

Ψηφιακή Επεξεργασία Εικόνας

Εργασία 1: Φίλτρο Bayer

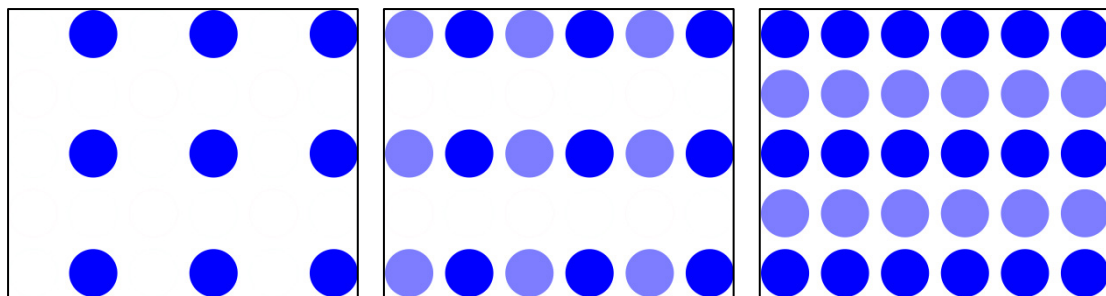
Στόχος της εργασίας

Η εργασία αυτή έχει ως στόχο την ανακατασκευή εικόνων με χρήση ενός φίλτρου Bayer. Για την υλοποίηση της εν λόγω διαδικασίας, χωρίζουμε την εικόνα σε τρεις πίνακες, έναν για κάθε χρώμα: R-G-B. Οι πίνακες αυτοί αποθηκεύονται σε έναν τρισδιάστατο πίνακα, με όνομα `image`. Σε αυτό το στάδιο, ο πίνακας είναι μερικώς γεμισμένος. Για τον υπολογισμό των υπόλοιπων τιμών του, χρησιμοποιούνται δύο τεχνικές: αυτή του κοντινότερου γείτονα, και η διγραμμική παρεμβολή. Παρακάτω αναλύεται η θεωρητική και προγραμματιστική τους υλοποίηση.

Μέθοδος Κοντινότερου Γείτονα (Nearest Neighbor)

Θεωρητική ανάλυση

Στην υλοποίηση αυτή, για τον υπολογισμό του χρώματος των «άδειων» στοιχείων του πίνακα, για τα χρώματα μπλε και κόκκινο, αντιγράφουμε το χρώμα του αριστερού ή πάνω χρώματος. Παρακάτω φαίνεται η διαδικασία για τον υπολογισμό του μπλε χρώματος. Αντίστοιχα λειτουργούμε και στο κόκκινο, ενώ στο πράσινο εκτελούμε μόνο το πρώτο βήμα.



Προγραμματιστική ανάλυση

Για την υλοποίηση της παραπάνω διαδικασίας, μετατοπίζουμε τον πίνακα κατά μια θέση δεξιά, και προσθέτουμε τον μετατοπισμένο πίνακα με τον αρχικό. Εκμεταλλευόμαστε, έτσι, το γεγονός ότι στα σημεία που δεν υπάρχει χρώμα, η τιμή του πίνακα είναι 0. Έπειτα, μετατοπίζουμε τον νέο πίνακα μια θέση προς τα κάτω, και αθροίζουμε ξανά.

Για τον υπολογισμό των σημείων της περιφέρειας κατά την ολίσθηση, ακολουθούμε μια διαδικασία `padding`: αντιγράφουμε τη δεύτερη (ή προτελευταία) στήλη (ή γραμμή) στην αρχή (ή στο τέλος). Με τον τρόπο αυτό, διατηρούμε τη συνέπεια στο φίλτρο Bayer.

Μέθοδος Διγραμμικής Παρεμβολής (Bilinear Interpolation)

Θεωρητική ανάλυση

Για την υλοποίηση της διγραμμικής παρεμβολής, χρησιμοποιούμε τα τέσσερα άμεσα γειτονικά σημεία, για κάθε pixel. Όπως και στη μέθοδο του κοντινότερου γείτονα, η διαδικασία ξεκινάει με τον υπολογισμό των σημείων στις γραμμές που υπάρχει πληροφορία για το χρώμα. Έπειτα, συμπληρώνονται και οι υπόλοιπες γραμμές. Ενδεικτικά,

για το σημείο [3,1] του φίλτρου (με 0,0 την πάνω αριστερή γωνία), το κόκκινο χρώμα θα υπολογιστεί ως:

$$R_{3,1} = (R_{3,0} + R_{4,1} + R_{3,2} + R_{2,1})/4$$

Ωστόσο, επειδή οι τιμές $R_{3,0}$ και $R_{3,2}$ δεν περιέχουν πληροφορία, τις αγνοούμε τελείως. Οπότε το κόκκινο χρώμα του [3,1] σημείου θα υπολογιστεί ως:

$$R_{3,1} = (R_{4,1} + R_{2,1})/2$$

Το ίδιο ισχύει και για το σημείο $R_{4,2}$. Τέλος, για το σημείο $R_{3,2}$, ξέρουμε όλα τα γειτονικά του σημεία. Μετά από πράξεις, το αποτέλεσμα θα είναι:

$$R_{3,2} = (R_{2,1} + R_{4,1} + R_{2,3} + R_{4,3})/4$$

Τα ίδια βήματα ακολουθούμε για το μπλε, ενώ στο πράσινο χρώμα εκτελούμε μόνο το τελευταίο βήμα.

Προγραμματιστική ανάλυση

Στην υλοποίηση του Matlab, πραγματοποιούμε διαδοχικές ολισθήσεις του πίνακα προς την κατάλληλη κατεύθυνση κάθε φορά, και τελικά προσθέτουμε τους ολισθημένους πίνακες. Για τον υπολογισμό των σημείων της περιφέρειας, ακολουθούμε την ίδια μεθοδολογία με αυτή του κοντινού γείτονα.

Αποτελέσματα



Nearest Neighbor



Bilinear Interpolation

Παρατηρούμε ότι στην περίπτωση του Bilinear Interpolation, η εικόνα έχει πιο ομαλές μεταβάσεις χρωμάτων, πράγμα που περιμέναμε σύμφωνα με τον αλγόριθμο. Ακόμη, η παραγόμενη εικόνα έχει σημαντικά μεγαλύτερο μέγεθος από την αρχική (3 φορές μεγαλύτερο). Οπότε θα μπορούσαμε να θεωρήσουμε ότι η εικόνα με δομή Bayer αποτελεί μια μορφή συμπίεσης με ελεγχόμενη παραμόρφωση της αρχικής εικόνας.