



ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΚΟΝΑΣ

Αναφορά 1^{ης} Εργαστηριακής Άσκησης




















«ΧΡΩΜΑΤΙΚΑ ΜΟΝΤΕΛΑ»



 Καραμαϊλής Παντελής 2016030040
 Βλάχος Κωνσταντίνος 2016030042
 Γαλάνης Μιχάλης 2016030036

ΠΕΡΙΕΧΟΜΕΝΑ

*Οι σύνδεσμοι για τις παρακάτω ενότητες είναι διαδραστικοί.
Πιέστε πάνω στο επιθυμητό τμήμα της άσκησης για τη μετάβαση σε αυτό.*

	ΕΙΣΑΓΩΓΗ.....	2
	Σκοπός Εργαστηρίου	2
	1 – GRAYSCALE CONVERSION.....	2
	Επισκόπηση & Θεωρητική Ανάλυση.....	2
	Υλοποίηση	2
	Αποτελέσματα & Συμπεράσματα.....	3
	2 – HSI TO RGB MODEL CONVERSION.....	3
	Επισκόπηση & Θεωρητική Ανάλυση.....	3
	Υλοποίηση	4
	Αποτελέσματα & Συμπεράσματα.....	5
	3 – IMAGE DESATURATION	5
	Επισκόπηση & Θεωρητική Ανάλυση.....	5
	Υλοποίηση	5
	Αποτελέσματα & Συμπεράσματα.....	6
	4 – WHITE BALANCE TECHNIQUE.....	6
	Επισκόπηση & Θεωρητική Ανάλυση.....	7
	Υλοποίηση	7
	Αποτελέσματα & Συμπεράσματα.....	8
	ΠΑΡΑΡΤΗΜΑ – ΚΩΔΙΚΑΣ	8

ΕΙΣΑΓΩΓΗ

Σκοπός Εργαστηρίου

Σκοπός της άσκησης αυτής είναι η εξοικείωσή μας με τα χρωματικά μοντέλα στις ψηφιακές εικόνες. Θα μελετήσουμε συγκεκριμένα το:

- **RGB model:** που κάθε pixel μιας εικόνας περιγράφεται απόλυτα από το συνδυασμό του κόκκινου, πράσινου και μπλε καναλιού.
- **HSI model:** που κάθε pixel μιας εικόνας περιγράφεται από την απόχρωση (H), τον διαποτισμό (S), και την έντασή του (I).

Θα ασχοληθούμε επίσης με διάφορες τεχνικές όπως white balance και grayscaling.

1 – GRAYSCALE CONVERSION

Στο πρώτο ερώτημα μας ζητήθηκε να διαβάσουμε μια 8-bit εικόνα και να τη μετατρέψουμε σε εικόνα αποχρώσεων του γκρι.

Επισκόπηση & Θεωρητική Ανάλυση

Η μαθηματική ανάλυση μιας indexed (**m x n**) εικόνας περιγράφεται από έναν πίνακα **X** και έναν πίνακα **map** (χρωματική παλέτα). Ο πίνακας **X** είναι μεγέθους (**m x n**) και περιέχει ως τιμές διευθύνσεις στον (**m x 3**) πίνακα **map** που αυτός με τη σειρά του περιέχει «πιθανές τριάδες» τιμών RGB.

Για τη μετατροπή της εικόνας σε αποχρώσεων του γκρι, πρέπει τα τρία χρωματικά κανάλια να περιέχονται σε ίση ποσότητα. Ένας επαληθευμένος τρόπος σήμερα είναι το ITU standard που χρησιμοποιεί έναν ελαφρά «προσαρμοσμένο» αριθμητικό μέσο όρο που έχει τα βέλτιστα αποτελέσματα με την ευαισθησία του ανθρώπινου ματιού και παρουσιάζεται παρακάτω:

$$Y = (222 * R + 707 * G + 71 * B) / 1000 \quad (\text{ITU standard})$$

Υλοποίηση

Στο matlab, η διαδικασία μετατροπής της εικόνας σε πίνακες χρωματικών τιμών επιτυγχάνεται με τη συνάρτηση **imread()**. Αφού εξάγουμε τους πίνακες **X** και **map**, η μετατροπή γίνεται ως εξής.

Κατασκευάζουμε αρχικά μια καινούργια παλέτα (**grayscale_map**). Στη συνέχεια, καθ' όλο το μήκος της παλιάς παλέτας βρίσκουμε τις τιμές του κάθε καναλιού και

υπολογίζουμε το μέσο όρο που παρουσιάστηκε προηγουμένως. Τέλος αποθηκεύουμε το αποτέλεσμα στη καινούργια παλέτα και τελικά παράγουμε την grayscale εικόνα με την εντολή `imwrite()`.

Μετάβαση στον κώδικα MATLAB ➡

Αποτελέσματα & Συμπεράσματα

ΑΡΧΙΚΗ ΕΙΚΟΝΑ



ΕΙΚΟΝΑ ΑΠΟΧΡΩΣΕΩΝ ΓΚΡΙ



Επειδη η RGB εικόνα είναι τύπου `uint8`, το αντίστοιχο διάστημα τιμών είναι το $[0,255]$. Για αυτό το `grayscale_map` που παράχθηκε έχει μέγεθος 256×3 . Η πρώτη διάσταση περιγράφει τις δυνατές τιμές του κάθε καναλιού, ενώ η δεύτερη αφορά κάθε κανάλι.

2 – HSI TO RGB MODEL CONVERSION

Το δεύτερο ερώτημα απαιτούσε να κατασκευάσουμε μια παλέτα 256 χρωμάτων σύμφωνα με το HSI μοντέλο. Στη συνέχεια κληθήκαμε να μετατρέψουμε τα χρώματα αυτά σε RGB και να δημιουργήσουμε την παλέτα ως μια νέα εικόνα.

Επισκόπηση & Θεωρητική Ανάλυση

Το χρωματικό μοντέλο HSI αποτελείται από 3 συνιστώσες. Το Hue (H), Saturation (S) και Intensity (I). Το πρώτο είναι η απόχρωση και περιγράφει ένα γνήσιο χρώμα. Ο διαποτισμός παριστάνει την ανάμειξη αυτού του χρώματος με το άσπρο και η ένταση είναι μια εναλλακτική της φωτεινότητας ενός χρώματος που πρακτικά δε μπορεί να υπολογιστεί.

Η μετατροπή του μοντέλου HSI σε RGB αποτελεί ελαφρώς πολύπλοκη διαδικασία, καθώς η τιμή του κάθε καναλιού RGB υπολογίζεται διαφορετικά αναλόγως τη τιμή της απόχρωσης H.

Hue	Κόκκινο	Πράσινο	Μπλε
<60	$m + (M - m)\frac{H}{60}$	m	M
<120	M		$m + (M - m)\frac{120 - H}{60}$
<180		$m + (M - m)\frac{H - 120}{60}$	m
<240	$m + (M - m)\frac{240 - H}{60}$	M	
<300	m		$m + (M - m)\frac{360 - H}{60}$
<360			

Όπου:

$$M = \begin{cases} I(1 + S), & I \leq 0.5 \\ I + S - IS, & I > 0.5 \end{cases}$$

Και:

$$m = 2I - M$$

Υλοποίηση

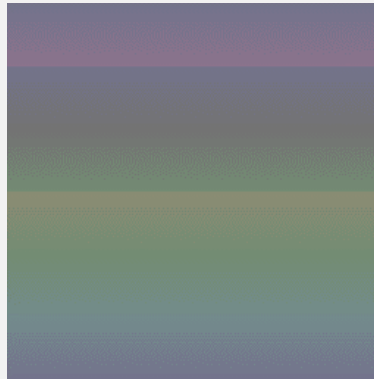
Αρχικά υπολογίζουμε τις τιμές k, S και I που παράγονται μοναδικά σύμφωνα με τον κωδικό αριθμό της ομάδας μας. Δημιουργούμε έναν πίνακα H όπου τοποθετούμε ισάξια 256 κατανεμημένες τιμές μεταξύ 0 και 360. Υπολογίζουμε έπειτα τη μέγιστη και την ελάχιστη τιμή του RGB και τέλος, για κάθε δυνατή τριάδα H, S, I δημιουργούμε την καινούργια τριάδα R, G, B και την αποθηκεύουμε στο νέο μας map_2.

Για να παράγουμε την παλέτα ως μια νέα εικόνα, δημιουργούμε έναν νέο δυσδιάστατο πίνακα X_new με μέγεθος ίσο με τον αριθμό των χρωμάτων που συλλέξαμε έτσι ώστε κάθε γραμμή να αναπαριστά ένα χρώμα.

Μετάβαση στον κώδικα MATLAB ➡

Αποτελέσματα & Συμπεράσματα

ΧΡΩΜΑΤΙΚΗ ΠΑΛΕΤΑ ($S=0.1$, $I=0.5$)



3 – IMAGE DESATURATION

Στο τρίτο ερώτημα μας ζητήθηκε να ξεθωριάσουμε με σταδιακά βήματα μια εικόνα χρησιμοποιώντας το βολικό HSI μοντέλο λόγω του χαρακτηριστικού του διαποτισμού που διαθέτει αλλά εμφανίζοντας κάθε φορά (σε RGB color space) την εικόνα που προκύπτει από τις αλλαγές.

Επισκόπηση & Θεωρητική Ανάλυση

Σε αυτό το ερώτημα δεν υπάρχει κάποιο καινούργιο φαινόμενο της θεωρίας. Αξίζει όμως να τονίσουμε ότι λαμβάνουμε την εικόνα με χαρακτηριστικά RGB και πρέπει να τη μετατρέψουμε σε HSI προτού μεταβάλουμε τη συνιστώσα S και να την επαναφέρουμε σε RGB μοντέλο προκειμένου να την εμφανίσουμε.

Υλοποίηση

Αρχικά διαβάζουμε την επιθυμητή εικόνα και μετατρέπουμε το `map` σε HSI. Αυτή τη φορά έχουμε διαθέσιμες έτοιμες matlab εντολές που μας επιτρέπουν να μετατρέψουμε ένα RGB μοντέλο σε HSI και αντιστρόφως (`rgb2hsv`, `hsv2rgb`). Στη συνέχεια, όσο η εικόνα δε διαθέτει **μόνο** αποχρώσεις του γκρι, μειώνουμε τη συνιστώσα S του `map` κατά ένα συντελεστή 0.2. Τη μετατρέπουμε αργότερα και πάλι σε μοντέλο RGB και την εμφανίζουμε με την εντολή `imshow`.

Μετάβαση στον κώδικα MATLAB ➡

Αποτελέσματα & Συμπεράσματα

Saturation 100%



Saturation 80%



Saturation 60%



Saturation 40%



Saturation 20%



Saturation 0%



⚠ Παρατήρηση: Με κάθε μείωση του Saturation channel στο HSI μοντέλο, η εικόνα φαίνεται να ξεθωριάζει έως ότου τα χρώματα της παλέτας να γίνουν μόνο αποχρώσεις του γκρι.

4 – WHITE BALANCE TECHNIQUE

Στο τέταρτο και τελευταίο υποερώτημα της άσκησης αυτής, έπρεπε να εφαρμόσουμε την τεχνική white balance σε μια εικόνα της επιλογής μας και να εξηγήσουμε τα αποτελέσματά της.

👁️ Επισκόπηση & Θεωρητική Ανάλυση

Ισορροπία λευκού είναι μια τεχνική που χρησιμοποιούμε για να διορθώσουμε τη θερμοκρασία μιας εικόνας. Οι αλγόριθμοι που χρησιμοποιούνται είναι διάφοροι, εμείς όμως επικεντρωνόμαστε σε αυτόν που ακολουθεί την υπόθεση **gray world**, που στην ουσία υποστηρίζουμε ότι ο μέσος όρος του κάθε καναλιού πρέπει να ισοδυναμεί με τους υπόλοιπους. Όταν δεν ικανοποιείται η παραπάνω συνθήκη «διορθώνουμε» τα κανάλια με νέες τιμές ανάλογα με την απόκλιση που είχαν μεταξύ τους. Συγκεκριμένα:

$$R_{ch_new}(x, y) = \frac{G_{avg}}{R_{avg}} R_{ch_old}(x, y)$$

$$G_{ch_new}(x, y) = G_{ch_old}(x, y)$$

$$B_{ch_new}(x, y) = \frac{G_{avg}}{B_{avg}} B_{ch_old}(x, y)$$

Όπου:

$$R_{avg} = \frac{1}{n^2} \sum_{x=1}^n \sum_{y=1}^n R_{ch_old}(x, y)$$

$$G_{avg} = \frac{1}{n^2} \sum_{x=1}^n \sum_{y=1}^n G_{ch_old}(x, y)$$

$$B_{avg} = \frac{1}{n^2} \sum_{x=1}^n \sum_{y=1}^n B_{ch_old}(x, y)$$

📁 Υλοποίηση

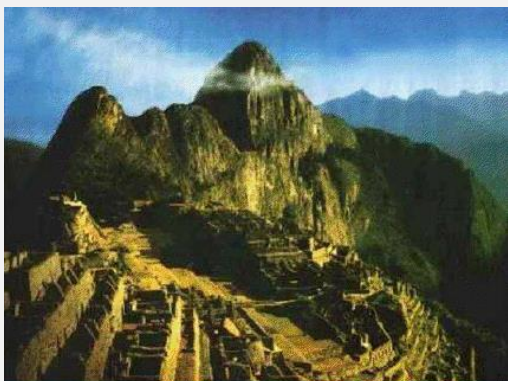
Όπως και με τα άλλα ερωτήματα, ξεκινάμε διαβάζοντας μια επιθυμητή εικόνα. Συνεχίζουμε συγχωνεύουμε τον πίνακα `X_4 & map_4` σε έναν καινούργιο πίνακα `Im`. Χωρίζουμε τα τρία κανάλια σε τρεις ξεχωριστούς πίνακες και στη συνέχεια βρίσκουμε το μέσο όρο του κάθε καναλιού.

Για να εφαρμόσουμε την ισοστάθμιση στην εικόνα μας, βρίσκουμε τους διορθωμένους μέσους όρους με τις σχέσεις που περιγράφηκαν παραπάνω και ενώνουμε πάλι τα χρωματικά κανάλια σε μια ενιαία εικόνα με την εντολή **cat**. Τέλος γράψαμε στο δίσκο τη τελική αυτή εικόνα.

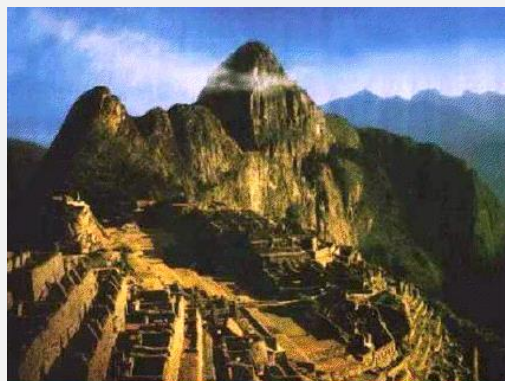
Μετάβαση στον κώδικα MATLAB ➡

Αποτελέσματα & Συμπεράσματα

ΕΙΚΟΝΑ ΧΩΡΙΣ WB



ΕΙΚΟΝΑ ΜΕ WB



Παρατηρούμε ότι με τη τεχνική του white balance, διορθώσαμε την εικόνα και έγινε πιο θερμή για να αντισταθμίσει το έντονο μπλε κανάλι της αρχικής εικόνας.

ΠΑΡΑΡΤΗΜΑ – ΚΩΔΙΚΑΣ

Σε περίπτωση που χρειαστεί ο κώδικας τον παραθέτουμε παρακάτω:

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Exercise 1 - Convert Color to Grayscale Image %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%Image Resolution is 428 x 320
[X1, map1] = imread('image.png');

%Creating new grayscale map
grayscale_map = zeros;

%Applying Grayscale effect
for i=1:length(map1)
    R = map1(i,1); G = map1(i,2); B = map1(i,3); %Acquiring Color Values
    Y = (222*R + 707*G + 71*B)/1000; %Generating shade of gray
    for j = 1:3
        grayscale_map(i,j) = Y; %Applying shades of gray to grayscale_map
    end
end

%Generates Grayscale Image
imwrite(X1, grayscale_map, 'image_grayscale.png');

```

Μετάβαση στα αποτελέσματα της άσκησης 1 

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Exercise 2 - HSI TO RGB MODEL CONVERSION %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%LAB31239720
k = 7 + 2 + 0; %k = 9
S = (1-k/10); %S = 0.1
I = 0.5;

H = zeros;
j = 0;
for i=0:255
    H(i+1) = j;
    j = j + 360/255;
end

%Determining Max & min values
M = I*(1 + S);
m = 2*I - M;

map_2 = zeros; %256 x 3

%Generating RGB Values for each H
for i=1:length(H)
    if (H(i) < 60)
        R = m + (M - m) * (H(i)/60);
        G = m;
        B = M;
    elseif (H(i) < 120)
        R = m;
        G = m;
        B = m + (M - m)*((120-H(i))/60);
    elseif (H(i) < 180)
        R = m;
        G = m + (M - m) * ((H(i)-120)/60);
        B = m;
    elseif (H(i) < 240)
        R = m + (M - m)*((240-H(i))/60);
        G = M;
        B = m;
    elseif (H(i) < 300)
        R = m;
        G = M;
        B = m + (M - m) * ((H(i)-240)/60);
    elseif (H(i) < 360)
        R = m;
        G = m + (M - m)*((360-H(i))/60);
        B = M;
    end

    map_2(i,1) = R;
    map_2(i,2) = G;

```

```

        map_2(i,3) = B;
    end

    %Creating RGB Color Palette
    X_new = zeros(256); %256 x 256
    for i=1:length(X_new)
        X_new(i,:)=i;
    end

    %Generating image based on RGB values
    imwrite(X_new, map_2, 'palette.bmp');

```

Μετάβαση στα αποτελέσματα της άσκησης 2 ➡

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Exercise 3 - IMAGE DESATURATION                                     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[X, map] = imread('image2.png');
map_3 = rgb2hsv(map);

while (mean(map_3(:,2)) > 0)
    figure;
    imshow(X, map)
    for i = 1:length(map_3)
        %map_3(i,2) is saturation channel of HSI model
        if (map_3(i,2) > 0.2)
            map_3(i,2) = map_3(i,2) - 0.2;
        else
            map_3(i,2) = 0;
        end
    end
    map = hsv2rgb(map_3);
end

```

Μετάβαση στα αποτελέσματα της άσκησης 3 ➡

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Exercise 4 - White Balance Technique                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[X_4, map_4] = imread('image.png');

if ~isempty(map_4)
    Im = ind2rgb(X_4, map_4);
end

redChannel = Im(:, :, 1);
greenChannel = Im(:, :, 2);

```

```
blueChannel = Im(:, :, 3);

meanR = mean2(redChannel);
meanG = mean2(greenChannel);
meanB = mean2(blueChannel);

redChannel = ((redChannel) * meanG / meanR);
greenChannel = ((greenChannel));
blueChannel = ((blueChannel) * meanG / meanB);

% Recombine separate color channels into a single, true color RGB image.
Ima = cat(3, redChannel, greenChannel, blueChannel);

imwrite(Ima, 'image_whitebalance.png')
```

Μετάβαση στα αποτελέσματα της άσκησης 4 ➡