Εργασία 1 στην

Ψηφιακή επεξεργασία εικόνας

Σημειακός μετασχηματισμός και διόρθωση ιστογράμματος

Χρυσούλα Μόσχου
14/5/2020
moschouc@ece.auth.gr
AEM 9045

Εισαγωγή

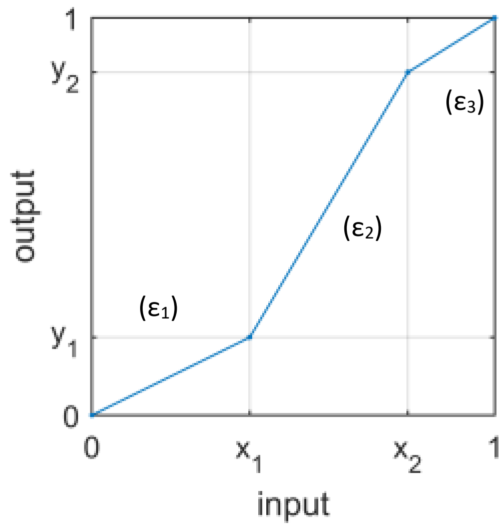
Η παρούσα εργασία αφορά στο πρώτο κομμάτι τον σημειακό μετασχηματισμό μιας εικόνας και στο δεύτερο τον μετασχηματισμό της με σκοπό την βελτίωση ιστογράμματος. Παρακάτω παρουσιάζεται η λογική με την οποία προσεγγίστηκε σε κάθε ενότητα και το πως έγινε η υλοποίηση της στο πρόγραμμα matlab. Η εργασία πέρα από αυτή την αναφορά συνοδεύεται και από τα εξής αρχεία:

- editHistogram.m (Ενότητα 2.3)
- pointtransfom.m (Ενότητα 1)
- histtransform.m (Ενότητα 2.1 και ενότητα 2.3)
- pdf2hist.m (Ενότητα 2.2 και ενότητα 2.3)
- partFirst.m (Ενότητα 1)
- partSecond.m (Ενότητα 2.1)

Τέλος, θα πρέπει να σημειωθεί ότι σε κάποιες ενότητες ελέγχονται διάφορες περιπτώσεις. Όπου αυτό είναι εφικτό οι διαφορετικές περιπτώσεις που παρουσιάζονται παρακάτω περιέχονται στον κώδικα με τη μορφή σχολίων. Κάθε φορά που εκτελείται ένα αρχείο μπορεί να γίνει αποεπιλογή του αντίστοιχου σχολίου για να ελεγχθεί η συγκεκριμένη περίπτωση. Σε διαφορετική περίπτωση θα σημειώνονται και θα δίνονται οι αλλαγές στα δεδομένα των μεταβλητών.

1. Σημειακός μετασχηματισμός

Σε αυτή την ενότητα ζητείται η υλοποίηση ενός σημειακού μετασχηματισμού εικόνας σύμφωνα με το παρακάτω σχήμα:



Σχήμα 1. Σημειακός μετασχηματισμός για την ενότητα 1 (γενική περίπτωση)

Ο σημειακός μετασχηματισμός δέχεται ως είσοδο μια εικόνα X υπολογίζει τις τιμές της εικόνας εξόδου Y όπως ορίζουν οι ευθείες του σχήματος. Τιμές φωτεινότητας στο διάστημα $[0,x_1)$ υπολογίζονται από την (ε_1), στο διάστημα $[x_1,x_2)$ από την (ε_2) και στο υπόλοιπο διάστημα $[x_2,1]$ από την (ε_3). Σημειώνεται ότι έχουμε φροντίσει να ανάγουμε όλες τις τιμές φωτεινότητας της εικόνα X στο διάστημα [0,1].

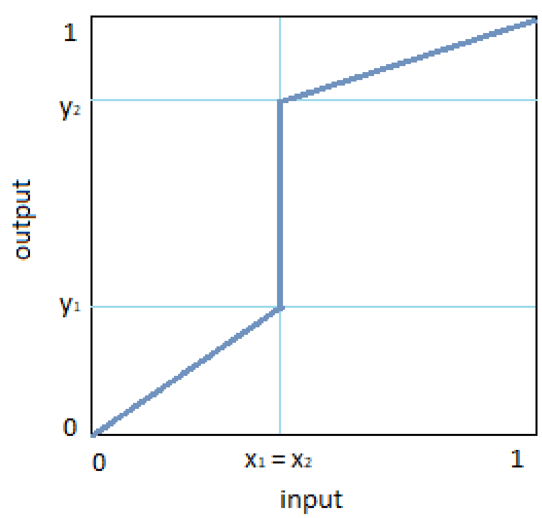
Για κάθε ευθεία γνωρίζουμε τουλάχιστον 2 σημεία από τα οποία διέρχεται οπότε μπορούμε να βρούμε τις εξισώσεις που τις περιγράφουν:

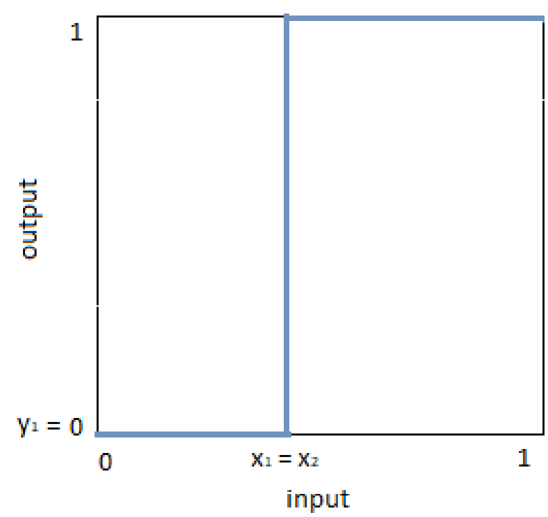
- Ευθεία 1 → διέρχεται από τα (0,0) και (x_1,y_1)
$$(\boldsymbol{\varepsilon_1}): \boldsymbol{y = a_1 x}, \text{ό}\boldsymbol{\pi o \upsilon}\ \boldsymbol{a_1 = \frac{y_1}{x_1}}$$
- Ευθεία 2 → διέρχεται από τα (x_1,y_1) και (x_2,y_2)
$$(\boldsymbol{\varepsilon_2}): \boldsymbol{y = a_2 x + b_2}, \text{ό}\boldsymbol{\pi o \upsilon}\ \boldsymbol{a_2 = \frac{y_1 - y_2}{x_1 - x_2}}\ \ \boldsymbol{\kappa\alpha\iota}\ \boldsymbol{b_2 = \frac{x_1 * y_2 - x_2 * y_1}{x_1 - x_2}}$$
- Ευθεία 3 → διέρχεται από τα (x_2,y_2) και (1,1)
$$(\boldsymbol{\varepsilon_3}): \boldsymbol{y = a_3 x + b_3}, \text{ό}\boldsymbol{\pi o \upsilon}\ \boldsymbol{a_3 = \frac{y_2 - 1}{x_2 - 1}}\ \ \boldsymbol{\kappa\alpha\iota}\ \boldsymbol{b_3 = \frac{x_2 - y_2}{x_2 - 1}}$$

Ανάλογα με τις τιμές των x_1,x_2 ,y_1 και y_2 είναι πιθανό οι ευθείες να έχουν άλλη μορφή. Για παράδειγμα στις περιπτώσεις που $x_1 = 0$, $x_1 - x_2 = 0$ ή $x_2 - 1 = 0$ οι παραπάνω εξισώσεις δεν ορίζονται. Στα πλαίσια της εργασίας μας ενδιαφέρει η γενική περίπτωση που δεν μηδενίζονται οι παρονομαστές και η περίπτωση της κατωφλοίωσης.

Συγκεκριμένα για την περίπτωση κατωφλοίωσης ισχύει ότι $x_1 = x_2$, $y_1 = 0$ και $y_2 = 1$. Η περίπτωση αυτή καλύπτεται από την γενική περίπτωση όπου $x_1 = x_2$ από τις παρακάτω εξισώσεις :

- Διάστημα $[0,x_1) \equiv [0,x_2)$
  Ευθεία 1 → διέρχεται από τα (0,0) και (x_1,y_1)
  $$(\boldsymbol{\varepsilon_1}): \boldsymbol{y = a_1 x}, \text{ό}\boldsymbol{\pi o\upsilon}\ \boldsymbol{a_1} = \frac{\boldsymbol{y_1}}{\boldsymbol{x_1}}$$
- Διάστημα $[x_2,1] \equiv [x_1,1]$
  Ευθεία 3 → διέρχεται από τα (x_2,y_2) και (1,1)
  $$(\boldsymbol{\varepsilon_3}): \boldsymbol{y = a_3 x + b_3}, \text{ό}\boldsymbol{\pi o\upsilon}\ \boldsymbol{a_3} = \frac{\boldsymbol{y_2 - 1}}{\boldsymbol{x_2 - 1}}\ \ \boldsymbol{\kappa\alpha\iota}\ \boldsymbol{b_3} = \frac{\boldsymbol{x_2 - y_2}}{\boldsymbol{x_2 - 1}}$$





Σχήμα 2. (Αριστερά) περίπτωση όπου x1=x2 , (Δεξιά) περίπτωση κατωφλοίωσης

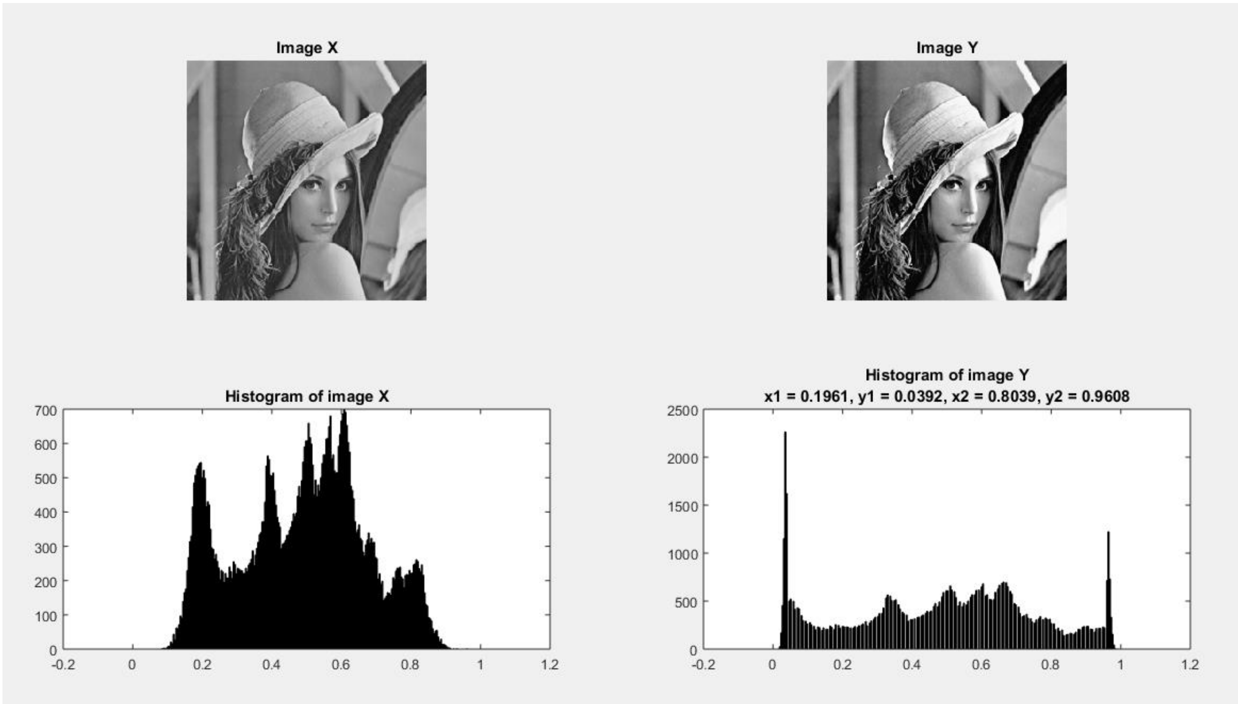
Οι παραπάνω εξισώσεις ευθειών για τον σημειακό μετασχηματισμό υλοποιούνται στο αρχείο pointtransform.m . Η συνάρτηση pointtransform δέχεται σαν ορίσματα την εικόνα X, τα x_1,x_2 ,y_1 και y_2 και δημιουργεί την εικόνα Y από τον παραπάνω μετασχηματισμό της εικόνας X. Στο αρχείο η περίπτωση της κατωφλοίωσης (binarization case) υλοποιείται σύμφωνα με το σχήμα 2 (αριστερά) και με κατάλληλη επιλογή των y_1 και y_2.

Επομένως ο κώδικας της συνάρτησης pointtransform λειτουργεί σε γενικά πλαίσια ως εξής:

1. Συγκρίνει τα x_1 και x_2 για να επιλέξει την γενική ή την περίπτωση της κατωφλοίωσης
2. Μετασχηματίζει τις τιμές των x σε y ανάλογα με το διάστημα στο οποίο ανήκουν και την εξίσωση ευθείας με την οποία συνδέονται.
3. Προβάλει τις εικόνες x και y καθώς και τα αντίστοιχα ιστογράμματα τους για σύγκριση.

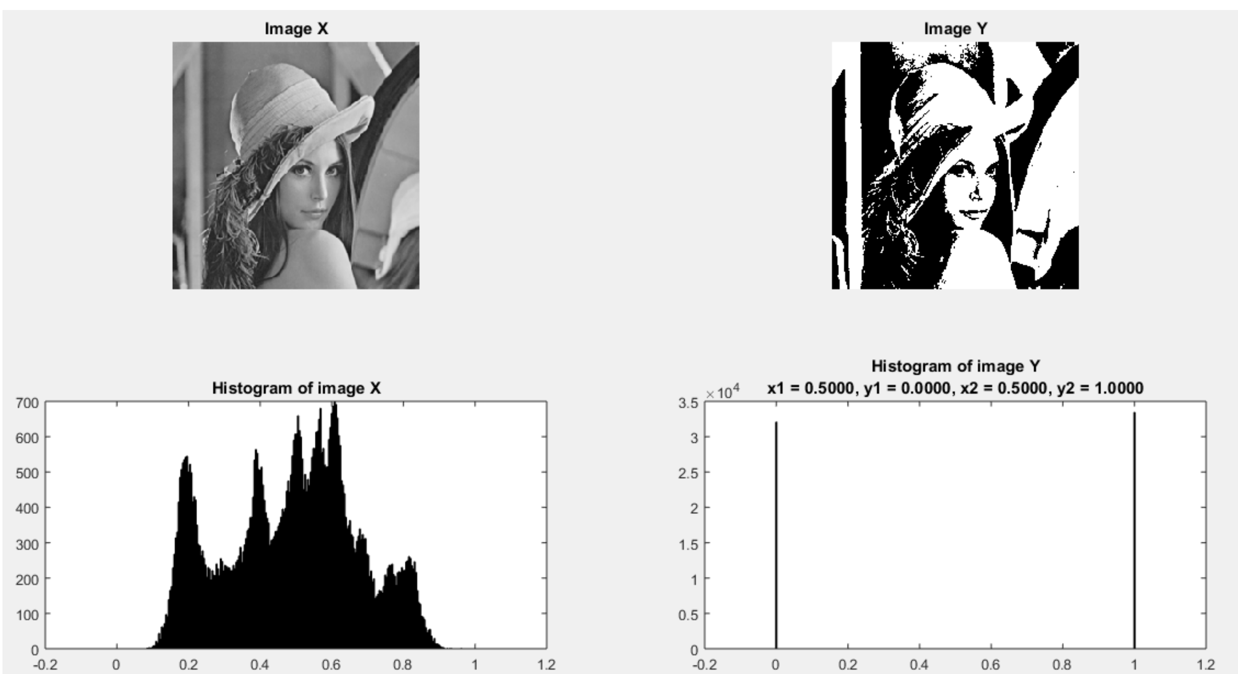
Στο script partFirst.m επιλέγεται η εικόνα X, μετατρέπεται σε gray-scale εικόνα ,καθορίζονται τα x_1,x_2 ,y_1 και y_2 και καλείται η συνάρτηση pointtransform(X, x_1,y_1 ,x_2 ,y_2) . Τα αποτελέσματα από δύο διαφορετικές περιπτώσεις για την εικόνα lena.bmp παρουσιάζονται στα παρακάτω σχήματα :

A. [x1, y1, x2, y2] = [0.1961, 0.0392, 0.8039, 0.9608]



Σχήμα 3. Σημειακός μετασχηματισμός εικόνας X περίπτωση (α)

B. Κατωφλοίωση στην τιμή φωτεινότητας 0.5 → [x1, y1, x2, y2] = [0.5, 0, 0.5, 1]



Σχήμα 4. Σημειακός μετασχηματισμός εικόνας X περίπτωση (β)

2.1 Μετασχηματισμός με βάση το ιστόγραμμα

Στην ενότητα αυτή θα πρέπει να υλοποιηθεί η συνάρτηση histtransform η οποία θα δέχεται σαν ορίσματα μια εικόνα X και κάποια στοιχεία για ένα ιστόγραμμα με σκοπό το ιστόγραμμα της μετασχηματισμένης εικόνας Y να προσεγγίζει το δοσμένο. Το δοσμένο ιστόγραμμα θα περιγράφεται από τα διανύσματα h και v μήκους n όπου n ο αριθμός των bins του ιστογράμματος. Το διάνυσμα v περιέχει τις τιμές φωτεινότητας σε αύξουσα σειρά και το h τα αντίστοιχα ποσοστά εμφάνισης τους.

Η υλοποίηση της histtransform ακολουθεί ένα greedy αλγόριθμο, επομένως είναι αναμενόμενο να υπάρχει κάποια απόκλιση στο ιστόγραμμα της εικόνας Y από το δοσμένο ιστόγραμμα.

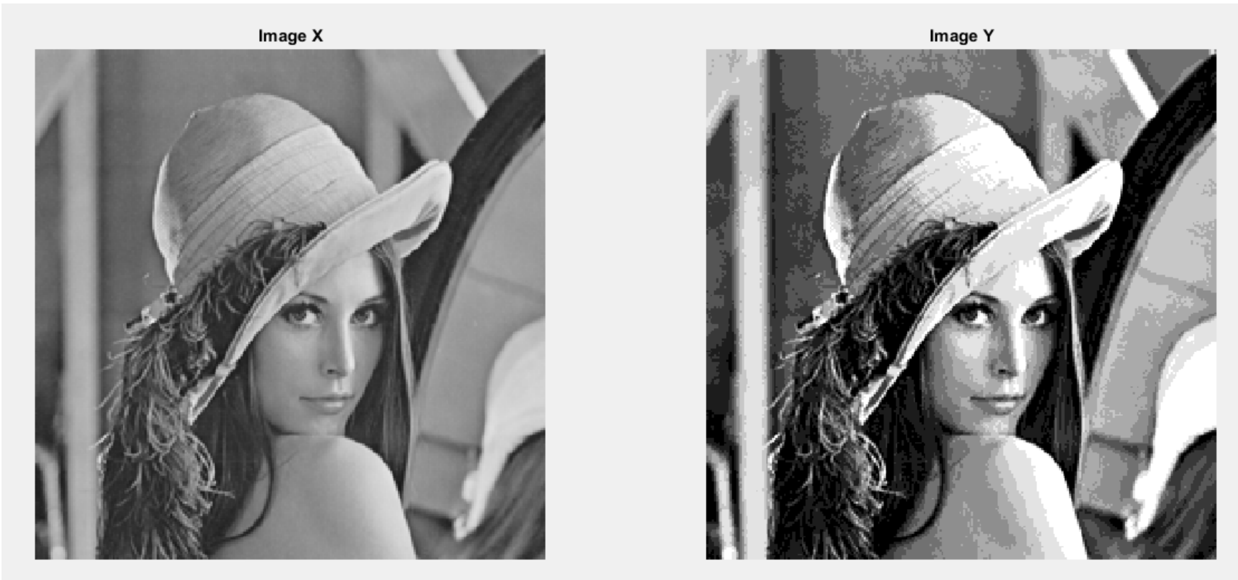
Ο κώδικας για αυτή την ενότητα βρίσκεται στο αρχείο histtransform.m. Η λογική που ακολουθείται είναι η εξής:

1. Η εικόνα X αποθηκεύεται προσωρινά σε μια εικόνα Z.
2. Για μια στάθμη φωτεινότητας k από τις n εντοπίζεται η ελάχιστη τιμή φωτεινότητας της εικόνας X. Ελέγχεται αν ισχύει η συνθήκη
$$\frac{αριθμός\ των\ pixels\ που\ έχουν\ ανατεθεί\ στην\ τιμή\ v(k)}{συνολικός\ αριθμός\ pixels\ της\ εικόνας} < h(k)$$
   Εάν ισχύει, εντοπίζονται όλα τα pixels που έχουν τιμή ίση με την ελάχιστη στην εικόνα X και ανατίθεται η τιμή v(k) στα αντίστοιχα pixel στην εικόνα Y. Στη συνέχεια για να μαρκάρουμε τα pixel της εικόνα X που έχουν μετασχηματιστεί τους δίνουμε προσωρινά μια αρκετά μεγάλη τιμή (π.χ. 10) ώστε να μην εντοπιστούν ξανά σαν ελάχιστα. Επίσης αυξάνεται αντίστοιχα και ο αριθμός των pixels που έχουν ανατεθεί στην τιμή v(k). Εντοπίζεται εκ νέου η ελάχιστη τιμή φωτεινότητας της εικόνας X και επανελέγχεται η συνθήκη.
   Εάν η συνθήκη δεν ισχύει, προχωράμε στην διαδοχικά επόμενη στάθμη φωτεινότητας και επαναλαμβάνουμε την παραπάνω διαδικασία.
3. Εάν το ελάχιστο που εντοπιστεί ισούται με την τιμή 10 η διαδικασία σταματάει καθώς θα έχουν μετασχηματιστεί όλα τα στοιχεία της εικόνα X.
4. Μέσω της προσωρινής εικόνας Z, η εικόνα X επανέρχεται στις αρχικές της τιμές για να προβληθεί και να συγκριθεί με την μετασχηματισμένη εικόνα Y.
5. Ο κάθετος άξονας του ιστογράμματος της εικόνα Y κανονικοποιείται διαιρώντας όλες τις τιμές του με τον συνολικό αριθμό των pixels για να είναι πλέον σε ποσοστά %. Έτσι προβάλλοντας το δοσμένο ιστόγραμμα στο ίδιο διάγραμμα με το κανονικοποιημένο ιστόγραμμα της Y μπορούμε να τα συγκρίνουμε και να ελέγξουμε κατά πόσο προσεγγίζεται το ζητούμενο.

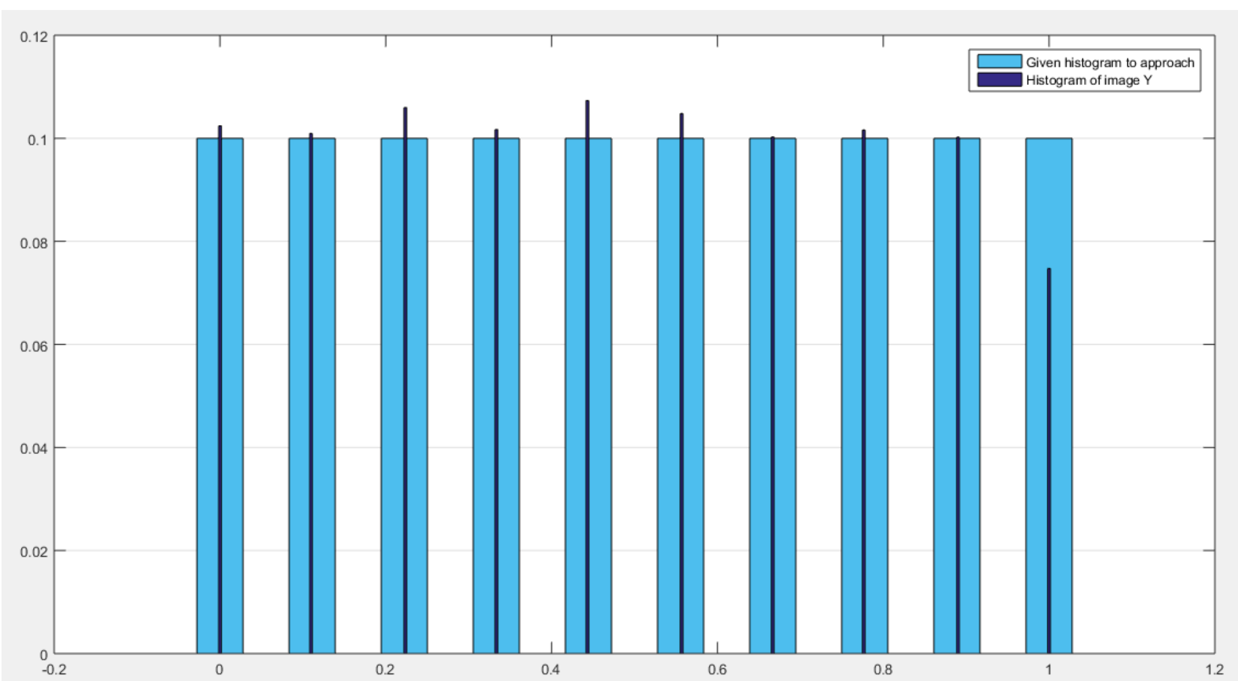
Παρακάτω παρουσιάζονται τα αποτελέσματα για τις 3 περιπτώσεις που μας ζητούνται.

Στο script partSecond.m επιλέγεται η εικόνα X, μετατρέπεται σε gray-scale εικόνα ,καθορίζονται τα v και h και καλείται η συνάρτηση histtransform(X, h, v). Τα αποτελέσματα από τρεις διαφορετικές περιπτώσεις που μας ζητούνται για την εικόνα lena.bmp παρουσιάζονται στα παρακάτω σχήματα :

Case .1 L = 10 , v = linspace(0, 1, L), h = ones([1, L]) / L

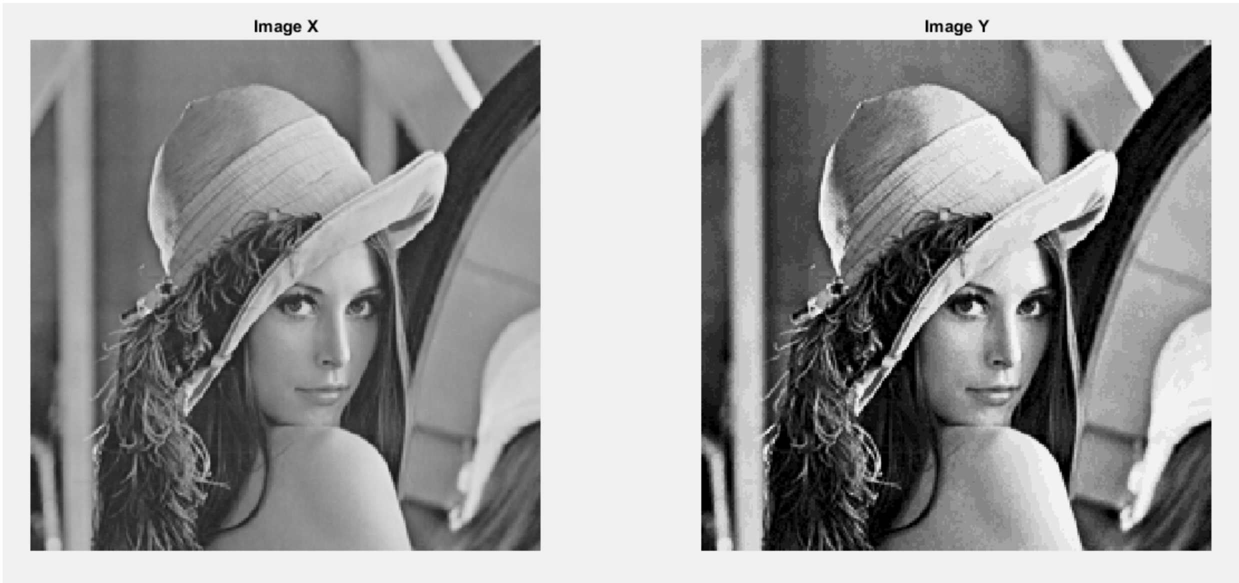


Σχήμα 5. Η εικόνα Y σε σύγκριση με την εικόνα X μετά τον μετασχηματισμό του case.1

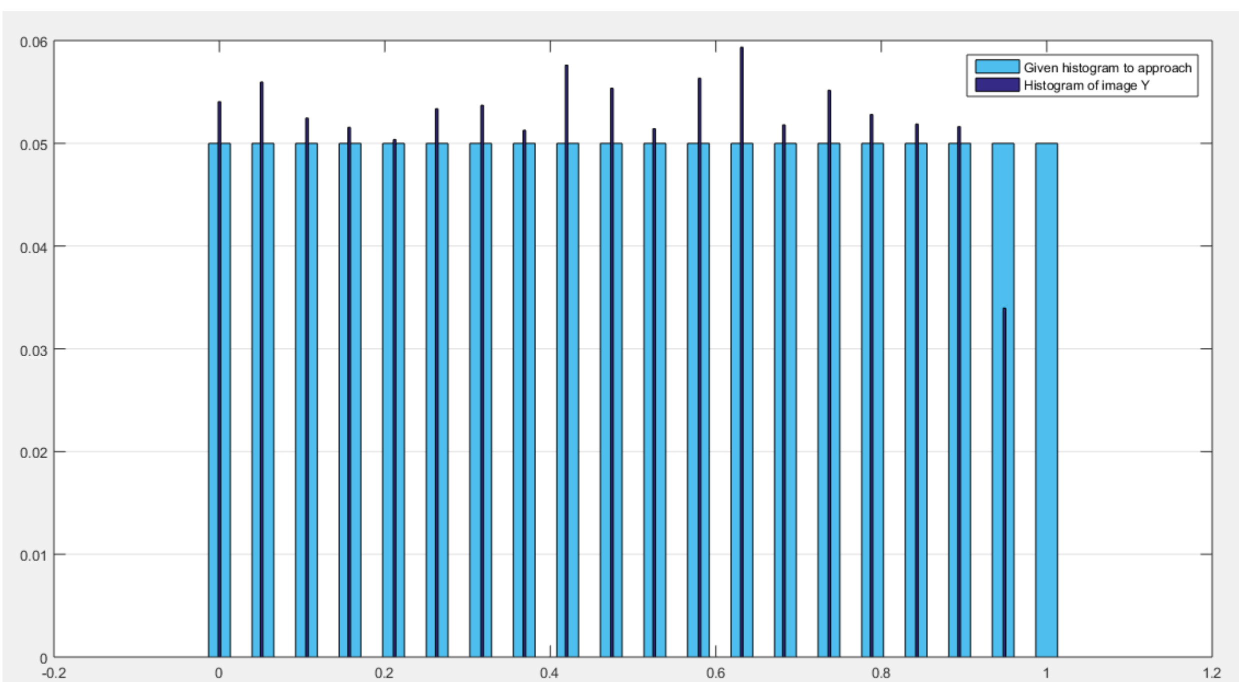


Σχήμα 6. Το ιστόγραμμα της εικόνας Y και το δοσμένο ιστόγραμμα που πρέπει να προσεγγίζει στο case.1

Case .2 L = 20 , v = linspace(0, 1, L), h = ones([1, L]) / L

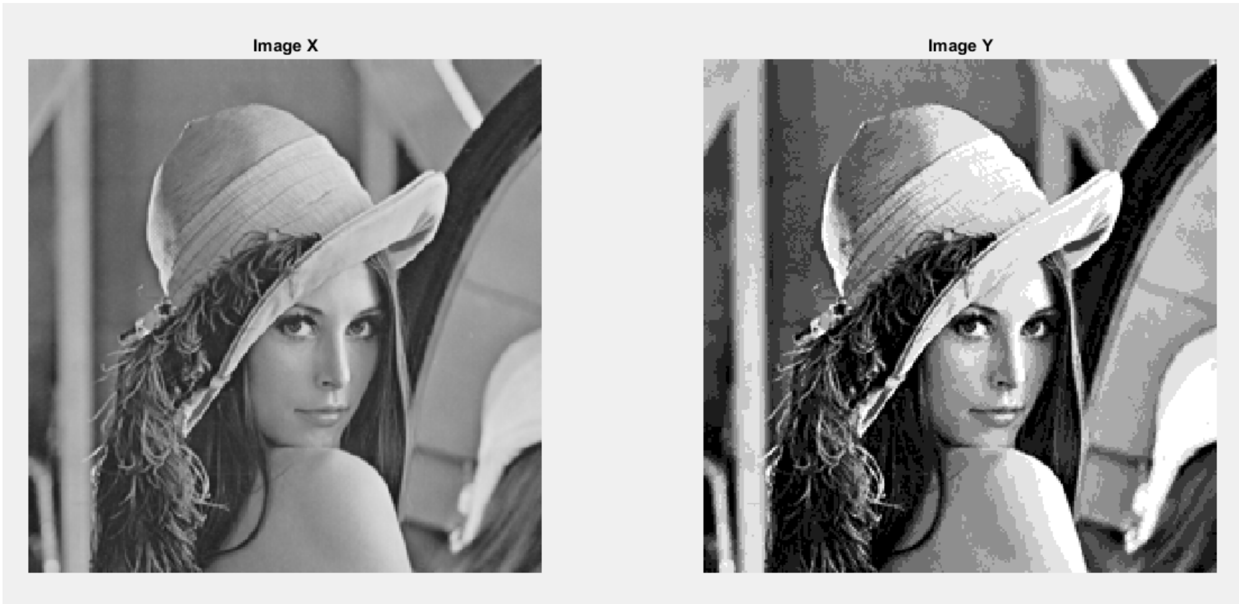


Σχήμα 7.Η εικόνα Y σε σύγκριση με την εικόνα X μετά τον μετασχηματισμό του case.2

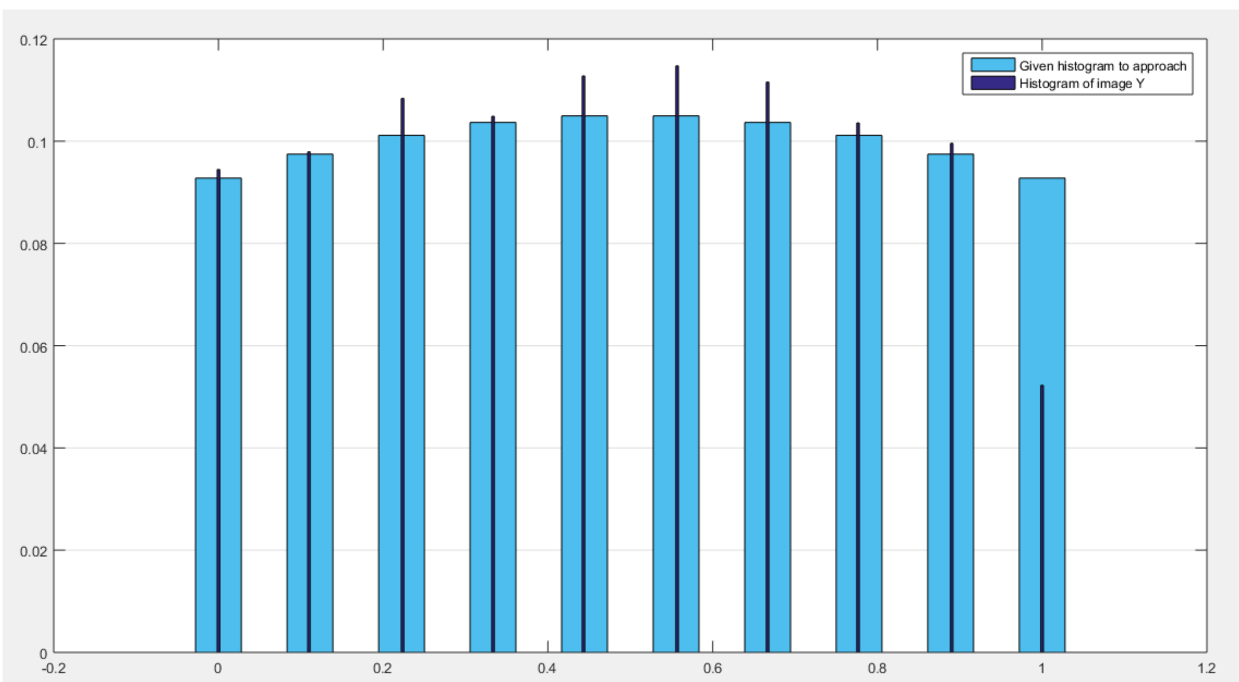


Σχήμα 8.Το ιστόγραμμα της εικόνας Y και το δοσμένο ιστόγραμμα που πρέπει να προσεγγίζει στο case.2

Case .3 L = 10 , v = linspace(0, 1, L), h = normpdf(v, 0.5) / sum(normpdf(v, 0.5))



Σχήμα 9. Η εικόνα Y σε σύγκριση με την εικόνα X μετά τον μετασχηματισμό του case.3



Σχήμα 10. Το ιστόγραμμα της εικόνας Y και το δοσμένο ιστόγραμμα που πρέπει να προσεγγίζει στο case.3

Σχόλια

Όλα τα ιστογράμματα όπως φαίνεται από τα αποτελέσματα παρουσιάζουν αποκλίσεις. Αυτό έχει να κάνει με την φύση του greedy αλγόριθμου ο οποίος δεν εξασφαλίζει ότι κάθε στάθμη φωτεινότητας θα περιέχει ακριβώς όσα Pixel θα έπρεπε. Μάλιστα όσο αυξάνεται το L ο αλγόριθμος εξαντλεί τον έλεγχο των pixel (case 2) πριν φτάσει στη τελευταία στάθμη με αποτέλεσμα αυτή να μην περιέχει κανένα Pixel. Επίσης κάθε στάθμη ξεπερνάει είτε λίγο (case 1) είτε πολύ (case 2) την σταθερή τιμή που θα έπρεπε να έχει καθιστώντας την κατανομή μη ομοιόμορφη. Για μεγαλύτερο L βέβαια έχουμε μεγαλύτερη ποικιλία αποχρώσεων για αυτό και δημιουργούνται περισσότερα bins στο ιστόγραμμα.

Το case 3 αν θεωρητικά λόγο της κανονικής κατανομής θα έπρεπε να παρουσιάζει περισσότερες αποχρώσεις κοντά στο 0.5 κάτι τέτοιο δεν είναι οπτικά εμφανές. Αυτό συμβαίνει λόγο της τυπικής απόκλισης (standard deviation) η οποία έχει την προεπιλεγμένη τιμή 1 (by-default) στη matlab δηλαδή είναι σχετικά μεγάλη. Στην ενότητα 2.3 που η τυπική απόκλιση είναι μικρή (0.1) το "γκριζάρισμα" της εικόνας είναι αρκετά πιο έντονο.

2.2 Εκτίμηση ιστογράμματος από κατανομή

Σε αυτή την ενότητα υλοποιείται η συνάρτηση pdf2hist στο αρχείο pdf2hist.m η οποία δημιουργεί στην έξοδο της ένα ιστόγραμμα h με βάση τα διαστήματα που δέχεται και την τιμή μιας συνάρτησης f σε αυτά τα διαστήματα. Πιο συγκεκριμένα δέχεται ένα διάνυσμα d μήκους n το οποίο ορίζει n-1 διαστήματα ως εξής:

$$\begin{gathered} 1^{o}\, διάστημα\ [d(1), d(2)] \\ 2^{o}\, διάστημα\ [d(2), d(3)] \\ \vdots \\ (n-1)^{o}\ διάστημα\ [d(n-1), d(n)] \end{gathered}$$

Το κάθε στοιχείο του h δημιουργείται ολοκληρώνοντας την f στο αντίστοιχο διάστημα. Συνεπώς το h που προκύπτει θα είναι μήκους n-1.

Για την υλοποίηση της αριθμητικής ολοκλήρωσης χρησιμοποιείται η σύνθετη μέθοδος τραπεζίου. Σύμφωνα με την μέθοδο τραπεζίου το ορισμένο ολοκλήρωμα μιας συνάρτησης εφόσον γνωρίζουμε τα όρια ολοκλήρωσης και τον τύπο της συνάρτησης προσεγγίζεται όπως φαίνεται παρακάτω :

$$\int_{a}^{b} f(x)\, dx \approx (b-a) * (\frac{f(a)+f(b)}{2})$$

Για μια πιο ακριβής προσέγγιση στο [a,b] μπορούμε να το σπάσουμε σε n μικρότερα ολοκληρώματα υπολογίζοντας και αυτά τα ολοκληρώματα με προσέγγιση. Προσθέτοντας αυτά τα μικρότερα ολοκληρώματα μεταξύ τους το αρχικό ολοκλήρωμα προσεγγίζεται με μεγαλύτερη ακρίβεια. Η μέθοδος αυτή είναι γνωστή ως σύνθετη μέθοδος τραπεζίου και η παραπάνω προσέγγιση μετατρέπεται σε αυτή :

$$\int_a^b f(x)\,dx \approx \frac{(b-a)}{n} * \left(\frac{f(a)}{2} + \sum_{k=1}^{n-1}\left(f\left(a + k * \frac{b-a}{n}\right)\right) + \frac{f(b)}{2}\right) \quad (1)$$

Είναι εμφανές ότι όσο μεγαλύτερο είναι το n τόσο καλύτερα προσεγγίζεται το ολοκλήρωμα. Στα πλαίσια της εργασίας το n επιλέγεται 100 καθώς είναι αρκετό για να εξασφαλίσει πολύ καλή προσέγγιση των συναρτήσεων της ενότητας 2.3. Σε κάθε περίπτωση μπορούμε να το αυξομειώσουμε ανάλογα τις ανάγκες ακρίβειας.

Συνοπτικά ο κώδικας της pdf2hist.m λειτουργεί με τον εξής τρόπο:

1. Ορίζεται η τιμή του n
2. Υπολογίζονται τα στοιχεία του h. Για κάθε στοιχείο h(i) ορίζονται τα α και b ως : $\alpha = d(i), b = d(i+1)$ και υπολογίζεται το εσωτερικό άθροισμα του τύπου (1) μέσω ενός επαναληπτικού βρόγχου (στον κώδικα subinterval). Τέλος υπολογίζεται το αρχικό ολοκλήρωμα από τον τύπο (1) το οποίο ισούται με το h(i) για το αντίστοιχο διάστημα. (Εναλλακτικά σε αυτό το σημείο μπορεί να χρησιμοποιηθεί και η έτοιμη συνάρτηση της matlab integral για αριθμητική ολοκλήρωση)
3. Επειδή το h θέλουμε να είναι κανονικοποιημένο, δηλαδή $\sum_{i=1}^{n} h(i) = 1$, ελέγχουμε το συνολικό άθροισμα των στοιχείων του h. Σε περίπτωση που δεν είναι ίσο με 1 διαιρούμε κάθε στοιχείο του h με το συνολικό άθροισμα για να ικανοποιήσουμε το ζητούμενο.

2.3 Μετασχηματισμός με βάση την πυκνότητα πιθανότητας

Σε αυτή την ενότητα αξιοποιούνται οι παραπάνω υλοποιήσεις της κατηγορίας 2 με σκοπό να μετασχηματιστεί μια εικόνα εισόδου X στην εικόνα Y της οποίας το ιστόγραμμα πρέπει να προσεγγίζει μια συγκεκριμένη κατανομή.

Η διαδικασία αυτή γίνεται στο αρχείο editHistogram.m από την αντίστοιχη συνάρτηση editHistogram. Τα βήματα είναι τα εξής:

1. Επιλογή της εικόνας (εδώ lena.bmp) και μετατροπή της σε gray-scale
2. Προσδιορισμός της συνάρτησης f , δηλαδή της κατανομής που θέλουμε να προσεγγίσουμε
3. Προσδιορισμός του d (όρια και διαστήματα)
4. Υπολογισμός του ιστογράμματος προς προσέγγιση καλώντας την pdf2hist(d, f)
5. Υπολογισμός του διανύσματος v που περιέχει τις τιμές φωτεινότητας ως το μέσο του κάθε διαστήματος που ορίζει το d
6. Μετασχηματισμός της εικόνα X σε εικόνα Y καλώντας τη συνάρτηση Y = histtransform(x, h, v)
7. Η εικόνα Y και το ιστόγραμμα της προβάλλονται μέσα στη συνάρτηση histtransform θεωρώντας ότι οι τιμές φωτεινότητας μετά τον μετασχηματισμό παραμένουν στο διάστημα [0,1]. Σε κάποιες περιπτώσεις (π.χ. εδώ case 2) που μπορεί αυτό να αλλάξει μπορεί να χρησιμοποιηθεί ο κώδικας στο τέλος της συνάρτησης edit_histogram για να προβληθεί σωστά η εικόνα Y και το ιστόγραμμα της.

Θα μελετηθούν τρία διαφορετικά cases για κάθε ένα από τα οποία παρουσιάζονται διαφορετικές εναλλακτικές για το διάστημα d με σκοπό να βρεθεί μια όσο το δυνατόν καλύτερη επιλογή.

Τα διαστήματα του d επιλέγονται ισομήκη και αυτό για να μπορέσει να δημιουργηθεί σωστά η ζητούμενη κατανομή από την pdf2hist. Αν τα διαστήματα στα οποία ολοκληρώνουμε είναι άνισα τότε δεν μπορεί να προσεγγιστεί η ομοιόμορφη ή η κανονική κατανομή καθώς δεν δημιουργείται σωστά η h. Ένα τέτοιο παράδειγμα παρουσιάζεται στο case 1.

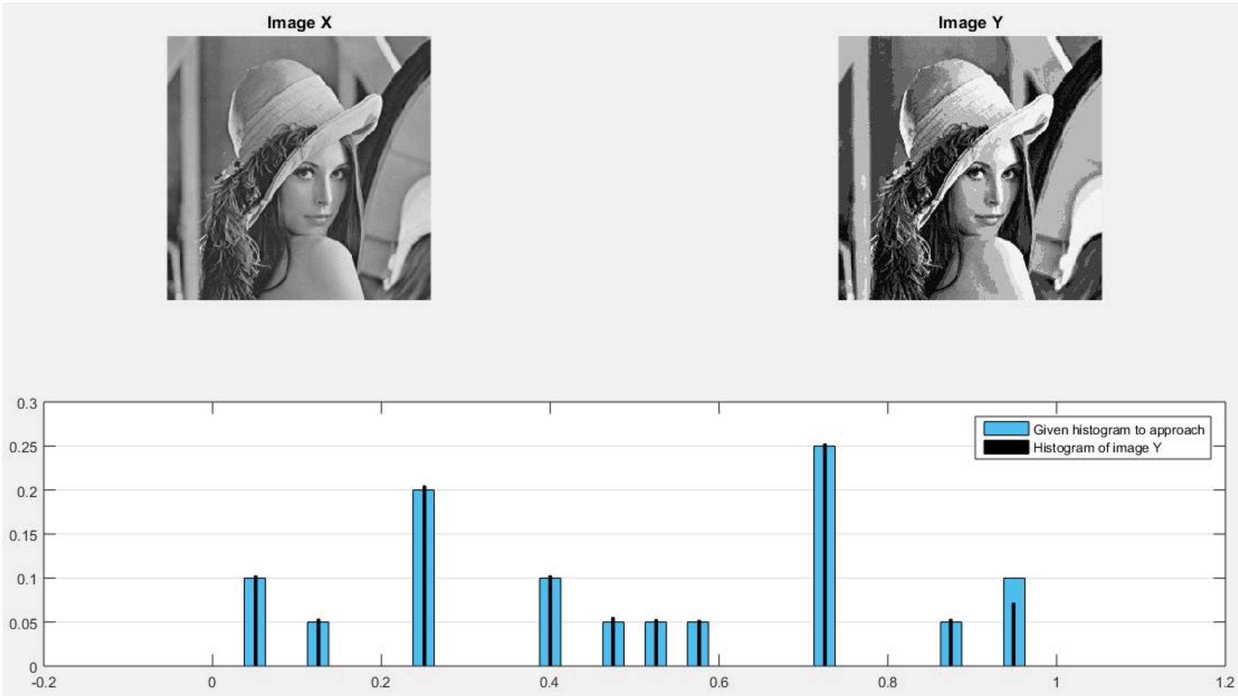
Επιπλέον για την επιλογή του d προτιμήθηκαν διαστήματα στα όρια των συναρτήσεων κατανομών για να μην δημιουργούνται bins με χαμηλή πιθανότητα εκτός του [0,1] και να προσεγγίζονται καλύτερα οι κατανομές. Ένα τέτοιο παράδειγμα παρουσιάζεται στο case 1.

Σε κάθε περίπτωση παρουσιάζονται οι εικόνες X,Y και το ιστόγραμμα της εικόνας Y στο ίδιο διάγραμμα με το επιθυμητό.

Case .1 Ομοιόμορφη κατανομή στο [0,1]

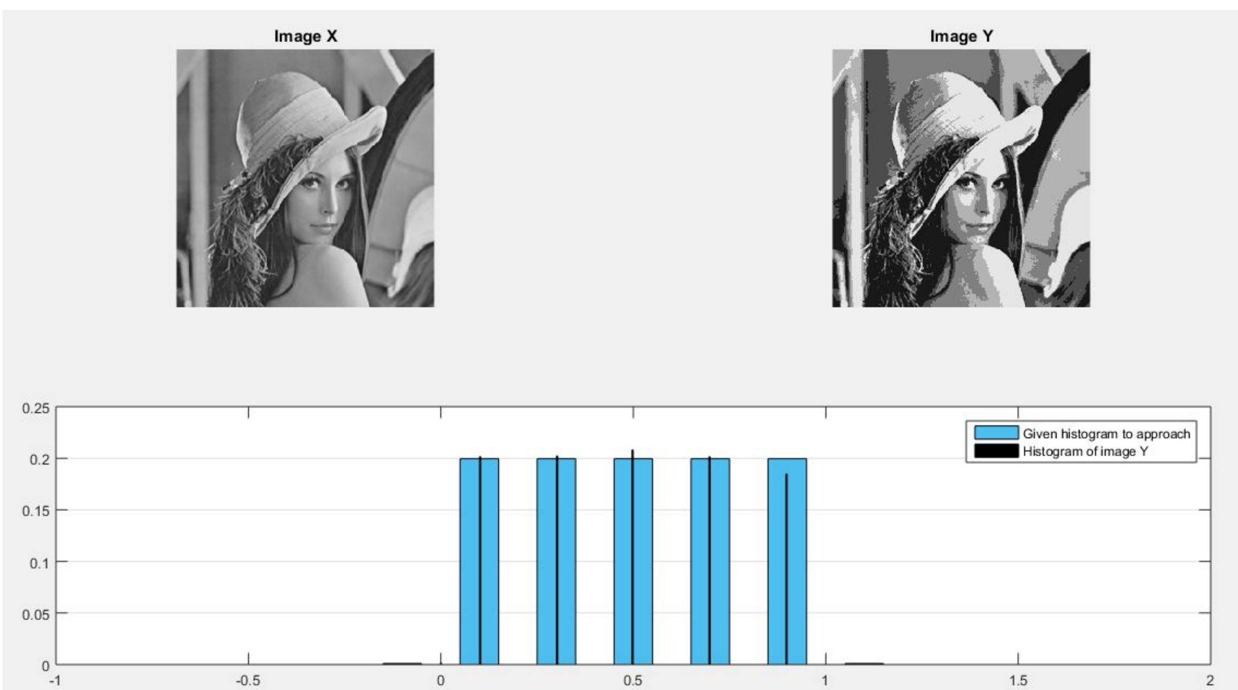
Για αυτή την περίπτωση βγάζουμε από σχόλιο τη γραμμή 10: f = @(x)unifpdf(x,0,1); (case 1) στο αρχείο editHistogram.m

- Τυχαία επιλογή των διαστημάτων, d = [0,0.1,0.15,0.35,0.45,0.5,0.55,0.6,0.85,0.9,1]



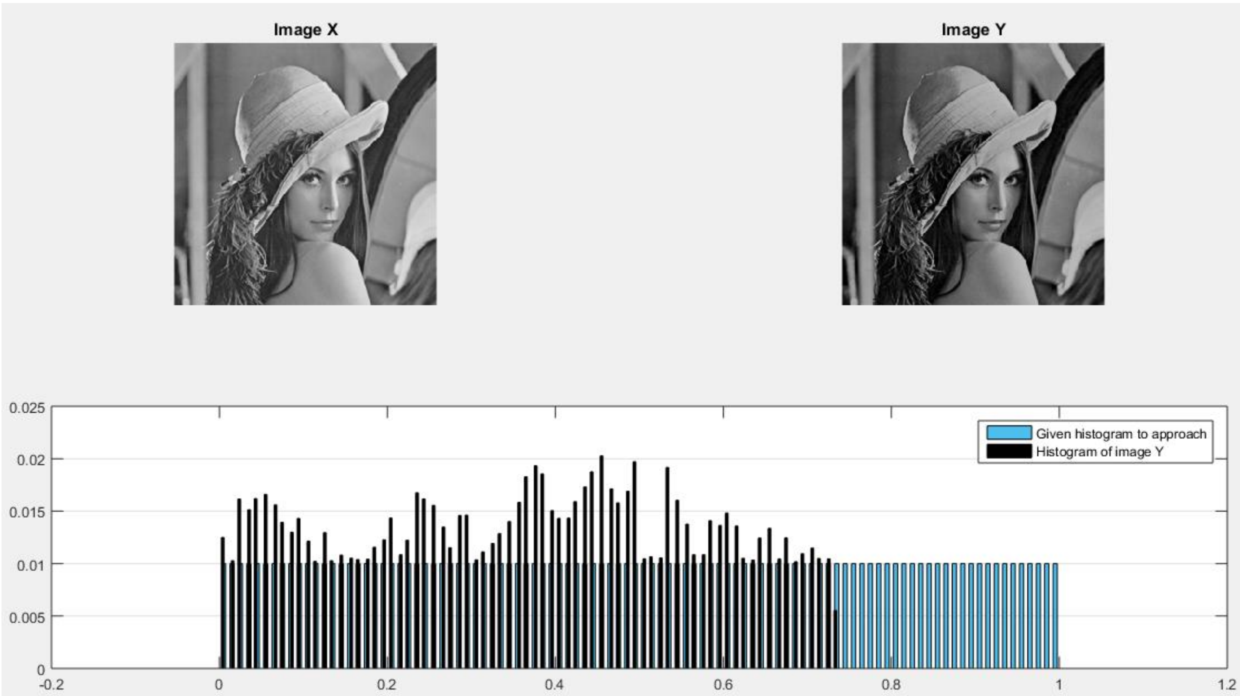
Σχήμα 11

- Επιλογή διαστήματος που ξεπερνάει τα όρια της συνάρτησης κατανομής, d = -1:0.2:2



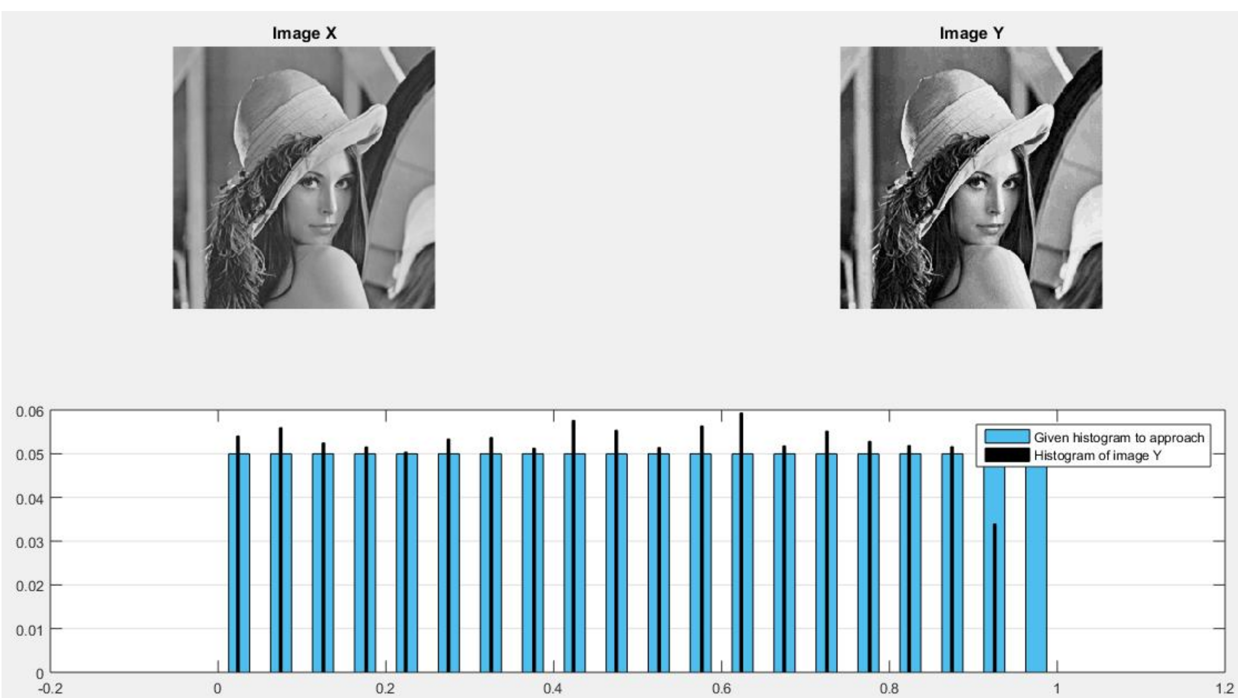
Σχήμα 12

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.01, d = 0:0.01:1



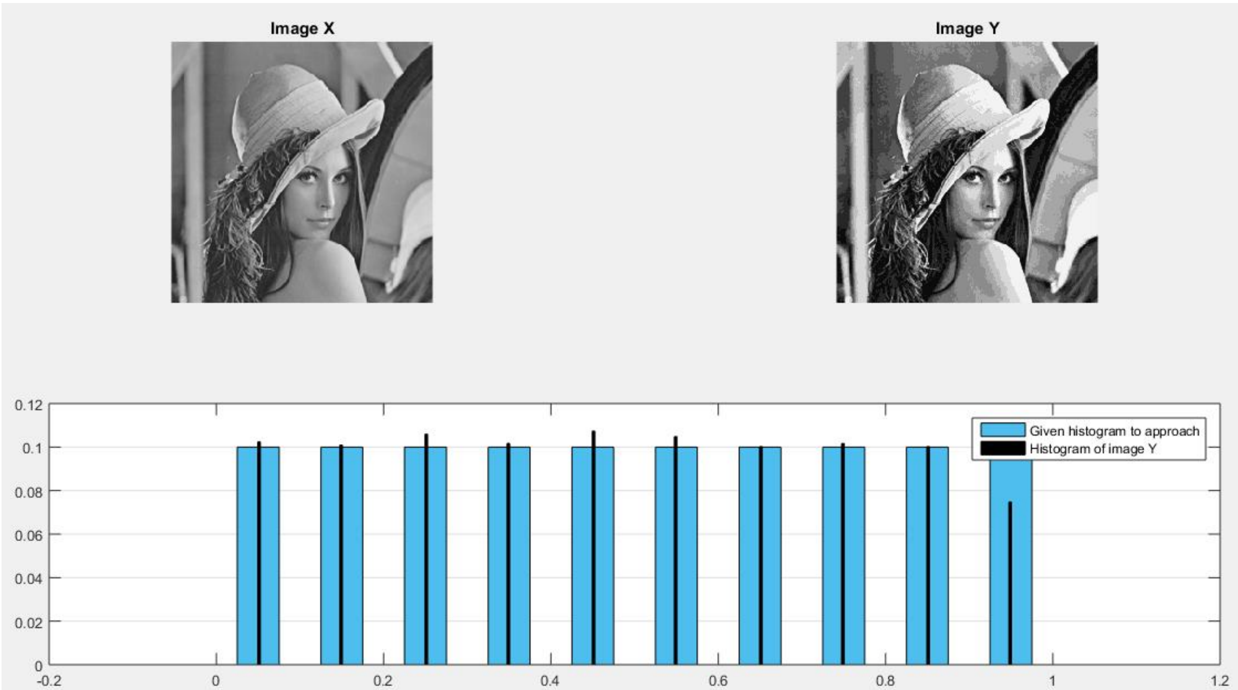
Σχήμα 13

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.05, d = 0:0.05:1



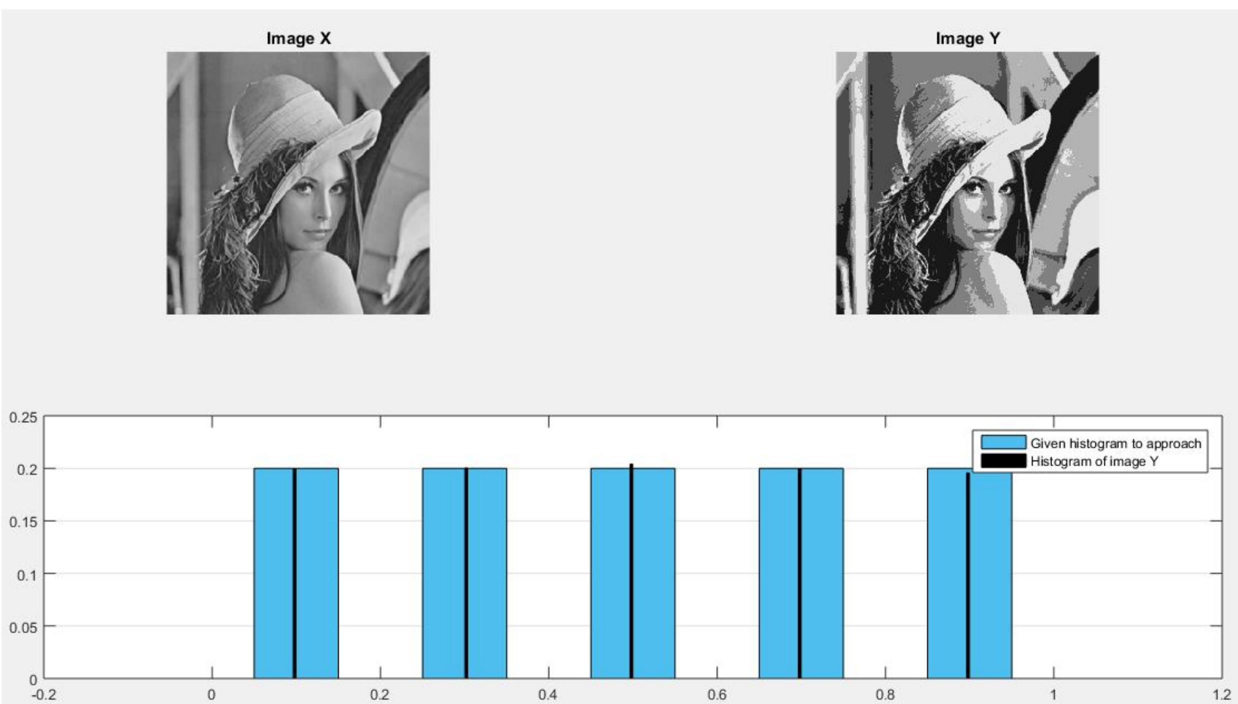
Σχήμα 14

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.1, d = 0:0.1:1



Σχήμα 15

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.2, d = 0:0.2:1



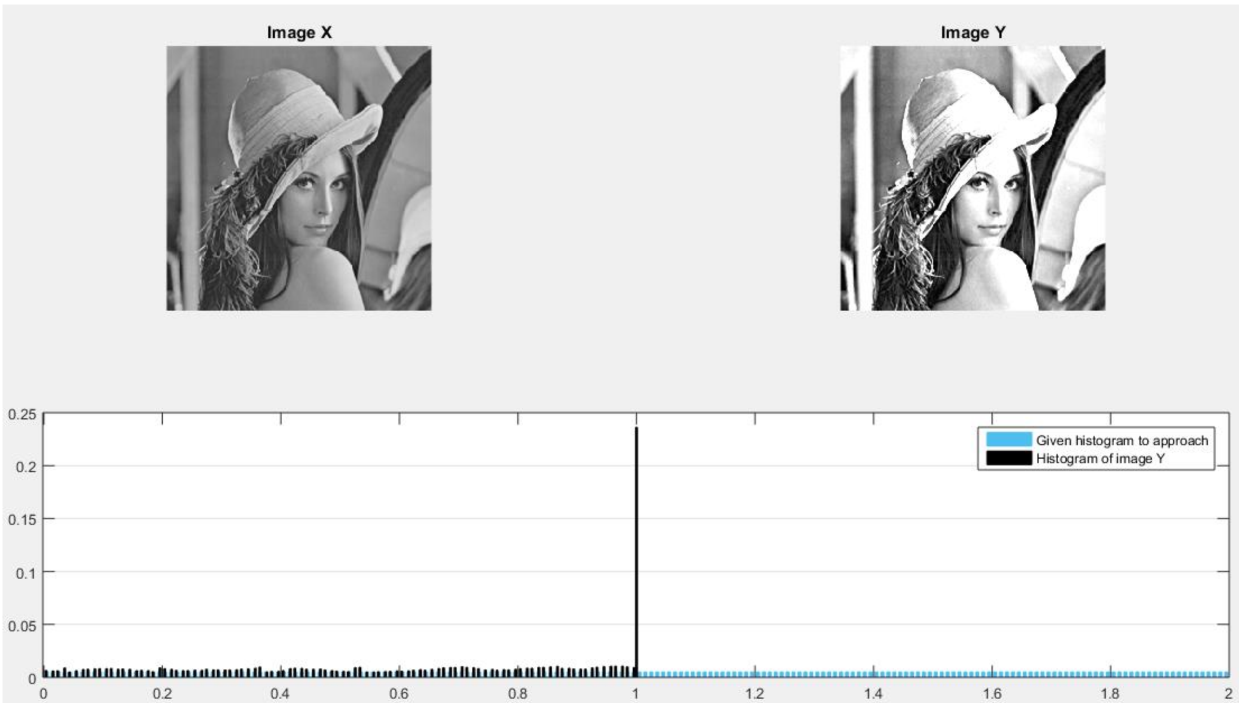
Σχήμα 16

Case .2 Ομοιόμορφη κατανομή στο [0,2]

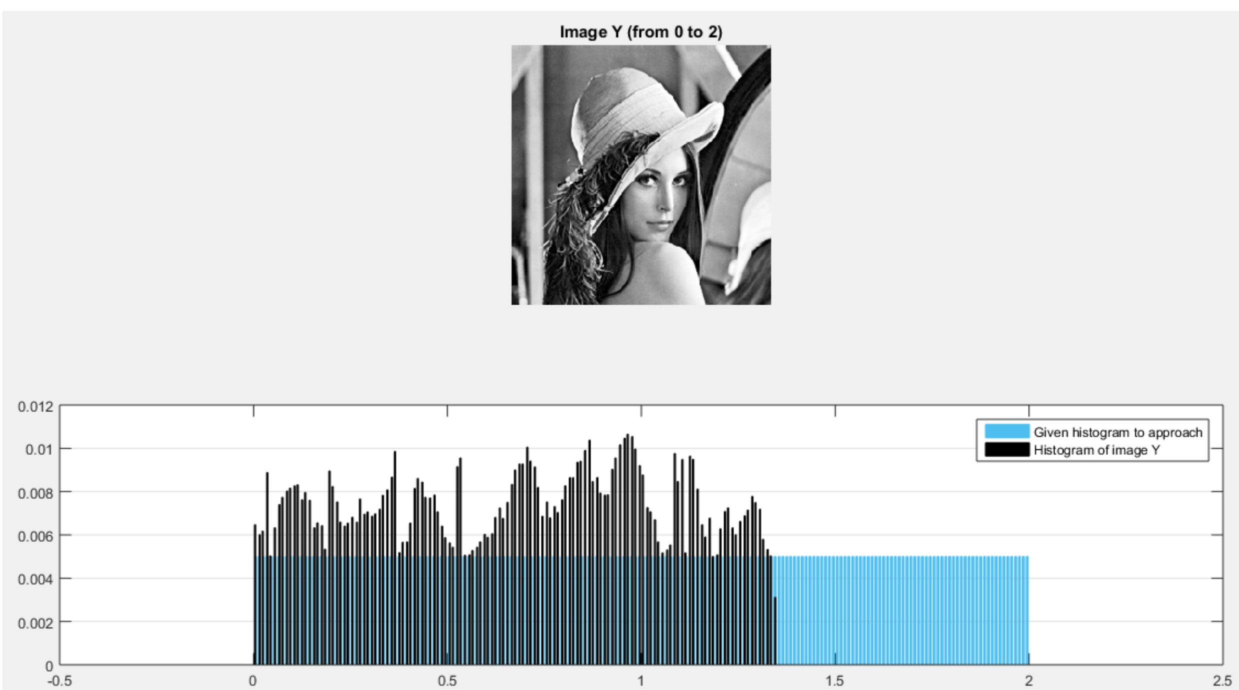
Στην περίπτωση αυτή παρατηρούμε ότι τα όρια της ομοιόμορφης κατανομής που θέλουμε να προσεγγίσουμε είναι εκτός του διαστήματος φωτεινότητας [0,1]. Αυτό θα έχει ως αποτέλεσμα η μετασχηματισμένη εικόνα Y να έχει τιμές φωτεινότητας μεγαλύτερες του 1. Επομένως αν εμφανίσουμε την εικόνα Y και το ιστόγραμμα της όπως την προηγούμενη περίπτωση θα παρατηρήσουμε συγκέντρωση όλων των τιμών μεγαλύτερων της μονάδας στο 1 (θα έχουμε κορεσμό). Για να εμφανίσουμε πλήρως την εικόνα Y και το ιστόγραμμα της και να μην χάσουμε πληροφορία βγάζουμε από τα σχόλια τον κώδικα του «extra section» στο τέλος του αρχείου edit_histogram.m. Έτσι πλέον το μαύρο θα συνεχίσει να αντιστοιχεί στο 0 αλλά το άσπρο θα αντιστοιχεί στο 2. Επίσης, για αυτή την περίπτωση βγάζουμε από σχόλιο τη γραμμή 12: f = @(x)unifpdf(x,0,2); (case 2) στο αρχείο editHistogram.m.

Παρακάτω για κάθε επιλογή του d θα παρουσιάζεται πρώτα η εικόνα Y με κορεσμό στο 1 και στη συνέχεια όπως είναι χωρίς κορεσμό.

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.01, d = 0:0.01:2

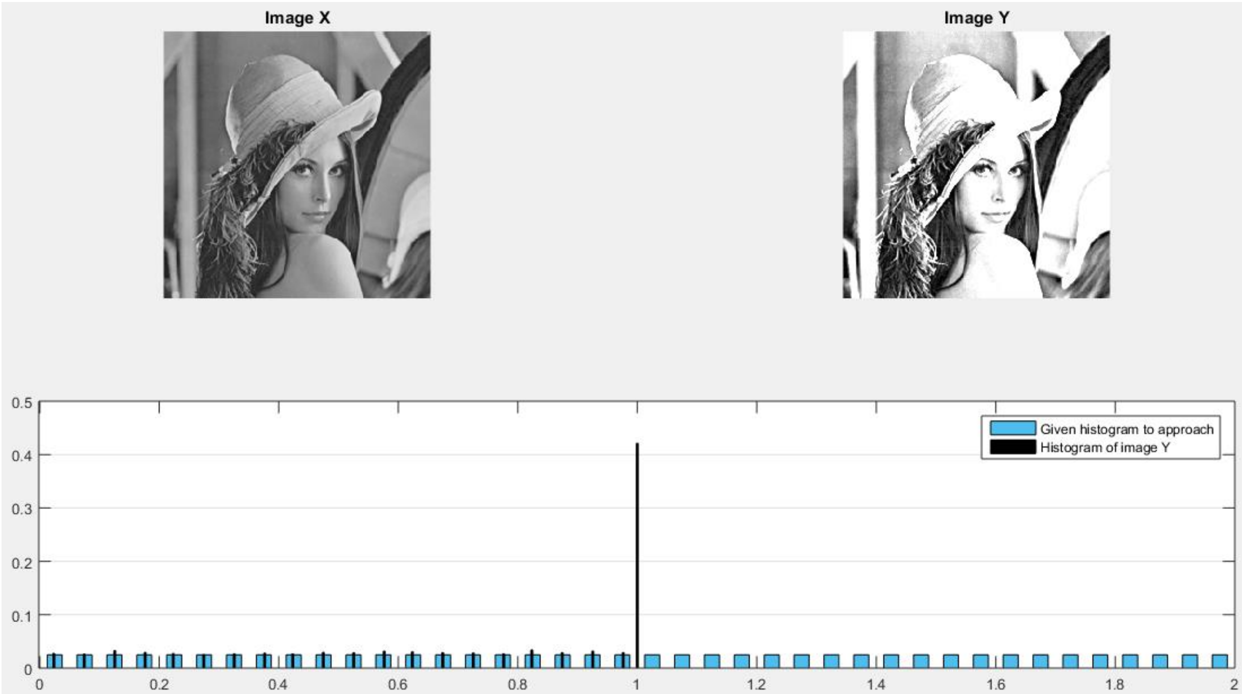


Σχήμα 17. Εικόνα με κορεσμό

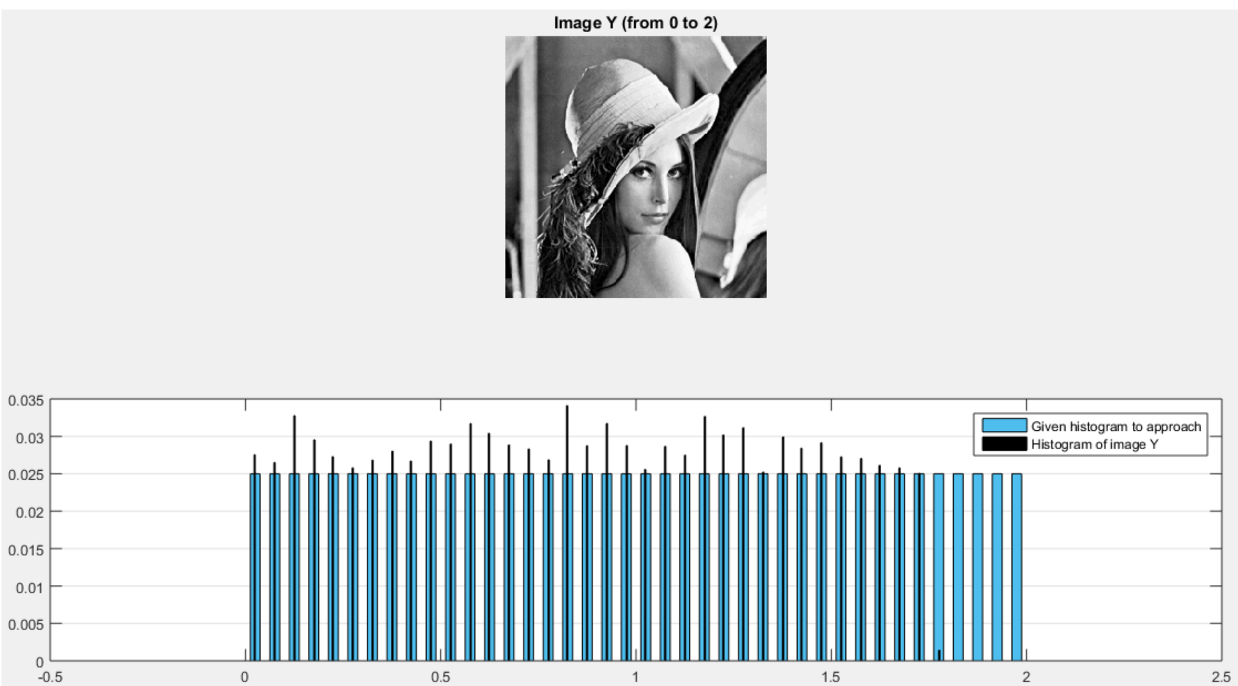


Σχήμα 18.Εικόνα χωρίς κορεσμό

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.2, d = 0:0.05:2

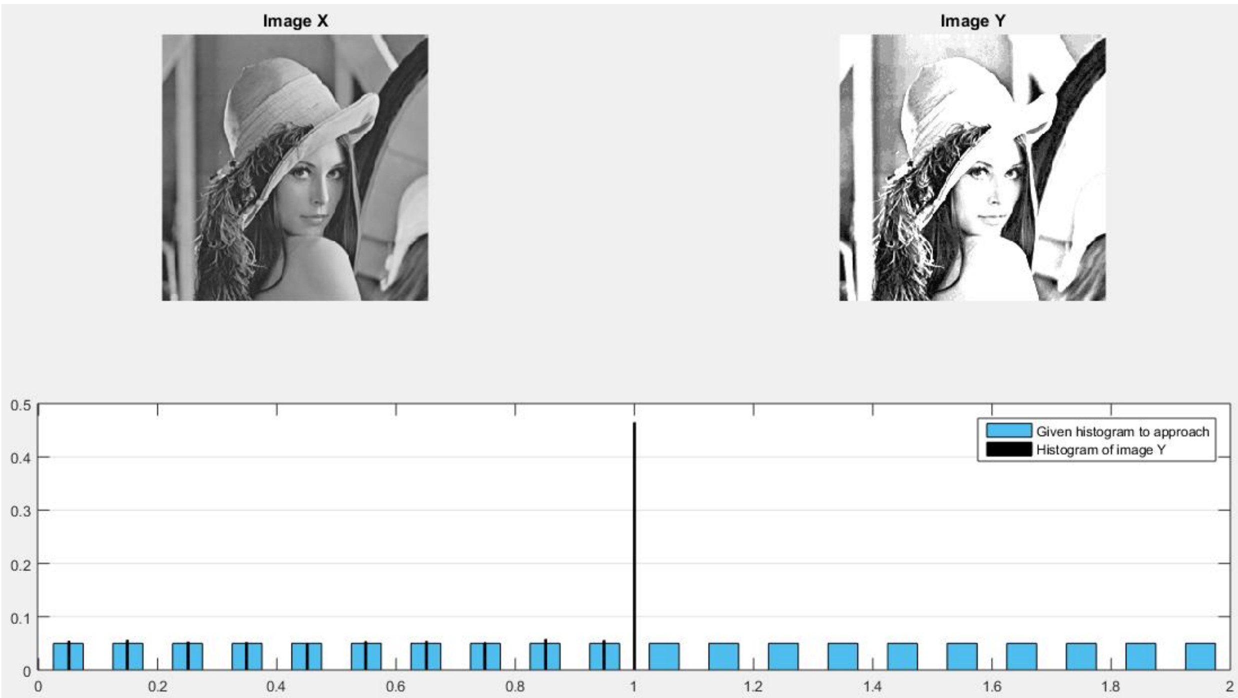


Σχήμα 19. Εικόνα με κορεσμό

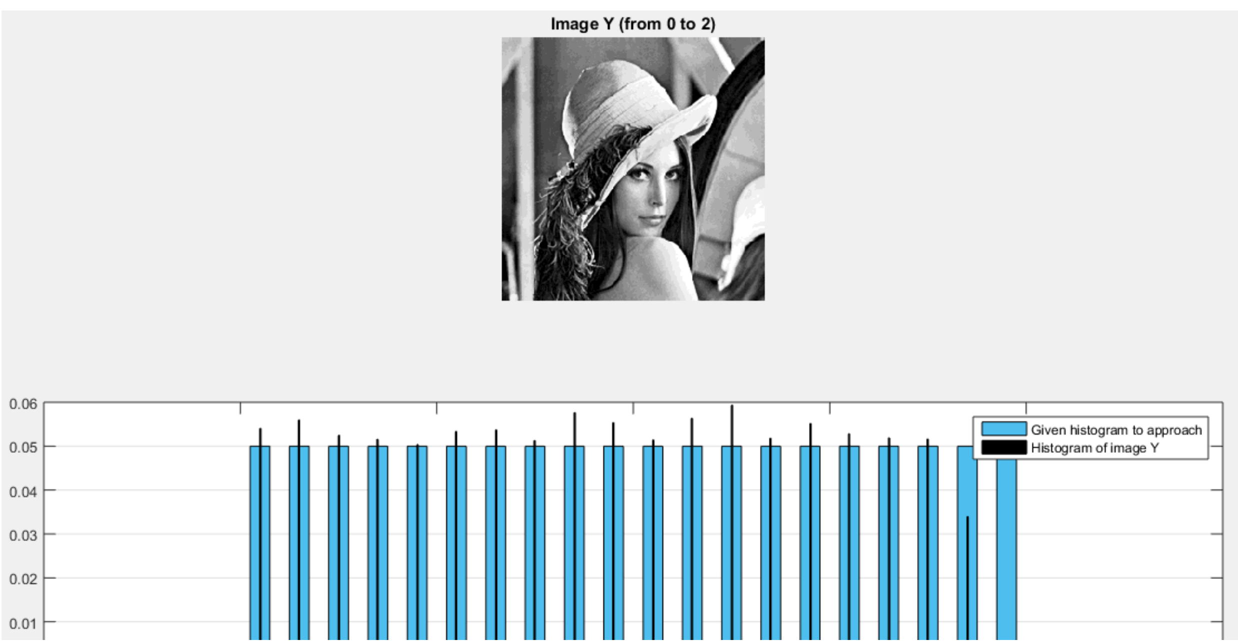


Σχήμα 20.Εικόνα χωρίς κορεσμό

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.2, d = 0:0.1:2

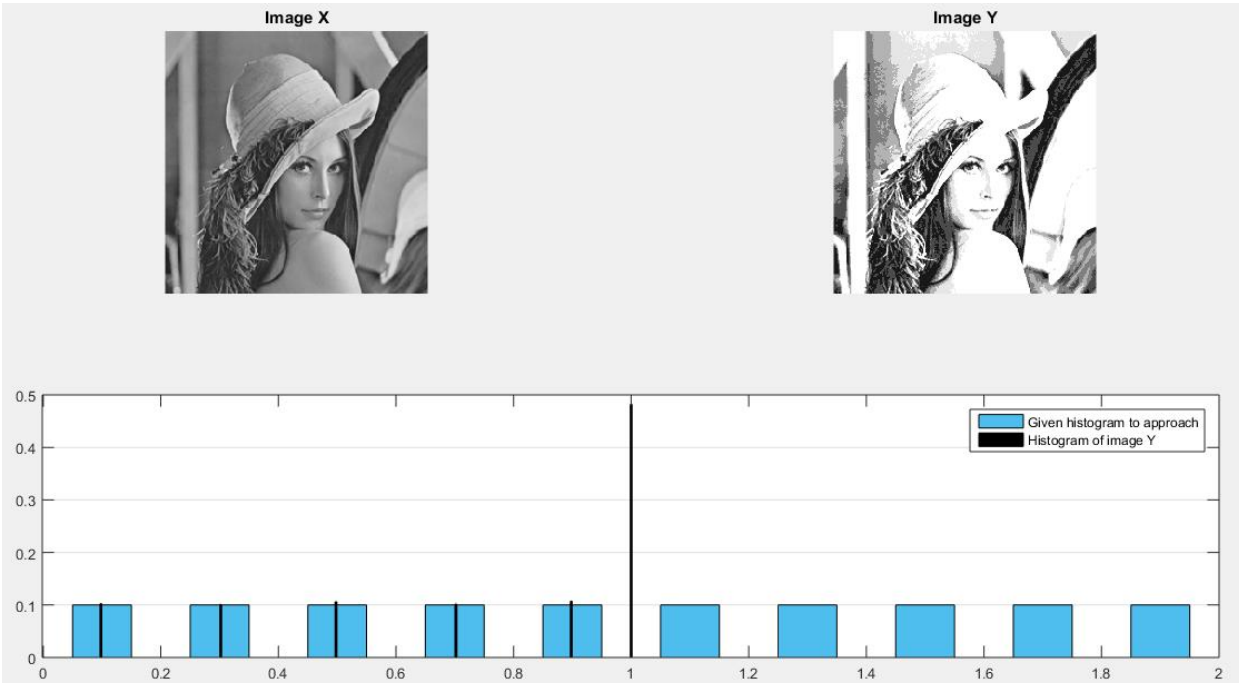


Σχήμα 21.Εικόνα με κορεσμό

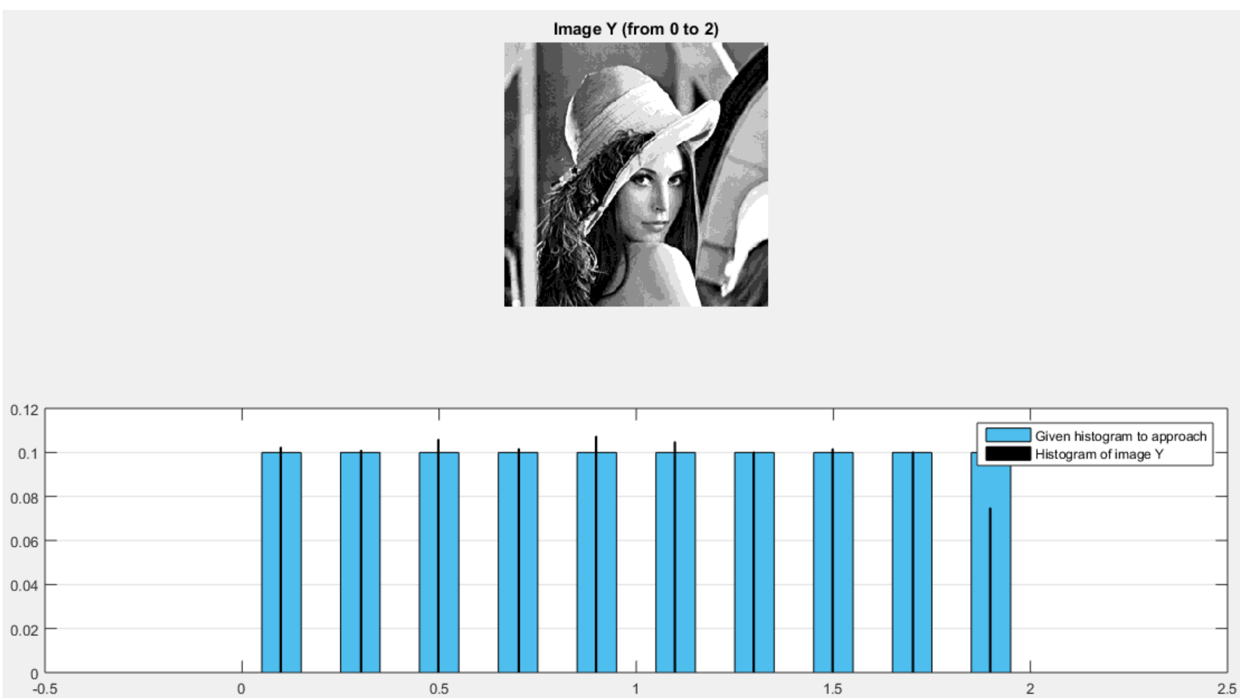


Σχήμα 22.Εικόνα χωρίς κορεσμό

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.2, d = 0:0.2:2

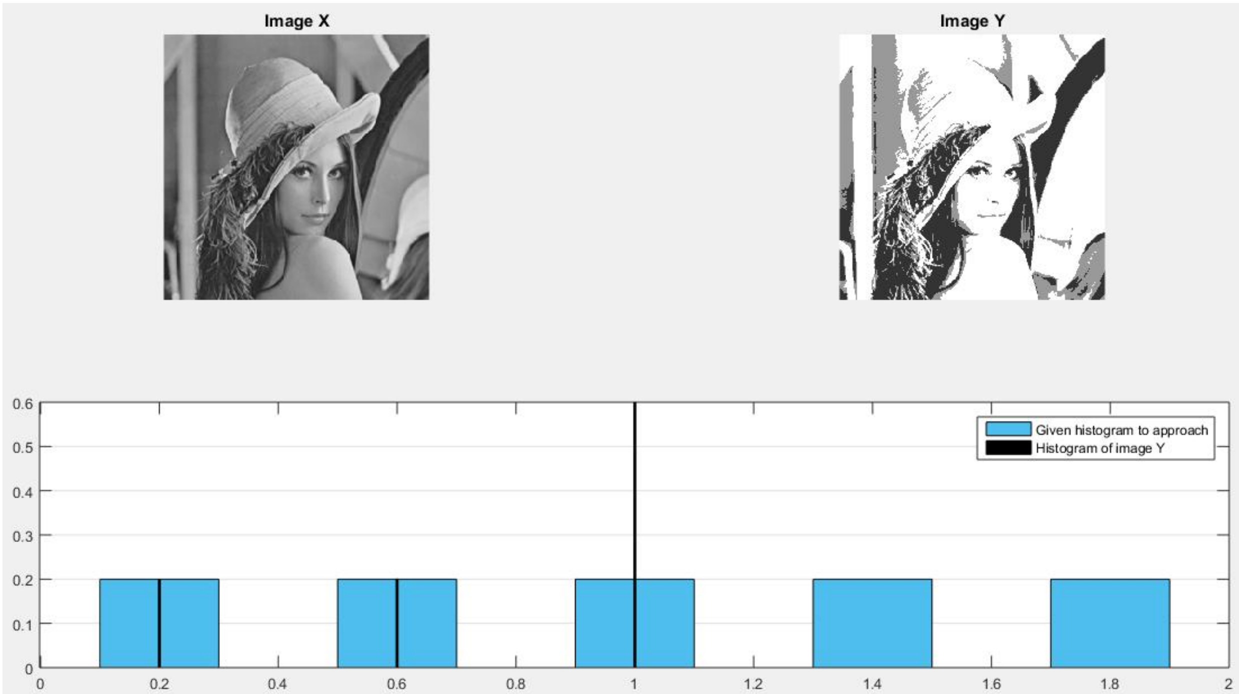


Σχήμα 23.Εικόνα με κορεσμό

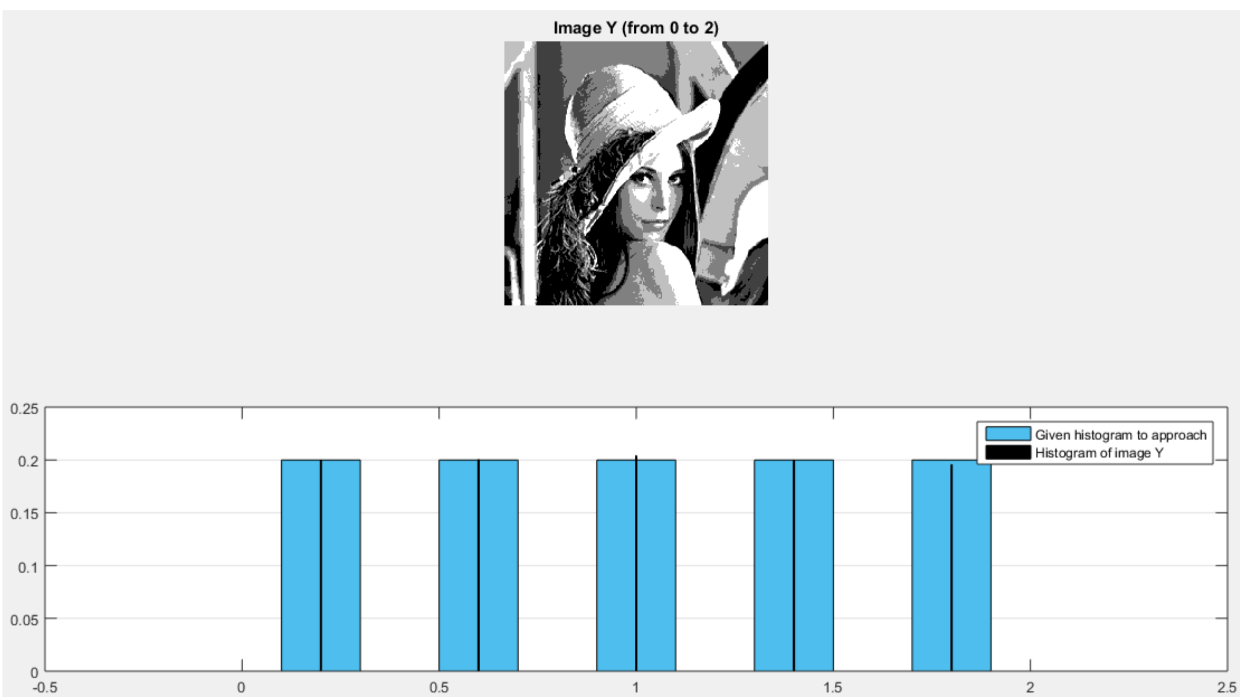


Σχήμα 24.Εικόνα χωρίς κορεσμό

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.2, d = 0:0.4:2



Σχήμα 26.Εικόνα με κορεσμό

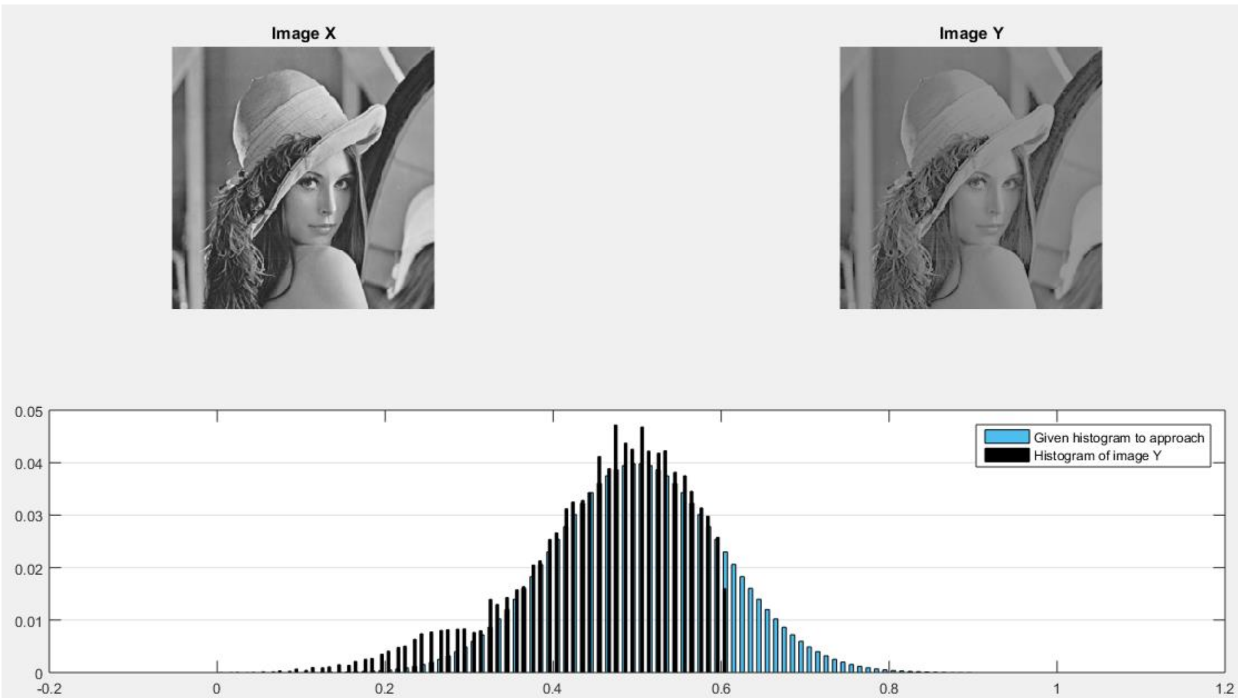


Σχήμα 25.Εικόνα χωρίς κορεσμό

Case .3 Κανονική κατανομή με μέση τιμή 0.5 και τυπική απόκλιση 0.1.

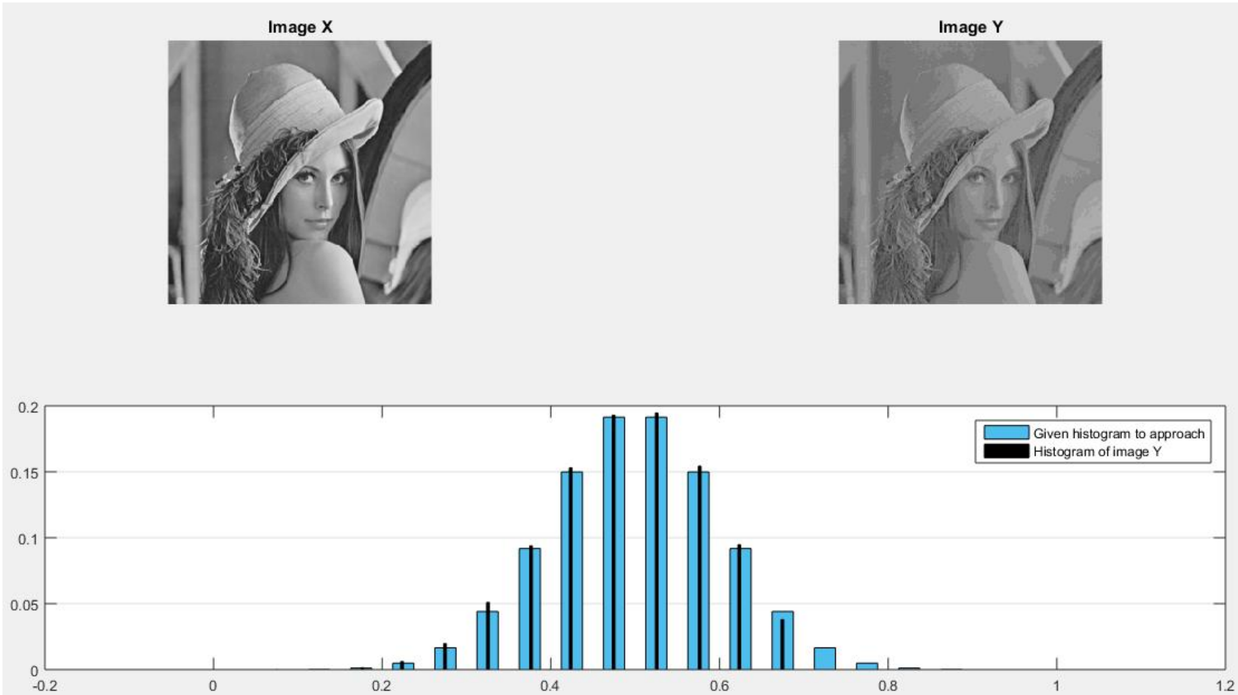
Για αυτή την περίπτωση βγάζουμε από σχόλιο τη γραμμή 14: f = @(x)unifpdf(x,0,2); (case 2) στο αρχείο editHistogram.m.

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.01, d = 0:0.01:1



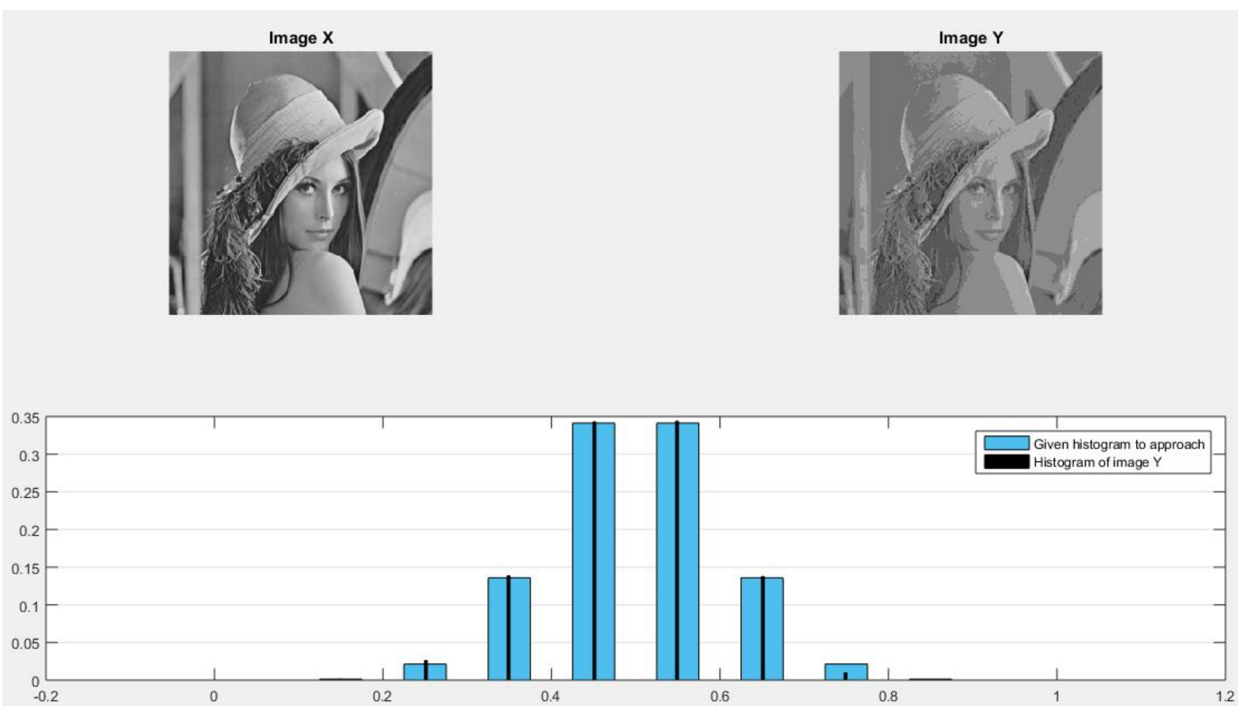
Σχήμα 27

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.05, d = 0:0.05:1



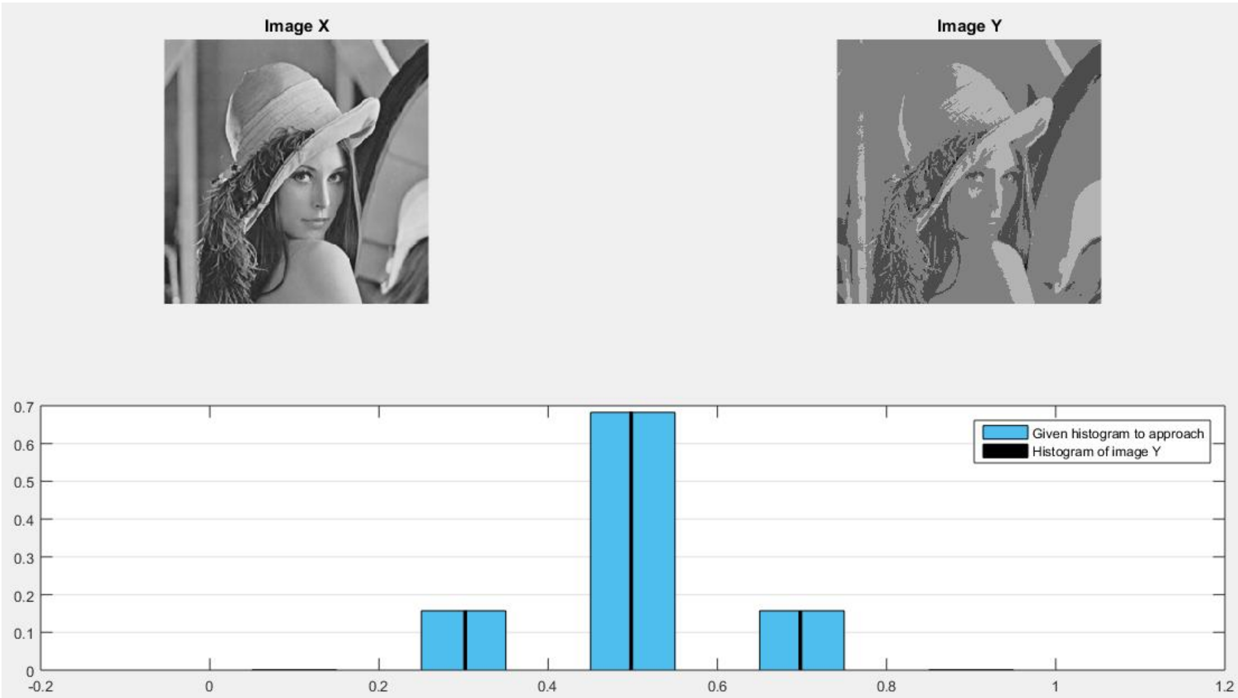
Σχήμα 28

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.1, d = 0:0.1:1



Σχήμα 29

- Διάστημα στα όρια της συνάρτησης κατανομής με σταθερό βήμα 0.1, d = 0:0.2:1



Σχήμα 30

Σχολιασμός των αποτελεσμάτων

Από τα παραπάνω φαίνεται ότι όσο περισσότερα είναι τα διαστήματα του d (μικρό βήμα) τόσο μεγαλύτερη απόκλιση έχουμε από τις ζητούμενες κατανομές ειδικά στο άνω άκρο τους. Αυτό είναι λογικό λόγω του greedy αλγόριθμου. Το πρόβλημα εμφανίζεται καθώς ο αλγόριθμος δεν χωρίζει ακριβώς τα pixel της εικόνας που αντιστοιχούν σε κάθε στάθμη φωτεινότητας. Αντ’ αυτού σε pixel ίδιας φωτεινότητας ανατίθεται η ίδια τιμή v με αποτέλεσμα σε κάθε στάθμη να προστίθενται περισσότερα Pixel από όσα θα θέλαμε. Το ιστόγραμμα βελτιώνεται και μοιάζει περισσότερο με το ζητούμενο καθώς το βήμα του d αυξάνεται. Για μεγαλύτερο βήμα βέβαια έχουμε και μικρότερη ποικιλία στις στάθμες φωτεινότητας (και άρα και στις αποχρώσεις) οι οποίες περιορίζονται σε περίπου 3 με 10 ανά περίπτωση.

Όσον αφορά τον κορεσμό των εικόνων στο case 2 αυτός είναι εμφανής και από τα αποτελέσματα. Στον κορεσμό όσες τιμές είναι μεγαλύτερες του 1 προβάλλονται ως άσπρο για αυτό και οι εικόνες φαίνονται πιο λευκές. Όταν το βήμα του d είναι πολύ μικρό το πρόβλημα οπτικά δεν είναι τόσο εμφανές καθώς το πλήθος των τιμών που ξεπερνούν τη μονάδα είναι αναλογικά μικρότερο των υπολοίπων. Όσο όμως το βήμα αυξάνει και οι στάθμες φωτεινότητας μοιράζονται όλο και καλύτερα στο διάστημα [0,2], η συγκέντρωση των Pixel στο 1 αυξάνεται σημαντικά. Όταν ανάγουμε τις φωτεινότητες αυτών των εικόνων στο [0,2] έχουμε όμοια αποτελέσματα με την ομοιόμορφη κατανομή στο [0,1] για ίσο πλήθος διαστημάτων (βλ. Σχήμα 16 σελ.14 και Σχήμα 25 σελ.20). Επομένως το case 1 και case 2 αν ελέγξουμε τον κορεσμό και τα διαστήματα είναι πρακτικά ίδια.

Το case 3 διαφοροποιείται και αυτό φαίνεται και οπτικά. Οι φωτεινότητες πλέον δεν ισομοιράζονται στο [0,1] αλλά συγκεντρώνονται προς τη μέση τιμή της κατανομής. Με δεδομένο ότι έχουμε και μικρή τυπική απόκλιση, για αυξανόμενο βήμα στο d οι εικόνα αρχίζει να "γκριζάρει'' όλο και περισσότερο έχοντας μεγάλη συγκέντρωση τιμών φωτεινότητας προς το 0.5.