

ΘΕΜΑΤΑ ΨΗΦΙΑΚΗΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΕΙΚΟΝΑΣ:
ΣΥΜΠΙΕΣΗ ΚΑΙ ΥΠΕΡΑΝΑΛΥΣΗ ΒΙΝΤΕΟ
ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2017-2018

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΑΣΚΗΣΗ 1

ΜΕΡΟΣ 1

Θα αναπτύξετε μία γρήγορη υλοποίηση του Διακριτού Μετασχηματισμού Συνημιτόνου (Discrete Cosine Transform, DCT) χρησιμοποιώντας το Matlab.

Ο μονοδιάστατος DCT, $F(u)$ μίας ακολουθίας $f(x)$, $x = 0, \dots, N - 1$ δίνεται από την εξίσωση

$$F(u) = w(u) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{(2x+1)\pi u}{2N}\right], \quad (1)$$

όπου

$$w(u) = \begin{cases} \frac{1}{\sqrt{N}} & \text{αν } u = 0 \\ \sqrt{\frac{2}{N}} & \text{διαφορετικά,} \end{cases} \quad (2)$$

και $u = 0, \dots, N - 1$.

Είναι δυνατόν να υπολογιστεί ο $F(u)$, ο DCT της ακολουθίας $f(x)$ μέσω του $G(u)$, του DFT της ακολουθίας $g(x)$, όπου

$$g(x) = \begin{cases} f(x), & 0 \leq x \leq N - 1 \\ f(2N - 1 - x), & N \leq x \leq 2N - 1 \end{cases} \quad (3)$$

Ο DFT της $g(x)$ είναι

$$G(u) = \sum_{x=0}^{2N-1} g(x) W_{2N}^{xu}, \quad (4)$$

όπου $W_N = e^{-j(2\pi/N)}$ και $u = 0, \dots, 2N - 1$.

Οι $f(x)$ και $F(u)$ έχουν μήκος N ενώ οι $g(x)$ και $G(u)$ έχουν μήκος $2N$. Υπάρχει όμως ο Fast Fourier Transform (FFT), που είναι μία γρήγορη υλοποίηση του DFT. Άρα, υπολογίζοντας το $G(u)$ και σχετίζοντάς το με το $F(u)$, μπορούμε να μειώσουμε σημαντικά την υπολογιστική πολυπλοκότητα της υλοποίησης, σε σύγκριση με υλοποίηση χρησιμοποιώντας τον ορισμό του DCT.

Πρέπει να κάνετε τα εξής:

1. Βρείτε αναλυτικά τη σχέση μεταξύ $G(u)$ και $F(u)$. Δώστε όλες τις λεπτομέρειες της μαθηματικής απόδειξης.

2. Γράψτε μία συνάρτηση του Matlab `mydct.m` η οποία για μία ακολουθία εισόδου $f(x)$, υπολογίζει την $g(x)$, μετα υπολογίζει το $G(u)$ με τη συνάρτηση `fft` του Matlab, και τέλος υπολογίζει το $F(u)$, το DCT της $f(x)$ από το $G(u)$.
3. Γράψτε μία συνάρτηση Matlab `mydct2.m`, η οποία υπολογίζει το διδιάστατο DCT ενός διδιάστατου πίνακα $f(x, y)$. Χρησιμοποιήστε τη συνάρτηση `mydct.m` σε συνδυασμό με row-column decomposition. Αυτό σημαίνει ότι ο διδιάστατος DCT υπολογίζεται σε δύο βήματα. Πρώτα, κάθε γραμμή του πίνακα $f(x, y)$ θεωρείται ως ένα μονοδιάστατο σήμα και λαμβάνεται ο μονοδιάστατος DCT κάθε γραμμής. Έπειτα, κάθε στήλη του αποτελέσματος θεωρείται ως μονοδιάστατο σήμα και λαμβάνεται ο μονοδιάστατος DCT κάθε στήλης.
4. Δημιουργήστε έναν οποιοδήποτε πίνακα μεγέθους 8×8 και υπολογίστε τον DCT χρησιμοποιώντας τη συνάρτηση `mydct2.m` και την ενσωματωμένη συνάρτηση `dct2.m`. Δείξτε ότι το αποτέλεσμα είναι το ίδιο.

ΜΕΡΟΣ 2

1. Φορτώστε την εικόνα *Cameraman* που υπάρχει στο Matlab Image Processing toolbox με την εντολή: `f=imread('cameraman.tif');`.
2. Χωρίστε την εικόνα σε μπλοκ μεγέθους 8×8 και υπολογίστε τον 8×8 DCT κάθε μπλοκ. Χρησιμοποιήστε τη συνάρτηση `mydct2.m`.
3. Για κάθε μπλοκ, κρατήστε τους 20 συντελεστές DCT με τα μεγαλύτερα μέτρα και μηδενίστε τους υπόλοιπους συντελεστές.
4. Πάρτε τον αντίστροφο DCT κάθε μπλοκ χρησιμοποιώντας τη συνάρτηση `idct2.m`. Εμφανίστε στην οθόνη και εκτυπώστε την εικόνα που δημιουργήθηκε. Υπολογίστε και εκτυπώστε το PSNR της εικόνας που δημιουργήθηκε. Για τον υπολογισμό του PSNR πρέπει οι δύο εικόνες που συγκρίνονται να είναι τύπου `uint8`.
5. Επαναλάβετε το πείραμα κρατώντας μόνο τους 10 συντελεστές DCT με τα μεγαλύτερα μέτρα και μηδενίζοντας τους υπόλοιπους.

ΜΕΡΟΣ 3

1. Φορτώστε την εικόνα *Cameraman* όπως στο Μέρος 2 και χωρίστε την σε μπλοκ μεγέθους 8×8 .

2. Διατάζετε κάθε μπλοκ μεγέθους 8×8 σε ένα διάνυσμα μεγέθους 64×1 χρησιμοποιώντας λεξικογραφική διάταξη, δηλ. κάθε διάνυσμα θα αποτελείται από τις γραμμές του μπλοκ ενωμένες στη σειρά. Θα έχετε έτσι δημιουργήσει 1024 διανύσματα \underline{s}_i , $i = 1, \dots, 1024$ (η αρχική εικόνα είναι μεγέθους 256×256).
3. Εκτιμήστε τη μέση τιμή των διανυσμάτων ως

$$\underline{\eta}_s = \frac{1}{1024} \sum_{i=1}^{1024} \underline{s}_i$$

Εκτιμήστε τον πίνακα συνδιακύμανσης των δεδομένων ως

$$[C]_s = \frac{1}{1024} \sum_{i=1}^{1024} [(\underline{s}_i - \underline{\eta}_s)(\underline{s}_i - \underline{\eta}_s)^H]$$

4. Βρείτε τον πίνακα του αντίστροφου KLT $[\Phi]$.
5. Υπολογίστε τον KLT για κάθε διάνυσμα \underline{s}_i .
6. Για κάθε διάνυσμα, κρατήστε μόνο τους 20 συντελεστές KLT με τα μεγαλύτερα μέτρα και μηδενίστε τους υπόλοιπους.
7. Πάρτε τον αντίστροφο KLT κάθε διανύσματος. Διατάζετε τα διανύσματα σε μπλοκ μεγέθους 8×8 αντιστρέφοντας τη διαδικασία της λεξικογραφικής διάταξης που χρησιμοποιήσατε προηγουμένως. Εμφανίστε στην οθόνη και εκτυπώστε την εικόνα που δημιουργήθηκε. Υπολογίστε και εκτυπώστε το PSNR της εικόνας που δημιουργήθηκε.
8. Επαναλάβετε το πείραμα κρατώντας μόνο τους 10 συντελεστές DCT με τα μεγαλύτερα μέτρα και μηδενίζοντας τους υπόλοιπους.
9. Συγκρίνετε τα αποτελέσματα του Μέρους 2 και του Μέρους 3. Ποιος από τους δύο μετασχηματισμούς οδηγεί σε καλύτερη ποιότητα εικόνας χρησιμοποιώντας τον ίδιο αριθμό συντελεστών; Γιατί;

Η άσκηση θα παραδοθεί στο μάθημα την Τρίτη 28 Νοεμβρίου 2018.

Παραδώστε εκτυπωμένο τον κώδικα που γράφατε. Επίσης, για το Μέρος 1, παραδώστε τη μαθηματική απόδειξη που ζητείται στο βήμα 1 και το αποτέλεσμα της κλήσης των συναρτήσεων στο βήμα 4. Για το Μέρος 2 και το Μέρος 3, παραδώστε εκτυπωμένες τις εικόνες που δημιουργήθηκαν και τα αντίστοιχα PSNR για καθένα από τα τέσσερα συνολικά πειράματα. Επίσης απαντήστε στο ερώτημα του βήματος 9 του Μέρους 3.