

Ψηφιακή Επεξεργασία Εικόνας

Εργασία 1

Function readdng.m.....	2
Function dng2rgb.m.....	3
Interpolation: Method ‘Nearest’.....	3
Bayrer Pattern: ‘bggr’	3
Ιστογράμματα.....	4
Εικόνες.....	5
Bayrer Pattern: ‘gbrg’	7
Ιστογράμματα.....	7
Εικόνες.....	8
Bayrer Pattern: ‘grbg’	9
Ιστογράμματα.....	10
Εικόνες.....	11
Bayrer Pattern: ‘rggb’	13
Ιστογράμματα.....	13
Εικόνες.....	14
Interpolation: Method ‘Linear ’	16
Bayrer Pattern: ‘bggr’	16
Ιστογράμματα.....	17
Εικόνες.....	18
Bayrer Pattern: ‘gbrg’	20
Ιστογράμματα.....	20
Bayrer Pattern: ‘grbg’	23
Ιστογράμματα.....	23
Εικόνες.....	24
Bayrer Pattern: ‘rggb’	26
Ιστογράμματα.....	26
Εικόνες.....	27
Δοκιμή με άλλες εικόνες.....	29
Σχολιασμός ιστογραμμάτων.....	29
Σχολιασμός εικόνων.....	30
Σύγκριση διαφορετικών εικόνων.....	30
Σύγκριση διαφορετικών patterns.....	30
Σύγκριση διαφορετικών μεθόδων.....	30

Η παρούσα εργασία έχει σκοπό την μετατροπή εικόνων από RAW FORMAT σε RGB. Συγκεκριμένα, δουλεύουμε με τις μετρήσεις των αισθητήρων μηχανής Adobe που έχει μετατραπεί σε .DNG (Digital Negative). Υλοποιήθηκε σε matlab με τις εξής συναρτήσεις

Function readdng.m

Η συνάρτηση αυτή δόθηκε στην εκφώνηση και έχει σκοπό κυρίως να παράγει τα δεδομένα της εικόνας, τα οποία θα υποστούν την επεξεργασία. Σαν είσοδο δέχεται το όνομα του αρχείου, που μπορεί να είναι οποιοδήποτε με κατάληξη .DNG. Αρχικά, διαβάζει τη raw εικόνα και παράγει 2D πίνακα με τις μετρήσεις των αισθητήρων. Έπειτα, διαβάζει τα metadata της εικόνας που θα χρειαστούν.

Οι μεταβλητές `y_origin` και `x_origin` έχουν την τιμή των άκρων για το χρήσιμο μέρος της εικόνας και οι μεταβλητές `width` και `height` αντίστοιχα το πλάτος και το ύψος. Όντως στην σειρά 30 του κώδικα, η εικόνα με βάση αυτά τα δεδομένα υφίσταται περικοπή, έτσι ώστε να κρατήσουμε το κομμάτι που είναι πιο χρήσιμο. Επιπλέον οι τιμές μετατρέπονται σε `tupou double` για να μπορέσει να συνεχίσει η επεξεργασία στο matlab.

Οι μεταβλητές `blacklevel` και `whitelevel` χρησιμοποιούνται στη σειρά 38 για την κανονικοποίηση των τιμών του λευκού και του μαύρου στην εικόνα. Θέλουμε όλες οι τιμές να βρίσκονται ενδιάμεσα από αυτές τις 2, καθώς η τιμή `blacklevel` είναι η χαμηλότερη και η `whitelevel` η υψηλότερη.

Οι μεταβλητές `wbcoefs` και `XYZ2Cam` είναι και ουτρούς της συνάρτησης και θα χρησιμοποιηθούν στην επόμενη.

Η εικόνα που παράγεται από αυτή τη συνάρτηση είναι σε `grayscale` και το χρώμα θα εφαρμοστεί αργότερα.



Εικόνα 1: Output της συνάρτησης `readdng (.DNG)`

Function dng2rgb.m

Η συνάρτηση αυτή δέχεται σαν είσοδο τον 2D πίνακα rawim από την προηγούμενη συνάρτηση, τις διαστάσεις του M, N, τον πίνακα XYZ2Cam που χρησιμεύει στη μετατροπή της εικόνας από camera color space στο XYZ color space, αν αντιστραφεί και τον πίνακα wbcoeffs που βοηθάει στην ρύθμιση του white balance στην εικόνα. Αυτούς τους πίνακες τους βρίκουμε στα metadata της εικόνας.

Δέχεται επίσης σαν είσοδο τις μεταβλητές method που δηλώνει τον τρόπο της παρεμβολής και την bayertype που δηλώνει το pattern των χρωμάτων στον αισθητήρα. Η χρήση τους θα εξηγηθεί αργότερα.

Αρχικά, χρησιμοποιώντας την συνάρτηση wbmask από το άρθρο ‘Processing RAW Images in MATLAB’ του Rob Sumner δημιουργείται μια μάσκα με τον πίνακα wbcoeffs στις διαστάσεις της εικόνας και εφαρμόζεται στο rawim. Αφήνουμε το χρώμα πράσινο όπως έχει και τα άλλα 2 τα διορθώνουμε έτσι ώστε να έχουν συγκεκριμένη αναλογία.

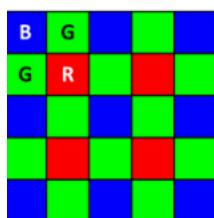
Έπειτα δημιουργούμε 2 2D πλέγματα για τα 3 χρώματα, στις διαστάσεις της εικόνας.

Χρησιμοποιώ τις μεταβλητές x_ones όπου αποθηκεύω τους περιπτούς αριθμούς στην x διάσταση (μέχρι το M), αντίστοιχα το y_ones για την y και οι y_zeros, x_zeros αποθηκεύουν τους άρτιους αριθμούς. Αυτά τα διανύσματα θα με βοηθήσουν στην παρεμβολή.

Interpolation: Method ‘Nearest’

Οι μετρήσεις που έχουμε στο raw image, ανάλογα με το pattern αντιστοιχούν σε ένα μόνο χρώμα για κάθε pixel. Αυτό σημαίνει ότι σε κάθε pixel 2 από τα χρώματα δεν έχουν καμία μέτρηση και πρέπει αυτές τις τιμές να τις βρούμε για να συμπληρώσουμε τα grids red, green, blue. Με τη μέθοδο παρεμβολής ‘nearest’ κάθε pixel χρώματος χωρίς τιμή, παίρνει την τιμή του πιο κοντινού που περιέχει το αντίστοιχο χρώμα.

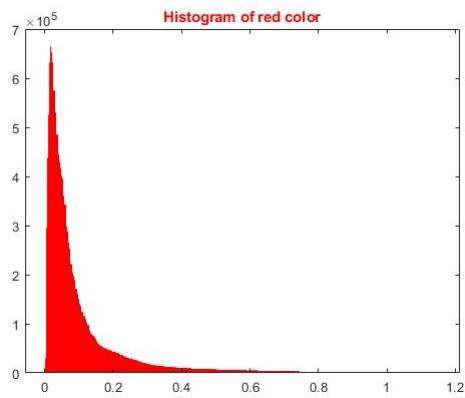
Bayrter Pattern: ‘bggr’



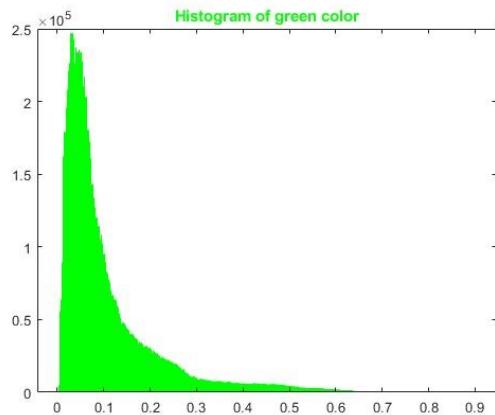
Εικόνα 2: Pattern BGGR

Στο συγκεκριμένο pattern όρισα στον πίνακα red στα στοιχεία που έχουν άρτιο x και y τις αντίστοιχες τιμές του raw image, καθώς θεωρούνται μετρήσεις του κόκκινου. Στον πίνακα blue τα στοιχεία με περιπτό x και y , ενώ στον πίνακα green τα στοιχεία που έχουν περιπτό x και άρτιο y και το αντίθετο. Και έπειτα συμπλήρωσα τις τιμές για τα υπόλοιπα στοιχεία σε κάθε πίνακα δίνοντας την τιμή του πιο κοντινού. Για παράδειγμα στο κόκκινο κάθε τετράδα θα είχε την ίδια τιμή, όπως και το μπλε, ενώ το ίδιο συμβαίνει για το πράσινο σε ζευγάρια. Προέκυψαν τα εξής ιστογράμματα για κάθε χρώμα

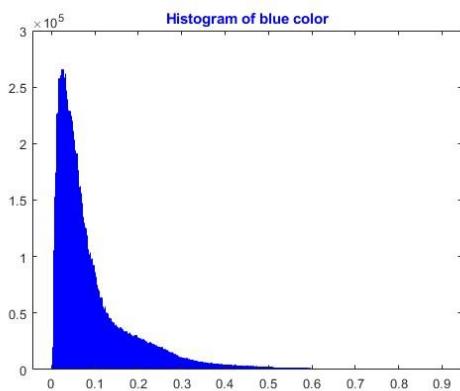
Ιστογράμματα



Εικόνα 3: Ιστόγραμμα για το κόκκινο, με nearest method και BGGR pattern



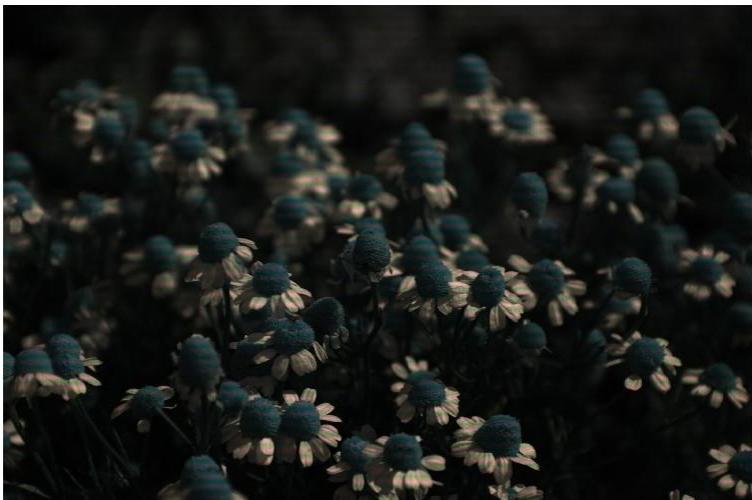
Εικόνα 4: Ιστόγραμμα για το πράσινο, με nearest method και BGGR pattern



Εικόνα 5: Ιστόγραμμα για το μπλε, με nearest method και BGGR pattern

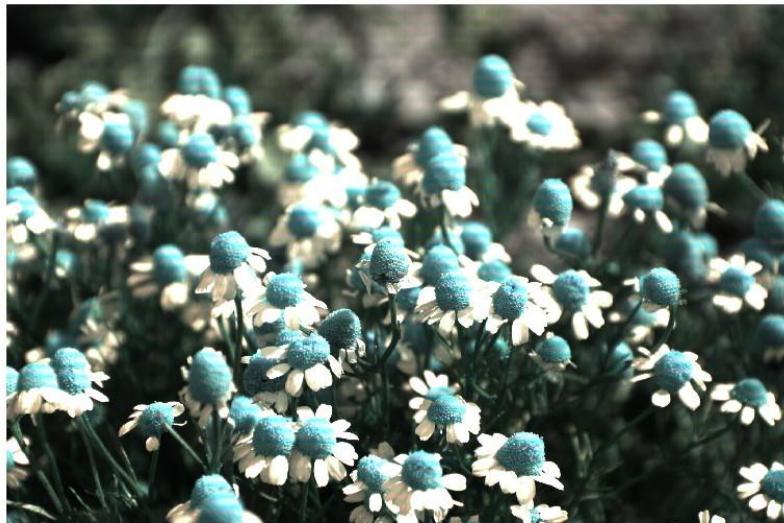
Εικόνες

Στη σειρά 279 τα 3 grids ενώνονται για να δημιουργήσουν την εικόνα Ccam που είναι στο color space της κάμερας.



Εικόνα 6: Ccam, με method nearest και pattern BGGR

Στη σειρά 283 εφαρμόζεται ο πίνακας που μετατρέπει την εικόνα στο XYZ color space και έχουμε την εικόνα Cxyz.



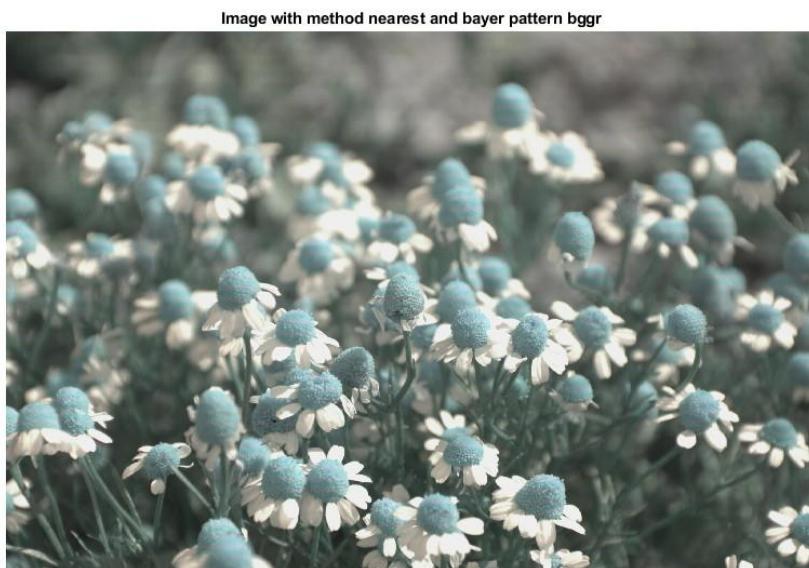
Εικόνα 7: Cxyz, method nearest, pattern BGGR

Αυτό που θέλουμε βέβαια είναι από το color space της κάμερας να μεταβούμε στο RGB, οπότε συνδυάζουμε τον πίνακα που μετατρέπει από camera space στο XYZ και αυτόν που μετατρέπει από XYZ σε RGB και αφού τον κανονικοποιούμε και τον αντιστρέφουμε τον εφαρμόζουμε στην εικόνα Ccam χρησιμοποιώντας τη συνάρτηση imfilter. Φροντίζουμε οι τιμές να είναι ανάμεσα στο 0 και το 1 και έχουμε την εικόνα Clinear.



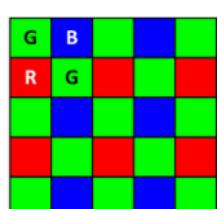
Εικόνα 8: Clinear, method nearest, pattern BGGR

Τέλος χρειάζεται να κάνουμε κάποιες διορθώσεις στην φωτεινότητα. Όπότε ρυθμίζω τη μέση φωτεινότητα, της grayscale εικόνας και εφαρμόζω τον συντελεστή Γάμμα. Έτσι παίρνω την τελική εικόνα Csrgb, η οποία δεν έχει τα πραγματικά χρώματα λόγω του διαφορετικού pattern.



Εικόνα 9: Τελική εικόνα, method nearest, pattern BGGR

Bayrer Pattern: 'gbrg'

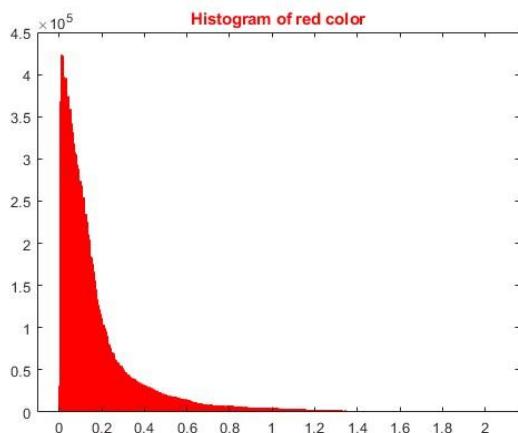


Εικόνα 10 : Pattern GBRG

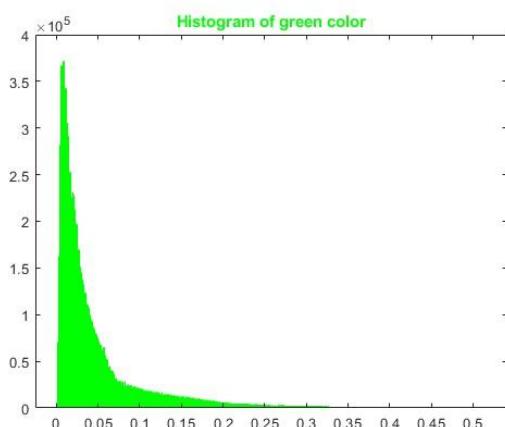
Εδώ τα στοιχεία με άρτιο x και περιττό y είναι τα κόκκινα, αυτά με περιττό x και άρτιο y, τα μπλε, ενώ αυτά με άρτιο x, y, ή περιττό x, y τα πράσινα. Ακολουθεί η ίδια μέθοδος σε τετράδες για τα κόκκινα και τα μπλε και ζευγάρια για τα πράσινα.

Ο κώδικας για παραγωγή των 4 εικόνων είναι ίδιος για κάθε pattern και μέθοδο.

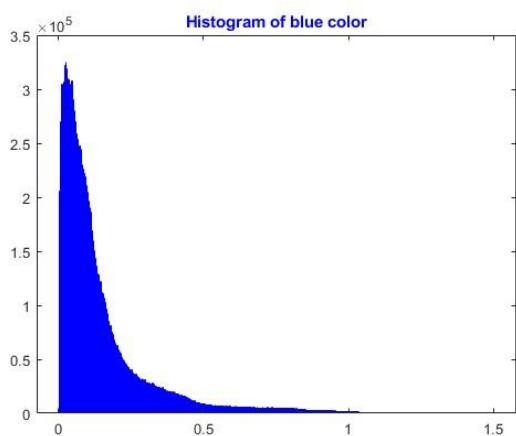
Ιστογράμματα



Εικόνα 11: Ιστόγραμμα για το κόκκινο χρώμα, method nearest, pattern GBRG

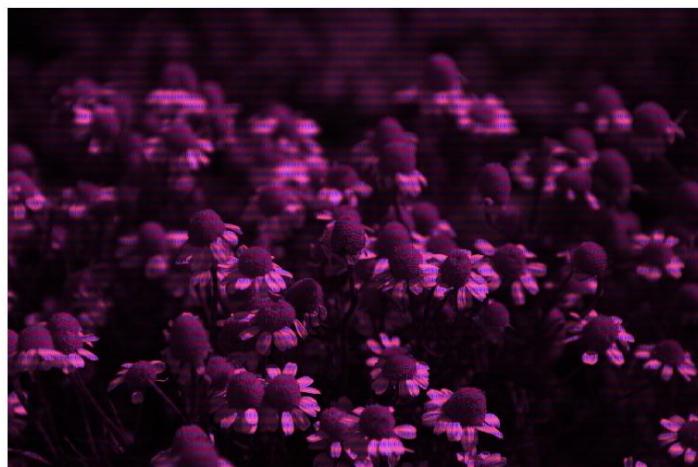


Εικόνα 12: Ιστόγραμμα για το πράσινο χρώμα, method nearest, pattern GBRG



Εικόνα 13: Ιστόγραμμα για το μπλε χρώμα, method nearest, pattern GBRG

Εικόνες



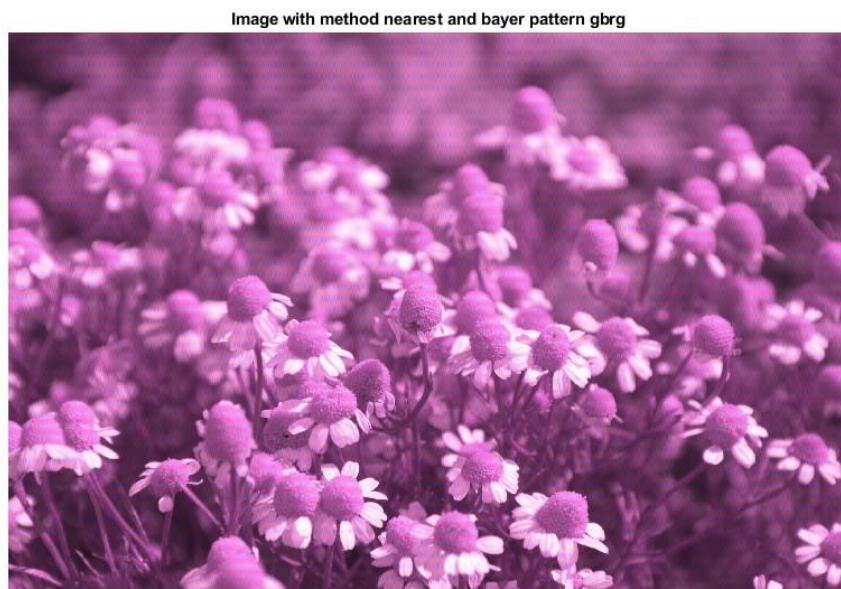
Εικόνα 14: Ccam image, method nearest, pattern GBRG



Εικόνα 15: Cxyz image, method nearest, pattern GBRG



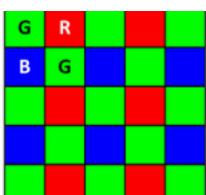
Εικόνα 16: Clinear image, method nearest, pattern GBRG



Εικόνα 17: Τελική εικόνα, method nearest, pattern GBRG

Παρατηρείται ότι τα χρώματα δεν είναι σωστά ουτε τώρα, διότι πάλι το pattern είναι διαφορετικό και δίνονται οι τιμές σε διαφορετικά χρώματα από αυτά που έπρεπε.

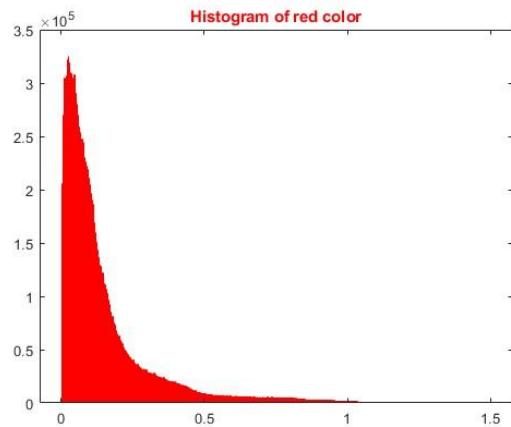
Bayrer Pattern: 'grbg'



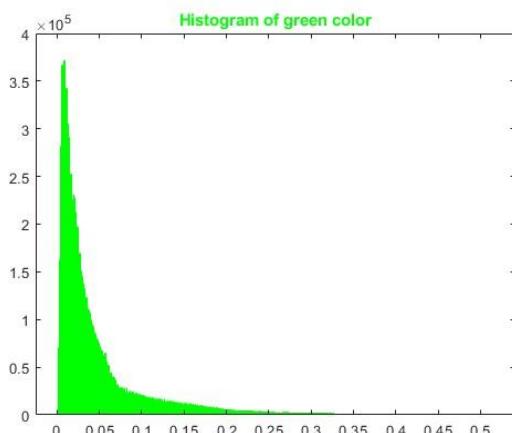
Εικόνα 18: Pattern GRBG

Κόκκινα είναι τα pixels με περιττό x και άρτιο y, μπλε με άρτιο x και περιττό y, ενώ πράσινα τα υπόλοιπα. Η διαδικασία που ακολουθεί είναι ίδια με πριν.

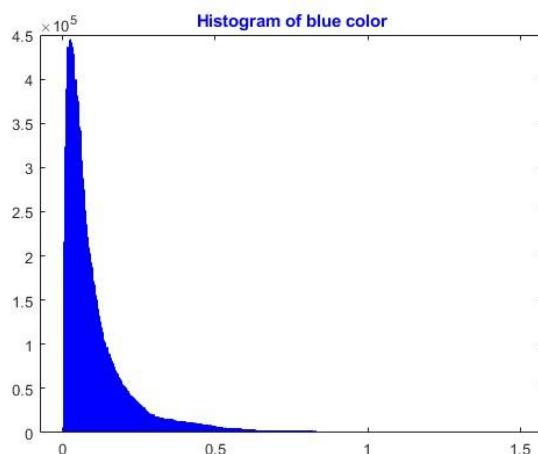
Ιστογράμματα



Εικόνα 19: Ιστόγραμμα για το κόκκινο χρώμα με nearest method και GRBG pattern



Εικόνα 20: Ιστόγραμμα για το πράσινο χρώμα με nearest method και GRBG pattern

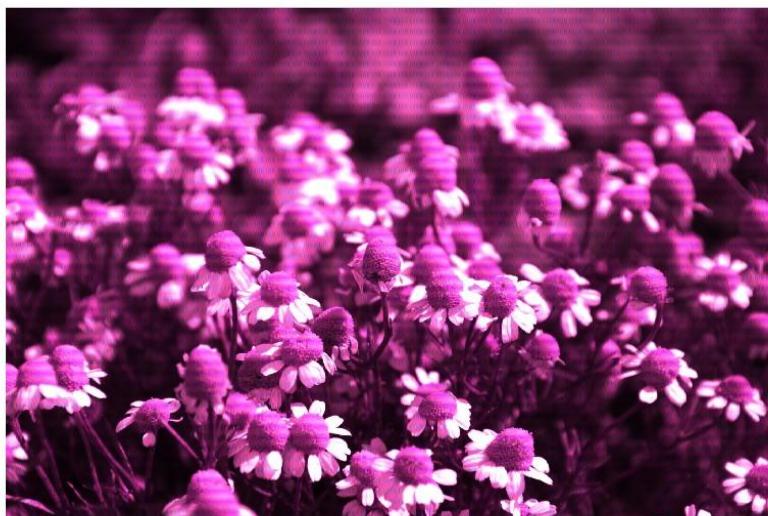


Εικόνα 21: Ιστόγραμμα για το μπλε χρώμα με nearest method και GRBG pattern

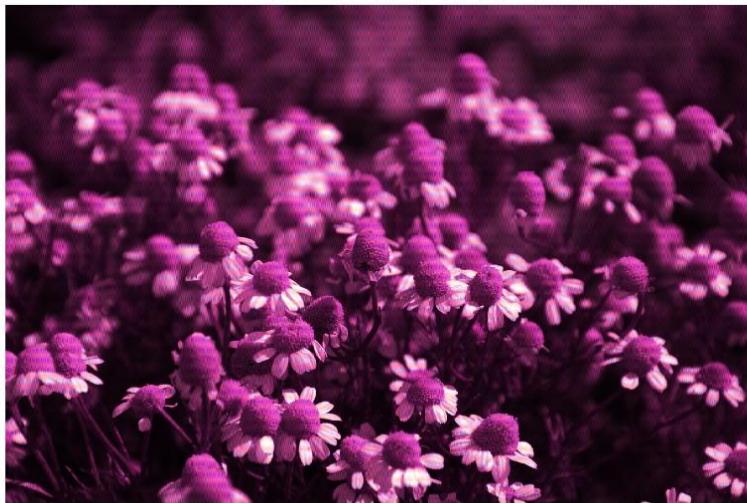
Εικόνες



Εικόνα 22: Ccam image με nearest method και GRBG pattern



Εικόνα 23: Cxyz image με nearest method και GRBG pattern

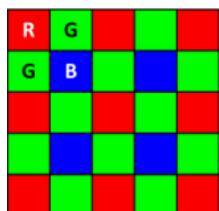


Εικόνα 24: Clinear image με nearest method και GRBG pattern



Εικόνα 25: Τελική εικόνα με nearest method και GRBG pattern

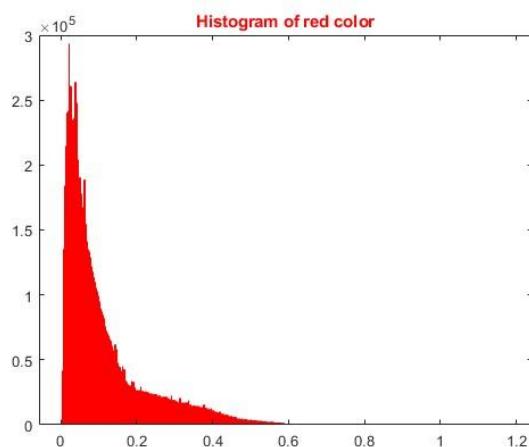
Bayrer Pattern: ‘rggb’



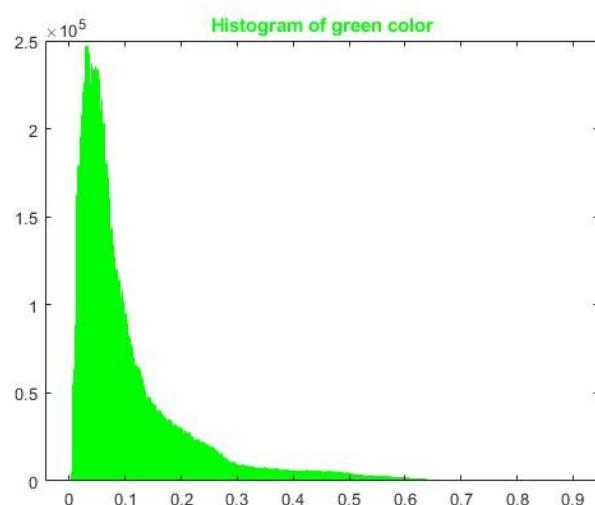
Εικόνα 26: Pattern RGGB

Αυτό το pattern είναι και αυτό με τα σωστά χρώματα, που θα δώσει την σωστή εικόνα. Τα περιττά x,y γίνονται κόκκινα. Τα άρτια x,y μπλε και τα υπόλοιπα πράσινα και η διαδικασία ίδια.

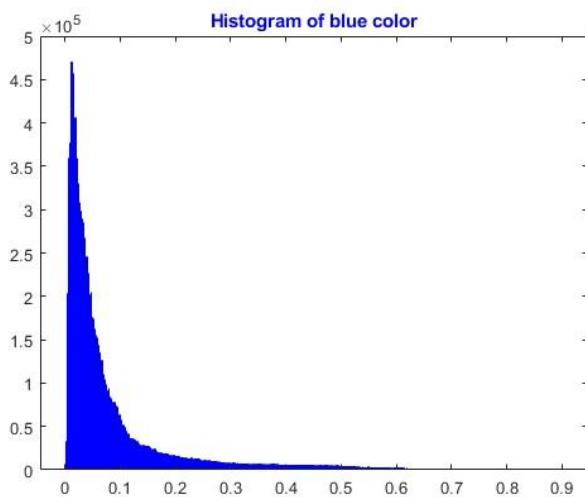
Ιστογράμματα



Εικόνα 27: Ιστόγραμμα για το κόκκινο χρώμα με method nearest, pattern RGGB.

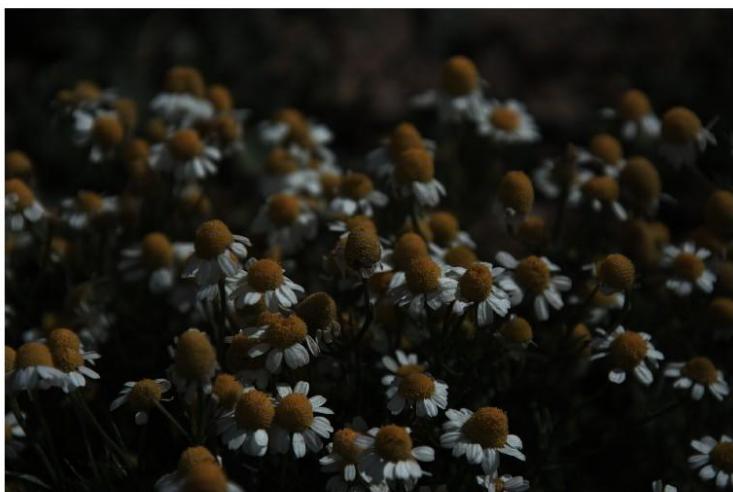


Εικόνα 28: Ιστόγραμμα για το πράσινο χρώμα με method nearest, pattern RGGB.



Εικόνα 29: Ιστόγραμμα για το μπλε χρώμα με method nearest, pattern RGGB.

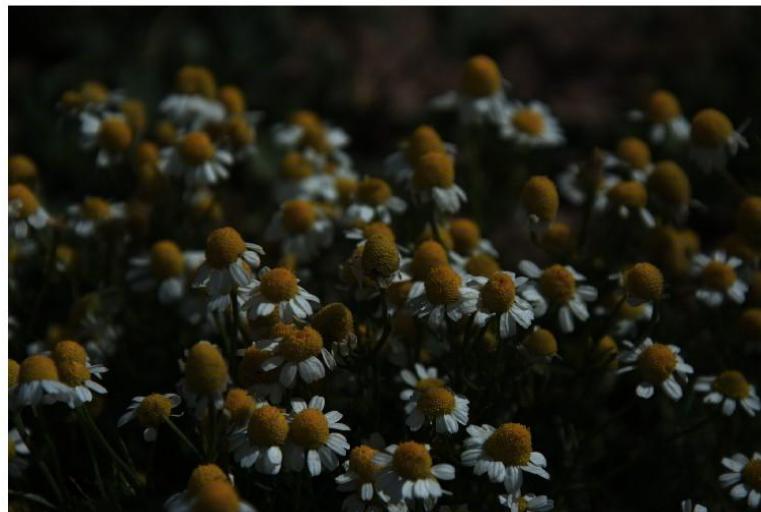
Εικόνες



Εικόνα 30: Ccam image με method nearest, pattern RGGB.



Εικόνα 31: Cxyz image με method nearest, pattern RGGB.



Εικόνα 32: Clinear image με method nearest, pattern RGGB.

Για την τελική εικόνα αντί για την imfilter χρησιμοποιείται η συνάρτηση apply_cmatrix από το άρθρο 'Processing RAW Images in MATLAB' του Rob Sumner καθώς παρατηρήθηκε καλύτερο αποτέλεσμα. Όντως η εικόνα φαίνεται να έχει σχηματιστεί σωστά.

Image with method nearest and bayer pattern rggb



Εικόνα 33: Τελική εικόνα με method nearest, pattern RGGB.

Interpolation: Method ‘Linear’

Σε αυτή τη μέθοδο παρεμβολής, το κάθε pixel χωρίς τιμή υπολογίζεται με γραμμική παρεμβολή από τα γειτονικά του. Έτσι υπάρχει μια συνέχεια και τα αποτελέσματα πλησιάζουν πιο πολύ στην πραγματικότητα.

Bayer Pattern: ‘bggr’

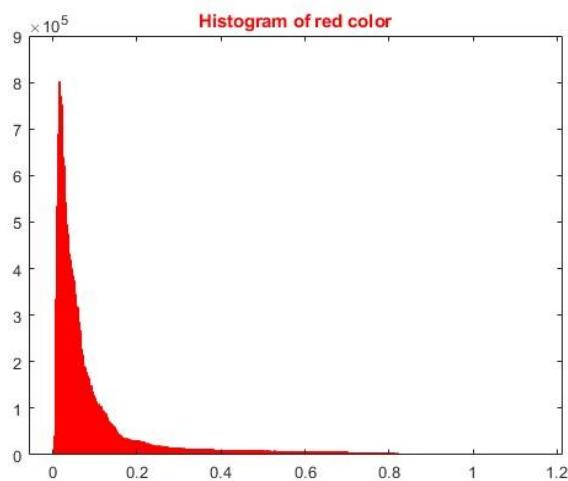
Σε αυτή τη μέθοδο η διαδικασία είναι διαφορετική. Αρχικά δίνονται οι τιμές στα pixels ανάλογα με το pattern. Έπειτα ασχολούμαι με τα υπόλοιπα pixels που είναι μηδενικά ακόμα. Ανάλογα με τη θέση που έχουν στο pattern σε σχέση με τα διπλανά προσέχω διαφορετικά πράγματα.

Γενικά, στα περιττά x και y, φροντίζω να δώσω πρώτα τιμή σε όλα εκτός από αυτά στη θέση 1, αφού δεν έχει προηγούμενο pixel έτσι ώστε να χρησιμοποιήσει την τιμή του με γραμμική παρεμβολή. Στα άρτια, παραμελώ αρχικά σε αυτά που βρίσκονται στην τελευταία θέση, για τον αντίστοιχο λόγο της έλλειψης επόμενου pixel. Στην ουσία ασχολούμαι αργότερα με τα pixels που βρίσκονται στο περίγραμμα.

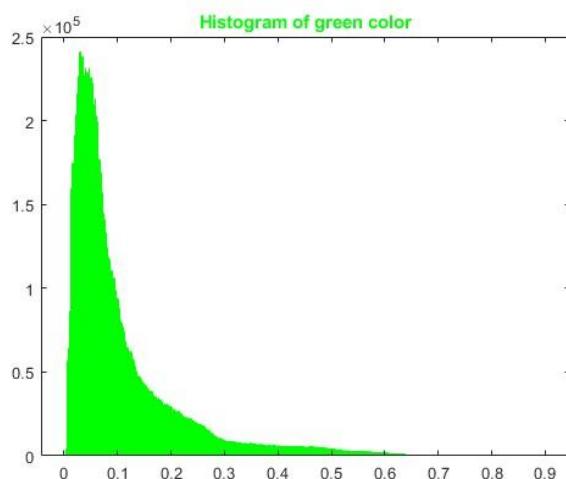
Σε όλα τα patterns ισχύει η ίδια διαδικασία. Κάθε pixel παίρνει για τιμή το μέσο όρο των διπλανών του, τα οποία θα είναι ή 4 ή 2. Στα πράσινα pixels, ενώ όλα έχουν 4 γύρω τους, τους δίνω μόνο την τιμή με βάση τα 2 διπλανά.

Για τα pixels του περιγράμματος κάνω το ίδιο με ό,τι υπάρχει δίπλα τους. Σε κάποια τυχαίνει να παίρνουν απλά την τιμή του διπλανού τους.

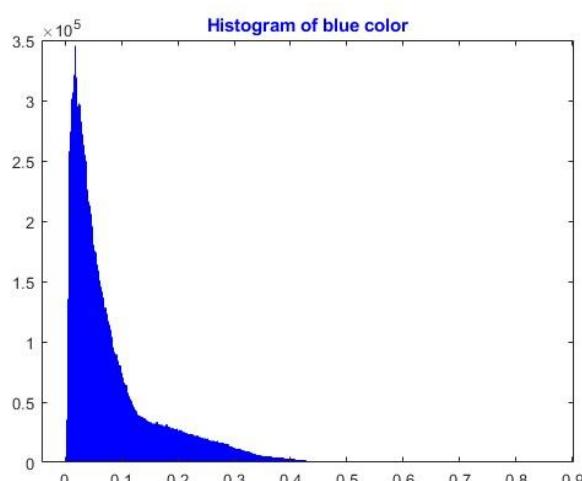
Ιστογράμματα



Εικόνα 34: Ιστόγραμμα για το χρώμα κόκκινο με method linear, pattern BGGR

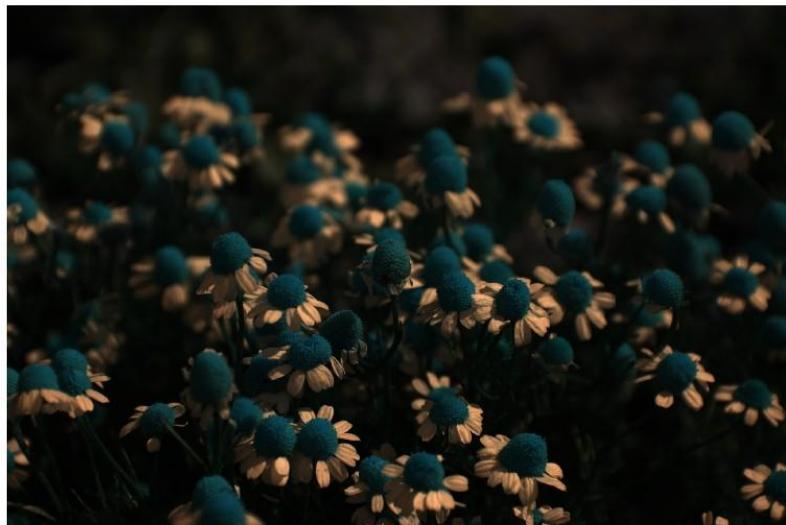


Εικόνα 35: Ιστόγραμμα για το χρώμα πράσινο με method linear, pattern BGGR



Εικόνα 36: Ιστόγραμμα για το χρώμα μπλε με method linear, pattern BGGR

Εικόνες



Εικόνα 37: Ccam image με method linear, pattern BGGR



Εικόνα 38: Cxyz image με method linear, pattern BGGR



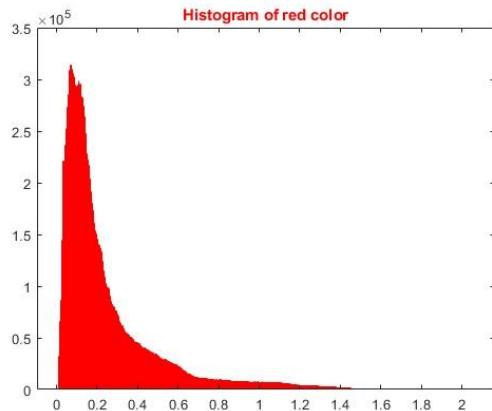
Εικόνα 39: Clinear image με method linear, pattern BGGR



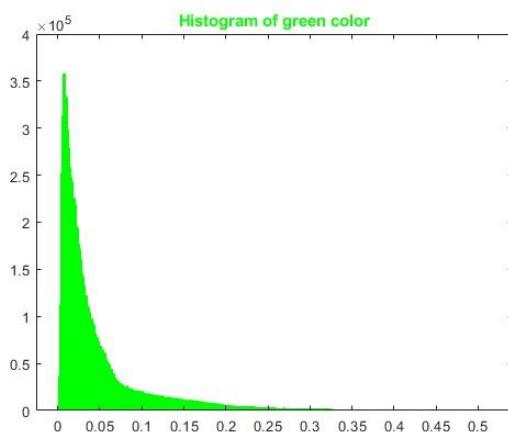
Εικόνα 40: Τελική εικόνα με method linear, pattern BGGR

Bayrer Pattern: 'gbrg'

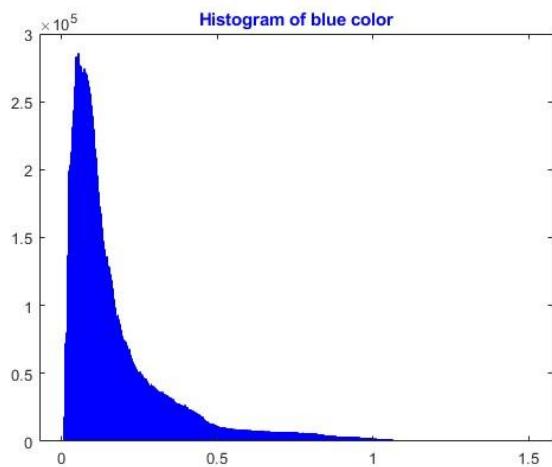
Ιστογράμματα



Εικόνα 41: Ιστόγραμμα για το χρώμα κόκκινο με μέθοδο linear, pattern GBRG



Εικόνα 42: Ιστόγραμμα για το χρώμα πράσινο με μέθοδο linear, pattern GBRG

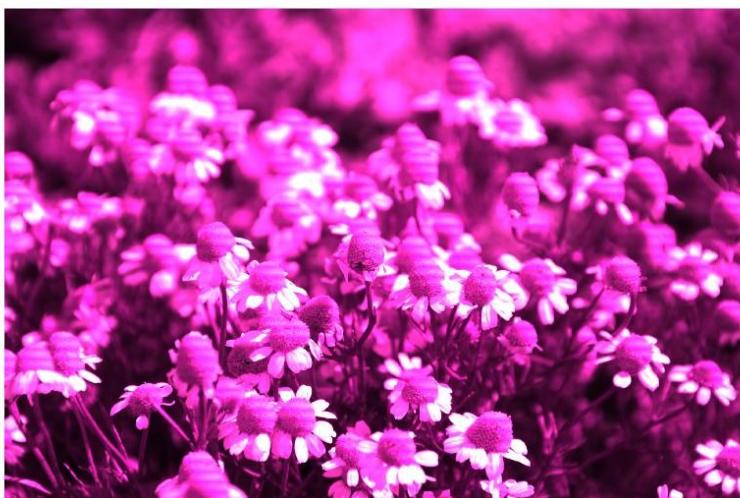


Εικόνα 43: Ιστόγραμμα για το χρώμα μπλε με μέθοδο linear, pattern GBRG

Εικόνες



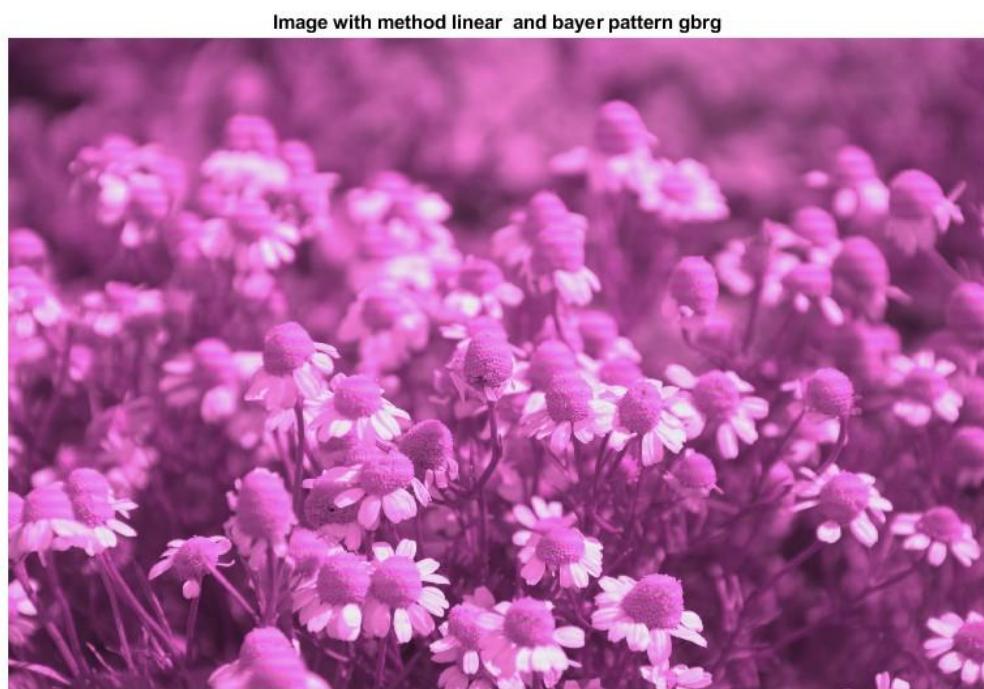
Εικόνα 44: Ccam image με μέθοδο linear, pattern GBRG



Εικόνα 45: Cxyz image με μέθοδο linear, pattern GBRG



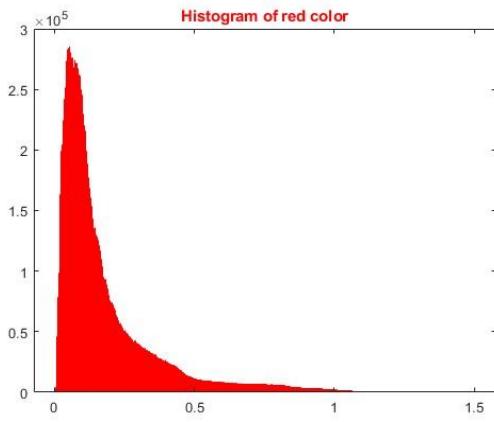
Εικόνα 46: Clinear image με μέθοδο linear, pattern GBRG



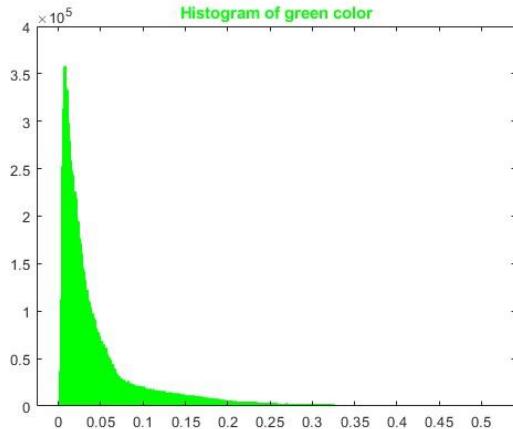
Εικόνα 47: Τελική εικόνα με μέθοδο linear, pattern GBRG

Bayrer Pattern: 'grbg'

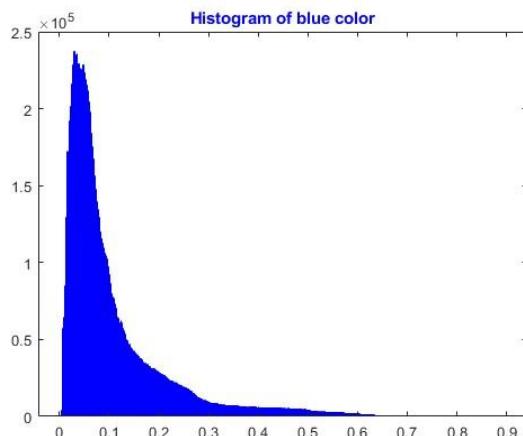
Ιστογράμματα



Εικόνα 48: Ιστόγραμμα για το κόκκινο χρώμα, method linear, pattern GRBG



Εικόνα 49: Ιστόγραμμα για το πράσινο χρώμα, method linear, pattern GRBG

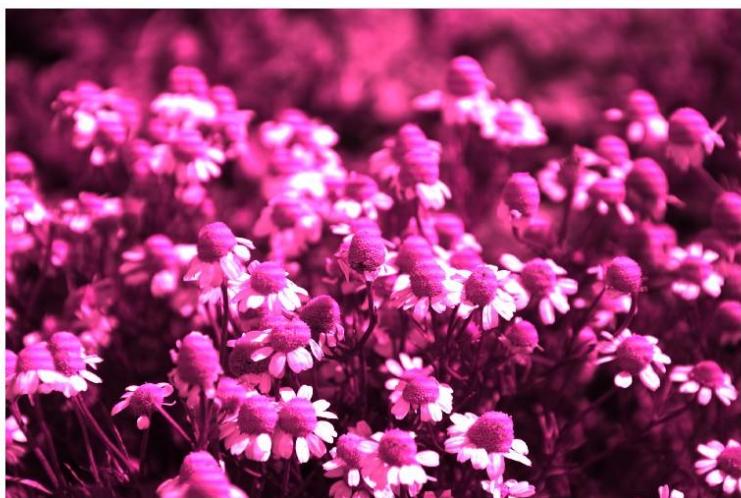


Εικόνα 50: Ιστόγραμμα για το μπλε χρώμα, method linear, pattern GRBG

Εικόνες



Εικόνα 51: Ccam, method linear, pattern GRBG

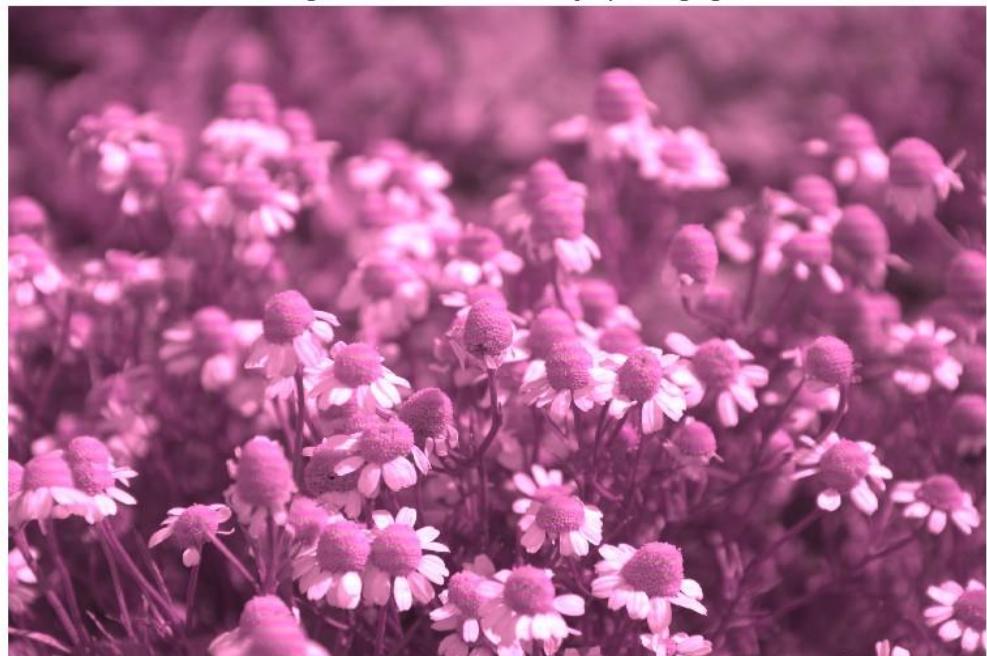


Εικόνα 52: Cxyz, method linear, pattern GRBG



Εικόνα 53: Clinear, method linear, pattern GRBG

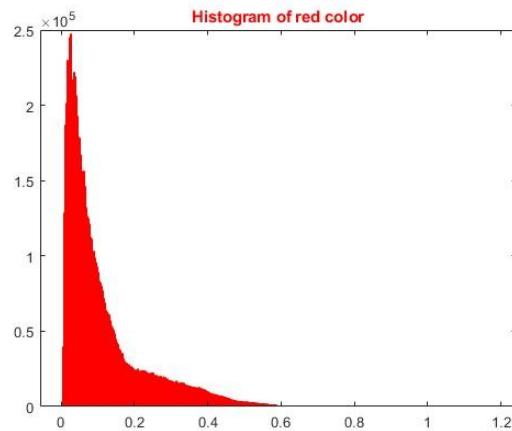
Image with method linear and bayer pattern grbg



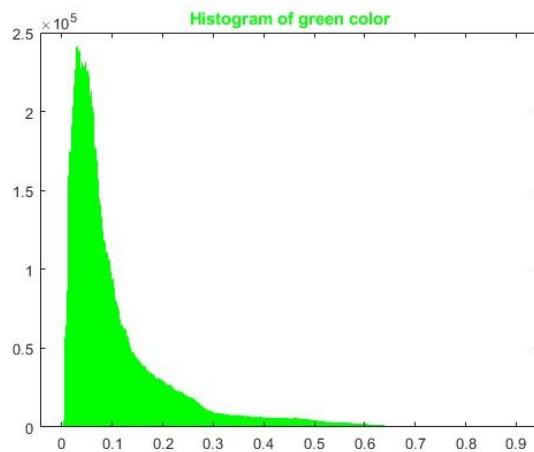
Εικόνα 54: Τελική εικόνα, method linear, pattern GRBG

Bayrer Pattern: 'rggb'

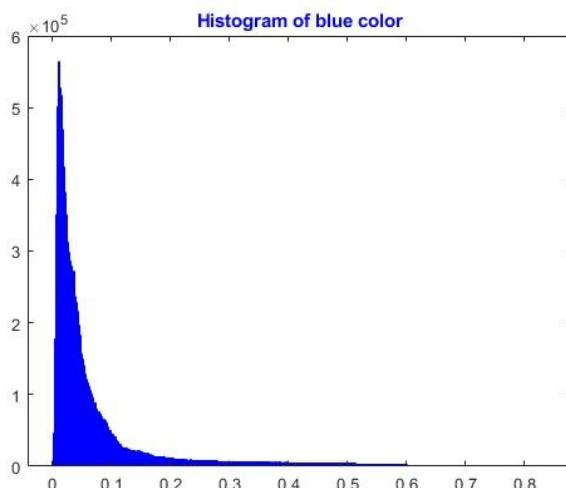
Ιστογράμματα



Εικόνα 55: Ιστόγραμμα για το κόκκινο χρώμα, method linear, pattern RGGB



Εικόνα 56: Ιστόγραμμα για το πράσινο χρώμα, method linear, pattern RGGB

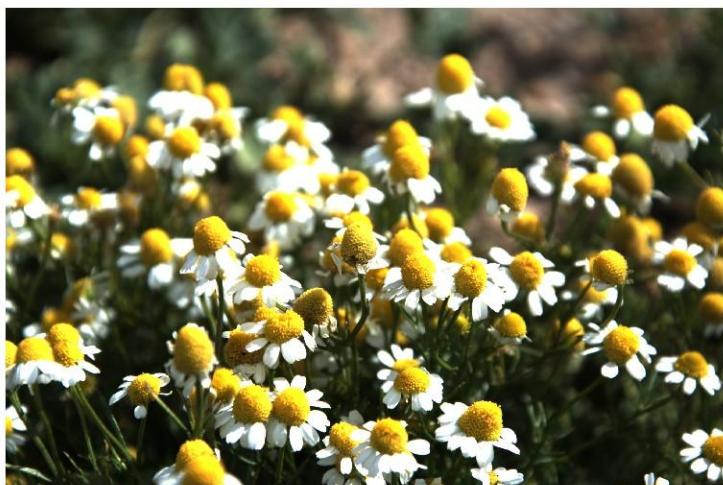


Εικόνα 57: Ιστόγραμμα για το μπλε χρώμα, method linear, pattern RGGB

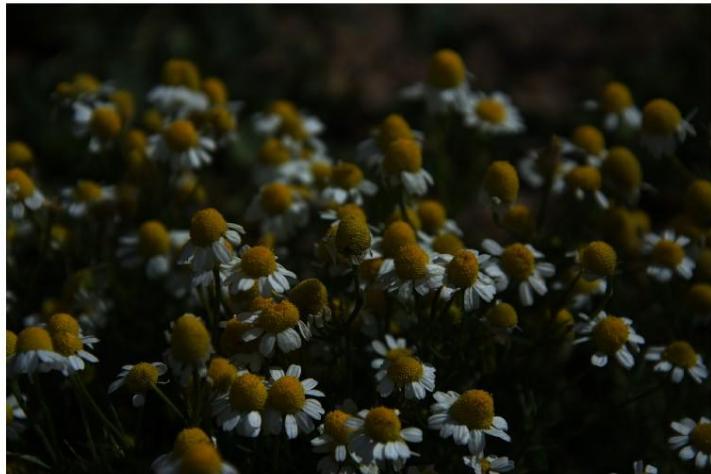
Εικόνες



Εικόνα 58: Ccam, method linear, pattern RGGB



Εικόνα 59: Cxyz, method linear, pattern RGGB



Εικόνα 60: Clinear, method linear, pattern RGGB

Image with method linear and bayer pattern rggb



Εικόνα 61: Τελική εικόνα, method linear, pattern RGGB

Δοκιμή με άλλες εικόνες

Ο κώδικας δοκιμάστηκε και με άλλες εικόνες εκτός από αυτήν και φάνηκε να έχει το ίδιο αποτέλεσμα. Παρουσιάζονται οι 2 εικόνες τύπου .DNG και η τελική εικόνα που προέκυψε με linear method, pattern RGGB. Οι εικόνες αυτές βρέθηκαν σε σελίδα με samples τύπου .DNG και άλλων raw format.



Εικόνα 62: Σύγκριση αρχικής .DNG εικόνας (αριστερά) με το αποτέλεσμα της μετατροπής (δεξιά)



Εικόνα 63: Σύγκριση αρχικής .DNG εικόνας (αριστερά) με το αποτέλεσμα της μετατροπής (δεξιά)

Σχολιασμός ιστογραμμάτων

Τα ιστογράμματα για τα 3 κανάλια χρωμάτων έχουν παρόμοια μορφή σε όλες τις περιπτώσεις. Τα περισσότερα pixels έχουν τιμές σε ένα συγκεκριμένο διάστημα, το οποίο διαφέρει για κάθε περίπτωση. Σε όλα υπάρχουν και κάποια με πιο ακραίες τιμές.

Σχολιασμός εικόνων

Σύγκριση διαφορετικών εικόνων

Οι εικόνες που δημιουργήθηκαν με τόσους διαφορετικούς τρόπους έχουν κάποια κοινά. Η Ccam είναι πάντα μια πιο σκοτεινή εικόνα, ενώ το αντίθετο συμβαίνει στην Cxyz που έχει μεγαλύτερη φωτεινότητα, πιο υψηλές τιμές, κοντά στο λευκό. Έπειτα η Clinear, εμφανίζεται πιο σκοτεινή, αλλά μόνο στα σημεία που χρειάζεται, έτσι είναι πιο ευδιάκριτες οι λεπτομέρειες. Τέλος η Csrgb γίνεται ξανά πιο φωτεινή προσπαθώντας να προσεγγίσει την πραγματική.

Σύγκριση διαφορετικών patterns

Αρχικά παρατηρήθηκε ότι μόνο με το RGGB εμφανίστηκε η σωστή εικόνα. Αυτό είναι λογικό γιατί αντιστοιχήθηκαν οι σωστές τιμές στα σωστά χρώματα. Με το BGGR οι τιμές των μπλε δόθηκαν στα κόκκινα και το αντίθετο. Οπότε στην εικόνα που σχηματίστηκε κυριαρχεί το μπλε χρώμα, ενώ το πράσινο πειράχτηκε ελάχιστα. Στο GBRG κανένα χρώμα δεν πήρε σωστή τιμή, ούτε στο GRBG. Και στα 2 κυριαρχεί ένα κόκκινο χρώμα, αφού πήρε τιμές που θα έπρεπε να έχει το πράσινο.

Σύγκριση διαφορετικών μεθόδων

Στην μέθοδο linear όπως ήταν αναμενόμενο οι εικόνες έχουν καλύτερη αναλυτικότητα και είναι πιο συνεχείς σε σχέση με αυτές της μεθόδου nearest, όπως ανέφερα και πιο πάνω.