

Domagoj Rojnić

Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek

Obrada slike i računalni vid

# **SEMINARSKI RAD**

„Implementacija kompresije slike“

Osijek, ožujak 2021.

# SADRŽAJ

<b>1. UVOD.....</b>	<b>1</b>
<b>2. STATE OF ART .....</b>	<b>2</b>
<b>3. ZADATAK .....</b>	<b>3</b>
<b>3.1. DCT jednadžba .....</b>	<b>3</b>
<b>3.2. DCT matrica.....</b>	<b>4</b>
<b>3.3. Proces DCT-a na 8x8 bloku .....</b>	<b>4</b>
<b>3.4. Kvantizacija.....</b>	<b>6</b>
<b>3.5. Kodiranje.....</b>	<b>7</b>
<b>3.6. Dekompresija .....</b>	<b>8</b>
<b>3.7. Usporedba matrica .....</b>	<b>9</b>
<b>3.8. Primjeri.....</b>	<b>9</b>
<b>4. ZAKLJUČAK .....</b>	<b>13</b>
<b>5. LITERATURA.....</b>	<b>14</b>
<b>6. POPIS NAPISANIH PROGRAMA I UPUTE ZA KORIŠTENJE .....</b>	<b>15</b>

# 1. UVOD

Kako ljudsko korištenje i oslanjanje na tehnologiju raste, tako raste i potreba za efikasnim načinima pohrane velike količine podataka. Na primjer, netko tko koristi web stranicu koja sadržava desetke, pa čak i stotine slika, najvjerojatno će trebati koristiti nekakvu metodu kompresije slike u svrhu pohrane navedenih slika. Potreba se javlja zato što prostor potreban za pohranu slika može doseći veliku cijenu. Srećom, danas je dostupno nekoliko metoda kompresije slike.

## 2. STATE OF ART

Kompresija slike može se podijeliti u dvije skupine: kompresija s gubitcima i kompresija bez gubitaka. JPEG kompresija slike danas predstavlja najrašireniju i najkorišteniju kompresiju slike s gubitcima, gdje se kompresija osniva oko diskretne kosinusove transformacije (engl. Discrete Cosine Transform, DCT). Ona predstavlja konačan skup podataka kao sumu kosinus funkcija koje osciliraju različitim frekvencijama. Korištenje kosinus funkcije u odnosu na sinus funkciju omogućava predstavljanja signala i podataka sa puno manje pojedinačnih signala.

### 3. ZADATAK

Proces JPEG kompresije slike, na kojoj se rad bazira, sastoji se od 5 osnovnih koraka. Ovim koracima će se opisati način izrade zadatka te omogućiti opsežno razumijevanje samoga procesa.

1. Slika se dijeli na 8x8 blokove
2. Na svaki blok primjenjuje se DCT
3. Svaki blok kompresiran je metodom kvantizacije
4. Polje kompresiranih blokova pohranjuje se u drastično smanjenom obliku
5. Slika se rekonstruira dekompresijom tj. procesom koji koristi inverznu diskretnu kosinus transformaciju (engl. Inverse Discrete Cosine Transform, IDCT)

#### 3.1. DCT jednadžba

Jednadžba diskretne kosinus transformacije računa vrijednost (i, j) piksela prema formuli prikazanoj na slici 3.1.1.

$$D(i,j) = \frac{1}{\sqrt{2N}} C(i)C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x,y) \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right]$$
$$C(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } u = 0 \\ 1 & \text{if } u > 0 \end{cases} \quad (1)$$

Slika 3.1.1. Jednadžba diskretne kosinus transformacije

$p(x, y)$  predstavlja (x, y) – ti element slike koja je predstavljena matricom p. N je veličina bloka na kojemu se radi diskretna kosinus transformacija. Jednadžba računa izlaz (i, j) transformiranog piksela na mjestu (x, y) originalne slike. Prilikom korištenja standardnog 8x8 bloka, kojeg JPEG kompresija također koristi, N je jednak 8 dok su vrijednosti x i y u rasponu od 0 do 7. Prethodno navedenim,  $D(i, j)$  se također može i zapisati kao:

$$D(i,j) = \frac{1}{4} C(i)C(j) \sum_{x=0}^7 \sum_{y=0}^7 p(x,y) \cos\left[\frac{(2x+1)i\pi}{16}\right] \cos\left[\frac{(2y+1)j\pi}{16}\right] \quad (2)$$

Iz razloga što DCT koristi kosinus funkcije, rezultirajuća matrica D ovisi o horizontalnim, dijagonalnim i vertikalnim frekvencijama. Tako će blok slike s puno promjena u frekvenciji

imati nasumičan oblik u rezultirajućoj matrici, dok će slika, koja se sastoji od samo jedne boje, imati rezultirajuću matricu takvu da će element prvog stupca i prvog retka biti iznimno velik dok će sve ostale vrijednosti biti 0.

### 3.2. DCT matrica

Kako bi se dobila matrica iz jednadžbe (1), koristit će se sljedeća jednadžba:

$$T_{i,j} = \begin{cases} \frac{1}{\sqrt{N}} & \text{ako je } i = 0 \\ \sqrt{\frac{2}{N}} \cos \left[ \frac{(2j+1)i\pi}{2N} \right] & \text{ako je } i > 0 \end{cases} \quad (3)$$

8x8 blok rezultira sljedećom matricom:

$$T = \begin{bmatrix} 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 & 0,3536 \\ 0,4904 & 0,4157 & 0,2778 & 0,0975 & -0,0975 & -0,2778 & -0,4157 & -0,4904 \\ 0,4619 & 0,1913 & -0,1913 & -0,4619 & -0,4619 & -0,1913 & 0,1913 & 0,4619 \\ 0,4157 & -0,0975 & -0,4904 & -0,2778 & 0,2778 & 0,4904 & 0,0975 & -0,4157 \\ 0,3536 & -0,3536 & -0,3536 & 0,3536 & 0,3536 & -0,3536 & -0,3536 & 0,3536 \\ 0,2778 & -0,4904 & 0,0975 & 0,4157 & -0,4157 & -0,0975 & 0,4904 & -0,2778 \\ 0,1913 & -0,4619 & 0,4619 & -0,1913 & -0,1913 & 0,4619 & -0,4619 & 0,1913 \\ 0,0975 & -0,2778 & 0,4157 & -0,4904 & 0,4904 & -0,4157 & 0,2778 & -0,0975 \end{bmatrix}$$

Prvi redak (i=0) ima sve vrijednosti jednake  $\frac{1}{\sqrt{8}}$ , kao što je i očekivano jednadžbom (3).

Matrica T je ortogonalna matrica. Prilikom procesa inverzne kosinus transformacije vrlo je važno da je ona ortogonalna kako bi se mogao izračunati inverz matrice T'.

### 3.3. Proces DCT-a na 8x8 bloku

Prije početka opisivanja procesa važno je napomenuti kako je raspon vrijednosti piksela crno-bijelih slika od 0 do 255, gdje je crna boja prezentirana 0, a bijela vrijednošću 255. Iz te činjenice je vidljivo kako se slika može precizno predstaviti sa 256 nijansi sive.

Slika se sastoji od stotine, pa čak i tisuće blokova piksela veličine 8x8 stoga je sljedeće navedeni opis procesa primjenjiv na pojedini blok pojedinačno, te shodno tome se primjenjuje na sve blokove zadane slike koracima opisanim u poglavlju 3.

Početak procesa se odvija na određenom bloku. Recimo da je taj određeni blok originalne slike sljedeći:

$$Original = \begin{bmatrix} 154 & 123 & 123 & 123 & 123 & 123 & 123 & 136 \\ 192 & 180 & 136 & 154 & 154 & 154 & 136 & 110 \\ 254 & 198 & 154 & 154 & 180 & 154 & 123 & 123 \\ 239 & 180 & 136 & 180 & 180 & 166 & 123 & 123 \\ 180 & 154 & 136 & 167 & 166 & 149 & 136 & 136 \\ 128 & 136 & 123 & 136 & 154 & 180 & 198 & 154 \\ 123 & 105 & 110 & 149 & 136 & 136 & 180 & 166 \\ 110 & 136 & 123 & 123 & 123 & 136 & 154 & 136 \end{bmatrix}$$

Pošto je DCT dizajniran da radi na vrijednostima piksela centriranih oko 0 tj. u rasponu od -128 do 127, svakoj vrijednosti originalnog bloka oduzima se 128. Ovo rezultira sljedećom matricom:

$$M = \begin{bmatrix} 26 & -5 & -5 & -5 & -5 & -5 & -5 & 8 \\ 64 & 52 & 8 & 26 & 26 & 26 & 8 & -18 \\ 126 & 70 & 26 & 26 & 52 & 26 & -5 & -5 \\ 111 & 52 & 8 & 52 & 52 & 38 & -5 & -5 \\ 52 & 26 & 8 & 39 & 38 & 21 & 8 & 8 \\ 0 & 8 & -5 & 8 & 26 & 52 & 70 & 26 \\ -5 & -23 & -18 & 21 & 8 & 8 & 52 & 38 \\ -18 & 8 & -5 & -5 & -5 & 8 & 26 & 8 \end{bmatrix}$$

Sada je sve spremno kako bi se izvršila diskretna kosinus transformacija, što se ostvaruje matričnim množenjem:

$$D = T M T' \quad (4)$$

Rezultat matričnog množenja je:

$$D = \begin{bmatrix} 162,3 & 40,6 & 20,0 & 72,3 & 30,3 & 12,5 & -19,7 & -11,5 \\ 30,5 & 108,4 & 10,5 & 32,3 & 27,7 & -15,5 & 18,4 & -2,0 \\ -94,1 & -60,7 & 12,3 & -43,4 & -31,3 & 6,1 & -3,3 & 7,1 \\ -38,6 & -83,4 & -5,4 & -22,2 & -13,5 & 15,5 & -1,3 & 3,5 \\ -31,3 & 17,9 & -5,5 & -12,4 & 14,3 & -6,0 & 11,5 & -6,0 \\ -0,9 & -11,8 & 12,8 & 0,2 & 28,1 & 12,6 & 8,4 & 2,9 \\ 4,6 & -2,4 & 12,2 & 6,6 & -18,7 & -12,8 & 7,7 & 12,0 \\ -10,0 & 11,2 & 7,8 & -16,3 & 21,5 & 0,0 & 5,9 & 10,7 \end{bmatrix}$$

Matrica D sastoji se od 64 DCT koeficijenta,  $c_{ij}$ , gdje su  $i$  i  $j$  u rasponu od 0 do 7. Koeficijent prvog retka i prvog stupca,  $c_{00}$ , predstavlja niske frekvencije bloka originalne slike. Kako se

udaljava u svim smjerovima od  $c_{00}$ , DCT koeficijenti predstavljaju sve više i više frekvencije originalnog bloka, gdje  $c_{77}$  predstavlja najvećoj frekvenciji. Važno je napomenuti kako je ljudsko oko osjetljivo na niske frekvencije na temelju čega se temelji korak kvantizacije.

### 3.4. Kvantizacija

Prethodno definirani 8x8 blok DCT koeficijenata spreman je za kompresiju putem kvantizacije. Kvantizacija predstavlja značajan i veoma koristan korak JPEG kompresije u kojemu se određuje stupanj kompresije slike kao i kvaliteta tako da se koristi određena kvantizacijska matrica. Ovo omogućava korisniku kompresije odlučivanje o stupnju kompresije koji prima vrijednosti iz intervala od 1 do 100. Odabir stupnja 1 rezultira najlošijom kvalitetom slike i najvećom kompresijom, dok odabir stupnja 100 daje najbolju kvalitetu i malu kompresiju. Omjer kompresije i kvalitete prilagođava se specifičnim potrebama korisnika.

Eksperimenti izvršeni na ljudskom vidnom sustavu rezultirali su standardnom JPEG kvantizacijskom matricom. Sa stupnjem kompresije 50, sljedeća matrica omogućava visoku kompresiju kao i izvrsnu kvalitetu dekomprimirane slike.

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Ukoliko se želi odabrati druga kvalitet i omjer kompresije, koristi se skalarni umnožak standardne kvantizacijske matrice. Za kvalitetu veću od 50 (manja kompresija, veća kvaliteta slike), standardna kvantizacijska matrica množi se sa  $\frac{100-kvaliteta}{50}$ . Za kvalitetu manju od 50 (veća kompresija, manja kvaliteta slike), standardna kvantizacijska matrica množi se s  $\frac{50}{kvaliteta}$ . Rezultirajuća matrica se tada zaokružuje na pozitivne cjelobrojne vrijednosti u rasponu od 0 do 255.



Kvantizacija se ostvaruje dijeljenjem matrica D i Q tako da se element na određenoj poziciji dijeli sa elementom na istoj toj poziciji u drugoj matrici (engl. element-wise matrix division) prema formuli:

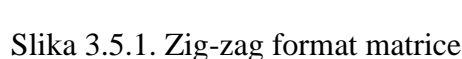
$$C_{i,j} = \text{round} \left( \frac{D_{i,j}}{Q_{i,j}} \right) \quad (5)$$

$$C = \begin{bmatrix} 10 & 4 & 2 & 5 & 1 & 0 & 0 & 0 \\ 3 & 9 & 1 & 2 & 1 & 0 & 0 & 0 \\ -7 & -5 & 1 & -2 & -1 & 0 & 0 & 0 \\ -3 & -5 & 0 & -1 & 0 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Prisjetivši se prethodno navedenog, da koeficijenti blizu gornjeg-lijevog ugla odgovaraju nižim frekvencijama, onima na koje je ljudsko oko najosjetljivije, jasno je uočljiv razlog izgleda matrice C. Nule predstavljaju manje važne tj. visoke frekvencije koje se odbacuju, čime dolazi do gubitka informacija radi čega je JPEG kompresija slike kompresija s gubitcima. Samo koeficijenti koji su različiti od 0 će se koristiti pri rekonstrukciji, odnosno dekompresiji slike. Zanimljivo je napomenuti kako bi kvaliteta kvantizacijske matrice od 10 rezultirala sa puno više nula u matrici C, dok bi na primjer kvaliteta od 90 generirala matricu C u kojoj je malen broj nula.

### 3.5. Kodiranje

Kvantizirana matrica C spremna je za završni korak kompresije. Prije pohrane, svi koeficijenti  $C_{i,j}$  se pretvaraju u niz binarnih podataka (011010...) putem enkodera. Prikazano slikom 3.5.1, matrica C se pretvara u takozvani zig-zag format nakon čega se dobija 1-D polje podataka. Na tako definirano jednodimenzionalno polje primjenjuje se kodiranje. Postoje različite vrste kodiranja: engl. run-length encoding, engl. Huffman encoding... U ovom radu korišteno je Huffman-ovo kodiranje. Ono se izvršava tako da se za polje brojeva izračuna frekvencija pojavljivanja svakog broja pojedinačno. Zatim se svim brojevima dodjeljuje binarni kod tako da se češće pojavljujućim brojevima dodjeljuje kraći kod.



Rekonstrukcija bloka originalne slike započinje dekodiranjem binarnog podatkovnog niza, koji predstavlja matricu  $C$ , u cjelobrojni jednodimenzionalni niz. Svaki element kvantiziranje matrice  $C_{i,j}$  se tada množi odgovarajućim elementom kvantizacijske matrice  $Q_{i,j}$  prema formuli:

$$R = \begin{bmatrix} 160 & 44 & 22 & 80 & 24 & 0 & 0 & 0 \\ 36 & 108 & 14 & 38 & 26 & 0 & 0 & 0 \\ -98 & -65 & 16 & -48 & -40 & 0 & 0 & 0 \\ -42 & -85 & 0 & -29 & 0 & 0 & 0 & 0 \\ -36 & 22 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

8

$$N = \text{round}(T'R T) + 128 \quad (7)$$

### 3.7. Usporedba matrica

$$Original = \begin{bmatrix} 154 & 123 & 123 & 123 & 123 & 123 & 123 & 136 \\ 192 & 180 & 136 & 154 & 154 & 154 & 136 & 110 \\ 254 & 198 & 154 & 154 & 180 & 154 & 123 & 123 \\ 239 & 180 & 136 & 180 & 180 & 166 & 123 & 123 \\ 180 & 154 & 136 & 167 & 166 & 149 & 136 & 136 \\ 128 & 136 & 123 & 136 & 154 & 180 & 198 & 154 \\ 123 & 105 & 110 & 149 & 136 & 136 & 180 & 166 \\ 110 & 136 & 123 & 123 & 123 & 136 & 154 & 136 \end{bmatrix}$$

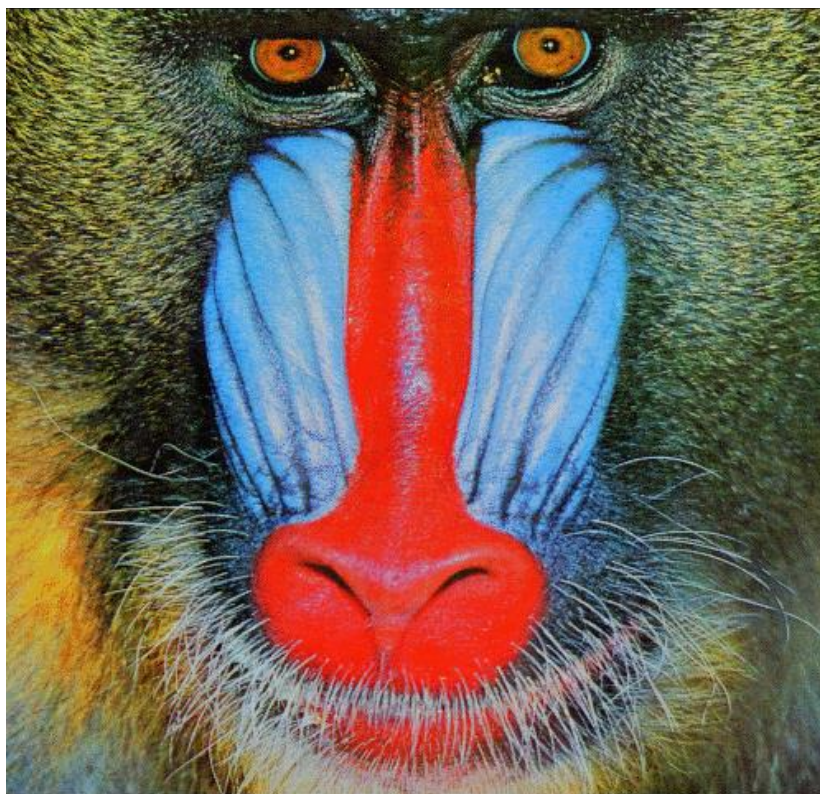
$$Dekompresirana = \begin{bmatrix} 149 & 134 & 119 & 116 & 121 & 126 & 127 & 128 \\ 204 & 168 & 140 & 144 & 155 & 150 & 135 & 125 \\ 253 & 195 & 155 & 166 & 183 & 165 & 131 & 111 \\ 245 & 158 & 148 & 166 & 184 & 160 & 124 & 107 \\ 188 & 149 & 132 & 155 & 172 & 159 & 141 & 136 \\ 132 & 123 & 125 & 143 & 160 & 166 & 168 & 171 \\ 109 & 119 & 126 & 128 & 139 & 158 & 168 & 166 \\ 111 & 127 & 127 & 114 & 118 & 141 & 147 & 135 \end{bmatrix}$$

Uspoređujući matrice vidljiv je odličan rezultat s obzirom da je gotovo 70% DCT koeficijenata odbačeno prethodno dekompresiji bloka. Slični rezultati pojavit će se i na ostalim blokovima koji definiraju originalnu sliku te će se JPEG kompresirana slika jedva raspoznati u odnosu na početno zadanu sliku.

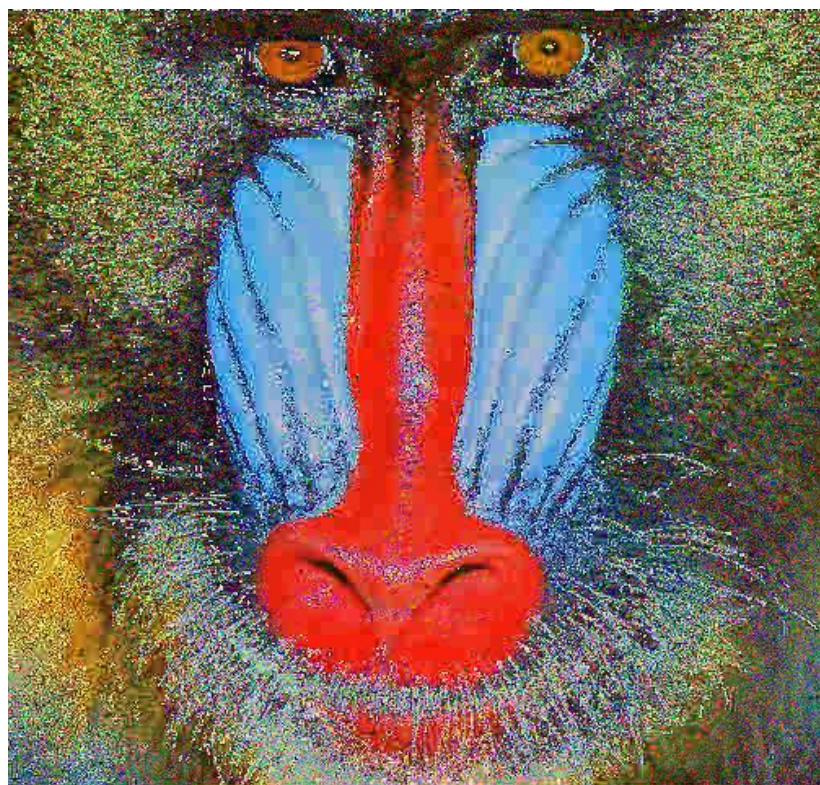
Crno-bijela slika ima 256 mogućih nijansi sive, te je razlika od otprilike 10 ljudskom oku teško vidljiva.

### 3.8. Primjeri

Program je izveden na poznatoj slici 'baboo.bmp' u omjere kompresije 10, 30, 50, 70 i 90. Slike će biti prikazane u većem formatu kako bi bile uočljivije razlike za različite omjere kompresije.

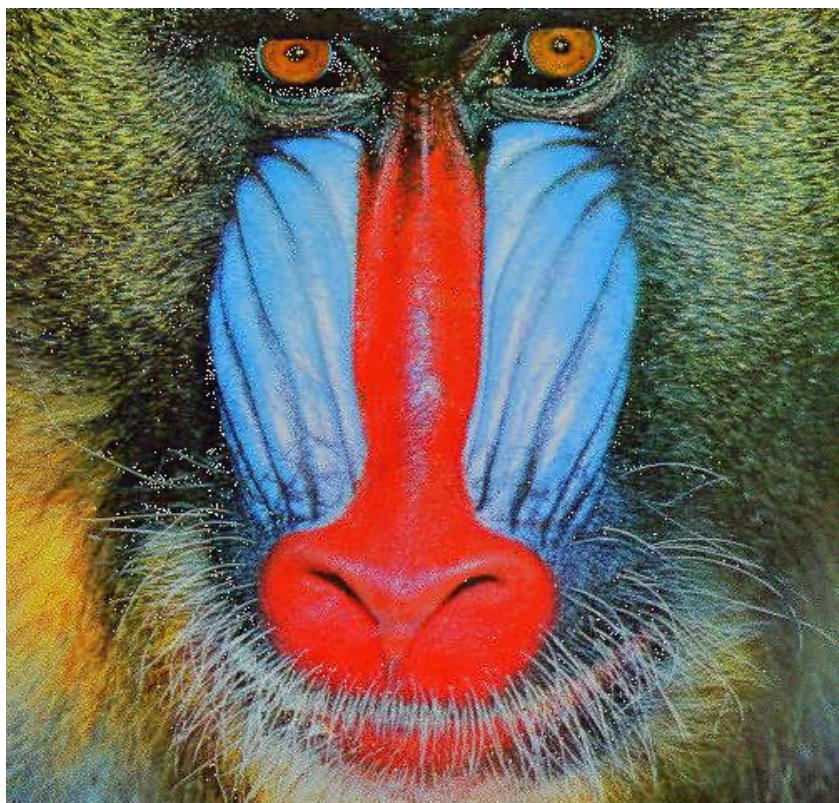


Slika 3.8.1. Originalna slika baboon

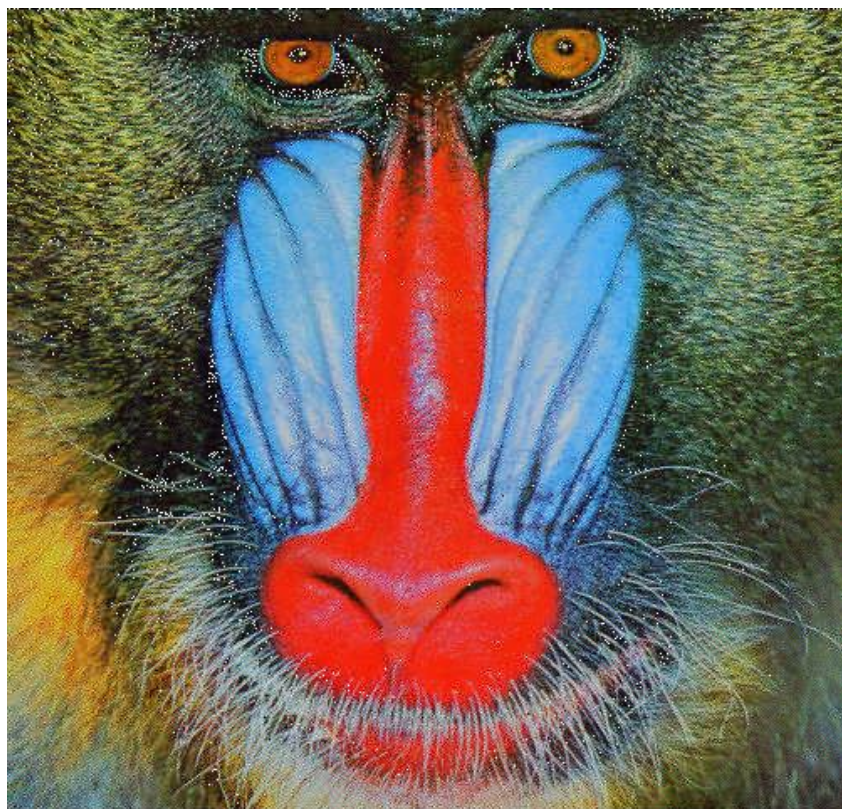


Slika 3.8.2. Omjer kompresije = 10



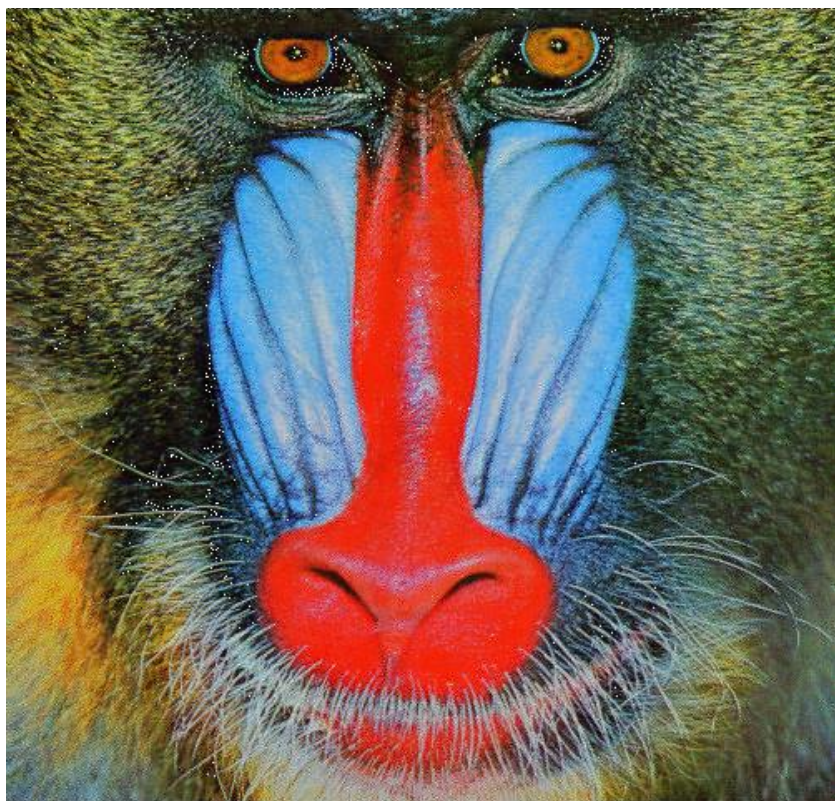


Slika 3.8.3. Omjer kompresije = 30

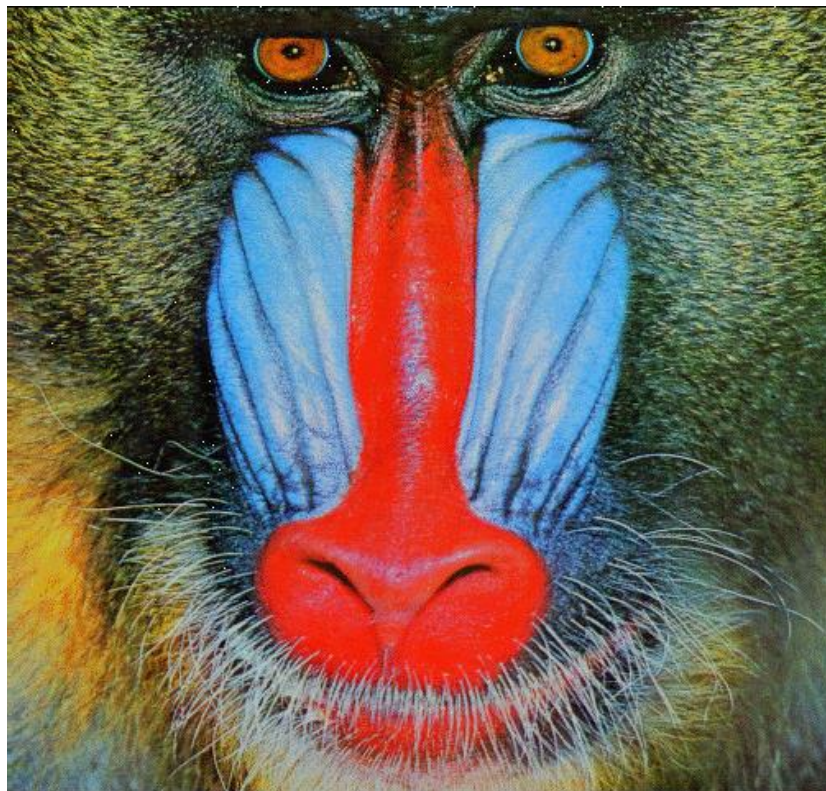


Slika 3.8.4. Omjer kompresije = 50





Slika 3.8.5. Omjer kompresije = 70



Slika 3.8.6. Omjer kompresije = 90

## 4. ZAKLJUČAK

Kompresija slike omogućuje pohranu velikih skupova slika uz malu potrošnju dostupne memorije. Odabirom omjera kompresija i/ili kvalitete slike moguće je povećati ili smanjiti kompresiju, odnosno kvalitetu dekompresirane slike. Obradom slike i signala te korištenjem osnovnih formulacija kao što su diskretna kosinus transformacija, Huffman-ovo kodiranje te matrično množenje ostvaren je efektivan algoritam kojim se uvelike štedi prostor za pohranu.

## 5. LITERATURA

- [1] <https://www.openbookproject.net/py4fun/huffman/huffman.html>
- [2] <http://pi.math.cornell.edu/~web6140/TopTenAlgorithms/JPEG.html>
- [3] <https://www.whymath.org/node/wavlets/basicjpg.html>
- [4] <https://wowelec.wordpress.com/2017/04/08/building-a-jpeg-compression-engine/>
- [5] <https://www.mathworks.com/matlabcentral/fileexchange/27078-zig-zag-scan>
- [6] <https://www.openbookproject.net/py4fun/huffman/huffman.html>



## **6. POPIS NAPISANIH PROGRAMA I UPUTE ZA KORIŠTENJE**

Napisane su dvije skripte u Python programskom jeziku: `huffman.py` i `imagCompression.py`. Program se izvodi pokretanjem `imageCompression.py` skripte unutar koje se postavlja vrijednost omjera kompresije (argument funkcije `calculateQuantizationMatrix()`).