

Προγραμματισμός Συστημάτων Υψηλής Επίδοσης (ECE415)
Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Πανεπιστήμιο Θεσσαλίας

Διδάσκων: Χρήστος Δ. Αντωνόπουλος

4η Εργαστηριακή Άσκηση

Στόχος: Παραλληλοποίηση και βελτιστοποίηση εφαρμογής στη GPU.

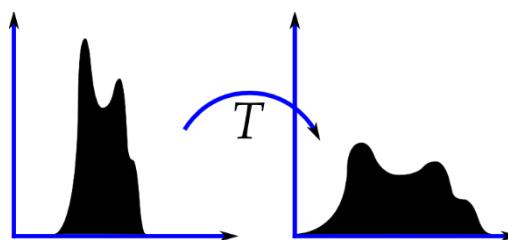
Εισαγωγή:

Η βελτίωση της αντίθεσης είναι πολύ συνηθισμένη διαδικασία στην επεξεργασία εικόνας. Στα πλαίσια αυτής της εργασίας θα πρέπει να υλοποιήσετε βελτίωση αντίθεσης με χρήση της διαδικασίας εξίσωσης ιστογράμματος σε εικόνες με τόνους του γκρι ("ασπρόμαυρες").

Ως σημείο αναφοράς και εκκίνησης σας δίνεται μια ακολουθιακή υλοποίηση της διαδικασίας σε απλή C. Επίσης, σας δίνονται εικόνες που μπορείτε να χρησιμοποιήσετε ως είσοδο στον κώδικα σας.

Υπόβαθρο – Εξίσωση ιστογράμματος:

Μια απλή μέθοδος βελτίωσης της αντίθεσης σε "ασπρόμαυρες" εικόνες είναι η εξίσωση ιστογράμματος. Το ιστόγραμμα μιας τέτοιας εικόνας αντιστοιχεί στην κατανομή των pixels σε διαφορετικούς τόνους του γκρι. Η εξίσωση ιστογράμματος ουσιαστικά "απλώνει" τις πιο συνηθισμένες τιμές τονικότητας, κάτι το οποίο ουσιαστικά αντιστοιχεί σε εξομάλυνση του ιστογράμματος. Στο επίπεδο της εικόνας αυτό έχει ως αποτέλεσμα περιοχές με μικρή τοπική αντίθεση να κερδίζουν μεγαλύτερη αντίθεση, χωρίς όμως να αναγκαζόμαστε να μεταβάλλουμε τη συνολική αντίθεση της εικόνας.



Παρακάτω μπορείτε να διακρίνετε τα αποτελέσματα της εφαρμογής της εξίσωσης ιστογράμματος σε μια εικόνα.



Συνοπτικά, η εφαρμογή του μετασχηματισμού εξίσωσης ιστογράμματος συνοψίζεται στα παρακάτω βήματα:

- Υπολογισμός του ιστογράμματος της αρχικής εικόνας.
- Υπολογισμός της σωρευτικής πυκνότητας (CDF) κάθε απόχρωσης του γκρι.
- Κατασκευή ενός πίνακα αναφοράς ο οποίος θα χρησιμοποιηθεί για την απεικόνιση της κάθε

απόχρωσης του γκρι στην αρχική εικόνα στην αντίστοιχη απόχρωση στην εξισωμένη εικόνα.
 δ) Κατασκευή της τελικής εικόνας.

Αναλυτική περιγραφή της διαδικασίας της εξίσωσης ιστογράμματος, μαζί με ένα αρκετά κατατοπιστικό παράδειγμα μπορείτε να βρείτε στη σελίδα της wikipedia: (http://en.wikipedia.org/wiki/Histogram_equalization).

Υπόβαθρο – Προσαρμοστική εξίσωση ιστογράμματος – Αλγόριθμος CLAHE:

Η απλή εξίσωση ιστογράμματος πάσχει από ένα σημαντικό μειονέκτημα: χρησιμοποιεί μία ενιαία καμπύλη μετασχηματισμού για όλη την εικόνα. Αυτό συχνά οδηγεί σε απώλεια λεπτομέρειας, καθώς περιοχές με διαφορετικό φωτισμό (π.χ. σκιές έναντι έντονου φωτός) απαιτούν διαφορετική αντιμετώπιση.

Η λύση είναι η Προσαρμοστική Εξίσωση Ιστογράμματος (Adaptive Histogram Equalization - AHE). Σε αυτήν, χωρίζουμε την εικόνα σε ένα πλέγμα (grid) από μικρά ορθογώνια πλακίδια (tiles), π.χ. 32 x 32 pixels, και υπολογίζουμε ξεχωριστό ιστόγραμμα για το καθένα. Για να αποφύγουμε την υπερβολική ενίσχυση του θορύβου σε ομοιογενείς περιοχές (όπως ο ουρανός), χρησιμοποιούμε την παραλλαγή CLAHE (Contrast Limited AHE), η οποία "ψαλιδίζει" (clips) τις κορυφές του ιστογράμματος πριν τον υπολογισμό της σωρευτικής πυκνότητας πιθανότητας (CDF).



Original Image



Histogram Equalized Image



CLAHE output

Εικόνα 1: Αρχική εικόνα (αριστερά), εικόνα μετά την εφαρμογή του global histogram equalization (μέση), και εικόνα μετά την εφαρμογή του CLAHE (δεξιά).

Υλοποίηση:

Η υλοποίησή σας πρέπει να ακολουθεί τα παρακάτω 3 διακριτά στάδια. Μπορείτε να τα υλοποιήσετε ως ξεχωριστούς kernels ή ως ένα μονολιθικό kernel, αλλά πρέπει να αιτιολογήσετε την επιλογή σας στην αναφορά.

Βήμα 1:

Υπολογισμός Ιστογράμματος & Ψαλίδισμα (Tiled Histogram Generation): Χωρίστε την εικόνα σε πλέγμα από πλακίδια (tiles). Για κάθε tile, υπολογίστε το ιστόγραμμα (256 bins).

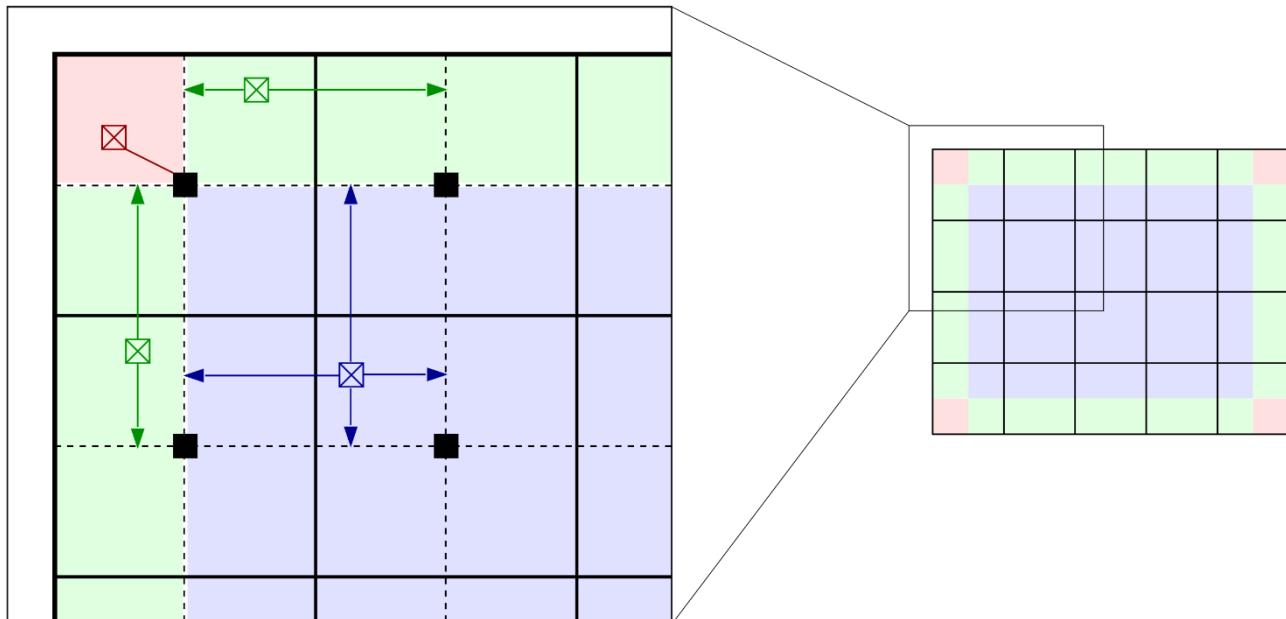
Clipping: Ελέγξτε αν κάποιο bin υπερβαίνει το όριο CLIP_LIMIT. Αν ναι, κόψτε την περίσσεια και αναδιανείμετε την ομοιόμορφα στα υπόλοιπα bins.

Υπολογισμός CDF: Μετατρέψτε το (ψαλιδισμένο) ιστόγραμμα σε σωρευτική πυκνότητα πιθανότητας (CDF) και δημιουργήστε τον πίνακα αντιστοίχισης (LUT). Αυτό το βήμα απαιτεί εκτενή χρήση Shared Memory και atomic operations. Προσοχή στα bank conflicts!

Βήμα 2:

Διγραμμική Παρεμβολή (Bilinear Interpolation): Αν εφαρμόσουμε απλά το ιστόγραμμα του κάθε tile στα pixels του, θα δημιουργηθούν τεχνονυργήματα (artifacts) "σκακιέρας" στα σύνορα των tiles. Για να το λύσουμε αυτό, για κάθε pixel (x,y):

- Εντοπίζουμε τα τέσσερα πλησιέστερα κέντρα tiles (Πάνω-Αριστερά, Πάνω-Δεξιά, Κάτω-Αριστερά, Κάτω-Δεξιά).
- Ψάχνουμε την τιμή του pixel στα 4 αντίστοιχα LUTs.
- Υπολογίζουμε την τελική τιμή παρεμβάλλοντας τα 4 αποτελέσματα με βάση την απόσταση του pixel από τα κέντρα των γειτονικών tiles.



Εικόνα 2: Το πλέγμα των tiles. Κάθε pixel ανήκει σε ένα tile, αλλά επηρεάζεται από τα κέντρα των 4 κοντινότερων tiles (μπλε pixel). Σημεία κοντά στα όρια της εικόνας μπορεί να μην έχουν τέσσερα γειτονικά tiles αλλά δύο (πράσινα pixels) ή και ένα (κόκκινο pixel). (πηγή εικόνας: Wikipedia, Adaptive Histogram Equalization)

Αναλυτική περιγραφή της διαδικασίας της εξίσωσης ιστογράμματος CLAHE μπορείτε να βρείτε στη σελίδα της wikipedia: (https://en.wikipedia.org/wiki/Adaptive_histogram_equalization).

Κώδικας:

Συνίσταται να μελετήσετε πολύ προσεκτικά τον κώδικα που σας δίνεται (ακολουθιακός κώδικας σε CPU). Ο κώδικας διαβάζει μια εικόνα σε pgm format, εφαρμόζει τον αλγόριθμο CLAHE και αποθηκεύει τη νέα εικόνα επίσης σε pgm format.

Μπορείτε να κατασκευάσετε δικές σας εικόνες με το gimp, αποθηκεύοντάς τις σαν pgm (raw format). Κατόπιν, θα πρέπει να τις ανοίξετε με έναν καλό editor (επειδή το μέγεθός τους είναι μεγάλο) και να σβήσετε το σχόλιο που προσθέτει το gimp στην αρχή του αρχείου (τη γραμμή που ξεκινάει με #).

Ζητούμενα:

Θα πρέπει να μεταφέρετε τον κώδικα που σας δίνεται στην GPU. Ο κώδικας θα πρέπει να είναι λειτουργικός για οποιοδήποτε μέγεθος εικόνας (με την προϋπόθεση ότι επαρκεί η device memory της GPU). Ο συγκεκριμένος κώδικας, παρότι μικρός, είναι αρκετά δεκτικός σε βελτιστοποιήσεις που έχουμε εξετάσει στο μάθημα και τη χρήση παράλληλων μοτίβων (parallel patterns). Μπορείτε να δείτε αυτό το lab και ως διαγωνισμό, όπου "κερδίζει" η γρηγορότερη υλοποίηση. Θεωρείται βέβαια αυτονόητο ότι η υλοποίηση θα πρέπει να είναι και ορθή.

Παράδοση:

Πρέπει να παραδώσετε:

- Τον τελικό κώδικα.
- Αναφορά στην οποία θα αναλύετε τη στρατηγική παραλληλοποίησης που ακολουθήσατε, τις βελτιστοποιήσεις που εφαρμόσατε και την επίδρασή τους στην επίδοση. Μην παραλείψετε να αναφέρετε και τυχόν βελτιστοποιήσεις που δοκιμάσατε χωρίς καλά αποτελέσματα στην επίδοση.
- Αντί για χρόνο, χρησιμοποιήστε ως μέτρο επίδοσης τα Mpixels/sec. Κατά την ανάπτυξη χρησιμοποιήστε τις μικρότερες εικόνες, αλλά για μετρήσεις επίδοσης χρησιμοποιήστε τις 2 μεγαλύτερες εικόνες.

Δημιουργήστε ένα αρχείο .tar.gz με τα παραπάνω περιεχόμενα και όνομα <όνομα1>_<ΑΕΜ1>_<όνομα2>_<ΑΕΜ2>_<όνομα3>_<ΑΕΜ3>_lab4.tar.gz. Αποστείλετε το στο e-class έως 23:59 της Παρασκευής 12/12/2025. Μην συμπεριλάβετε αρχεία εισόδου και εξόδου στο .tar.gz.