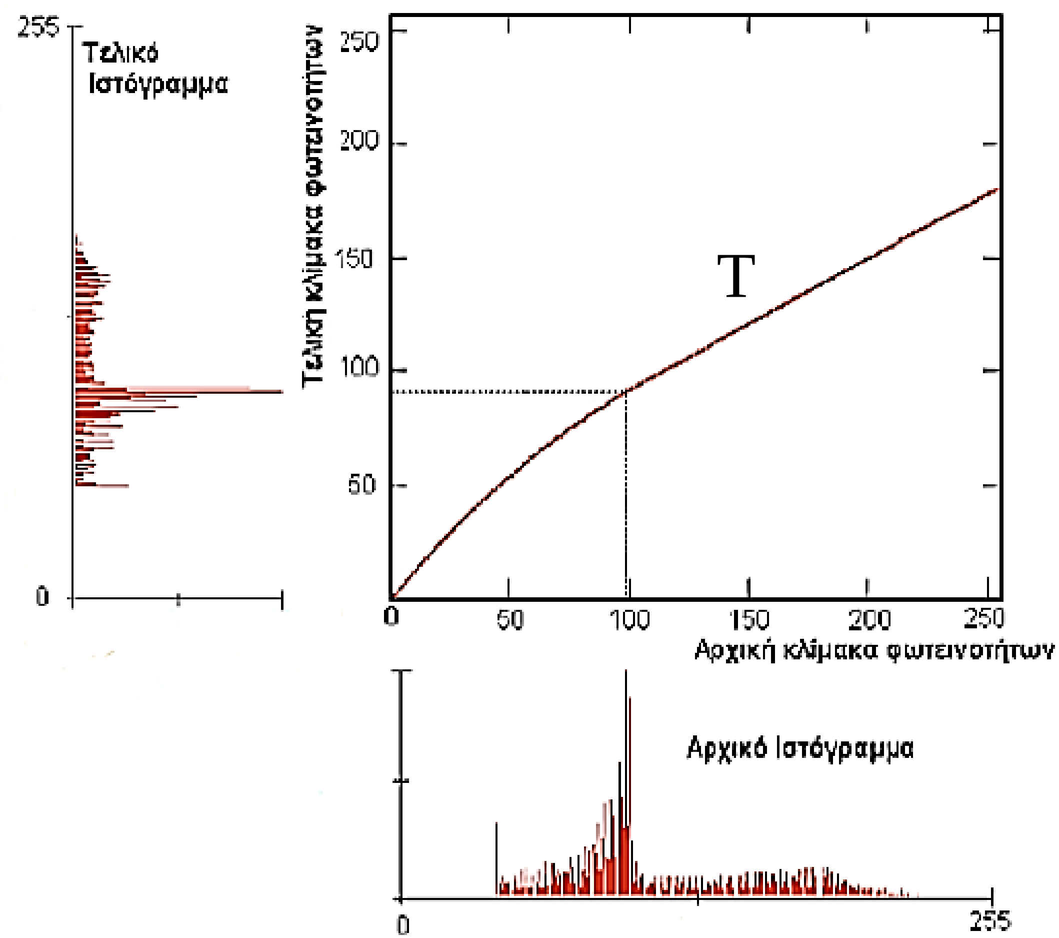
Histogram Processing

April 2020

ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ ΦΩΤΕΙΝΟΤΗΤΑΣ



ΑΡΝΗΤΙΚΗ ΕΙΚΟΝΑ

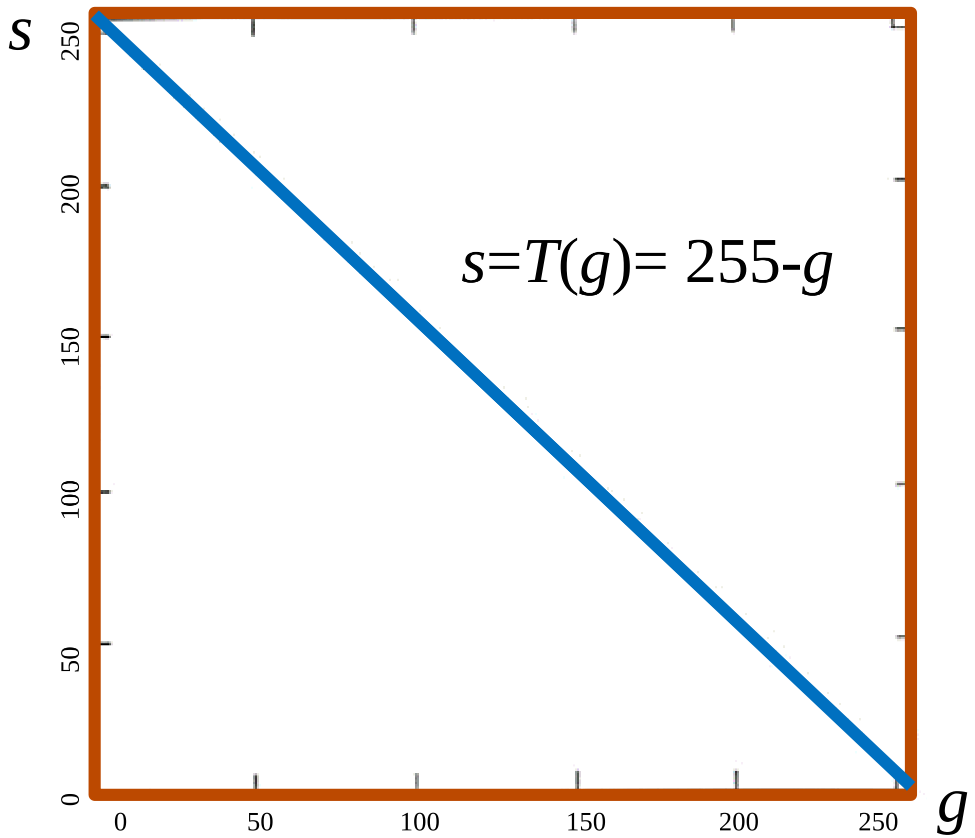
- Το αρνητικό μιας εικόνας 8bit παράγεται χρησιμοποιώντας τη συνάρτηση μετασχηματισμού η οποία είναι ίση με:

$$s=T(g)= 255-g$$

- Η βασική ιδέα είναι η αντιστροφή των φωτεινοτήτων.

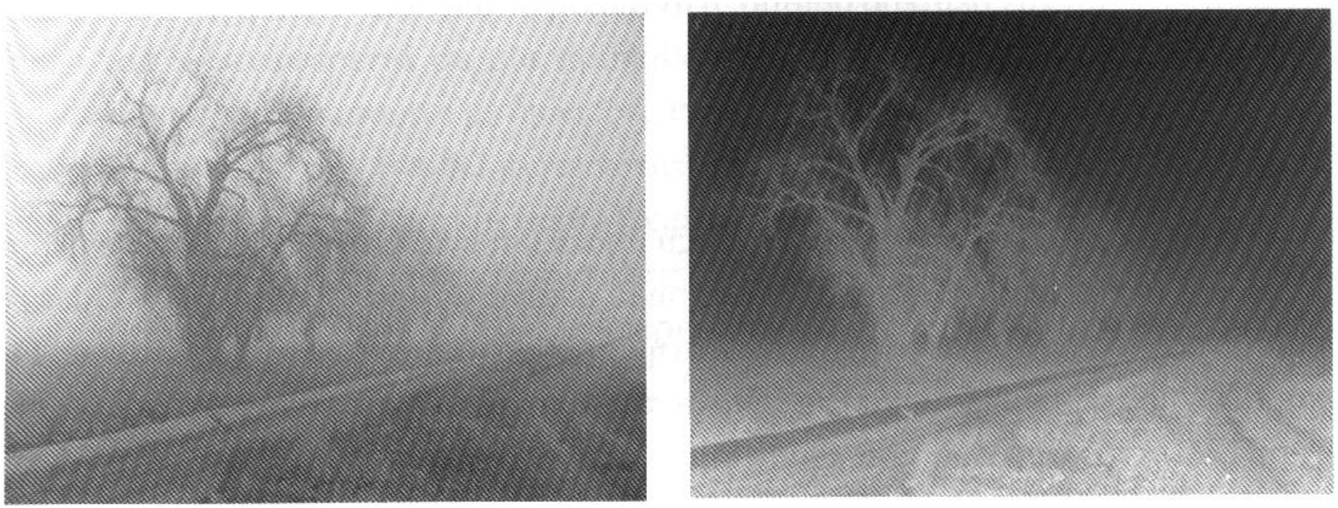
ΑΡΝΗΤΙΚΗ ΕΙΚΟΝΑ

- Η συνάρτηση μετασχηματισμού για το αρνητικό της εικόνας



ΑΡΝΗΤΙΚΗ ΕΙΚΟΝΑ

- μια αρχική εικόνα → το αρνητικό της.



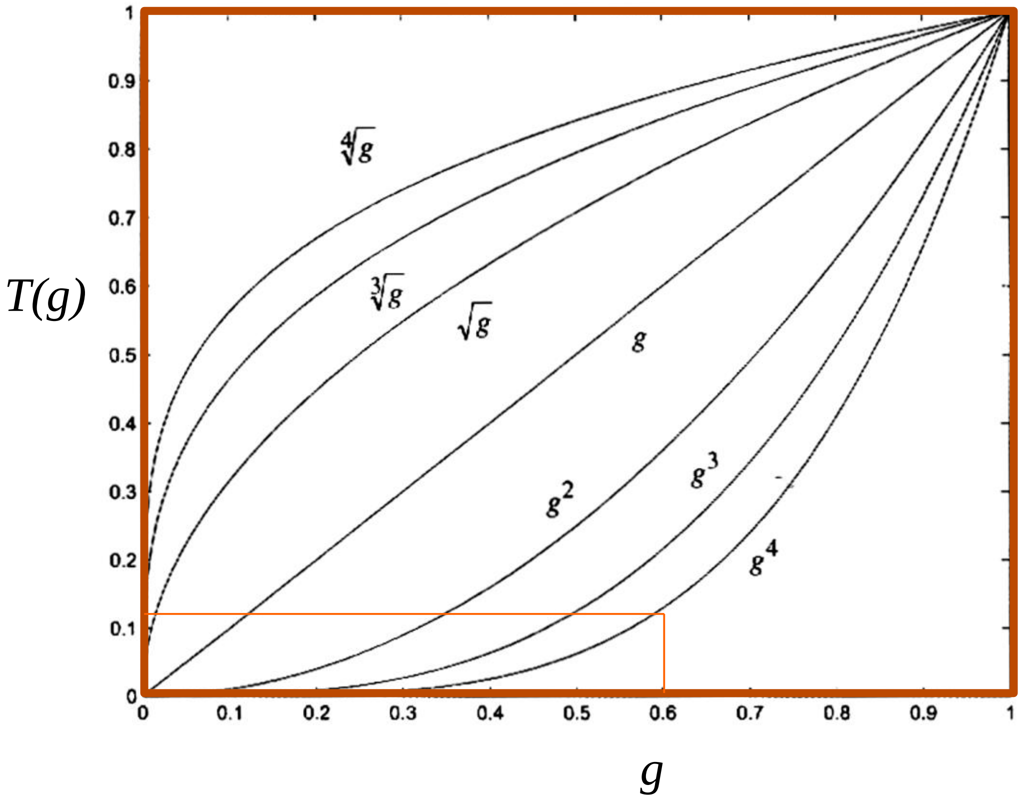
ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ ΔΥΝΑΜΗΣ

- Η πρώτη κατηγορία τέτοιων συναρτήσεων βασίζεται στη οικογένεια των συναρτήσεων δύναμης:

$$T(g) = 255 \cdot g^{\gamma}$$

- όπου 0≤ g ≤1 και το γ ο εκθέτης. Συνεπώς, για μία εικόνα $I(i,j)$ ∈ [0,255] έχουμε:
- $I(i,j) = int\,[255 \cdot g(i,j)^{\gamma}], οπου\; g(i,j) = \frac{I(i,j)}{255}$

ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ ΔΥΝΑΜΗΣ



Μορφές της $T(g)$ για διάφορες τιμές του εκθέτη *γ {4,3,2,1,1/2,…}*

Εφαρμογή των μετασχηματισμών δύναμης για διάφορες τιμές του γ

https://en.wikipedia.org/wiki/Gamma_correction

ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ ΔΥΝΑΜΗΣ σε matlab

```
f= imread('cameraman.gif');
max(max(f))
%rescale values to 0-1
f2=mat2gray(f);
max(max(f2))
figure, imshow(f2,[])
figure, imshow(f,[])
%gamma=2
I=(255*f2.^2);
max(max(I))
figure, imshow(I,[])
%gamma=1/2
I=(255*f2.^(1/2));
figure, imshow(I,[])
```

ΛΟΓΑΡΙΘΜΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ

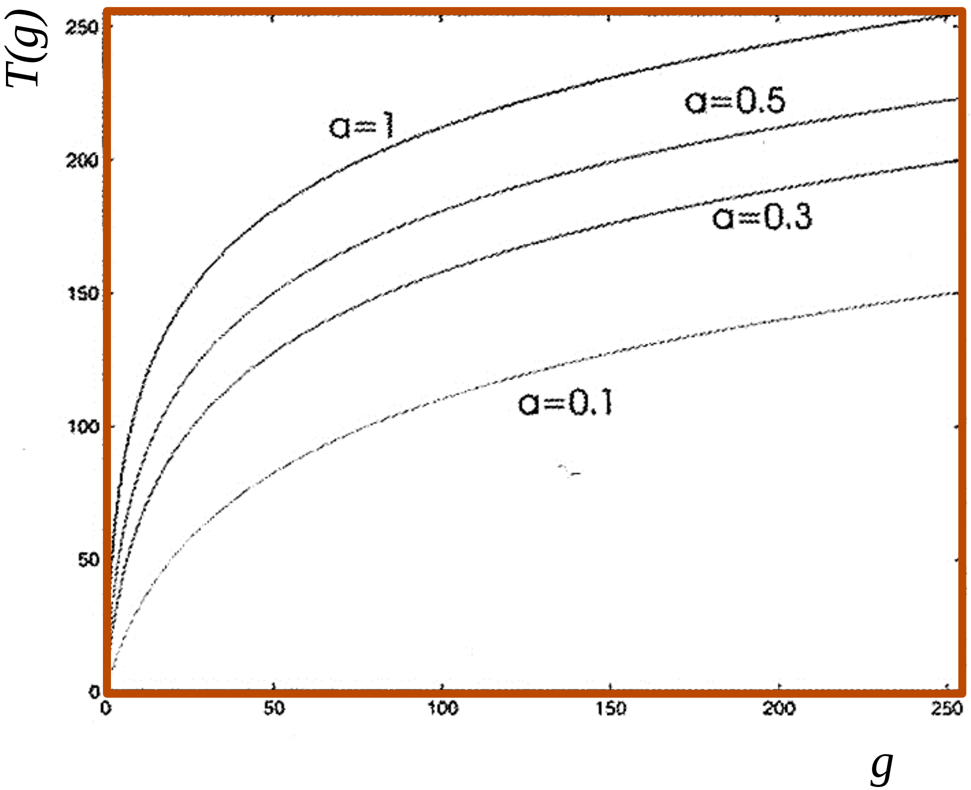
- Χρησιμοποιούνται για να βελτιστοποιούν σκοτεινές και φωτεινές εικόνες αντίστοιχα.
- Ο λογαριθμικός μετασχηματισμός έχει τη ακόλουθη γενική μορφή:

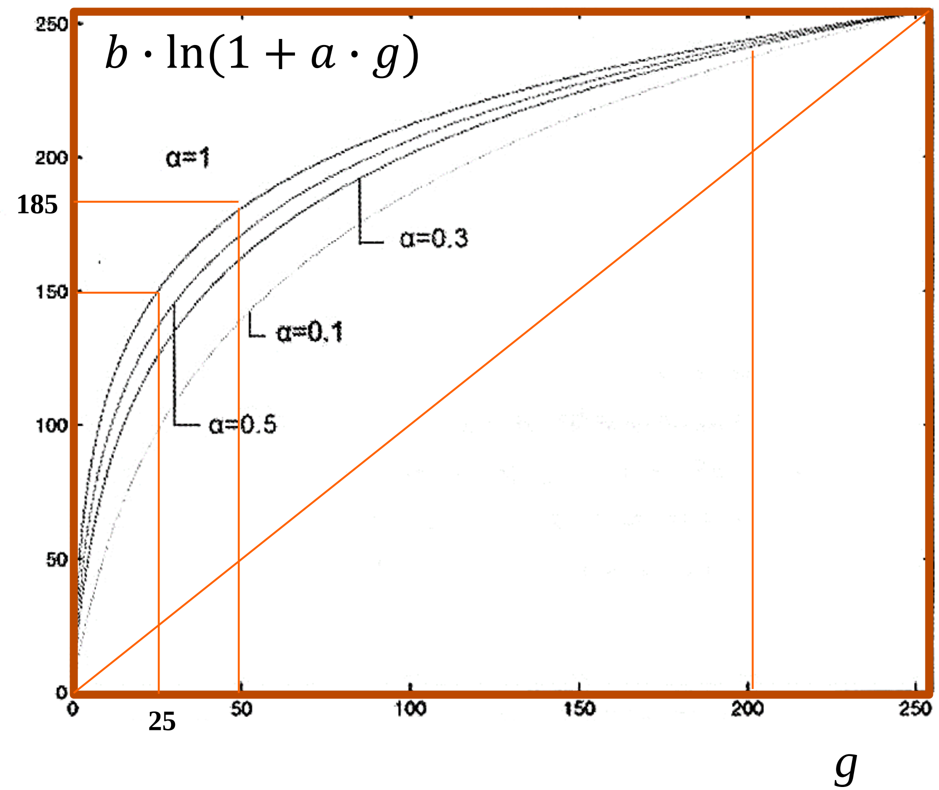
$$T(g) = b \cdot \ln(1 + a \cdot g)$$

- και αν θεωρήσουμε ως προϋπόθεση ότι T(0) = 0 και T(255) = 255, τότε

$$b = \frac{255}{\ln(1 + 255 \cdot a)}$$

ΛΟΓΑΡΙΘΜΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ





Καμπύλες λογαριθμικού μετασχηματισμού για b=45.986

Καμπύλες λογαριθμικού μετασχηματισμού για $b = \dfrac{255}{\ln(1 + 255 \cdot a)}$

ΛΟΓΑΡΙΘΜΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ

Παράδειγμα εφαρμογής του λογαριθμικού μετασχηματισμού

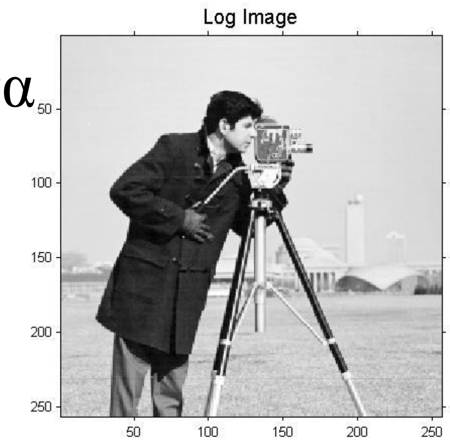
```
grayImage = imread('Cameraman.tif');
grayImage = double(grayImage);
subplot(2,1,1);
imshow(grayImage, []);
axis on;
title('Original Image', 'FontSize', 15);
% Take the log of it.  Add 1 to avoid taking log of zero.
logImage = log(grayImage+1);
% Normalize to the range 0-1.
normalizedImage = mat2gray(logImage);
% Display it.
subplot(2,1,2);
imshow(normalizedImage, []);
axis on;
title('Log Image', 'FontSize', 15);
% If you want a uint8 version, then you can multiply by 255
% uint8Image = uint8(255 * normalizedImage);
msgbox('Note how the coat has more details');
```

$$b = 255\,/\ln(1 + 255 \cdot a)$$

$$T\,(g) = b \cdot \ln(1 + a \cdot g)$$

Δοκιμάστε το ίδιο για την εικόνα grayImage = imread('pout.tif');

(α) Αρχική εικόνα.



(β) Τελική εικόνα για α=1.

ΛΟΓΑΡΙΘΜΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ

```
grayImage = imread('Cameraman.tif');
grayImage = double(grayImage);
subplot(2,1,1); imshow(grayImage, []);
axis on;
title('Original Image', 'FontSize', 15);
% Take the log of it. Add 1 to avoid taking log of zero.
a=1; b=255/(log(1+255*a));
logImage = b*log(1+a*grayImage);
% Display it.
subplot(2,1,2);
imshow(logImage , []);
axis on;
title('Log Image', 'FontSize', 15);
msgbox('Note how the coat has more details');
```

$$b = 255\,/\ln(1 + 255 \cdot a)$$

$$T\,(g) = b \cdot \ln(1 + a \cdot g)$$

Δοκιμάστε το ίδιο για την εικόνα
grayImage = imread('pout.tif');

ΕΚΘΕΤΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ

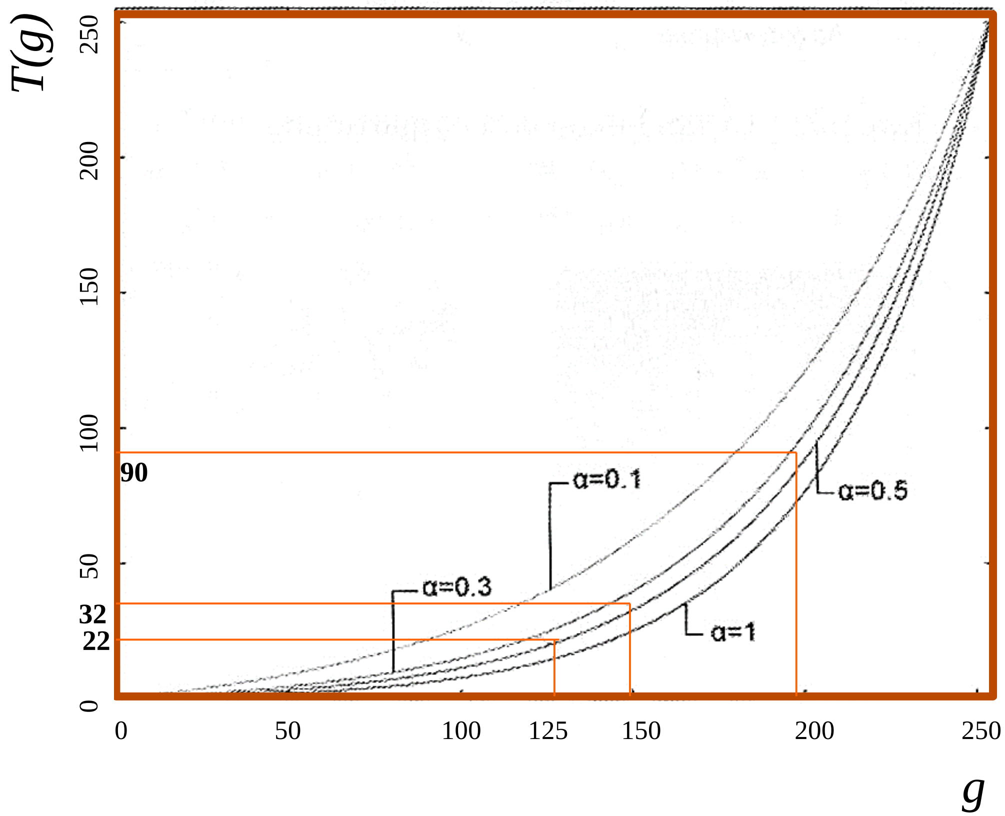
- Σε αντιστοίχηση με τον λογαριθμικό μετασχηματισμό, ο εκθετικός μετασχηματισμός έχει την ακόλουθη έκφραση:

$$T(\boldsymbol{g}) = \frac{1}{a}\left(e^{\frac{\boldsymbol{g}}{b}} - 1\right)$$

- Αν θέσουμε ως περιορισμό ότι T(0) = 0 και T(255) = 255, τότε το b μπορεί να υπολογισθεί πάλι από τη σχέση

$$b = \frac{255}{\ln(1 + 255 \cdot a)}$$

ΕΚΘΕΤΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ



Καμπύλες εκθετικού μετασχηματισμού για $b = \frac{255}{\ln(1+255 \cdot a)}$

ΕΚΘΕΤΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ

Παράδειγμα εφαρμογής του εκθετικού μετασχηματισμού

```
grayImage= imread('Cameraman.tif');
grayImage= double(grayImage);
subplot(2,1,1);
imshow(grayImage, []);
axis on;
title('Original Image', 'FontSize', 15);
% Take the log of it. Add 1 to avoid taking log of zero.
expImage=exp(grayImage/50)-1;
% Normalize to the range 0-1.
normalizedImage= mat2gray(expImage);
% Display it.
subplot(2,1,2);
imshow(normalizedImage, []);
axis on;
title('ExpImage', 'FontSize', 15);
```

$$b = 255/\ln(1 + 255 \cdot a)$$

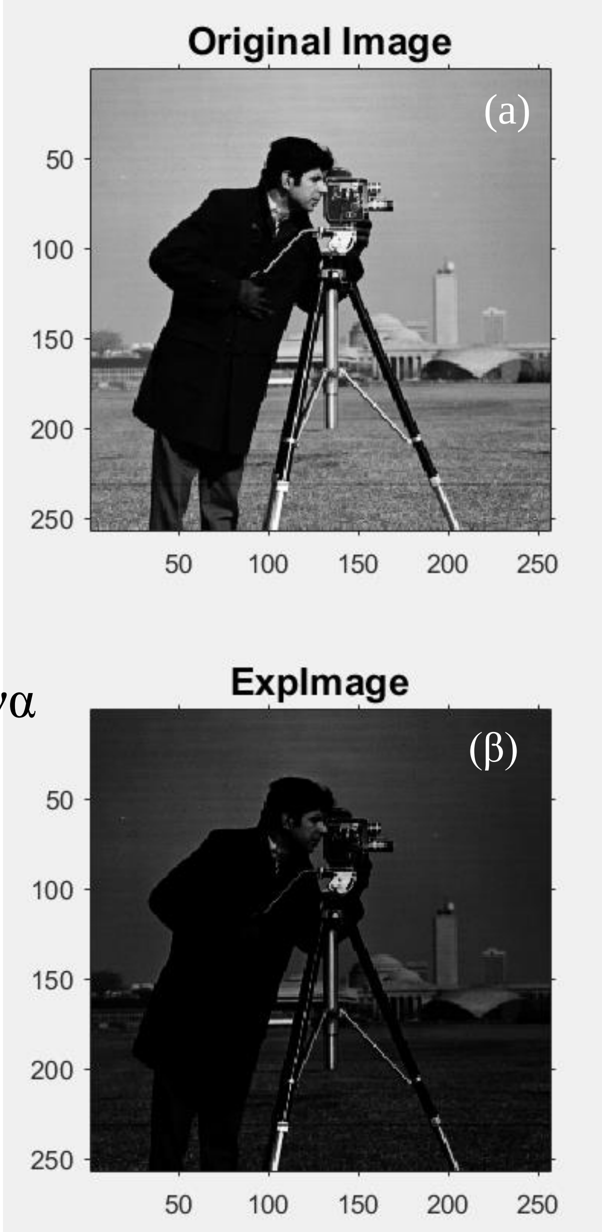
$$T(\boldsymbol{g}) = 1/a^{*}\left(e^{g/b} - 1\right)$$

Δοκιμάστε το ίδιο για την εικόνα

A=imread('office_6.jpg');

grayImage = A(:,:,1);

Τι παρατηρείτε;



Παράδειγμα εφαρμογής του εκθετικού μετασχηματισμού:
(α) Αρχική εικόνα.
(β) Τελική εικόνα για α=1.

ΕΚΘΕΤΙΚΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ

```
grayImage = imread('Cameraman.tif');
grayImage = double(grayImage);
subplot(2,1,1);
imshow(grayImage, []);
axis on;
title('Original Image', 'FontSize', 15);
% Take the log of it. Add 1 to avoid taking log of zero.
a=1; b=255/(log(1+255*a));
expImage =(1/a)* (exp(grayImage/b)-1);
% Display it.
subplot(2,1,2);
imshow(expImage, []);
axis on;
title('Exp Image', 'FontSize', 15);
```

$$b = 255/\ln(1 + 255 \cdot a)$$

$$T\,(\boldsymbol{g}) = 1/a^*\,(e^{g/b} - 1)$$

Δοκιμάστε το ίδιο για την εικόνα

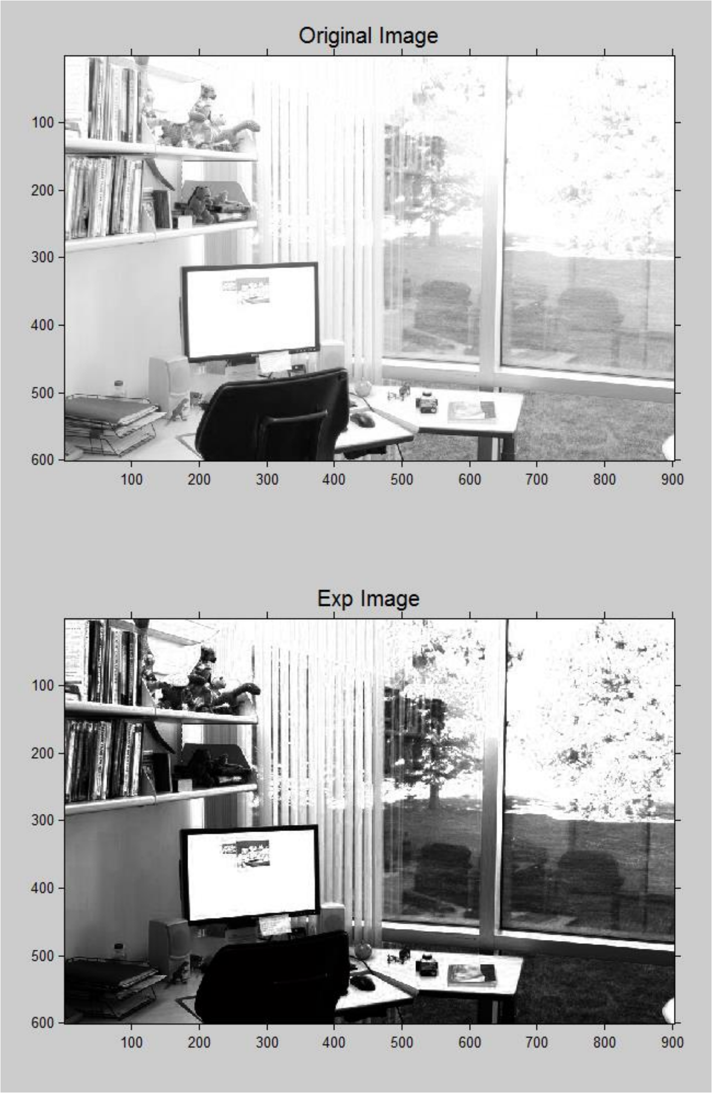
A=imread('office_6.jpg');

grayImage = A(:,:,1);

Τι παρατηρείτε;

```
A=imread('office.jpg');
grayImage = double(A(:,:,1));

subplot(2,1,1);
imshow(grayImage, []);
axis on;
title('Original Image', 'FontSize', 15);
% Take the log of it. Add 1 to avoid taking log of zero.
a=1; b=255/(log(1+255*a));
expImage =(1/a)* (exp(grayImage/b)-1);
% Display it.
subplot(2,1,2);
imshow(mat2gray(expImage), []);
axis on;
title('Exp Image', 'FontSize', 15);
```


Original Image
100
200
300
400
500
600
100
200
300
400
500
600
700
800
900
Exp Image
100
200
300
400
500
600
100
200
300
400
500
600
700
800
900

Ας δούμε και τα αντίστοιχα ιστογράμματα

```
grayImage = imread('office.jpg');
grayImage=grayImage(:,:,1);
grayImage = double(grayImage);
% Take the log of it. Add 1 to avoid taking log of zero.
a=0.3; b=255/(log(1+255*a));
expImage =(1/a)* (exp(grayImage/b)-1);

subplot(2,2,1); imshow(grayImage, []); title('Original Image', 'FontSize', 15);
subplot(2,2,2); imshow(expImage , []); title('Exp Image', 'FontSize', 15);
subplot(2,2,3);imhist(uint8(grayImage));title('Histogram of original image')
subplot(2,2,4);imhist(uint8(expImage));title('Histogram of EXP image')
```

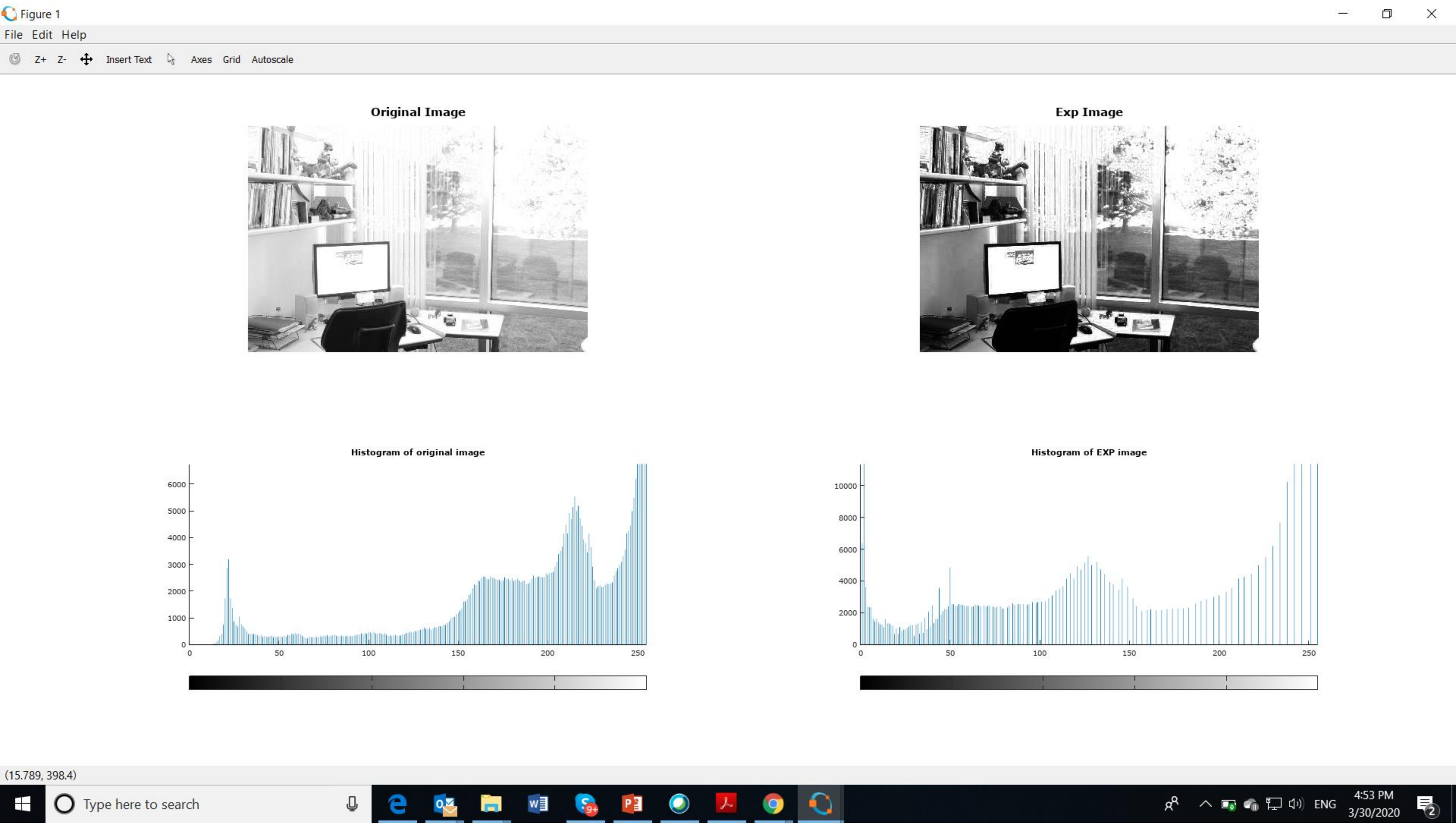
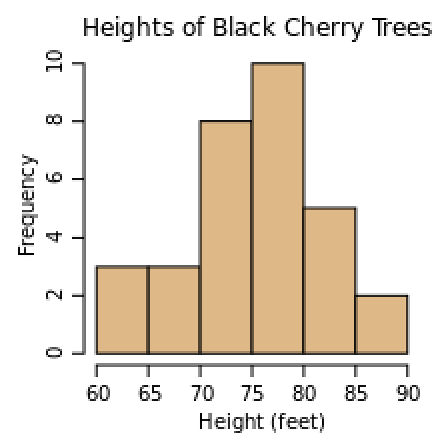


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
Original Image
Exp Image
Histogram of original image
Histogram of EXP image
(15.789, 398.4)
Type here to search
ENG
4:53 PM
3/30/2020

Ιστόγραμμα: Γενικά

https://el.wikipedia.org/wiki/Ιστόγραμμα

- Το Ιστόγραμμα είναι γραφική απεικόνιση στατιστικών συχνοτήτων περιοχών τιμών ενός μεγέθους.
- Πρόκειται για τη συνηθέστερη επιλογή γραφικής παράστασης συνεχών μεταβλητών.
- Στα συνεχή δεδομένα, οι τιμές της μεταβλητής ομαδοποιούνται και οι ομάδες διατάσσονται στον οριζόντιο άξονα κατ΄αύξουσα σειρά.
- Στη συνέχεια από κάθε ομάδα υψώνουμε ορθογώνια ,το ύψος των οποίων αντιστοιχεί στη συχνότητα κάθε ομάδας.



ΤΟ ΙΣΤΟΓΡΑΜΜΑ ΜΙΑΣ ΕΙΚΟΝΑΣ

```
%Διαβάζουμε μια εικόνα της Matlab
Im = imread('cameraman.tif');
%Δημιουργούμε μια άλλη με θόρυβο Gauss
I = imnoise(Im,'gaussian');
%Δείχνουμε τις δυο εικόνες με τα ιστογράμματά τους
subplot(2,2,1);imshow(Im,[ ]);title('Αρχική εικόνα');
subplot(2,2,2);imshow(I,[ ]);title('Εικόνα με θόρυβο Gauss');
subplot(2,2,3);imhist(Im);title('Ιστόγραμμα αρχικής εικόνας');
subplot(2,2,4);imhist(I);title('Ιστόγραμμα εικόνας με θόρυβο Gauss ')
```

J = histeq(I);

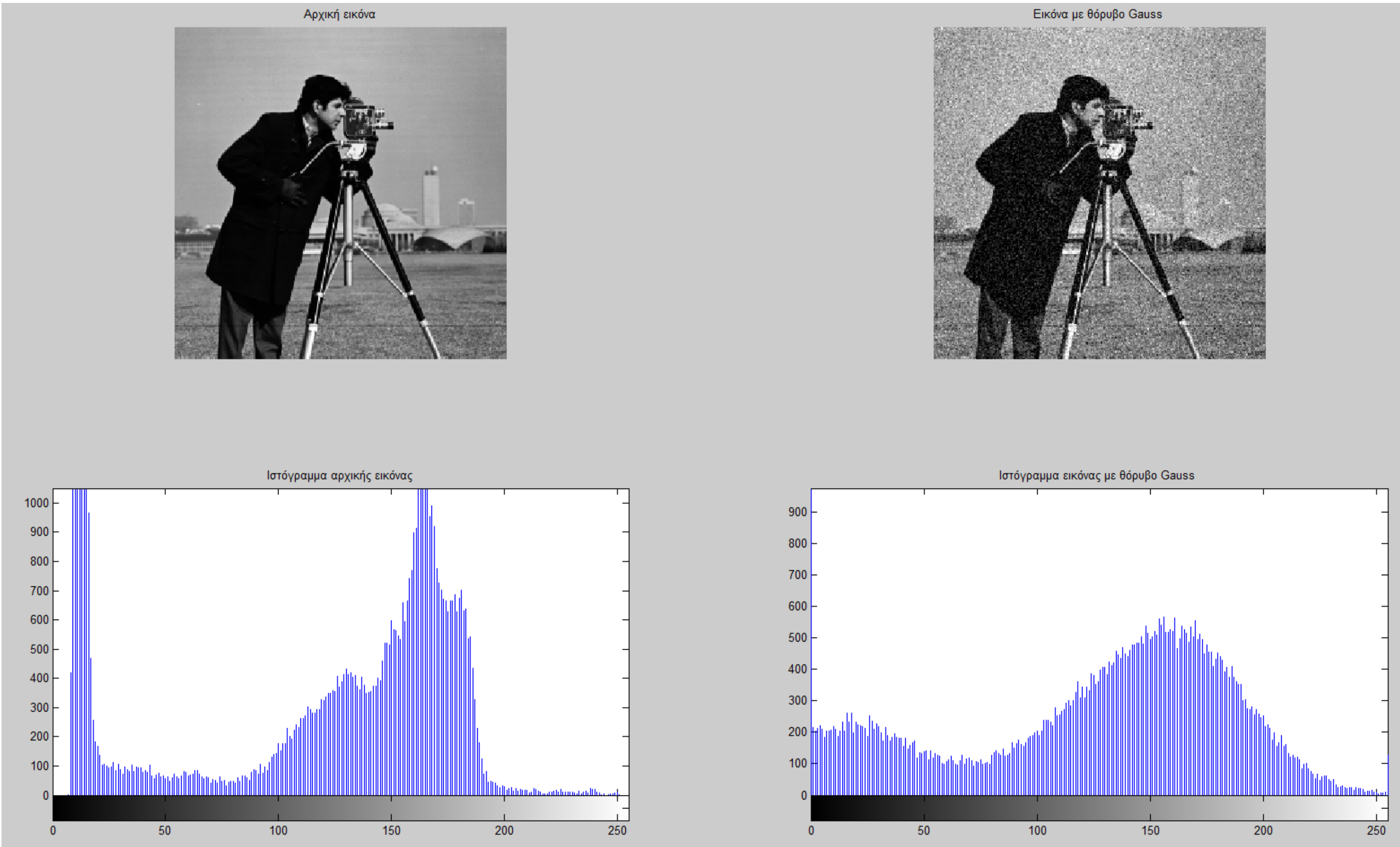
Σε αναλογία με το παραπάνω παράδειγμα δείξτε σε μια εικόνα α) την αρχική, β) τον λογαριθμικό μετασχηματισμό και γ,δ) τα αντίστοιχα ιστογράμματά τους. Τι παρατηρείτε ότι κάνει ο λογαριθμικός μετασχηματισμός στα ιστογράμματα?

ΤΟ ΙΣΤΟΓΡΑΜΜΑ ΜΙΑΣ ΕΙΚΟΝΑΣ

```
%Διαβάζουμε μια εικόνα της Matlab
Im = imread('cameraman.tif');
%Δημιουργούμε μια άλλη με LOG
grayImage = double(Im);
a=1; b=255/(log(1+255*a));
logImage = b*log(1+a*grayImage);
%Δείχνουμε τις δυο εικόνες με τα ιστογράμματά τους
subplot(2,2,1);imshow(Im,[ ]);title('Αρχική εικόνα');
subplot(2,2,2);imshow(logImage,[ ]);title('Εικόνα LOG');
subplot(2,2,3);imhist(Im);title('Ιστόγραμμα αρχικής εικόνας');
subplot(2,2,4);imhist(uint8(logImage));title('Ιστόγραμμα εικόνας LOG')
```

Σε αναλογία με το παραπάνω παράδειγμα δείξτε σε μια εικόνα α) την αρχική, β) τον λογαριθμικό μετασχηματισμό και γ,δ) τα αντίστοιχα ιστογράμματά τους. Τι παρατηρείτε ότι κάνει ο λογαριθμικός μετασχηματισμός στα ιστογράμματα?

ΤΟ ΙΣΤΟΓΡΑΜΜΑ ΜΙΑΣ ΕΙΚΟΝΑΣ



Χρησιμοποιείστε την εντολή J = histeq(I); Για εξισοπόηση εικόνας cameraman.tif (με και χωρίς θόρυβο) καθώς και tire.tif
Τι παρατηρείτε;

ΤΟ ΙΣΤΟΓΡΑΜΜΑ ΜΙΑΣ ΕΙΚΟΝΑΣ

```
%Διαβάζουμε μια εικόνα της Matlab
Im = imread('cameraman.tif');
I=histeq(Im);
%Δημιουργούμε μια άλλη
%Δείχνουμε τις δυο εικόνες με τα ιστογράμματά τους
subplot(2,2,1);imshow(Im,[ ]);title('Αρχική εικόνα');
subplot(2,2,2);imshow(I,[ ]);title('Εικόνα με Εξισορρόπηση Ιστογράμματος');
subplot(2,2,3);imhist(Im);title('Ιστόγραμμα αρχικής εικόνας');
subplot(2,2,4);imhist(I);title('Ιστόγραμμα εικόνας με ΕΙ')
```

J = histeq(I);

Σε αναλογία με το παραπάνω παράδειγμα δείξτε σε μια εικόνα α) την αρχική, β) τον λογαριθμικό μετασχηματισμό και γ,δ) τα αντίστοιχα ιστογράμματά τους. Τι παρατηρείτε ότι κάνει ο λογαριθμικός μετασχηματισμός στα ιστογράμματα?

```
%Διαβάζουμε μια εικόνα της Matlab
Im = imread('cameraman.gif');
%Δημιουργούμε μια άλλη με θόρυβο Gauss
I = imnoise(Im,'gaussian');
I=histeq(I);

%Δείχνουμε τις δυο εικόνες με τα ιστογράμματά τους
subplot(2,2,1);imshow(Im,[ ]);title('Αρχική εικόνα');
subplot(2,2,2);imshow(I,[ ]);title('Εικόνα με θόρυβο Gauss');
subplot(2,2,3);imhist(Im);title('Ιστόγραμμα αρχικής εικόνας');
subplot(2,2,4);imhist(I);title('Ιστόγραμμα εικόνας με θόρυβο
Gauss ')
```

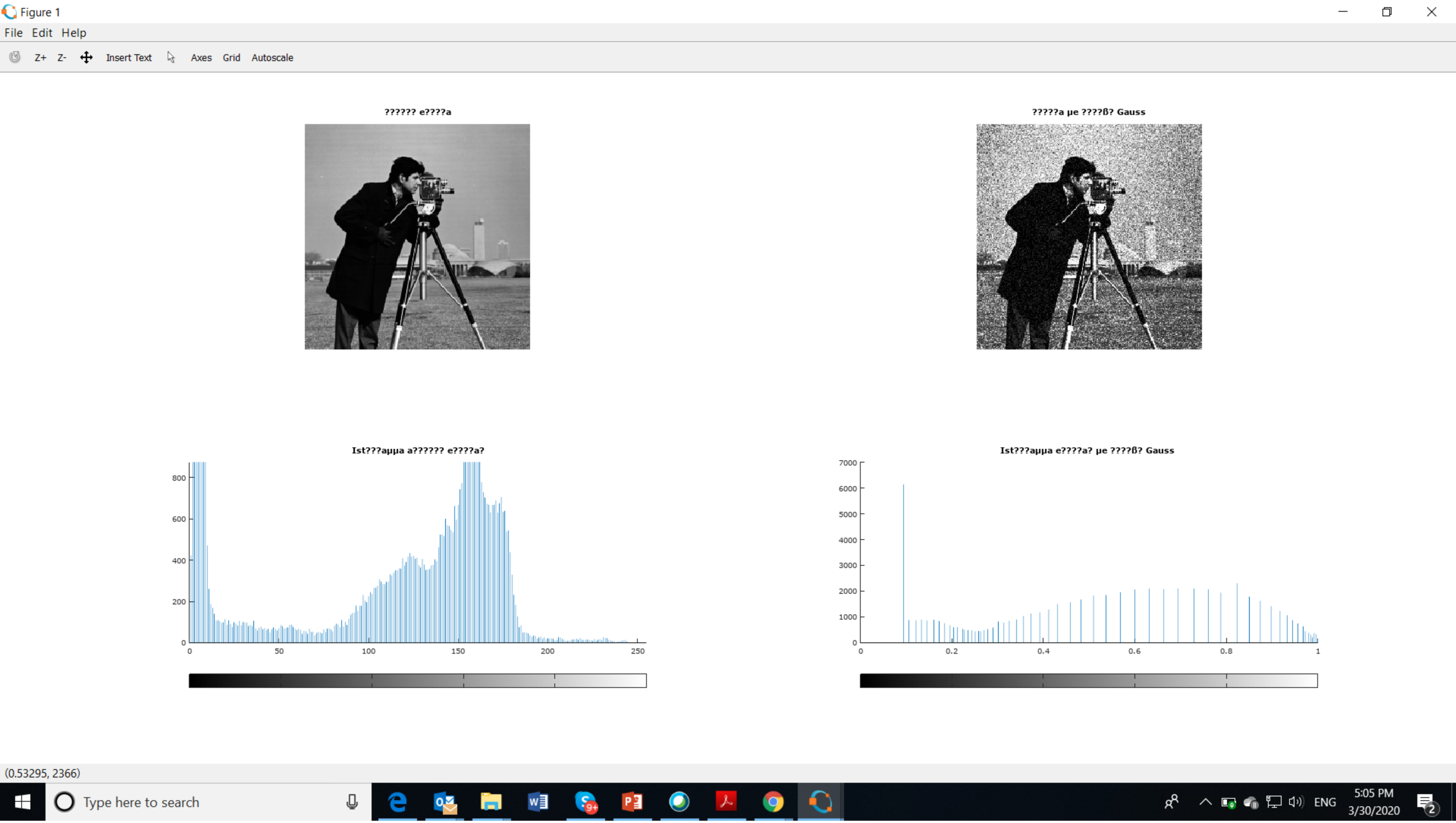


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
?????? e????a
?????a με ????β? Gauss
Ist???αμμα α?????? e????a?
Ist???αμμα e????a? με ????β? Gauss
0 50 100 150 200 250
0 200 400 600 800
0 1000 2000 3000 4000 5000 6000 7000
0 0.2 0.4 0.6 0.8 1
(0.53295, 2366)
Type here to search
ENG
5:05 PM
3/30/2020

Ιστόγραμμα: Εικόνα

https://el.wikipedia.org/wiki/Ιστόγραμμα

- Σε ένα ιστόγραμμα, ο Χ άξονας αντιπροσωπεύει την κλίμακα έντασης (0 έως 255 σε ένα σύστημα των 8 bit).
- Ο Υ άξονας μετρά τον αριθμό από pixels στην εικόνα που έχουν μια ορισμένη τιμή έντασης (την αντίστοιχη του Χ άξονα).
- Τα ιστογράμματα επεξηγούν, με μορφή γραφήματος, τη φωτεινότητα και τα χαρακτηριστικά αντίθεσης μιας εικόνας, δηλαδή αν υπάρχουν και πόσα pixel με κάποια δεδομένη τιμή έντασης.
- **Όμως δεν δίνει πληροφορίες για την θέση των pixel μέσα στην εικόνα!**

ΕΠΕΞΕΡΓΑΣΙΑ ΙΣΤΟΓΡΑΜΜΑΤΟΣ

- Το ιστόγραμμα μιας εικόνας αποχρώσεων του γκρι περιέχει σημαντικές πληροφορίες για την εικόνα και για το λόγο αυτό είναι ένα από τα σημαντικότερα εργαλεία στην επεξεργασία ψηφιακών εικόνων.
- Μπορεί να χρησιμοποιηθεί για τη βελτιστοποίηση της εικόνας, την τροποποίηση των χαρακτηριστικών της, την μετατροπή της σε εικόνα με λιγότερες αποχρώσεις, την εξαγωγή χαρακτηριστικών της εικόνας κ.α.

ΤΟ ΙΣΤΟΓΡΑΜΜΑ ΜΙΑΣ ΕΙΚΟΝΑΣ

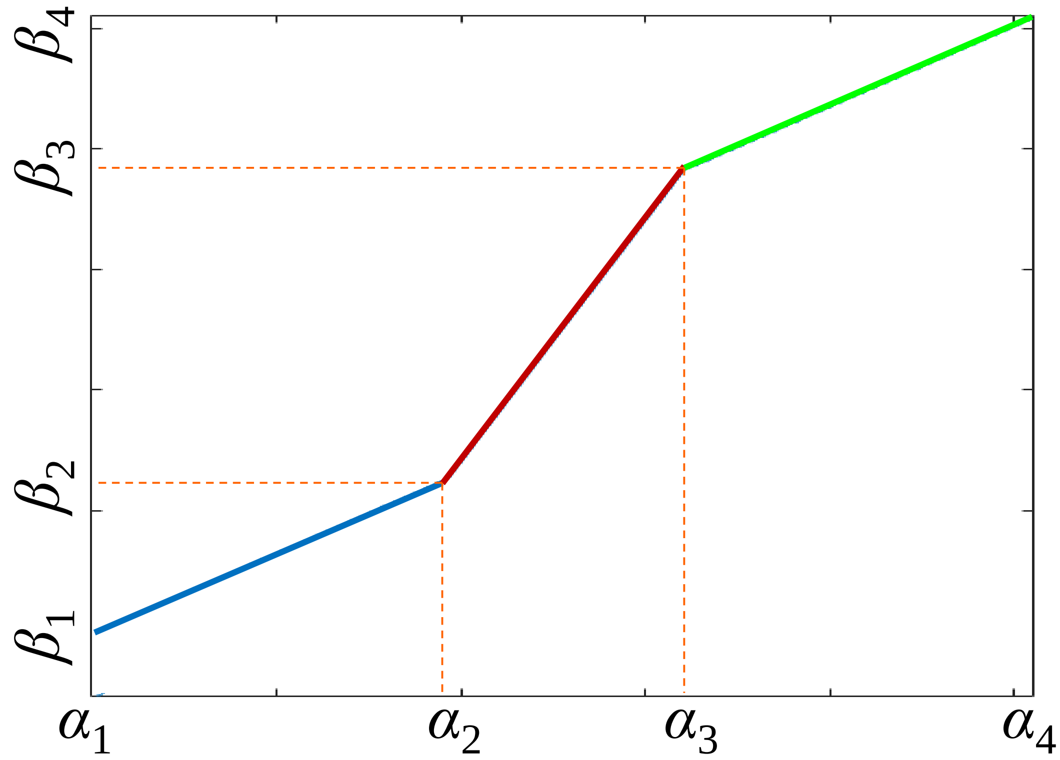
- Το ιστόγραμμα *h* της εικόνας *I* μας δείχνει τη συχνότητα κάθε έντασης του γκρίζου *g* στην εικόνα.
- Η τιμή του ιστογράμματος *h*(*g*), για την ένταση *g* ισούται με τον αριθμό των pixels της εικόνας *I* που έχουν ένταση *g.*
- Για μια εικόνα 8-bit το ιστόγραμμα *h* έχει *g=1..*256 τιμές συχνοτήτων από 0-255.

- $h_{\mathbf{I}}(g)$ = αριθμός pixels στην εικόνα **I** με τιμή g-1

In Matlab an array of length *n* has indices from 1 to *n*. In many computer languages, *e.g.* "C" or "C++" an n-element array is indexed from 0 to *n*-1.

Τμηματικά Γραμμική Μεταβολή Ιστογράμματος

Η Τμηματικά Γραμμική Μεταβολή Ιστογράμματος πραγματοποιείται με βάση μια τμηματικά γραμμική σχέση μετασχηματισμού όπως φαίνεται στο παρακάτω σχήμα:



TEI Κρήτης
Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Τμηματικά Γραμμική Μεταβολή Ιστογράμματος

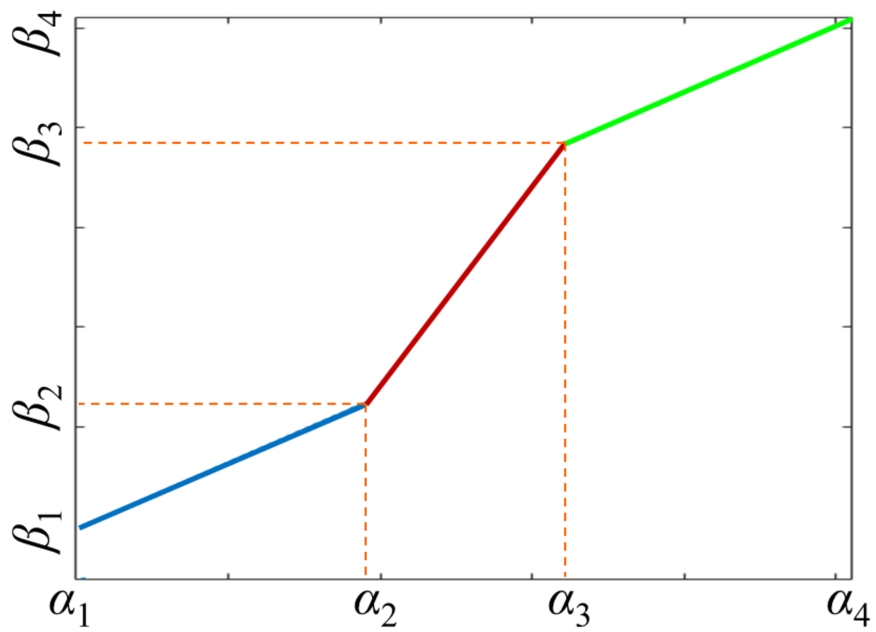
Η σχέση μετασχηματισμού μπορεί να περιγραφεί (για i=1:3) με την παρακάτω εξίσωση:

$$y = \frac{\beta_{i+1} - \beta_i}{\alpha_{i+1} - \alpha_i}(x - \alpha_i) + \beta_i$$

$$y_1 = \frac{\beta_2 - \beta_1}{\alpha_2 - \alpha_1}(x - \alpha_1) + \beta_1$$

$$y_2 = \frac{\beta_3 - \beta_2}{\alpha_3 - \alpha_2}(x - \alpha_2) + \beta_2$$

$$y_3 = \frac{\beta_4 - \beta_3}{\alpha_4 - \alpha_3}(x - \alpha_3) + \beta_3$$



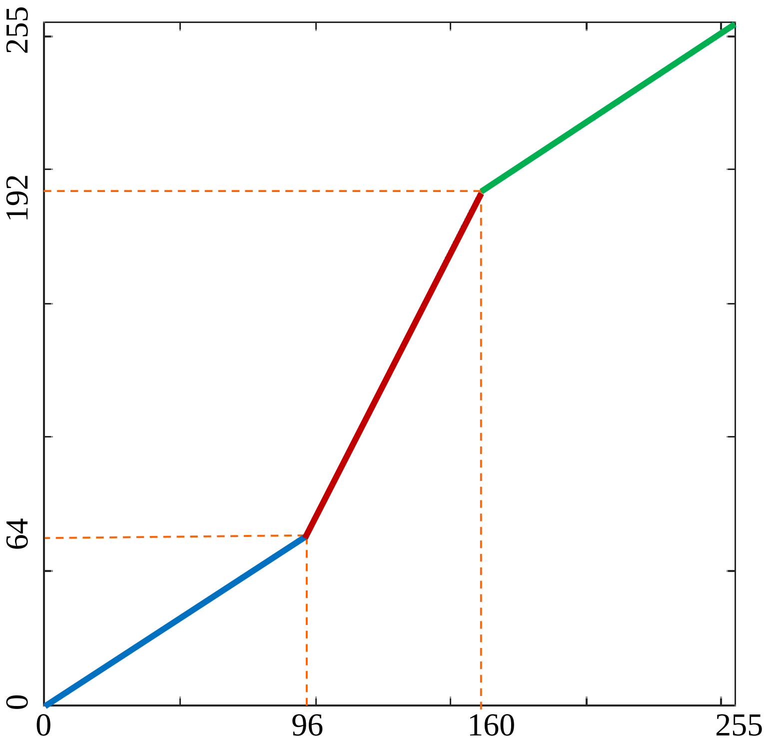
Τμηματικά Γραμμική Μεταβολή Ιστογράμματος

Ας δούμε το παρακάτω παράδειγμα

$$y_1 = \frac{64-0}{96-0}(x-0)+0$$

$$y_2 = \frac{192-64}{160-96}(x-96)+64$$

$$y_3 = \frac{255-192}{255-160}(x-160)+192$$



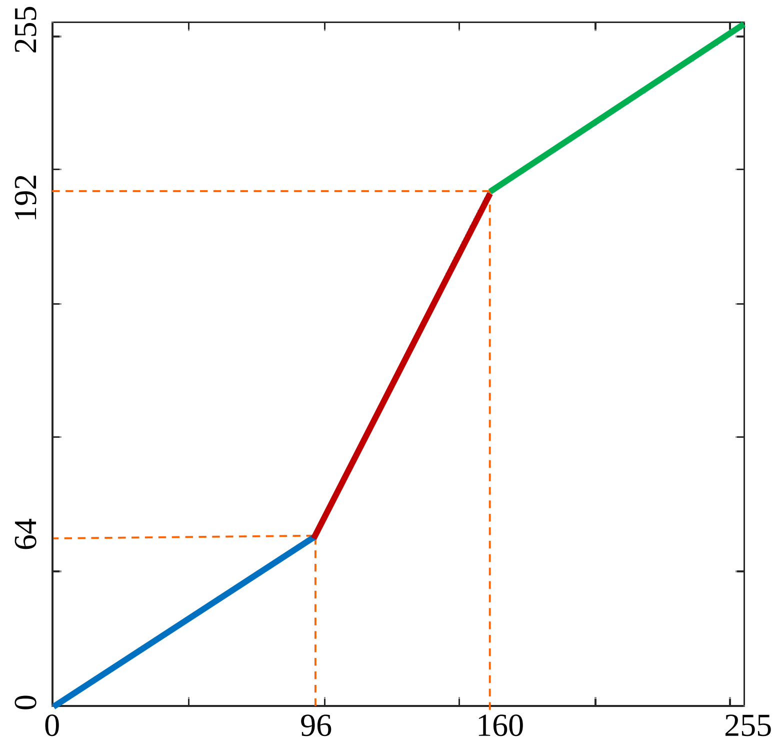
Τμηματικά Γραμμική Μεταβολή Ιστογράμματος

Ας δούμε το παρακάτω παράδειγμα

$$y_1 = 0.6667 \cdot x$$

$$y_2 = 2 \cdot x - 128$$

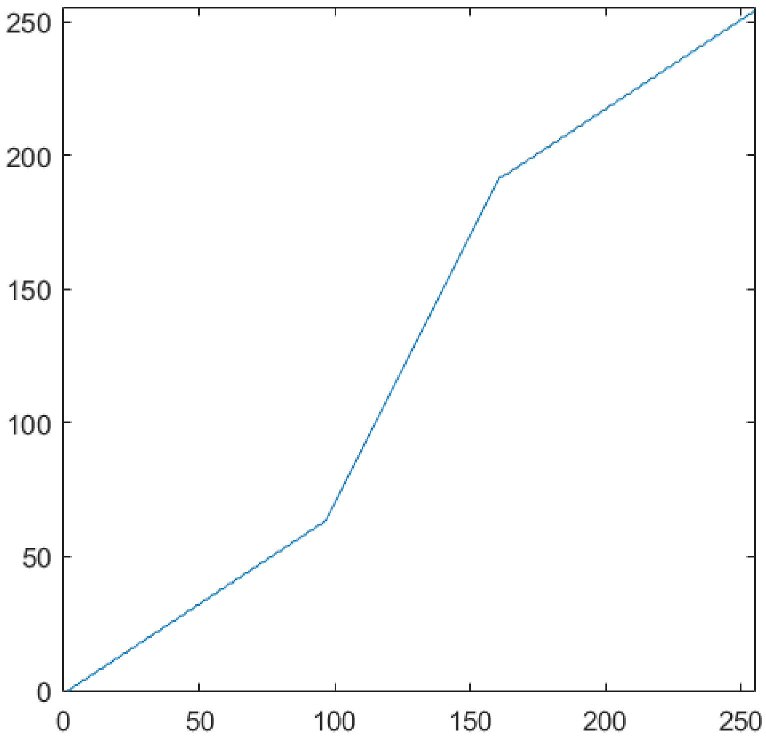
$$y_3 = 0.6632 \cdot x + 85.8947$$



Παρακάτω υλοποιούμε τον μετασχηματισμό σε Matlab

Τμηματικά Γραμμική Μεταβολή Ιστογράμματος

```
y1=64/96*[0:95];
y2=2*[96:160]-128;
y3=0.6632*[161:255]+85.8947;
T=uint8(floor([y1 y2 y3]));
plot(T)
```



Τμηματικά Γραμμική Μεταβολή Ιστογράμματος

```
y1=64/96*[0:95];
y2=2*[96:160]-128;
y3=0.6632*[161:255]+85.8947;
T=uint8(floor([y1 y2 y3]));
plot(T)

I=imread('pout.tif');
I2=T(I);
subplot(1,2,1), imshow(I), title('original image');
subplot(1,2,2), imshow(I2), title('Histogram stretched image');
```

Τμηματικά Γραμμική Μεταβολή Ιστογράμματος

```
y1=64/96*[0:95];
y2=2*[96:160]-128;
y3=0.6632*[161:255]+85.8947;
T=uint8(floor([y1 y2 y3]));
plot(T)

I=imread('moon.tiff');
I2=T(I+1);
subplot(1,2,1), imshow(I), title('original image');
subplot(1,2,2), imshow(I2), title('Histogram stretched image');
```

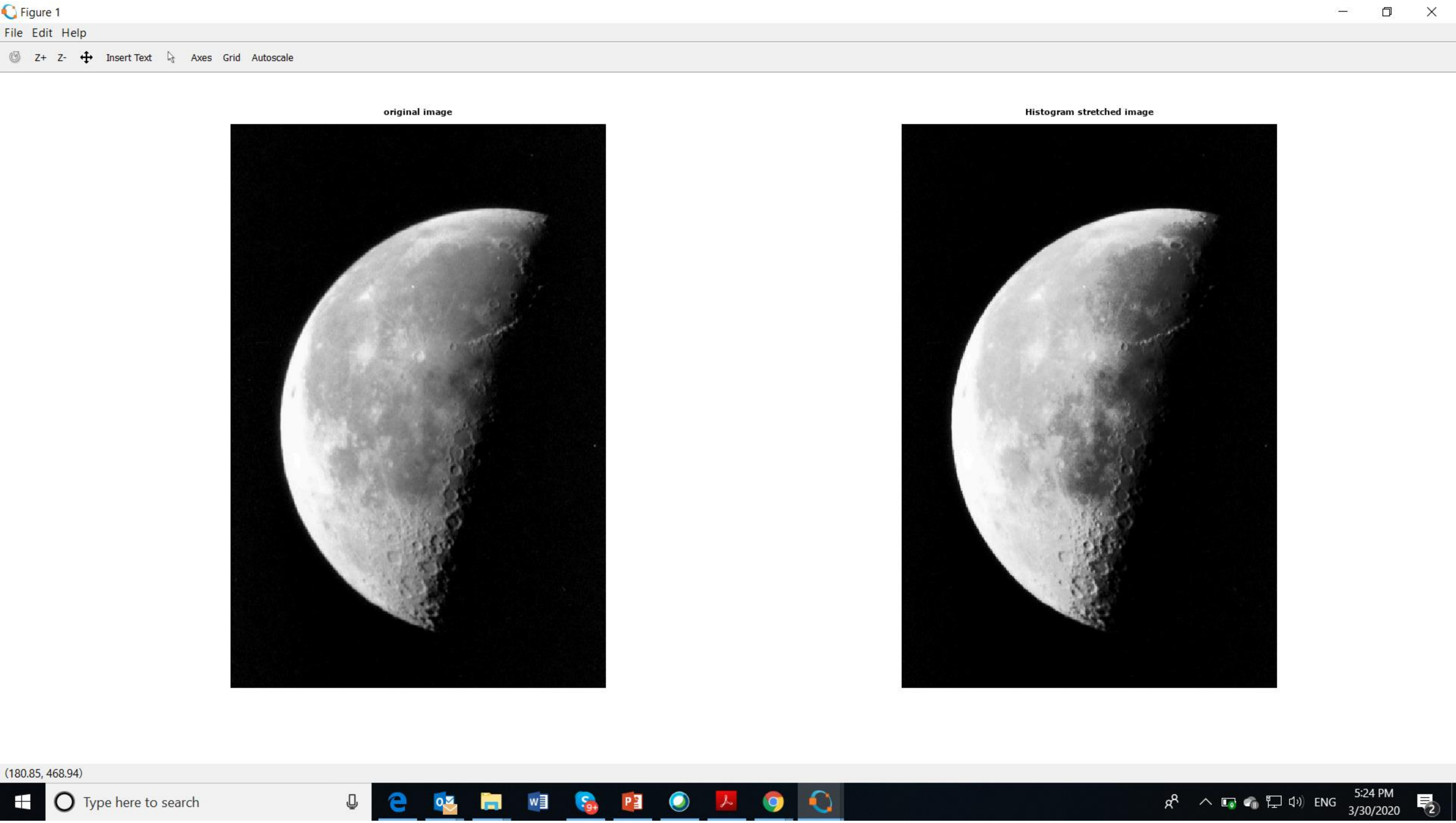
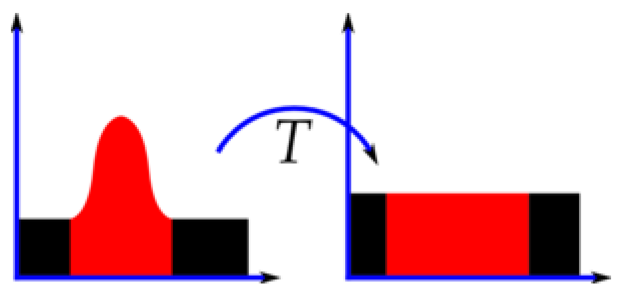


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
original image
Histogram stretched image
(180.85, 468.94)
Type here to search
ENG
5:24 PM
3/30/2020

ΕΞΙΣΟΡΡΟΠΗΣΗ ΙΣΤΟΓΡΑΜΜΑΤΟΣ

- Η τεχνική της εξισορρόπησης ιστογράμματος (histogram equalization) μετασχηματίζει τις γκρι φωτεινότητες μιας εικόνας έτσι ώστε αυτές να κατανέμονται ομοιόμορφα σ' όλη την κλίμακα φωτεινοτήτων.
- Η εικόνα που προκύπτει με τον τρόπο αυτό είναι αυξημένης αντίθεσης σε σχέση με την αρχική.



https://en.wikipedia.org/wiki/Histogram_equalization

ΕΞΙΣΟΡΡΟΠΗΣΗ ΙΣΤΟΓΡΑΜΜΑΤΟΣ

Για την ανάπτυξη της μεθόδου έστω ότι έχουμε μια γκρι εικόνα *I*, διαστάσεων $M_{γραμμες}$ x $N_{στήλες}$ με L αποχρώσεις του γρι: {0,...,L-1}.

Έστω $h(g) = ng$, το ιστόγραμμα της εικόνας *I*, όπου g = 0..L-1

Ουσιαστικά το n_g μας δίνει τη συχνότητα εμφάνισης της κάθε τιμής g στην εικόνα. Αν διαιρεθεί με τον συνολικό αριθμό pixels μας δίνει την πιθανότητα εμφάνισης του κάθε pixel:

$$p(g) = \frac{n_g}{M \cdot N}$$

Η γραφική απεικόνιση του $p(g)$ έναντι του g είναι το γνωστό ιστόγραμμα

ΕΞΙΣΟΡΡΟΠΗΣΗ ΙΣΤΟΓΡΑΜΜΑΤΟΣ

Ο μετασχηματισμός *T* ονομάζεται εξισορρόπηση ιστογράμματος (ΕΙ) και μας δίνει νέο ιστόγραμμα εικόνας:

$$T(g) = (L-1)\sum_{g'=0}^{g} p(g')$$

ή

$$T(g) = \frac{(L-1)}{M \cdot N}\sum_{g'=0}^{g} n_{g'}$$

Παράδειγμα εξισορρόπησης ιστογράμματος

Μια εικόνα 3-bit με διαστάσεις 64x64 έχει:

g	$h(g)$	$p(g)=h(g)/MN$
0	790	0.19
1	1023	0.25
2	850	0.21
3	656	0.16
4	329	0.08
5	245	0.06
6	122	0.03
7	81	0.02

Εφαρμόζοντας την εξίσωση ΕΙ:

$$T(g) = (L-1)\sum_{g'=0}^{g} p(g')$$

$T(0) = (8-1)\sum_{g=0}^{0} p(0) = 7 \cdot 0.19 = 1.33 \rightarrow \mathbf{1}$

$T(1) = 7\sum_{g=0}^{1} p(g) = 7 \cdot (0.19 + 0.25) = 3.08 \rightarrow \mathbf{3}$

$T(2) = 7\sum_{g=0}^{2} p(g) = 7 \cdot (0.19 + 0.25 + 0.21) = 4.55 \rightarrow \mathbf{5}$

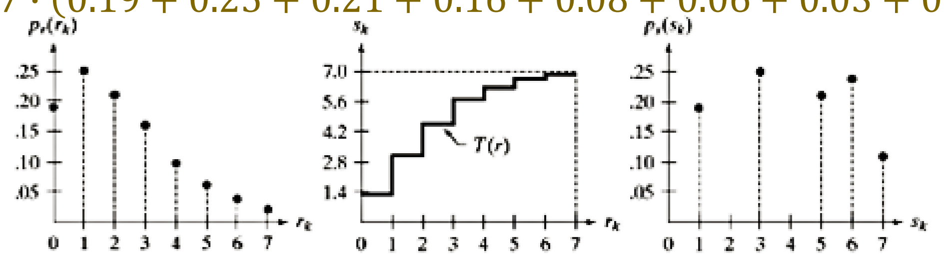
$T(3) = 7\sum_{g=0}^{3} p(g) = 7 \cdot (0.19 + 0.25 + 0.21 + 0.16) = 5.67 \rightarrow \mathbf{6}$

$T(4) = 7\sum_{g=0}^{4} p(g) = 7 \cdot (0.19 + 0.25 + 0.21 + 0.16 + 0.08) = 6.23 \rightarrow \mathbf{6}$

$T(5) = 7\sum_{g=0}^{5} p(g) = 7 \cdot (0.19 + 0.25 + 0.21 + 0.16 + 0.08 + 0.06) = 6.65 \rightarrow \mathbf{7}$

$T(6) = 7\sum_{g=0}^{6} p(g) = 7 \cdot (0.19 + 0.25 + 0.21 + 0.16 + 0.08 + 0.06 + 0.03) = 6.86 \rightarrow \mathbf{7}$

$T(7) = 7\sum_{g=0}^{7} p(g) = 7 \cdot (0.19 + 0.25 + 0.21 + 0.16 + 0.08 + 0.06 + 0.03 + 0.02) = 7 \rightarrow \mathbf{7}$



TEI Κρήτης
Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

“Digital Image Processing”, Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002

Παράδειγμα εξισορρόπησης ιστογράμματος

Μια εικόνα 3-bit με διαστάσεις 64x64=4096 pixel

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

Μετά τον μετασχηματισμό ΕΙ βρίσκουμε τις νέες συχνότητες:

$$T(0) \rightarrow \mathbf{1} \Rightarrow \boldsymbol{h'(1) = 790}, \quad \boldsymbol{p'(1) = 0.19}$$

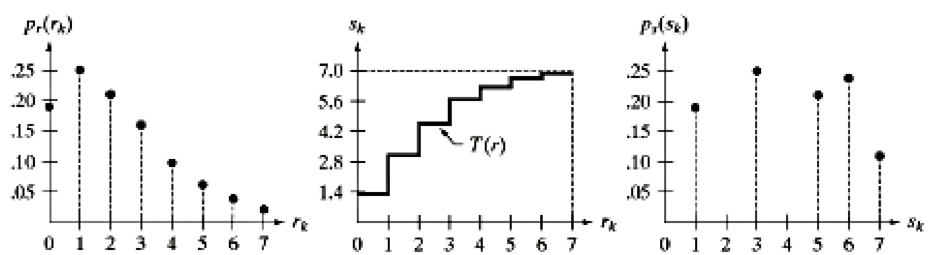
$$T(1) \rightarrow \mathbf{3} \Rightarrow \boldsymbol{h'^{(3)} = 1023}, \quad \boldsymbol{p'(3) = 0.25}$$

$$T(2) \rightarrow \mathbf{5} \Rightarrow \boldsymbol{h'(5) = 850}, \quad \boldsymbol{p'(5) = 0.21}$$

$$\left.\begin{matrix} T(3) \rightarrow \mathbf{6} \\ T(4) \rightarrow \mathbf{6} \end{matrix}\right. \Rightarrow \boldsymbol{h'(6) = 656 + 329 = 985}, \ \boldsymbol{p'(6) = \frac{985}{4096} = 0.24}$$

$$\left.\begin{matrix} T(5) \rightarrow \mathbf{7} \\ T(6) \rightarrow \mathbf{7} \\ T(7) \rightarrow \mathbf{7} \end{matrix}\right. \Rightarrow \boldsymbol{h'(7) = 245 + 122 + 81 = 448}, \ \boldsymbol{p'(7) = \frac{448}{4096} = 0.11}$$

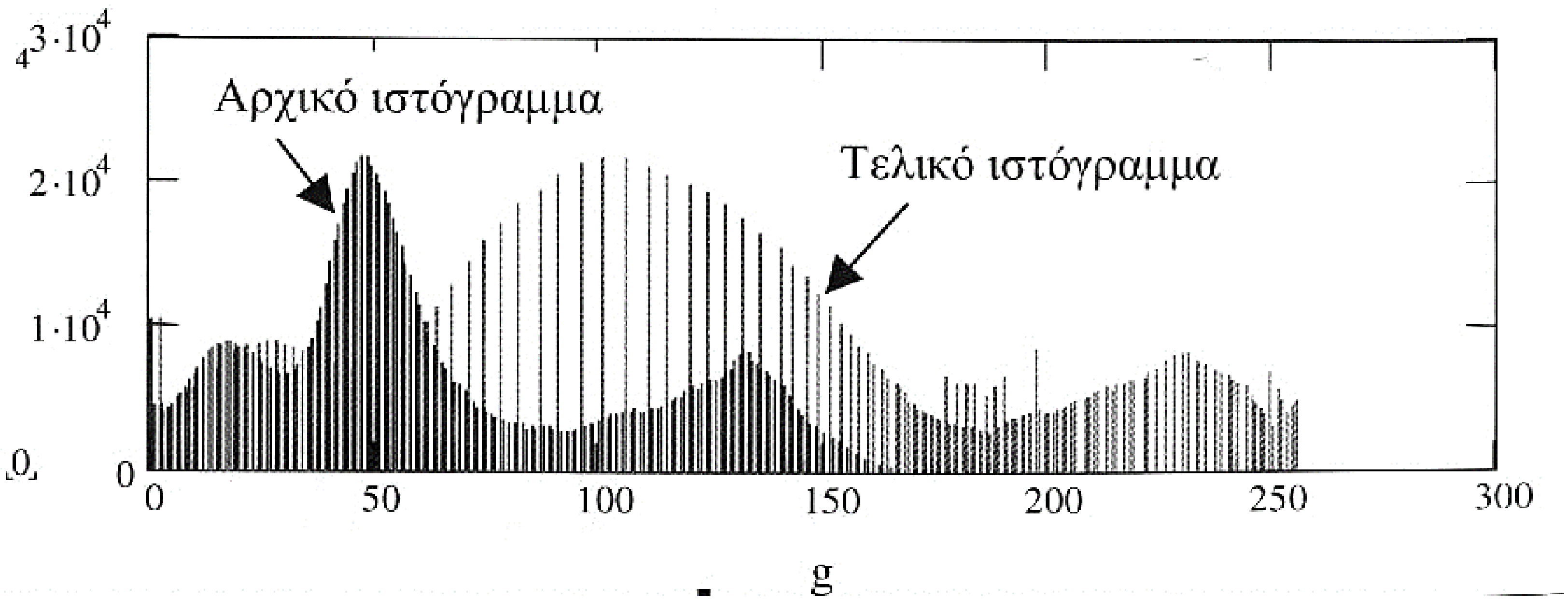
SOS! Εξαιτίας της διακριτοποίησης το ‘εξισορροπημένο’ ιστόγραμμα δεν είναι επίπεδο, σε κάθε περίπτωση όμως είναι διευρυμένο!



“Digital Image Processing”, Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002

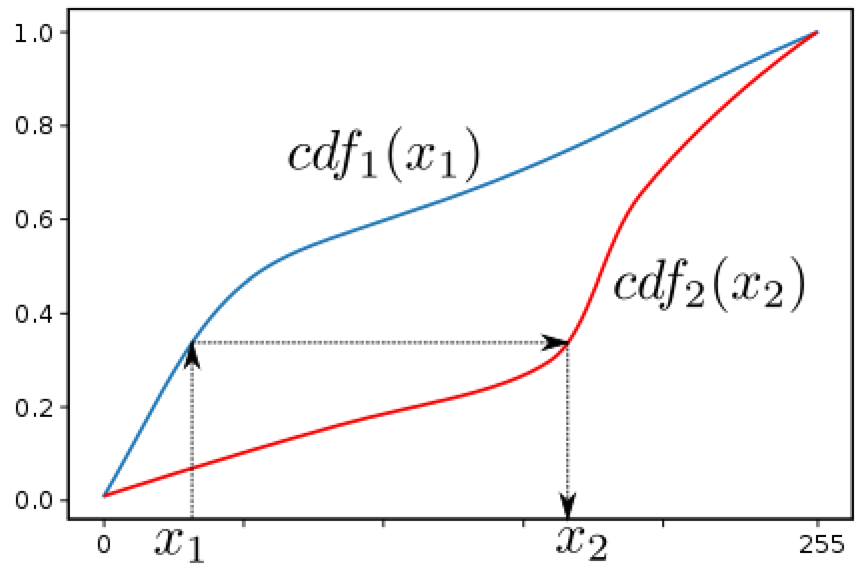
ΕΞΙΣΟΡΡΟΠΗΣΗ ΙΣΤΟΓΡΑΜΜΑΤΟΣ

Η διαδικασία εξισορρόπησης ιστογράμματος που περιγράψαμε αναφέρεται ως “ολική εξισορρόπηση ιστογράμματος” (global histogram equalization) σε αντίθεση με τεχνικές τοπικής εξισορρόπησης ιστο-ράμματος (local histogram equalization).



Histogram Matching

- histogram matching or histogram specification is the transformation of an image so that its histogram matches a specified histogram
- The well-known histogram equalization method is a special case in which the specified histogram is uniformly distributed.



https://en.wikipedia.org/wiki/Histogram_matching

Matching methodology

- Consider a grayscale input image X. It has a probability density function pr(r), where r is a grayscale value, and pr(r) is the probability of that value. This probability can easily be computed from the histogram of the image by

$$p_r(r_j) = \frac{n_j}{n}$$

- Where nj is the frequency of the grayscale value rj, and n is the total number of pixels in the image.
- Now consider a desired output probability density function $p_z(z)$. A transformation of $p_r(r)$ is needed to convert it to $p_z(z)$.

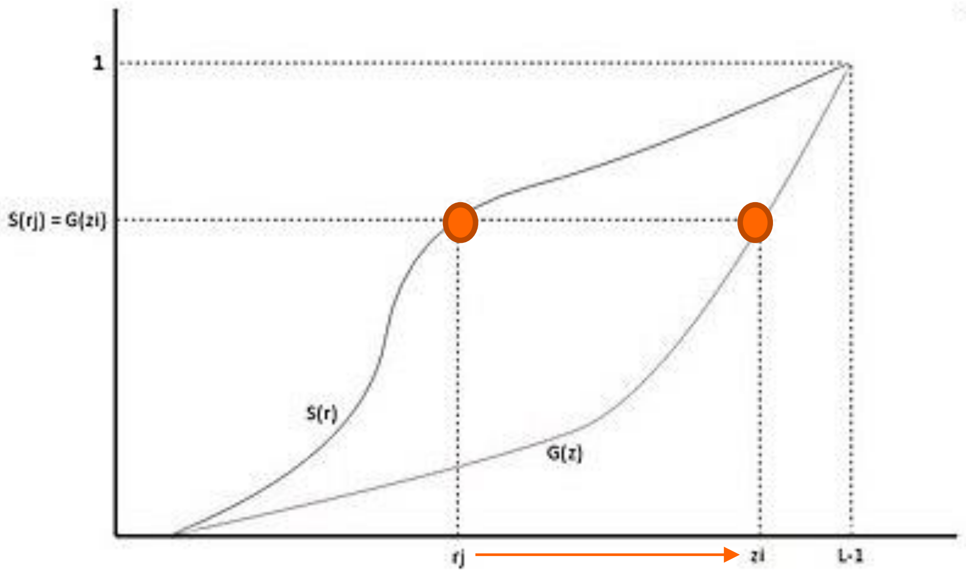
https://en.wikipedia.org/wiki/Histogram_matching

Matching methodology

- Now consider a desired output probability density function pz(z). A transformation of pr(r) is needed to convert it to pz(z).
- Input image CDF matched to desired output CDF
- Each pdf (probability density function) can easily be mapped to its cumulative distribution function by

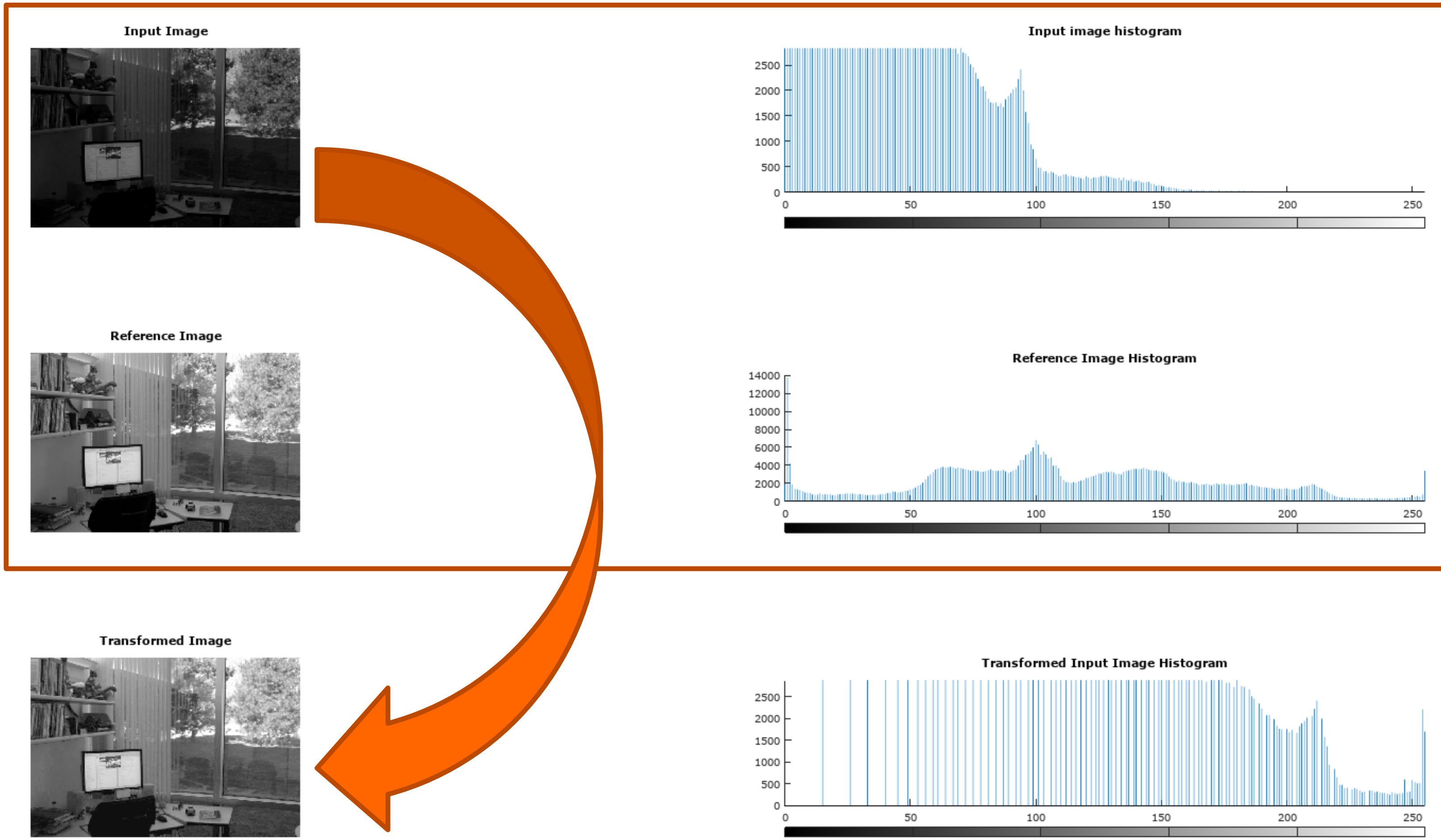
$$S(r_k) = \sum_{j=0}^{k} p_r(r_j), \qquad k = 0, 1, 2, 3, \ldots$$

$$G(z_k) = \sum_{j=0}^{k} p_z(z_j), \qquad k = 0, 1, 2, 3, \ldots, L$$



- Where L is the total number of gray level (256 for a standard image).
- The idea is to map each r value in X to the z value that has the same probability in the desired pdf i.e. $S(r_j) = G(z_i)$ or $z = G^{-1}(S(r))$.

Examples



Μετασχηματισμός αρχικής εικόνας (input) με histogram matching με βάση την εικόνα αναφοράς (ref)

End of today's lecture

Thank you for your attention!