

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΑΣΚΗΣΗ 1

ΜΕΡΟΣ 1

Σε αυτό το μέρος θα δείξουμε ότι ο DCT συγκεντρώνει την ενέργεια μίας εικόνας σε μικρό αριθμό συντελεστών. Με χρήση Matlab ή Octave:

1. Φορτώστε την εικόνα *Cameraman* που παρέχεται από το Matlab Image Processing toolbox χρησιμοποιώντας την εντολή: `f=imread('cameraman.tif');`. Μπορείτε να δείτε την εικόνα στην οθόνη χρησιμοποιώντας διάφορες συναρτήσεις όπως: `imagesc(f);`. Θα χρειαστεί επίσης να δώσετε την εντολή `colormap(gray);`.
2. Πάρτε τον DCT $F(u, v)$ της εικόνας χρησιμοποιώντας τη συνάρτηση `dct2`.
3. Δώστε τιμή μηδέν στους συντελεστές με $|F(u, v)| < 5$ και πάρτε τον αντίστροφο DCT χρησιμοποιώντας τη συνάρτηση `idct2.m`. Μετατρέψτε την εικόνα σε `uint8`. Προσέξτε τη μετατροπή αρνητικών τιμών και τιμών μεγαλύτερων του 255. Εκτυπώστε την εικόνα που αναδημιουργήθηκε. Υπολογίστε το PSNR. Πόσοι συντελεστές μηδενίστηκαν;
4. Επαναλάβετε το παραπάνω βήμα για $|F(u, v)| < 10$ και $|F(u, v)| < 20$.

Θα παραδώσετε τον κώδικα που γράψατε. Επίσης, για καθένα από τα τρία κατώφλια, θα εκτυπώσετε την εικόνα που αναδημιουργήθηκε, το PSNR και το πλήθος των συντελεστών που μηδενίστηκαν.

ΜΕΡΟΣ 2

Γράψτε πρόγραμμα Matlab ή Octave που:

1. Διαβάζει την εικόνα *Cameraman* χρησιμοποιώντας την εντολή: `f=imread('cameraman.tif');`.
2. Υπολογίζει την εντροπία της εικόνας χρησιμοποιώντας τη συνάρτηση `entropy`.
3. Χωρίζει την εικόνα σε μπλοκ διαστάσεων 8×8 .
4. Παίρνει τον DCT για κάθε μπλοκ χρησιμοποιώντας τη συνάρτηση `dct2`.
5. Κβαντίζει τους συντελεστές κάθε μπλοκ σύμφωνα με τη σχέση $\hat{F}(u, v) = \text{round}(F(u, v)/Q(u, v))$, όπου $Q(u, v)$ είναι στοιχεία ενός πίνακα κβάντισης Q .

6. Υπολογίζει την εντροπία της απόλυτης τιμής $|\hat{F}(u, v)|$ των κβαντισμένων συντελεστών χρησιμοποιώντας τη συνάρτηση `entropy`.
7. Κάνει ‘αντίστροφη κβάντιση’ για τους συντελεστές σύμφωνα με τη σχέση $\tilde{F}(u, v) = \hat{F}(u, v) \cdot Q(u, v)$.
8. Παίρνει τον αντίστροφο DCT κάθε μπλοκ χρησιμοποιώντας τη συνάρτηση `idct2`.
9. Μετατρέπει την εικόνα σε `uint8`. Προσέξτε τη μετατροπή αρνητικών τιμών και τιμών μεγαλύτερων του 255.
10. Υπολογίζει το PSNR της εικόνας.

Τρέξτε το πρόγραμμα για

$$Q = Q_1 = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

και για $Q = 2 \cdot Q_1$ και $Q = 3 \cdot Q_1$.

Θα παραδώσετε τον κώδικα που γράψατε. Επίσης, θα εκτυπώσετε την εντροπία της αρχικής εικόνας και, για κάθε πίνακα Q , θα εκτυπώσετε την εντροπία της απόλυτης τιμής των κβαντισμένων συντελεστών και το PSNR που υπολογίσατε, καθώς και την εικόνα που αναδημιουργήθηκε.

ΜΕΡΟΣ 3

Γράψτε ένα πρόγραμμα Matlab ή Octave που θα παίρνει ως είσοδο τους 63 κβαντισμένους AC συντελεστές ενός μπλοκ 8×8 (μετά από zig-zag scan) και θα εκτυπώνει τα ζεύγη (LEVEL, RUN) που πρέπει να κωδικοποιηθούν. Θα παραδώσετε τον κώδικα που γράψατε καθώς και το αποτέλεσμα που εκτυπώνει αν οι κβαντισμένοι AC συντελεστές είναι:

7, 0, 3, 0, 2, 3, 0, 0, 2, 4, 0, 1, 0, 0, 0, 1, 0, 0, -1, -1, 0, 1, 0, 0, 0, 1, 0, ..., 0.

Η προθεσμία για την παράδοση της άσκησης είναι στις 7 Νοεμβρίου 2019. Θα παραδώσετε τρία αρχεία κώδικα (ένα για κάθε μέρος της άσκησης) και ένα αρχείο pdf που περιέχει τις εικόνες και τα αποτελέσματα που ζητούνται. Η παράδοση θα γίνει με χρήση turnin και την εντολή: `turnin assignment1@mye025 code1.m code2.m code3.m results.pdf`