



ΠΑΝΕΠΙΣΤΗΜΙΟΥ ΠΕΛΟΠΟΝΝΗΣΟΥ  
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ  
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

**Συμπίεση και Μετάδοση Πολυμέσων**

5<sup>η</sup> Φροντιστηριακή Σειρά Ασκήσεων – Επεξεργασία και Συμπίεση Εικόνων

**Αντικείμενο Άσκησης:**

Αντικείμενο της εργασίας είναι η κατανόηση βασικών αρχών συμπίεσης εικόνας, των τεχνικών επεξεργασίας και συμπίεσης εικόνας και ιδιαίτερα του προτύπου συμπίεσης JPEG.

**Περιγραφή:**

**Άσκηση 1**

Δημιουργήστε πρόγραμμα σε MATLAB (im\_average.m) το οποίο διαβάζει μια εικόνα αποχρώσεων του γκρι, υπολογίζει τη μέση τιμή των τόνων του γκρι και εμφανίζει το αποτέλεσμα του υπολογισμού και την εικόνα στην οθόνη. Για διευκόλυνσή σας χρησιμοποιείστε την εικόνα “cameraman.tif” του MATLAB.

**Άσκηση 2**

Δημιουργήστε πρόγραμμα σε MATLAB (im\_invert.m) το οποίο διαβάζει μια εικόνα αποχρώσεων του γκρι “cameraman.tif”, δημιουργεί το «αρνητικό» της και το αποθηκεύει στο αρχείο “cameraman\_inv.tif”.

**Άσκηση 3**

Υλοποιήστε την παρακάτω συνάρτηση σε MATLAB (jpeg.m) η οποία προσομοιώνει την λειτουργία ενός κωδικοποιητή JPEG.

```
function y = jpeg(x, quality)
%JPEG Compresses an image using a JPEG approximation.
% Y = JPEG(X, QUALITY) compresses image X based on 8 x 8 DCT
% transforms, coefficient quantization, and DC DPCM coding.
% Input QUALITY determines the amount of information that
% is lost and compression achieved. Y is an encoding structure
% containing fields:
%
%   Y.size      Size of X
%   Y.numblocks Number of 8-by-8 encoded blocks
%   Y.quality   Quality factor (as percent)
%   Y.huffman  Huffman encoding structure, as returned by
%              MAT2HUFF
```

Για να υλοποιήσετε τον κωδικοποιητή θα χρειαστεί να ακολουθήσετε τα παρακάτω βήματα:

1. Διαβάστε μια εικόνα αποχρώσεων του γκρι και φορτώστε την (εντολή imread) στο περιβάλλον MATLAB. Για διευκόλυνσή σας χρησιμοποιείστε την εικόνα “cameraman.tif”.
2. Αποθηκεύστε το μέγεθος της εικόνας
3. Χρησιμοποιείστε την εντολή blkproc για να επεξεργαστείτε την εικόνα κατά μπλοκ 8X8

4. Εφαρμόστε μετασχηματισμό DCT σε κάθε μπλοκ της εικόνας (χρήση της εντολής `dct2` σε συνδυασμό με την εντολή `blkproc`)
5. Διαιρέστε το αποτέλεσμα του μετασχηματισμού με τον πίνακα κβαντισμού διά τον quality factor. Ο quality factor μπορεί να πάρει τιμές μέσα στο διάστημα (0, 1]. Ο πίνακας κβαντισμού Q βρίσκεται αποθηκευμένος στο αρχείο `QTable.mat`.
6. Στη συνέχεια μετατρέψτε το μπλοκ των κβαντισμένων συντελεστών σε διάνυσμα χρησιμοποιώντας την τεχνική zig-zag scanning. Η σειρά σάρωσης των συντελεστών βρίσκεται στο αρχείο “zigzag64.mat”.
7. Δημιουργήστε ένα διάνυσμα  $r$  και αποθηκεύστε στην αρχή του τους DC συντελεστές αφού πρώτα εφαρμόσετε διαφορική κωδικοποίηση χρησιμοποιώντας την συνάρτηση `dpcm` (αρχείο `dpcm.m`).
8. Σε κάθε διάνυσμα AC συντελεστών εξαλείψτε όλα τα μηδενικά στοιχεία μετά το τελευταίο μη μηδενικό στοιχείο και προσθέστε στο τέλος κάθε διανύσματος έναν ειδικό χαρακτήρα EOB που θα σημαίνει το τέλος του μπλοκ. Θεωρήστε ως ειδικό χαρακτήρα: τη μέγιστη τιμή όλου του πίνακα + 1.
9. Προσθέστε στο διάνυσμα  $r$  τους AC συντελεστές από όλα τα μπλοκ.
10. Εκτυπώστε στην οθόνη το μέγεθος του διανύσματος  $r$ .
11. Υλοποιήστε την κωδικοποίηση Huffman του διανύσματος  $r$  χρησιμοποιώντας την συνάρτηση `mat2huff.m`.
12. Δημιουργήστε τη δομή  $Y$  που αποτελεί την μεταβλητή που επιστρέφει η συνάρτηση.
13. Υπολογίστε και εκτυπώστε στην οθόνη:
  - a. Το αρχικό μέγεθος της ασυμπίεστης εικόνα
  - b. Το μέγεθος της συμπίεσμένης κατά JPEG εικόνας. Για να το υπολογίσετε χρησιμοποιείτε την έξοδο `Y.code` της συνάρτησης `mat2huff.m` και πολλαπλασιάστε επί 16.
  - c. Το λόγο συμπίεσης που επιτυγχάνεται.
14. Επαναλάβετε την παραπάνω διαδικασία συμπίεσης για διαφορετικές τιμές του quality factor. Τι παρατηρείται για το μέγεθος του διανύσματος  $r$  και το λόγο συμπίεσης;