

1. Napisati Huffmanov kod ako su date sledece verovatnoce pojavljivanja simbola A,B,C,D,E – 0,3 0,2 0,25 0,1 0,15 respektivno

A 0.3 11 B 0.2 10 C 0.25 011 D 0.1 010 E 0.15 00

2. Objasniti oznake 1080i i 720p.

- 1080i 1920x1080 interlaced scanning – Ima sliku rezolucije 1920 piksela u 1080 horizontalnih linija koje su obojene na ekranu kroz dva prolaska od 540 kroz svaku. Kvalitet slika je odlican za objekte koji se sporo kreću.
- 720p 1280x720 progressive scanning – Ima sliku rezolucije 1280 piksela u 720 linija, nije popularna zbog nize cene i boljeg kvaliteta 1080p.

3. Koje vrste korelacija medju podacima se koriste u algoritmima za kompresiju?

U algoritmima za kompresiju se koriste **prostorne, vremenske i spektralne** korelacije.

4. Razlike izmedju steganografije i watermarkinga.

Glavna razlika između steganografije i watermarkinga jeste da watermark ne mora nužno biti **nevidljiv**. Takođe, razlika je i u **primeni** – u steganografiji je primena isključivo **skrivanje** informacija, a u watermarking-u je **zaštita autorskih prava** ili dodavanje informacija vezanih za sadržaj. Takođe, **kommunikacija** je u steganografiji obično prirode jedan na jedan (tj. od pošiljaoca primaocu), a u watermarkingu jedan na više.

5. Objasniti HSV model boja

HSV model transformiše standardni RGB (Red, Green, Blue) model u **prostor** gde se boja definiše parametrima koji se nazivaju Hue, Saturation i Value (ili Intensity).

Hue se najčešće naziva „aktuelna” boja objekta

Saturation je mera čistoće boje. Ako Hue kaže da je boja zelena, Saturation će reći koliko je zapravo zelena

Intensity parametar definiše koliko je boja svetla ili tamna.

6. Navesti primer za konvolucioni filter za nalaženje ivica.

Edge detected horizontal - Može se svesti na konvolucioni filter prikazan sledećom matricom uz Offset = 0 i Factor = 1

-1	-1	-1	-1	-1	-1	-1
0	0	0	0	0	0	0
1	1	1	1	1	1	1

Crveni bit predstavlja **centar konvolucije** u odnosu na koga se konvolucija računa. Zeleni bit predstavlja mesto gde se **izračunati bajt** umeće u rezultatnoj slici.

Edge detected vertical - Može se svesti na konvolucioni filter prikazan sledećom matricom uz Offset = 0 i Factor = 1

Crveni bit predstavlja **centar konvolucije** u odnosu na koga se konvolucija računa i mesto gde se **izračunati bajt** umeće u rezultatnoj slici.

-1	0	1
-1	0	1
-1	0	1
-1	0	1
-1	0	1
-1	0	1

Edge detected homogeneity

Koristi matricu 3×3. Treba nam jedna prosleđena vrednost "threshold". Krećemo se kroz sliku i izdvajamo podmatrice 3×3. Izdvojimo centralni piksel i označimo ga sa S. Za svaki elemenat izdvojene matrice izračunajmo $\text{abs}(p[x,y] - S)$. Nađimo **maksimum** od svih izračunatih razlika i označimo ga sa m. Ako je $m < \text{threshold}$ onda je $m=0$. Centralni piksel dobije novu vrednost m.

Edge detected Difference

A	B	C
D	E	F
G	H	I

Koristi matricu 3×3. Izračunaju se sledeći izrazi $\text{abs}(C-G)$, $\text{abs}(I-A)$, $\text{abs}(H-B)$, $\text{abs}(F-D)$, i nađe se njihov **maksimum** (označimo ga sa m). Ako je $m < \text{threshold}$ onda je $m=0$. Vrednost piksela E se zameni izračunatom vrednošću m.

Edge Enhance

A	B	C
D	E	F
G	H	I

Koristi matricu 3×3. Izračunaju se sledeći izrazi $\text{abs}(C-G)$, $\text{abs}(I-A)$, $\text{abs}(H-B)$, $\text{abs}(F-D)$, i nađe se njihov **maksimum** (označimo ga sa m).

if (**nPixelMax** > nThreshold && **nPixelMax** > E) m = Max(E, nPixelMax);

Vrednost piksela E se zameni izračunatom vrednošću m.

7. Šta su False Color Images?

Slike sa „lošom“ paletom, ili lošim rasporedom boja. Najčešće se smatraju kao „artistic“ filteri. Najjednostavniji način za njihovo dobijanje je sledeći: Uzeti jednu sliku sa definisanom paletom (8-bitni BMP) i primeniti neki filter nad bajtovima slike.

8. Kojim redom se podaci nalaze u data delu WAV fajla?

- Od pozicije 36, 4 bajta: Subchunk2ID slova "data"
- Od 40, 4 bajta: Subchunk2Size = NumSamples * NumChannels * BitsPerSample/8. Ovo je ukupan broj bajtova u Data segmentu
- Od 44, pa do kraja: podaci koji opisuju zvukove

9. LZW algoritam za dekodiranje

```

1  Inicijalizacija tabele podstringova pojedinačnim karakterima
2  OLD = prvi ulazni kod
3  output translacije za OLD (tj S(OLD) )
4  WHILE not end of ulazni strim
5      NEW = sledeći ulazni kod
6      IF se NEW ne nalazi u tabeli podstringova
7          S = translacija od OLD (tj S(OLD) )
8          S = S + C (prvi karakter od S)
9          Output S
10         Upiši S u tabelu podstringova
9      ELSE
10         S = translacija od NEW
11         output S
12         C = prvi karakter od S
13         S(OLD) + C upiši u tabelu podstringova
14     OLD = NEW
15 END WHILE
```

LZW algoritam - kodiranje

```

1  Inicijalizuj kodnu tabelu pojedinačnim karakterima stringa
2  P = prvi ulazni karakter
3  WHILE not end of ulaznog strima
4      C = sledeći ulazni karakter
5      IF se P + C nalazi u tabeli podstringova
6          P = P + C
7      ELSE
8          output kod za P
9          dodaj P + C u tabelu podstringova
10         P = C
11     END WHILE
12     output kod za P
```

10. Objasniti watermarking i navesti osobine koje mora da poseduje

Watermarking je **ubacivanje informacija** u neki sadržaj, sa osnovnim zadatkom da mora biti **otporan**, tj. da se iz sadržaja ne može lako ukloniti. Šta je „lako“ i kako ga ukloniti, zavisi i od funkcije watermarka i od samog sadržaja.

Osobine: Nevidljiv, Kapacitet, Mala verovatnoća greške, Robustnost

11. Karakteristike PAL i NTSC standarda

PAL

- 625 linija se **ponavlja** 25 puta u sekundi
- **Interlaced scan** linije **dele frejm** u 2 polja od po 312.5 linija
- 20% više linija nego NTSC
- NTSC vs. PAL približno isti **bandwidth**

NTSC

- 525 scan linije **ponavljaju** se 29.97 puta u sekundi
- **Interlaced scan** linije **dele frame** u 2 polja, svako 262.5 linija
- 20 linija je rezervisano kao **kontrolna informacija**
- Samo 483 linije su **vidljivi podaci**

12. Ukratko opisati strukturu BMP fajla

BMP zaglavlje - osnovne info, **DIB zaglavlje** - detaljne info o slici, **Paleta boja** - lista boja koja se koristi, **Bitmapa** - podaci o slici, pixel po pixel, red po red

13. Navesti razliku izmedju konvolucionih i displacement filtera za slike

Konvolucionni filteri: Krecemo se kroz sliku i izdvajamo podslike iste dimenzije kao konvolucionna matrica. Mnozimo te dve matrice i sabiramo medjusobno, tako da dobijamo konvolucionu vrednost $k=(S/Factor)+Offset$, ova vrednost menja **centralni bajt** u rezultujucoj slici.

Displacement filteri: Ulaz u svaki displacement filter je slika predstavljena kao **bmp objekat** i matrica koja definise **offset**. Matrica treba da bude saglasna dimenzijama sa rezolucijom slike. Za svaki piksel se preračunavaju **komponente boja**.

14. Navesti glavne korake u kreiranju JPEG slike

Iako JPEG fajl može biti kreiran na različite načine, najkorišćeniji je tzv. "JFIF encoding". Proces:

- **Transformacija** modela boja iz **RGB** u **YCbCr**
- Broj boja se **redukuje** (50% boja se eliminiše). Ovo je moguće, ali nije neophodno.
- Slika se **podeli** na blokove od po 8×8 piksela. U svakom bloku se primeni **DCT** transformacija nad svakom od Y, Cb i Cr komponenti.
- Visoke frekvencije se **kvantizuju** manje precizno nego niske
- Rezultatni 8×8 blok se dalje **kompresuje** nekim metodom za kompresiju bez gubitaka

15. Navesti razlike izmedju CMYK i RGB modela boja

CMYK - Cyan, Magenta, Yellow i Key (Black). Koristi se u **štamarskoj industriji**, substraktivni model, boje se dobijaju oduzimanjem od bele

RGB - Aditivni model, red, green i blue. RGB model je danas najkorišćeniji model boja za **prikaz slika na TV-u i računarima**. Boje se dobijaju dodavanjem R, G i B komponenti crnoj boji.

16. Navesti vrednosti R, G i B komponente koje daju belu i crnu boju.

Kada sve tri komponente imaju maksimalnu vrednost (255), rezultat je bela boja, a kada sve tri imaju vrednost 0, rezultat je crna boja.

17. Robusnost watermarking-a

Savršeni watermark sistem morao bi podneti svako **modifikovanje** sadržaja u standardnim (npr. izoštravanje slike) ili zlonamernim (uništavanje watermarka) procesima. Neki od **napada** na koje watermark mora biti otporan su npr: Poboljšanje signala, Aditivni ili multiplikativni šum, Linearno filtriranje, Nelinearno filtriranje, Kompresije s gubitkom, Geometrijske transformacije, Redukcija i kompozicija podataka, D/A i A/D konverzija, Višestruki watermark, Mozaični napad

18. Napisati kod za konverziju boje predstavljene RGB modelom u odg. nijansu sive.

blue = p[0]; green = p[1]; red = p[2];

p[0] = p[1] = p[2] = (redR * red + greenR * green + blueR * blue);

Najbolji rezultat se dobije kada važi uslov $\text{redR} + \text{greenR} + \text{blueR} = 1.0$

Heuristički dobijeni koeficijenti koji daju najbolji utisak: $\text{redR}=0.299$, $\text{greenR}=0.587$, $\text{blueR}=0.114$

19. Koji je glavni nedostatak GIF formata za prikaz slika?

Originalni GIF je ograničen na osmobarbnu paletu, što znači 256 boja. Pogodan je za cuvanje slika sa relativno **malo boja** i sa relativno **jednostavnim oblicima**. Koristi „**lossless**” **kompresiju** koja je pogodna za slike koje imaju velike površine iste boje. Manje je efektna kod slika sa visokim nivoom detalja.

20. Opisati nacin funkcionisanja resize operacije.

Postoji **linear** i **bilinear** resize operacija. U oba slucaja su nove dimenzije slike: nWidth, nHeight. Stara slika je objekat Bitmap **bTemp**. Kreira se nova slika kao $b = \text{new Bitmap}(nWidth, nHeight, bTemp.PixelFormat)$. Izračunaju se **resize faktori**: $nXFactor = bTemp.Width/nWidth$; $nYFactor = bTemp.Height/nHeight$; Svaki **piksel** u odredišnoj slici se računa kao:
 $b.SetPixel(x, y, bTemp.GetPixel(x * nXFactor, y * nYFactor));$

21. Navesti glavne delove WAV fajla.

Glavni delovi WAV fajla su **RIFF zaglavlje** (ChunkID, ChunkSize, Format), **FMT** (subchunk 1- Subchunk1ID, Subchunk1Size, AudioFormat, NumChannels, SampleRate, ByteRate, BlockAlign, BitsPerSample) i **Data** (subchunk 2 – Subchunk2ID, Subchunk2Size, data).

22. Kolika je složenost LZW kodiranja i dekodiranja?

$O(N)$ tj. $O(M)$ gde je N dužina **ulaznog niza** a M dužina **koda**

23. Opisati aritmeticku kompresiju

- Celu poruku predstaviti **realnim brojem** iz intervala $[0,1)$
- Interval $(0, 1)$ se sukcesivno **deli na podintervale** u skladu sa frekvencom pojavljivanja **sledećeg simbola**.
- Svaki novi podinterval predstavlja jedan **simbol**.
- Po završetku procesa **najmanji realan broj** sadržan u **odredišnom intervalu** se bira kao **kod** date poruke

24. Sta uključuje jedan multimedijalni sistem?

Generisanje, predstavljanje, skladištenje, prenos, pretraživanje, izdvajanje i isporuku multimedijalnih informacija.

25. Poredjenje algoritama za kompresiju (LZ, Word Huffman, Character Huffman, Aritmeticki)

	Aritmetička	Character Huffman	Word Huffman	LZ
Stepen kompresije	Vrlo dobar	Loš	Vrlo dobar	Dobar
Brzina kompresije	Sporo	Brzo	Brzo	Vrlo brzo
Brzina dekompresije	Sporo	Brzo	Vrlo brzo	Vrlo brzo
Potrebna memorija	Mala	Mala	Velika	Osrednja
Pattern matching	Ne	Da	Da	Da
Random Access	Ne	Da	Da	Ne

26. Opisati funkcionisanje Contrast osnovnog filtera

- Ideja kontrasta je da tamnije tonove na slici učini tamnijim, a svetlije svetlijim
- Najčešće se zadaje kao **normalizovana celobrojna vrednost** u rasponu -100 do 100 ($nContrast$)
 - $contrast = (100.0 + nContrast) / 100.0;$
 - $contrast *= contrast;$
- Rezultat je broj iz opsega od 0.0 do 4.0
- Izvorna vrednost **pixelSrc** se iz opsega 0 do 255 prevede u opseg -0.5 do 0.5:
 - $pixel = pixelSrc / 255.0;$
 - $pixel -= 0.5;$
- Zatim se pomnoži dobijenom vrednošću **contrast** i zatim se vrati u interval 0 do 255
 - $pixel *= contrast;$
 - $pixel += 0.5;$
 - $pixel *= 255;$

27. Navesti transformacionu matricu jednog konvolucionog filtera za detektovanje ivica.
(isto kao sesto pitanje, drugi primeri)

- Emboss Laplacian

– Offset: 127

– Factor: 1

-1	0	-1
0	4	0
-1	0	-1

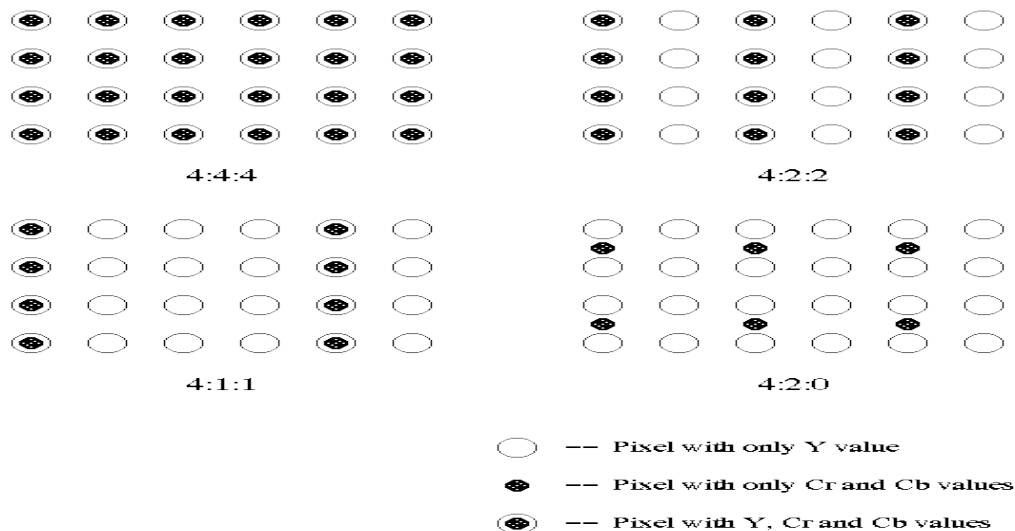
- Edge detect quick

– Offset: 0

– Factor: 1

-1	-1	-1
0	0	0
1	1	1

28. Nacrtati 4:2:2 downsampling matricu



29. Razlika vektorske i rasterske grafike

Vektorska grafika se zasniva na principu **vektora**. Svaki vektor ima svoj smer, pocetnu i krajnju tacku. Vektorsku sliku mozemo beskonacno povecavati i smanjivati bez gubitka kvaliteta zato sto se bazira na matematickim funkcijama.

Rasterska grafika se zasniva na **pikselu**. Raster je matrica piksela. Svaki od tih piksela moze prikazati samo jednu boju odredjene svetline. Povecavanjem grafike dolazi do gubitka kvaliteta, pikseli ce se videti golim okom.

30. Napisati jednu funkciju za Color i Brightness filter.

```
Void foo(double r, double g, double b, bool color)
{
    If(color){
        For(int i=0;i<img.Width;i++)
            { for(int j=0;j<img.Height;j++){
                var pixel = img.getPixel(i,j);
                img.setPixel(i,j, Color.FromArgb(pixel.R + r, pixel.G + g, pixel.B + b));}}
    }
    else
    {
        Var col = r+g+b;
        col=col/3;
        Random r = new Random();
        Var rand = r.Next();
        If(rand%2==0) col=col*(-1);
        For(int i=0;i<img.Width;i++)
        {
            for(int j=0;j<img.Height;j++)
            {
                var pixel = img.getPixel(i,j);
                img.setPixel(i,j, Color.FromArgb(pixel.R + col, pixel.G + col, pixel.B + col);
            }
        }
    }
}
```

31. Koje osobine mora da ima Watermark?

Neprimetnost - bitno je odrediti koliko watermark unosi distorzije u sistem

Autorska prava - potrebno je osigurati mogucnost detekcije redosleda ubacivanja watermarka

Robusnost - savrseni watermark morao bi podneti svako modifikovanje standardnim ili zlonamernim procesima

Sigurnost watermarka i kljucevi - zastitita od neovlascenog koriscenja ili detektovanja, dva osnovna nivoa sigurnosti:

i. **visok nivo** - izabrani mehanizam mora osiguravati da neovlasцени korisnik ne moze procitati informaciju koja je ubacena watermarkom I ne moze ni detektovati watermark

ii. **nizak nivo** - izabrani mehanizam mora osiguravati da neovlasцени korisnik ne moze procitati informaciju bez posedovanja tajnog kljuka, ali watermark moze detektovati

32. Navesti vrste video signala

Komponentni video signal – svaka od komponenti se salje kao odvojeni video signal. One mogu biti RGB ili neka njihova transformacija (YIQ, YUV). Najbolja reprezentacija boja zahteva veci protok I dobru sinhronizaciju te tri komponente.

Kompozitni video signal – boja I svetlina signala su pomesane u jedan prenosnik talasa. U ovom slucaju je neizbezna interferencija izmedju dva signala.

Odvojeni video signal – kompromis izmedju komponentnog I kompozitnog video signala. Koristi dve linije, jednu za svetlinu a drugu za hromaticnost.

33. Ukratko opisati proces konkatencije dva WAV fajla sa potpuno istim sadržajem

1. SubChunk2Size dobija veličinu SubChunk2Size * 2 (Ovde SubChunk2Size dobija veličinu duplo veću)
2. ChunkSize se ažurira tako što se postavi novi koji ima vrednost prethodnog + SubChunk2Size (jer su oba fajla iste dužine)
3. Data deo posle 40. bajta se samo duplira.

34. Objasniti multipoint videokonferenciju I opisati strukturu MCU

Koristeci MCU softver, IP, DSL i mobilnu konekciju, videokonferencijski sistem moze da bude konfigurisan da komunicira sa vise udaljenih sajtova istovremeno. On moze da se poveze sa vise drugih videokonferencijskih sistema da bi bilo omoguceno komuniciranje sa vecim brojem sajtova, pruzajuci bolji protok po sajtu na koji je povezan. Za bolju prilagodljivost, funkcije koje omogucavaju brze povezivanje omogucavaju da, na primer, IP mreze koje imaju vecu brzinu budu povezane sa dosta sporijom ISDN mrežom preko multipoint konferencije bez smanjivanja brzine na brzinu sporije mreze. Sa podrskom „MCU on the fly” korisnik moze da otpocne peer-to-peer videokonferenciju I prosiri je na multipoint konferenciju bilo kada, samo pozivanjem odgovarajucih kontakata.

35. Ako su frekvencije pojavljivanja simbola A,B,C i D redom 0.4, 0.3, 0.2, 0.1 odrediti aritmeticki kod ulaznog stringa CBAD.

A B C D 4:3:2:1
C (0-0.4-0.7-0.9-1)
B (0.7-0.78-0.84-0.88-0.9)
A (0.78-0.804-0.822-0.834-0.84)
D (0.78-0.7896-0.7968-0.8016-0.804)

36. Odrediti LZ kod za ulazni niz XXXYXYXXYYXXYXX

1.X 2.Y 3.YX 4.YXY 5.YY 6.XX 7.YXX
<0,0,X><0,0,Y><2,1,X><3,2,Y><2,1,Y><1,1,X><3,2,X>

37. Odrediti RL kod za ulazni niz XXXXXXZZZZZZZZXXYXYYYYYZZZZXYYYZZZXXX

6X7Z3X1Y1X5Y4Z1X3Y3Z3X

38. Prikazati sadržaj kodne tabele prilikom LZW kompresije niza simbola ABBABAAABAABBAAA

code representing | codeword string

65	A	256	AB
66	B	257	BB
66	B	258	BA
256	AB	259	ABA
65	A	260	AA
260	AA	261	AAB
258	BA	262	BAA
256	AB	263	ABB
262	BAA	264	BAAA
65	A		