

Η εργασία υλοποιήθηκε από τους φοιτητές:  
Χριστόφορος Αποστολόπουλος – ΑΜ: 1115201400322  
Παναγιώτης Ζάζος – ΑΜ: 1115201400325

Αρχικά, επιλέξαμε να χωρίσουμε την υλοποίηση του προγράμματός μας σε 2 header files και 3 source files (εκτών οποίων το ένα πηγαίο είναι η main) . Το ένα από τα άλλα δύο πηγαία αρχεία περιέχει μερικές βοηθητικές συναρτήσεις, οι οποίες βρίσκουν τους γείτονες και τα γωνιακά πιξελς, και το άλλο περιέχει τις δύο βασικές συναρτήσεις του προγράμματός μας: την greyscale convolution και την rgb convolution.

### **Main.c:**

Ξεκινώντας αναθέτουμε τις τιμές που έχουμε δώσει ως ορίσματα στο τερματικό σύστημα(όνομα, ύψος, πλάτος, grayscale ή rgb, επαναλήψεις), καθώς και έναν πίνακα με τους συντελεστές του φίλτρου. Θέτουμε μια σταθερά img\_side που είναι η ρίζα των threads και παρακάτω φτιάχνουμε τα μεγέθη των block στα οποία θα βάλουμε τα κομμάτια των εικόνων. Έπειτα εκκινούμε μια επικοινωνία με την νέα καρτεσιανή τοπολογία. Στην περίπτωση που η τοπολογία εμφανίσει mpi μήνυμα λάθους (να είναι δηλαδή NULL) τότε τερματίζουμε το mpi και το πρόγραμμα κλείνει.

Δημιουργούμε στην συνέχεια τις μετατοπίσεις ή αλλιώς τα σημεία στα οποία θα ξεκινάει το κάθε ένα από τα δημιουργούμενα blocks, καθώς και έναν counter, ο οποίος αναφέρεται στον αριθμό των blocks. Έχοντας ετοιμάσει τις μετατοπίσεις και τους αθροιστές των blocks, αυτό που μένει για να κάνουμε scatter τα μπλοκ αυτά στους επεξεργαστές είναι να φτιάξουμε μια βάση δεδομένων για τα blocks που πρόκειται να στείλουμε. Στο σημείο αυτό ανοίγουμε την εικόνα και κάνουμε allocate χώρο σε έναν buffer ίσο με το μέγεθος της εικόνας. Και επειδή θέλουμε να διαβάσουμε και να δημιουργήσουμε μόνο μια φορά αυτόν τον χώρο και όχι περισσότερες, φράζουμε αυτό το foren και το fread με το αν το rank είναι ίσο με το root rank. Εδώ σημειώνουμε πως βάλαμε μια συνθήκη για το άνοιγμα του αρχείου.

Θέτουμε και δύο προσωρινούς buffers με μέγεθος το 1/N της εικόνας, όπου N ο αριθμός των processes. Τα blocks στην συνέχεια είναι έτοιμα να γίνουν scatter στις διεργασίες, ώστε να υπολογίσουν τα κομμάτια της εικόνας.

Δημιουργήσαμε και έναν πίνακα με την προοπτική να κρατάει τους γείτονες του κάθε rank.

Εκκινούμε τον χρονομετρητή για την συνέλιξη, καλούμε την βασική συνάρτηση

**convolution function** (rgb ή grayscale), τερματίζουμε τον χρονομετρητή, κάνουμε το reduce, μαζεύουμε όλα τα αποτελέσματα από τις διεργασίες και τα τοποθετούμε στον buffer, τον οποίο στο τέλος γράφουμε σε ένα καινούργιο αρχείο που είναι η καινούργια μας εικόνα (edited.raw)

## Convolution.c:

Ξεκινάμε φτιάχνοντας χώρο για εξτρα πιξελς που θα χρησιμοποιηθούν στα receive ( τα εξωτερικά του "grid" τα οποία θα επικοινωνήσουν μεταξύ των διεργασιών).

Για κάθε επανάληψη, όπου αυτή ορίζεται από την γραμμή εντολής, κάνουμε τα παρακάτω:

- Κάνουμε 8 MPI\_Isend ( 4 πλευρές και 4 γωνίες) και 8 MPI\_Irecv αντίστοιχα. Τα κεντρικά πιξελς δεν χρειάζονται send και receive, υπολογίζονται ανεξάρτητα.

- Με την χρήση του OPENMP υπολογίζουμε όλα τα εσωτερικά πιξελς μιας εικόνας, χρησιμοποιώντας μερικές δικές μας συναρτήσεις που βρίσκονται στο extra\_functions.c.

- Κάνουμε MPI\_Waitall μέχρι να γίνουν όλα τα send και receive, καθώς μπορεί να έχει τελειώσει ο υπολογισμός των εσωτερικών πιξελς αλλά να μην έχουν ληφθεί τα εξωτερικά πιξελς.

- Υπολογίζουμε τα πιξελς των 4 πλευρών (παραπάνω πληροφορίες για τον τρόπο υλοποίησης είναι γραμμένες σε σχόλια πάνω στον κώδικα)

- Υπολογίζουμε τα πιξελς των 4 γωνιών (παραπάνω πληροφορίες για τον τρόπο υλοποίησης είναι γραμμένες σε σχόλια πάνω στον κώδικα)

- Κάνουμε MPI\_Waitall μέχρι να έχουν γίνει όλες οι πράξεις

- Κάνουμε swap τον αρχικό buffer με τον τελικό(φιλτραρισμένο) buffer, έτσι ώστε αν υπάρχει παραπάνω από 1 επανάληψη, ο τελικός να γίνει ο αρχικός κ.ο.κ.

- Ελέγχουμε ανά 20 επαναλήψεις (δική μας απόφαση ο αριθμός 20) αν έχει γίνει σωστά το φιλτράρισμα, δηλαδή αν έχουν γίνει αλλαγές στην εικόνα. Αν δεν έχει γίνει καμία, τότε δεν έγινε σωστά η συνέλιξη.

- Η συνάρτηση επιστρέφει τον αρχικό buffer (που στην ουσία είναι ο τελικός μετά το swap)

Τα παραπάνω βήματα ακολουθούνται και για την συνέλιξη της rgb εικόνας, με την διαφορά ότι μερικές τιμές έχουν πολλαπλασιαστεί με το 3, όπως height, μερικοί πίνακες και allocates στην μνήμη.

## extra\_functions.c:

Το αρχείο αυτό δημιουργήθηκε από εμάς για να υλοποιήσουμε και να ομαδοποιήσουμε μερικές συναρτήσεις που ήταν βοηθητικές για το πρόγραμμα. Οι πρώτες 6 συναρτήσεις κάνουν υπολογισμούς πλευρών και γωνιών για την εικόνα.

Ακολουθεί η column\_type στην οποία δημιουργούμε βάση δεδομένων strided, όπου τα άλματα είναι το πλάτος της εικόνας δια N (όπου N ο αριθμός διεργασιών).

Τέλος, η συνάρτηση get\_neighbours μας βρίσκει τους γείτονες του κάθε rank (θέτει τα coordination του grid, έτσι όπως τα έχουμε ορίσει εμείς). Χρησιμοποιώντας την συνάρτηση MPI\_Cart\_rank επικοινωνούν τα coordination μας με το καρτεσιανό (communicator).

**Ολόκληρη εικόνα waterfall\_1920\_2520.raw και waterfall\_grey\_1920\_2520.raw  
height=1920, width=2520**

Processes	Time for RGB- waterfall_1920_2520.raw	Time for B/W- waterfall_grey_1920_2520.raw
1	0.298197	0.101626
4	0.082008	0.027127
9	0.037213	0.012984
16	0.025492	0.009903
25	0.017527	0.009845
36	0.012603	0.011634

**½ της εικόνας waterfall\_1920\_2520.raw και waterfall\_grey\_1920\_2520.raw  
height=960, width=1260**

Processes	Time for 1/2 RGB- waterfall_1920_2520.raw	Time for 1/2 B/W- waterfall_gray_1920_2520.raw
1	0.077089	0.031218
4	0.020426	0.008660
9	0.011870	0.010797
16	0.023353	0.009251
25	0.005541	0.007141
36	0.011790	0.007209

**¼ της εικόνας waterfall\_1920\_2520.raw και waterfall\_grey\_1920\_2520.raw  
height=480, width=630**

Processes	Time for 1/4 RGB- waterfall_1920_2520.raw	Time for 1/4 B/W- waterfall_gray_1920_2520.raw
1	0.024112	0.009579
4	0.006830	0.002375
9	0.011667	0.006143
16	0.009548	0.009616
25	0.006014	0.004988
36	0.008486	0.017060

**Διπλάσιο της εικόνας waterfall\_1920\_2520.raw και waterfall\_grey\_1920\_2520.raw  
height=3840, width=5040**

Processes	Time for x2 RGB- waterfall_1920_2520.raw	Time for x2 B/W- waterfall_gray_1920_2520.raw
1	1.155946	0.386359
4	0.317446	0.106590
9	0.143265	0.048467
16	0.082663	0.031093
25	0.056403	0.020398
36	0.045949	0.018315

**Τετραπλάσιο της εικόνας waterfall\_1920\_2520.raw και waterfall\_grey\_1920\_2520.raw  
height=7680, width=10080**

Processes	Time for x4 RGB- waterfall_1920_2520.raw	Time for x4 B/W- waterfall_gray_1920_2520.raw
1	4.602973	1.531538
4	1.257885	0.420321
9	0.562060	0.107796
16	0.318936	0.107810
25	0.206661	0.070391
36	0.143832	0.056854

**Οκταπλάσιο της εικόνας waterfall\_1920\_2520.raw και waterfall\_grey\_1920\_2520.raw  
height=15360, width=20160**

Processes	Time for x8 RGB- waterfall_1920_2520.raw	Time for x8 B/W- waterfall_grey_1920_2520.raw
1	18.396041	6.114105
4	5.017215	1.669244
9	2.234035	0.743965
16	1.258502	0.421770
25	0.873504	0.275551
36	0.566262	0.197586