
*Εξαμηνιαία εργασία Θεωρίας Πληροφορίας
Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων –
Πανεπιστήμιο Αιγαίου*

Κωνσταντίνος Μαλιάτσος

Οδηγίες:

Μπορείτε να χρησιμοποιήσετε οποιαδήποτε γλώσσα προγραμματισμού προτιμάτε. Θα πρέπει επίσης να υποβάλετε μαζί με την εργασία πολύ σύντομες οδηγίες σχετικά με τον τρόπο εκτέλεσης/compiling του κώδικα. Αν αποφασίσετε να χρησιμοποιήσετε ένα εξειδικευμένο πακέτο-βιβλιοθήκη, πρέπει να δώσετε λεπτομέρειες σχετικά με τον τρόπο σύνδεσής/χρήσης του από το project σας.

Χρησιμοποιήστε εκτενώς σχόλια στον κώδικα σας. Η αποτελεσματικότητα του κώδικα δεν βαθμολογείται. Ωστόσο, η παρουσίασή του ναι. Αν ο κώδικας είναι ευκολοδιάβαστος, τότε ο βαθμός θα είναι υψηλότερος.

Παραδίδεται:

- Ο κώδικας μέσω του eclass.
- Μια παράγραφος ανά ερώτημα με εξήγηση/περιγραφή του τι κάνατε και γιατί.

Όταν χρησιμοποιείτε εξωτερικές πηγές, χρησιμοποιήστε αναφορές!

Εργασία:

Κατεβάστε το dataset [ColorMasks-FloodNetv1.0.zip](https://www.dropbox.com/scl/fo/k33qdif15ns2qv2jdxvhx/ANGaa8iPRhvlrvckXjnmNRc?rlkey=ao2493wzl1cltonowjdbnrp7f&e=2&dl=0) από τον παρακάτω σύνδεσμο:

<https://www.dropbox.com/scl/fo/k33qdif15ns2qv2jdxvhx/ANGaa8iPRhvlrvckXjnmNRc?rlkey=ao2493wzl1cltonowjdbnrp7f&e=2&dl=0>

Αρχικά εργάζεστε με τις εικόνες που περιέχονται στο ColorMasks-TrainSet.

Χρησιμοποιήστε οποιοδήποτε διαθέσιμο εργαλείο ή συνάρτηση της επιλεγμένης γλώσσας προγραμματισμού για να «διαβάσετε» προγραμματιστικά την εικόνα (π.χ., στο Matlab την imread). Όταν «διαβάζεται» η κάθε εικόνα προκύπτει ένας πίνακας διαστάσεων 1024x1024x3 – όπου η κάθε διάσταση αντιπροσωπεύει το πλάτος εικόνας σε pixel, το μήκος εικόνας σε pixel και το συνδυασμό χρωμάτων RGB (Red – Green – Blue). Το κάθε χρώμα «εκφράζεται από έναν ακέραιο αριθμό με 256 στάθμες (από 0 – 255).

Φτιάχνουμε το ακόλουθο σύστημα συμπίεσης και μετάδοσης των εικόνων. Για να επιλύσετε αυτό το θέμα, θα χρειαστεί να γράψετε ένα πρόγραμμα.

Ερώτημα 0: (Προπαρασκευαστικές εργασίες)

- Ανοίχτε και «διαβάστε» τα αρχεία εικόνας.

- Μετατρέψτε τον πίνακα εικόνας σε τρία streams διαστάσεων $1.048.576 \times 1$ ανά γραμμή. Δηλαδή για κάθε χρώμα δημιουργείται ένα διάνυσμα που περιέχει διαδοχικά τις 1024 τιμές pixel ανά γραμμή ($1024 \times 1024 = 1.048.576$).

Ερώτημα 1:

- Πραγματοποιήστε ένα χονδροειδή αλγόριθμο συμπίεσης με «επανα-κβαντισμό» των χρωμάτων. Αντί για 256 στάθμες ανά χρώμα, ορίστε και χρησιμοποιήστε μόνο 16. Αφού επανακβαντίσετε, παρουσιάστε το αποτέλεσμα με ενδεικτικές εικόνες.
- Υπάρχει αλλαγή ή απώλεια πληροφορίας? Αν ναι υπολογίστε ένα μέτρο Παραμόρφωσης (δηλαδή απόκλισης της αρχικής από την τελική εικόνα). Μπορείτε να επιλέξετε ή να ορίσετε ένα δικό σας μέτρο.
- Ποιο είναι το «ζευγάρι» ρυθμού – παραμόρφωσης για την εικόνα? Δηλαδή, πόσα bit χρειάζονται για να αποθηκεύσεις την αρχική εικόνα, πόσα bit την εικόνα με 16 στάθμες? Αλλά με τι κόστος παραμόρφωσης?
- Υπολογίστε το ζεύγος ρυθμού – παραμόρφωσης για επανα-κβαντισμό της εικόνας με 64 στάθμες και 8 στάθμες.

Ερώτημα 2:

- Χρησιμοποιήστε από εδώ και πέρα τις εικόνες με τις 16 στάθμες κβαντισμού.
- Καταμετρήστε πόσες φορές εμφανίζεται κάθε στάθμη ανά χρώμα στο σύνολο των εικόνων του trainset.
- Εκτιμήστε την κατανομή πιθανότητας $p_{level-red}(x_n)$ για τις στάθμες του κόκκινου και τις αντίστοιχες κατανομές για το πράσινο και το μπλε.
- Θεωρώντας τις στάθμες ανά pixel και χρώμα ως τυχαίες μεταβλητές ανεξάρτητες μεταξύ τους – βασιζόμενοι στο trainset.
 - ο Υπολογίστε την εντροπία της πηγής ανά χρώμα, π.χ. $H_{level-red}(x_n)$ που προκύπτει από τα αρχεία
 - ο Υπολογίστε τον πλεονασμό της πηγής ανά χρώμα.
- Καταμετρήστε πόσες φορές εμφανίζεται κάθε χαρακτήρας στο αρχείο matrix.txt.

Ερώτημα 3:

- Για το δοσμένο dataset (με 16 στάθμες):
 - ο Θεωρώντας ότι οι χαρακτήρες είναι ανεξάρτητοι και χρησιμοποιώντας την κατανομή του Ερωτήματος 1, υλοποιήστε τις κωδικοποιήσεις Shannon-Fano, και Huffman – ανά χρώμα ανεξάρτητα!
 - ο Εφαρμόστε τον κωδικοποιητή και προσδιορίστε το μέγεθος του συμπιεσμένου αρχείου.
 - ο Ποιο το μέσο μήκος συμβόλου του κώδικα?

Ερώτημα 4:

- Επαναλάβετε τα βήματα 1-3 αλλά δημιουργώντας τα διανύσματα ανά χρώμα *ανά γραμμή* και όχι ανά στήλη όπως πριν.

Ερώτημα 5:

- Για το δοσμένο dataset (με 16 στάθμες) και την ανά γραμμή διάταξη:
 - ο Υπολογίστε την κατανομή των σταθμών σε ζεύγη γειτονικών χαρακτήρων $p(x_n, x_{n+1})$ θεωρώντας την ακόλουθη διαδικασία: Καταμετρήστε τα ζεύγη σειριακά αλλά προσοχή! Ο συρμός rixel με τιμές ...14, 14, 13, 12, 11, 14... δεν περιέχει μόνο τα ζεύγη 14/14-13/12-11/14, αλλά τα ζεύγη 14/14-14/13-13/12-12/11-11/14,
 - ο Καταμετρήστε τον αριθμό των ζευγών που προέκυψαν.
 - ο Χρησιμοποιήστε το για να καταμετρήστε το ποσοστό εμφάνισης του κάθε ζεύγους. Συνεχίστε επιλέγοντας ένα άλλο ζεύγος κ.ο.κ.
 - ο Υπολογίστε την από κοινού εντροπία $H(x_n, x_{n+1})$
 - ο Είναι μικρότερη του $2H(x_n)$; Αν ναι γιατί;
 - ο Υπολογίστε την υπό συνθήκη εντροπία $H(x_n|x_{n+1})$.

Ερώτημα 6:

- Για το δοσμένο dataset (με 16 στάθμες):
 - ο Θεωρώντας ότι οι χαρακτήρες είναι ανεξάρτητοι και χρησιμοποιώντας την κατανομή του Ερωτήματος 1, υλοποιήστε την κωδικοποίηση Shannon-Fano, ή Shannon ή Huffman.
 - ο Εφαρμόστε τον κωδικοποιητή και προσδιορίστε το μέγεθος του συμπιεσμένου αρχείου.
 - ο Ποιο το μέσο μήκος συμβόλου του κώδικα?
- Επαναλάβετε για το αρχείο matrix.txt:

Ερώτημα 7:

- ο Επαναλάβετε την υλοποίηση Shannon-Fano και Huffman θεωρώντας τα ζεύγη χαρακτήρων ως σύμβολα του κώδικα.
- ο Εφαρμόστε τον κωδικοποιητή και προσδιορίστε το μέγεθος του συμπιεσμένου αρχείου.
- ο Ποιο το μέσο μήκος συμβόλου του κώδικα?
- ο Ποιος κώδικας αποδίδει καλύτερα? (Ερώτημα 3 vs. 7)

Ερώτημα 8:

- ο Επαναλάβετε το ερώτημα 7 για την ανά στήλη διάταξη.
- ο Υπάρχει διαφορά? Ποιος αποδίδει καλύτερα? Τι συμπέρασμα βγάζετε?

Ερώτημα 9:

- Εφαρμόστε τους κώδικες που φτιάξατε στο νέο dataset ColorMasks-TestSet αφού τους έχετε μετατρέψει πάλι σε 16 στάθμες.
- Υπάρχει περίπτωση κάποια ζεύγη τιμών, να μην εμφανίζονται (να έχουν μηδενική πιθανότητα) στο σχήμα κωδικοποίησης που αναπτύξατε. Πώς θα το διαχειριστείτε αυτό? Αναπτύξτε μια μέθοδο αντιμετώπισης του φαινομένου.
- Ποιος κώδικας από όλους αποδίδει καλύτερα στα νέα αρχεία?
- Πόσο χειρότερα ή καλύτερα συμπεριφέρεται ο κώδικας στα νέα αρχεία σε σχέση με τα αρχεία αναφοράς και γιατί?

Ερώτημα 10:

- Αναπτύξτε ένα σχήμα κωδικοποίησης συμπίεσης όπου κάθε σύμβολο είναι μια τριάδα χρωμάτων RGB. Δηλαδή αντί να κωδικοποιήσετε κάθε χρώμα ως συρμό ξεχωριστά, κάντε ένα συρμό (ανά γραμμή) με τριπλέτες RGB. Κάθε σύμβολο είναι τριάδα RGB.
- Θεωρώντας τα pixel και χρώμα ως τυχαίες μεταβλητές ανεξάρτητες μεταξύ τους – βασιζόμενοι στο trainset – αναπτύξτε
 - ο Υπολογίστε την εντροπία της πηγής $H_{colors}(x_n)$ που προκύπτει από τα αρχεία.

Ερώτημα 11:

- Θεωρώντας τα pixel και χρώμα ως τυχαίες μεταβλητές ανεξάρτητες μεταξύ τους – βασιζόμενοι στο trainset – αναπτύξτε ένα σχήμα συμπίεσης βασισμένο στον αλγόριθμο Huffman.

Ερώτημα 12:

- Θέλουμε να μεταδώσουμε τις εικόνες του testset πάνω από θορυβώδες κανάλι σηματοθορυβικού λόγου 6 dB.
- Θεωρώντας ότι για bit 0 αποστέλλεται η στάθμη σήματος -1 (π.χ. σε Volt) και για το bit 1 αποστέλλεται η στάθμη σήματος 1 (BPSK διαμόρφωση), φτιάξτε το σύστημα μετάδοσης και προσθέστε το θόρυβο σύμφωνα με την εκκώνηση της άσκησης.
- Εξομοιώστε τη διαδικασία λήψης με χρήση hard απόφασης (επιλογή του συμβόλου με την μικρότερη απόσταση). Υπάρχουν σφάλματα μετάδοσης? Αν ναι πόσα?

Ερώτημα 13:

- Εφαρμόστε κωδικοποίηση καναλιού Hamming με μήκος κωδικοποιημένης λέξης τα 15 bits.
- Εισάγετε θόρυβο και λήψη με hard απόφαση.
- Πραγματοποιήστε αποκωδικοποίηση Hamming ανά 15bits και μετρήστε τα σφάλματα.
- Ποιο το όφελος και ποιο το κόστος συγκριτικά με το ερώτημα 12.

Ερώτημα 14:

- Φτιάξτε ένα πρωτόκολλο, όπου σε περίπτωση λανθασμένης λήψης γίνεται επανάληψη της αποστολής του ίδιου Hamming block.
- Θεωρώντας κανάλι ταχύτητας 1 Msymbol/sec (για BPSK είναι ίδιο με Mbps) και εύρος ζώνη 1MHz, ποιος ο επιτεύξιμος ρυθμός μετάδοσης πληροφορίας και ποια η χωρητικότητα όπως υπολογίζεται θεωρητικά.
- Σε πόσο χρόνο αναμένεται να σταλεί όλο το test set?