ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΟΛΥΜΕΣΑ

ΑΝΑΦΟΡΑ 1ης ΑΣΚΗΣΗΣ

ΥΛΟΠΟΙΗΣΗ ΔΙΑΚΡΙΤΟΥ ΜΕΤΑΣΧΙΜΑΤΙΣΜΟΥ ΣΥΝΗΜΙΤΟΝΟΥ (DESCRETE COSINE TRANSFORM, DCT)

Παναγιώτης Μουζούρης 4561

Χειμερινό Εξάμηνο 2023-2024

Εισαγωγή

Θα αναπτύξουμε μία υλοποίηση του Διακριτού Μετασχηματισμού Συνημίτονου (Discrete Cosine Transform, DCT) χρησιμοποιώντας το Octave.

Μέρος 1ο

- Σχέση μεταξύ G(u) και F (u).
- Δημιουργία μίας συνάρτησης mydct.m η οποία για μία ακολουθία εισόδου f(x) υπολογίζει τη g(x), μετα υπολογίζει το G(u) με τη συνάρτηση fft του Octave, και τέλος υπολογίζει το F (u), τον DCT της f (x) από το G(u).
- Δημιουργία μίας συνάρτησης mydct2.m, η οποία υπολογίζει τον δισδιάστατο DCT ενός πίνακα f (x, y).Με την βοήθεια της συναρτησης mydct.m.

Μέρος 2ο

- Συμπίεση της εικόνας "cameraman.tif" με χρήση DCT και κβάντισης.

Μέρος 1ο

1.

1.1 Αναλυτική **σχέση μεταξύ G(u) και F(u)**

Τα μεγέθη των σημειωμάτων f(x) και F(u) είναι N, ενώ τα μεγέθη των σημειωμάτων g(x) και G(u) είναι 2N. Χρησιμοποιώντας τη γρήγορη μετασχηματισμό Fourier (FFT), ο οποίος αποτελεί υλοποίηση του διακριτού μετασχηματισμού Fourier (DFT), μπορούμε να υπολογίσουμε το G(u) και να το συσχετίσουμε με το F(u). Αυτή η προσέγγιση επιτρέπει σημαντική μείωση της υπολογιστικής πολυπλοκότητας της υλοποίησης σε σύγκριση με την πλήρη υλοποίηση χρησιμοποιώντας τον παραδοσιακό ορισμό του δισδιάστατου κοσινού μετασχηματισμού (DCT).

$$\bullet F(u) = w(u)\sum_{x=0}^{N-1} f(x)\cos[\frac{(2x+1)\pi u}{2N}] \qquad \bullet G(u) = \sum_{x-0}^{2N-1}(g\ \ x)e^{\frac{-j2\pi xu}{2N}}$$

$$g(x) = f(2N-1-x), N \leq x \leq 2N-1 \qquad g(x) = f(x), 0 \leq x \leq N-1$$

G(u) σε αναλυτική βάση θεωρίας:

$$G(u) = \sum_{x=0}^{N-1} f(x)e^{\frac{-j2\pi xu}{2N}} + \sum_{x=N}^{2N-1} f(2N-1-x)e^{\frac{-j2\pi(2N-1-x)u}{2N}}$$

$$x' = 2N-1-x \Rightarrow x = 2N-1-x'$$

Για $x = N \Rightarrow x' = N-1$

Για $x = 2N-1 \Rightarrow x' = 0$

$$\Rightarrow G(u) = \sum_{x=0}^{N-1} f(x)e^{\frac{-j2\pi xu}{2N}} + \sum_{x=0}^{N-1} f(x)e^{\frac{-j2\pi(2N-1-x)u}{2N}}$$

$$\Rightarrow G(u) = \sum_{x=0}^{N-1} f(x)[e^{\frac{-j2\pi xu}{2N}} + e^{\frac{-j2\pi(2N-1-x)u}{2N}}]$$

$$e^{\frac{-j2\pi(2N-1-x)u}{2N}} = e^{\frac{-j2\pi 2Nu}{2N}} e^{\frac{j2\pi u}{2N}} e^{\frac{-j2\pi xu}{2N}} = e^{\frac{j2\pi u}{2N}} e^{\frac{j2\pi xu}{2N}}$$

$$\Rightarrow G(u) = \sum_{x=0}^{N-1} f(x)[e^{\frac{-j2\pi xu}{2N}} + e^{\frac{j2\pi(x+1)u}{2N}}]$$

Αθροισμα της αγκύλης:

$$e^{\frac{-j2\pi xu}{2}} + e^{\frac{j2\pi(x+1)u}{2N}} = e^{\frac{-j(2\pi xu+\pi u-\pi u)}{2N}} e^{\frac{j(2\pi xu+2\pi u)}{2N}}$$

$$= e^{\frac{(-j(2x+1)\pi u-j(-\pi u))}{2N}} e^{\frac{j(2\pi xu+\pi u+\pi u)}{2N}}$$

$$= e^{\frac{-j(2x+1)\pi u}{2N}} e^{\frac{j\pi u}{2N}} e^{\frac{j(2x+1)\pi u}{2N}} e^{\frac{j\pi u}{2N}}$$

$$= e^{\frac{j\pi u}{2N}}[e^{\frac{j(2x+1)\pi u}{2}} + e^{\frac{-j(2x+1)\pi u}{2}}]$$

Από θεωρία έχουμε:

$$\cos(\theta) = \frac{1}{2}(e^{j\theta} + e^{-j\theta}) \quad \text{άρα} \quad e^{j\theta} + e^{-j\theta} = 2\cos(\theta)$$

$$\theta = \frac{(2+1)\pi u}{2N} \Rightarrow e^{\frac{j(2x+1)\pi u}{2N}} + e^{\frac{-j(2x+1)\pi u}{2N}} = 2\cos[\frac{(2x+1)\pi u}{2N}]$$

$$\Rightarrow e^{\frac{j\pi u}{2N}}[e^{\frac{j(2x+1)\pi u}{2N}} + e^{\frac{-j(2x+1)\pi u}{2N}}] = 2e^{\frac{j\pi u}{2N}} \cos[\frac{(2x+1)\pi u}{2N}]$$

$$G(u) = \sum_{x=0}^{N-1} f(x) 2e^{\frac{j\pi u}{2N}} \cos[\frac{(2x+1)\pi u}{2N}] = 2e^{\frac{j\pi u}{2N}} \sum_{x=0}^{N-1} f(x)\cos[\frac{(2x+1)\pi u}{2N}]$$

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

Αρά έχουμε: $G(u) = \dfrac{2e^{\frac{j\pi u}{2N}}}{w(u)} F(u)$

Για w(u)==0: $w(u) = \dfrac{1}{N}$

Υπόλοιπα: $w(u) = \sqrt{\dfrac{2}{N}}$

1.2 Ανάλυση mydct() κώδικα και αποτελέσματα

Η συνάρτηση mydct() δέχεται ένα διάνυσμα f ως είσοδο. Για κάθε τιμή x στο διάνυσμα f, ελέγχει αν ισχύει x >= 0 και x <= N. Αν αυτή η συνθήκη είναι αληθής, τότε η τιμή του g(x) ορίζεται ως f(x). Σε διαφορετική περίπτωση, αν x > N και x <= 2N, τότε η τιμή του g(x) υπολογίζεται ως f(2N - x + 1).Στη συνέχεια, η συνάρτηση χρησιμοποιεί την έτοιμη συνάρτηση fft() για τον υπολογισμό της G(u), που αντιστοιχεί στον Διακριτού μετασχηματισμό Συνημίτονου (DCT) του f(x).

Βάσει της απόδειξης που προσφέρατε, ενσωματώνεται η εξίσωση της G(u) σε σχέση με την F(u). Έπειτα, υπολογίζεται ξανά η G(u) χρησιμοποιώντας αυτήν τη σχέση.

Ουσιαστικά, η συνάρτηση mydct() υπολογίζει τον Διακριτού μετασχηματισμό Συνημίτονου (DCT) του διανύσματος f(x) χρησιμοποιώντας τη συνάρτηση fft() και τον αντίστροφο DCT για να υπολογίσει τον αρχικό DCT του διανύσματος.

Σαν είσοδος έχουμε αυτή την ακολουθία: [0 1 2 3 4 5 6 7 8]

```
ans = 0
ans =  1
ans =  2
ans =  3
ans =  4
ans =  5
ans =  6
ans =  7
ans =  8
ans =  8
ans =  7
ans =  6
ans =  5
ans =  4
ans =  3
ans =  2
ans =  1
ans = 0
FFT G(u): 72.000000
FFT G(u): -32.163437
FFT G(u): -0.000000
FFT G(u): -3.000000
FFT G(u): -0.000000
FFT G(u): -0.704088
FFT G(u): 0.000000
FFT G(u): -0.132474
FFT G(u): 0.000000
FFT G(u): 0.000000
FFT G(u): 0.000000
FFT G(u): -0.132474
FFT G(u): 0.000000
FFT G(u): -0.704088
FFT G(u): -0.000000
FFT G(u): -3.000000
FFT G(u): -0.000000
FFT G(u): -32.163437
```

```
G(u): 240.000000
G(u): 240.000000
G(u): 327.264935
G(u): 327.264935
G(u): 72.000000
G(u): 72.000000
G(u): 159.264935
G(u): 159.264935
G(u): -96.000000
G(u): -96.000000
G(u): -8.735065
G(u): -8.735065
G(u): -264.000000
G(u): -264.000000
G(u): -176.735065
G(u): -176.735065
G(u): -432.000000
G(u): -432.000000
```

1.3 Ανάλυση mydct2() και αποτελέσματα

Η συνάρτηση mydct2.m υπολογίζει τον διδιάστατο μετασχηματισμό κοσινού (DCT) ενός διδιάστατου πίνακα f(x, y). Αυτό γίνεται χρησιμοποιώντας τη συνάρτηση mydct.m σε συνδυασμό με τη μέθοδο "row-column decomposition". Συγκεκριμένα, ο διδιάστατος DCT υπολογίζεται σε δύο βήματα. Πρώτα, κάθε γραμμή του πίνακα f(x, y) θεωρείται ως ένα μονοδιάστατο σήμα, και υπολογίζεται ο μονοδιάστατος DCT για κάθε γραμμή. Στη συνέχεια, κάθε στήλη του αποτελέσματος θεωρείται ως μονοδιάστατο σήμα, και υπολογίζεται ο μονοδιάστατος DCT για κάθε στήλη.

Για να επαληθεύσουμε ότι η συνάρτηση mydct2.m λειτουργεί σωστά, δημιουργούμε έναν οποιοδήποτε πίνακα μεγέθους 8x8 και υπολογίζουμε τον DCT του χρησιμοποιώντας τη συνάρτηση mydct2.m. Έπειτα, χρησιμοποιούμε την ενσωματωμένη συνάρτηση dct2 για να υπολογίσουμε τον DCT του ίδιου πίνακα. Κατόπιν, συγκρίνουμε τα δύο αποτελέσματα για να επιβεβαιώσουμε ότι είναι ίδια. Οπού δεν είναι λόγο λάθος στους υπολογισμούς μου για την G(u).

Η είσοδος που χρησιμοποιήθηκε για αυτά τα αποτελέσματα είναι:

1	9	17	25	33	41	49	57
2	10	18	26	34	42	50	58
3	11	19	27	35	43	51	59
4	12	20	28	36	44	52	60
5	13	21	29	37	45	53	61
6	14	22	30	38	46	54	62
7	15	23	31	39	47	55	63
8	16	24	32	40	48	56	64

Αποτελέσματα από το Mydct2():

114048	300288	775296	1539072	2591616	3932928	5563008	7481856
281472	740608	1910400	3790848	6381952	9683712	13696128	18419200
662400	1741056	4484736	8893440	14967168	22705920	32109696	43178496
1256832	3301632	8498304	16846848	28347264	42999552	60803712	81759744
2064768	5422336	13951104	27651072	46522240	70564608	99778176	134162944
3086208	8103168	20843136	41306112	69492096	105401088	149033088	200388096
4321152	11344128	29174400	57811968	97256832	147508992	208568448	280435200
5769600	15145216	38944896	77168640	129816448	196888320	278384256	374304256

Αποτελέσματα από το dct2() της matlab :

260.0000	-145.7731	-0.0000	-15.2385	0.0000	-4.5459	-0.0000	-1.1473
-18.2216	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-1.9048	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-0.5682	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-0.1434	0	0	0	0	0	0	0

>

Μερος 2ο

1.4 Ανάλυση κώδικα meros2

Το πρόγραμμα Octave που δημιουργήσαμε περιλαμβάνει τα εξής βήματα:

Διαβάζει την εικόνα "Cameraman" χρησιμοποιώντας την εντολή: f=imread('cameraman.tif');.Υπολογίζει την εντροπία της εικόνας χρησιμοποιώντας τη συνάρτηση entropy.Χωρίζει την εικόνα σε μπλοκ διαστάσεων 8 × 8.Παίρνει τον DCT για κάθε μπλοκ χρησιμοποιώντας τη συνάρτηση dct2.Κβαντίζει τους συντελεστές κάθε μπλοκ σύμφωνα με τη σχέση $Fˆ(u, v) = round(F(u, v)/Q(u, v))$, όπου $Q(u, v)$ είναι στοιχεία ενός πίνακα κβάντισης Q. Υπολογίζει την εντροπία της απόλυτης τιμής $|Fˆ(u, v)|$ των κβαντισμένων συντελεστών ολόκληρης της εικόνας χρησιμοποιώντας τη συνάρτηση entropy.Υπολογίζει το πλήθος των συντελεστών της εικόνας που μηδενίστηκαν μετά την κβάντιση.Κάνει αντίστροφη κβάντιση για τους συντελεστές σύμφωνα με τη σχέση $F˜(u, v) = Fˆ(u, v) \cdot Q(u, v)$.Παίρνει τον αντίστροφο DCT κάθε μπλοκ χρησιμοποιώντας τη συνάρτηση idct2.Μετατρέπει την εικόνα σε uint8, λαμβάνοντας υπόψη τη μετατροπή αρνητικών τιμών και τιμών μεγαλύτερων του 255.Υπολογίζει το PSNR (Peak Signal-to-Noise Ratio) της εικόνας και εμφανίζει και εκτυπώνει την εικόνα με τη συνάρτηση imagesc. Επιπλέον, χρησιμοποιεί την εντολή colormap(gray) για την κατάλληλη απεικόνιση.

Το πρόγραμμα τρέχει για διάφορες τιμές του πίνακα κβάντισης Q, συμπεριλαμβανομένων των Q1, 3 * Q1 και 6 * Q1.

Entropy of original picture: 7.0097

Q1:

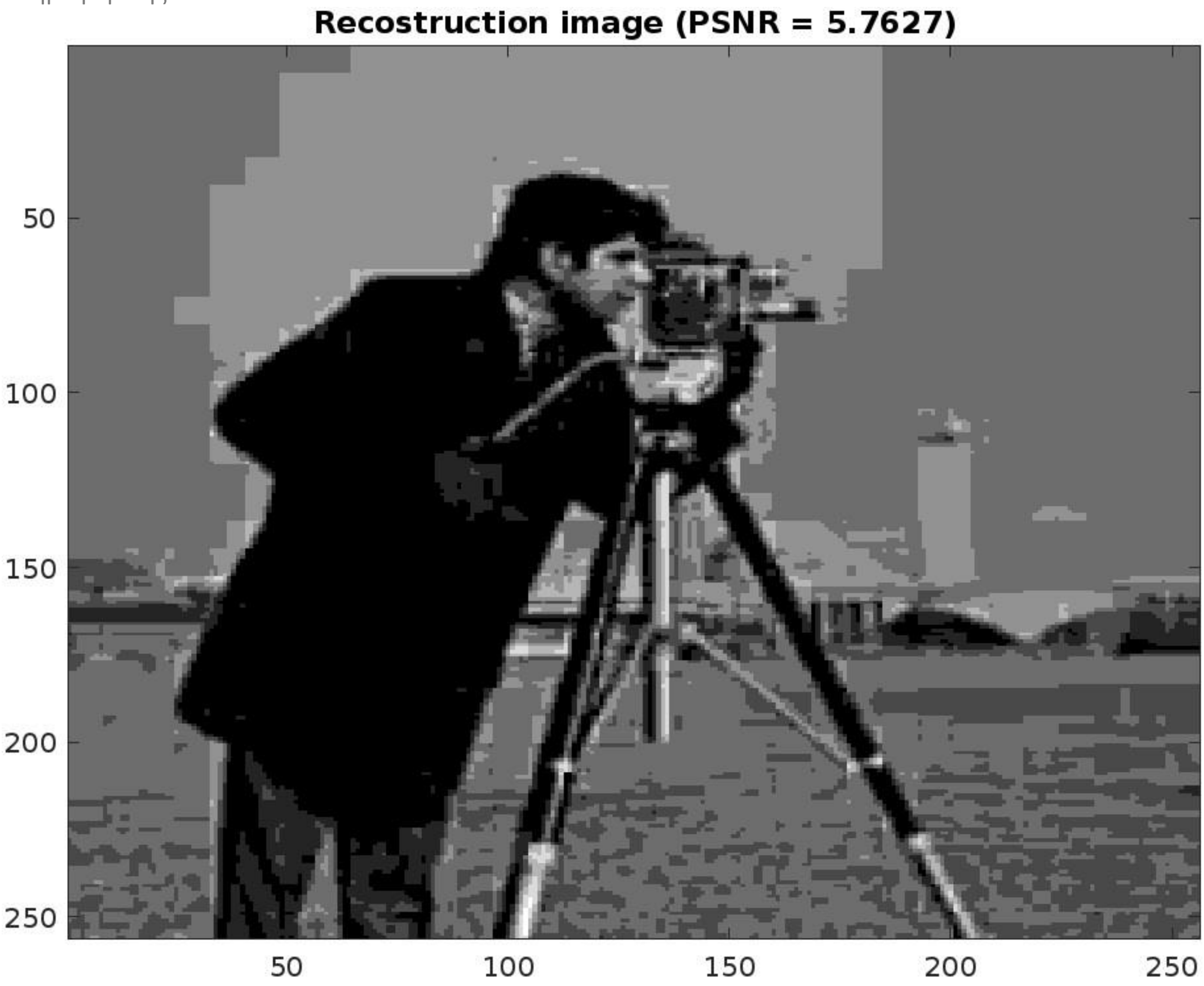
Entropy after the Q1:510.992009

SumOfZeros=4962

Q1*3:

Entropy after the Q2:510.992009

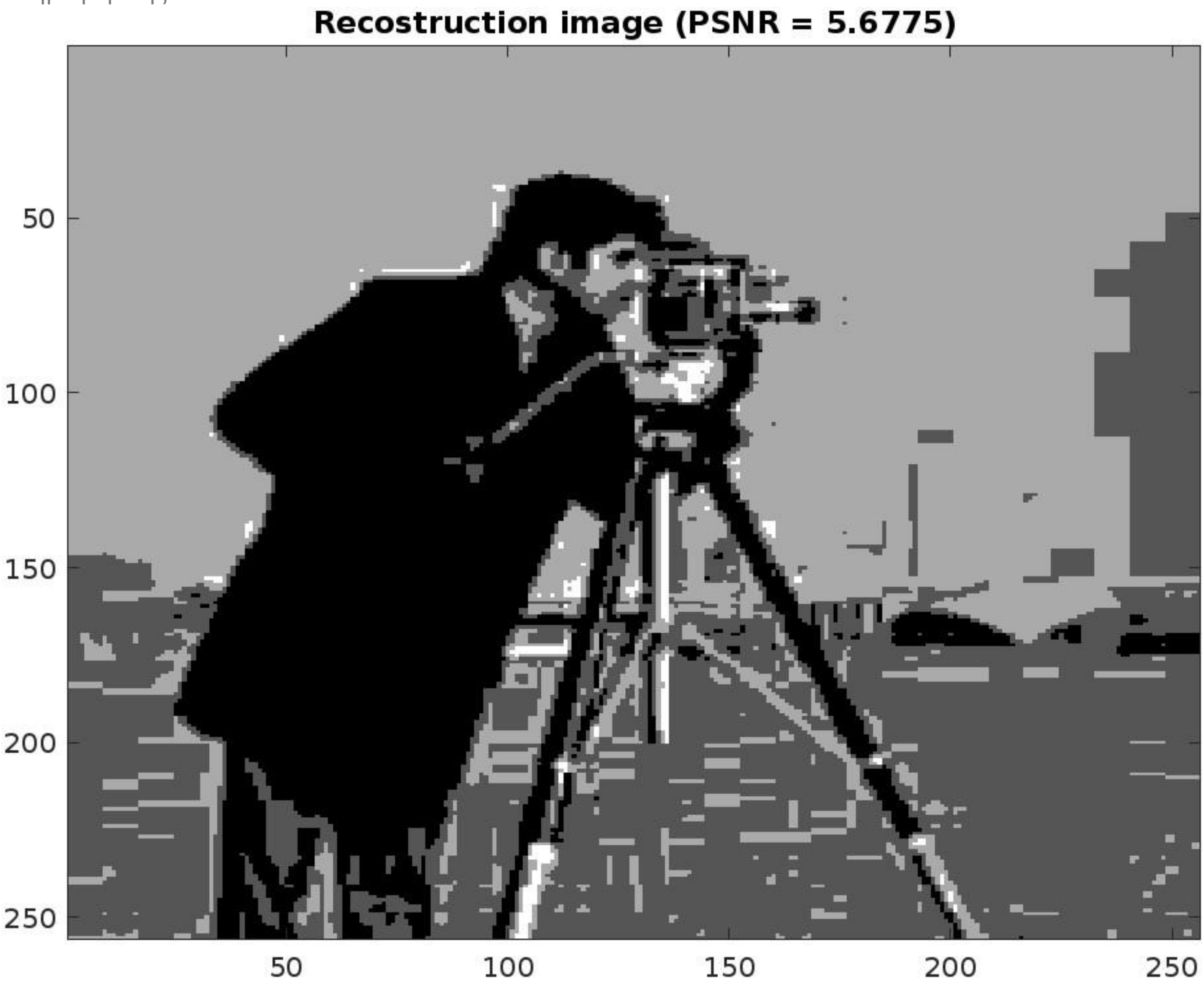
SumOfZeros=6092



Q1*6:

Entropy after the Q3:510.992009

SumOfZeros=6573


Recostruction image (PSNR = 5.6775)
50
100
150
200
250
50
100
150
200
250