

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.

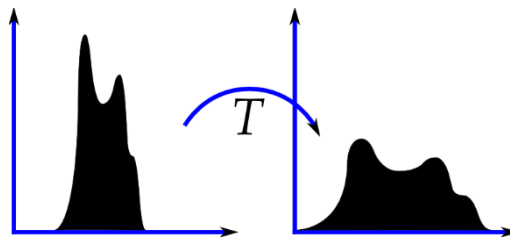
Εισαγωγή:

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 κώδικά σας.

Υπόβαθρο:

Μια απλή μέθοδος βελτίωσης της αντίθεσης σε "ασπρόμαυρες" εικόνες είναι η εξίσωση ιστογράμματος. Το ιστόγραμμα μιας τέτοιας εικόνας αντιστοιχεί στην κατανομή του αριθμού 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 αναγκάζομαστε 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 πιθανότητας 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).

Επιπρόσθετα, συνίσταται να μελετήσετε πολύ προσεκτικά τον κώδικα που σας δίνεται. Ο κώδικας διαβάζει μια εικόνα σε pgm format, υπολογίζει το ιστόγραμμα της, εφαρμόζει εξίσωση ιστογράμματος 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 στο χρόνο εκτέλεσης του kernel για καθεμιά από τις εικόνες που σας δίνονται. Μην παραλείψετε να αναφέρετε και τυχόν βελτιστοποιήσεις που δοκιμάσατε χωρίς καλά αποτελέσματα στο χρόνο εκτέλεσης.

Δημιουργήστε ένα αρχείο .tar.gz με τα παραπάνω περιεχόμενα και όνομα <όνομα1>_<AEM1>_<όνομα2>_<AEM2>_lab5.tar.gz. Αποστείλετέ το στο e-class έως 23:59 της Τετάρτης 14/12/2022.