

1. feladat Adjuk meg a 720 soros, valamint az 1080 soros HD videó formátum következő paramétereit 25 Hz-es progresszív képtartalom esetére: összes képsor száma, aktív képsorok száma, soronkénti összes pixel száma, soronkénti aktív pixelek száma, aktív képtartalom bitsebesség igénye 4:2:2 színkülönbségi jel alulmintavételezés, és 10 bites komponens ábrázolás esetén.

1. feladat, megoldás A HD mintavételi frekvencia 74.25 MHz.

	L_T	L_A	S_T	S_A	Mbit/s (4:2:2)
720p25	750	720	$74.25 \times 10^6 / 750 / 25 = 3960$	1280	$((720 \times 1280 + 2 \times 720 \times 1280 / 2) \times 25 \times 10) / 10^6 = 460.8$
1080p25	1125	1080	$74.25 \times 10^6 / 1125 / 25 = 2640$	1920	$((1080 \times 1920 + 2 \times 1080 \times 1920 / 2) \times 25 \times 10) / 10^6 = 1036.8$

2. feladat Írjuk fel az

$$\mathbf{X} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

2x2-es ($N = 2$) képblokk 2D DCT transzformáltját, ha az 1D transzformáció mátrixa a

$$\mathbf{A} = A[k, m] = \sqrt{\frac{2}{N}} \alpha(k) \cos \left[\frac{\pi}{N} \left(m + \frac{1}{2} \right) k \right],$$

ahol

$$\alpha(k) = \begin{cases} \frac{1}{\sqrt{2}} & \text{ha } k \equiv 0 \\ 1 & \text{ha } k \neq 0. \end{cases}$$

kifejezéssel adott, és k a $0 \dots 1$ együtthatók indexe, m a $0 \dots 1$ mintaindex.

2. feladat, megoldás Az 1D DCT az $\mathbf{y} = \mathbf{A}\mathbf{x}$ kifejezéssel számítható, ahol $N = 2$ esetére

$$\mathbf{A} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}$$

Ha \mathbf{X} oszlopai alkotják az \mathbf{x}_i oszlopvektorokat, akkor \mathbf{X} oszlopai szerinti transzformáltat \mathbf{Y}_c -vel jelölve: $\mathbf{Y}_c = \mathbf{A}\mathbf{X}$. Az 1D transzformációt ezután \mathbf{Y}_c transzponáltjára, \mathbf{Y}_c^T -re elvégezve $\mathbf{Y}_{r,c} = \mathbf{A}\mathbf{Y}_c^T$. A végeredmény $\mathbf{Y} = \mathbf{Y}_{r,c}^T$

$$\mathbf{Y} = \begin{bmatrix} 5 & -1 \\ -2 & 0 \end{bmatrix}$$

vagy a transzformált közvetlenül is számítható a $\mathbf{Y} = \mathbf{A}\mathbf{X}\mathbf{A}^T$ segítségével, mely szintén a fenti eredményre vezet.

3. feladat Hasonlítsuk össze az MPEG-2 és H.264/AVC tömörítési eljárásokat, ismertessük a lényegesebb különbségeket a kódolási eszközök, illetve azok tulajdonságainak szempontjából.

3. feladat, megoldás

- A MB-ok 16×16 -os méretűek, azon belül a mozgáskompensáció szub-MB-en történik, melyek 16×16 , 16×8 , 8×16 , 8×8 , 4×8 , 4×8 és 4×4 méretűek lehetnek
- A predikciós hiba kódolása 4×4 -es vagy 8×8 -as (FRExt) blokkokon történik
- Integer transzformáció a DCT helyett
- Alapvetően kétfajta képtípus: referencia és nem-referencia kép (predikció szempontjából)
- B kép is lehet referenciakép
- Több referenciakép használata (nem csak az adott képet közvetlenül megelőző, illetve közvetlenül követő referenciaképek lehetnek időbeli prediktorok, hanem akár 16 képpel korábbi/későbbi képek is)
- Képeken belül I,P,B típusú képszeletek

- Intra predikció I típusú MB esetében, többféle predikciós irány lehetséges
- Időbeli súlyozott predikció lehetősége
- Különböző méretű szub-MB alapú mozgáskompenzáció
- Blokkosodást csökkentő szűrés: jelentősen csökkenti a predikciós hiba maradványait, a blokk határokon megjelenő élek simításával.
- FReXt kiegészítések
 - 8 bit/mintánál nagyobb bitmélységek támogatása
 - 4:2:2 és 4:4:4 mintastruktúrák támogatása
 - Veszteségmentes tömörítés támogatása (lináris PCM, vagy transzformációs kódolás nélküli entrópiakódolás)
 - A szintér transzformációk numerikus hibáinak elkerülése (YCgCo szintér)

4. feladat Példa: Számítsuk ki az 1080 soros HD formátumhoz tartozó ideális vízszintes és függőleges képméretet (cm-ben, vagy méterben kifejezve), 3 méteres nézőtávolság esetére.

4. feladat, megoldás Jelöljük a kép magasságát pixelben: N_V -vel (V: vertical), a kép szélességet pixelben N_H -vel (H: horizontal), a szomszédos pixelekből a szemünkbe érkező fénysugarak által bezárt szöget Φ -vel. A méterben mért nézőtávolságot jelöljük D -vel, míg a méterben mért képmagasságot H -val (Height), és a méterben mért képszélességet W -vel (Width). Ideális nézőtávolság esetén fennáll, hogy $\tan(\frac{1}{60}) = \frac{H/N_V}{D}$ (a kép közepét merőlegesen nézve a kép közepén lévő két szomszédos képsorból a szemünkbe érkező fénysugarak által bezárt szög megegyezik a látás szögfelbontásának alsó határával.) Ebből: A képmagasság: $H = DN_V \tan \frac{1}{60}$, a képszélesség: $W = H \times \frac{A}{B}$, ahol $\frac{A}{B}$ a képarány (SD: 4:3, HD: 16:9). Ebből az 1080 soros HD rendszerre: $H=0,94$ m, $W=1,68$ m.

5. feladat Adott a következő 4×4 -es képblokk, és környezete:

42	58	58	50	50
58	58	64	51	58
60	52	64	56	66
63	62	63	61	64
61	59	51	63	69

Állapítsuk meg, hogy QP=6 - os kvantálási paraméter esetén a Horizontal, Vertical és DC intra-predikciós módok melyike eredményezheti várhatóan a legkisebb bitigényt transzformáció és újrakvantálás után (az entrópiakódolás hatásától tekintünk el)

A transzformáció mátrixa

$$\mathbf{T} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$$

míg a QP=6 kvantálási paraméterhez tartozó ortonormál (skalázó) és kvantáló mátrix a következő:

$$\mathbf{V}(QP = 6) = \begin{bmatrix} 6554 & 4033 & 6554 & 4033 \\ 4033 & 2622 & 4033 & 2622 \\ 6554 & 4033 & 6554 & 4033 \\ 4033 & 2622 & 4033 & 2622 \end{bmatrix}$$

5. feladat, megoldás Első lépésben számítsuk ki a vertikális predikció hibablokkját. A blokk feletti 4 pixel értékét függőlegesen a blokk minden pixelére függőlegesen kiterjesztve (extrapolálva), és a predikciós hibát kiszámítva a következő adódik:

$$\mathbf{R}_V = \begin{bmatrix} 58 & 64 & 51 & 58 \\ 52 & 64 & 56 & 66 \\ 62 & 63 & 61 & 64 \\ 59 & 51 & 63 & 69 \end{bmatrix} - \begin{bmatrix} 58 & 58 & 50 & 50 \\ 58 & 58 & 50 & 50 \\ 58 & 58 & 50 & 50 \\ 58 & 58 & 50 & 50 \end{bmatrix} = \begin{bmatrix} 0 & 6 & 1 & 8 \\ -6 & 6 & 6 & 16 \\ 4 & 5 & 11 & 14 \\ 1 & -7 & 13 & 19 \end{bmatrix}$$

Képezzük az intra-predikciós hibablokk transzformáltját a $\mathbf{TR}_V \mathbf{T}^T$ kifejezéssel, majd ezt a szorzatot elemenként szorozzuk össze a skálázó mátrixszal (\mathbf{V}). A kapott eredményt el kell osztani 2^{15} -el, majd a kerekítés után megmaradó újrakvantált együttható mátrix:

$$\mathbf{Y}_{R_V} = \frac{1}{2^{15}} [(\mathbf{TR}_V \mathbf{T}^T) * \mathbf{V}] = \begin{bmatrix} 19 & -17 & 3 & -2 \\ -4 & 6 & -4 & -8 \\ -3 & 0 & 3 & 3 \\ 2 & 6 & -1 & 1 \end{bmatrix}$$

Ezután számítsuk ki a horizontális predikció hibablokkját. A bloktól balra található 4 pixel értékét vízszintesen a blokk minden pixelére vízszintesen kiterjesztve (extrapolálva), és a predikciós hibát kiszámítva a következő adódik:

$$\mathbf{R}_H = \begin{bmatrix} 58 & 64 & 51 & 58 \\ 52 & 64 & 56 & 66 \\ 62 & 63 & 61 & 64 \\ 59 & 51 & 63 & 69 \end{bmatrix} - \begin{bmatrix} 58 & 58 & 58 & 58 \\ 60 & 60 & 60 & 60 \\ 63 & 63 & 63 & 63 \\ 61 & 61 & 61 & 61 \end{bmatrix} = \begin{bmatrix} 0 & 6 & -7 & 0 \\ -8 & 4 & -4 & 6 \\ -1 & 0 & -2 & 1 \\ -2 & -10 & 2 & 8 \end{bmatrix}$$

Hasonlóan az előző esethez:

$$\mathbf{Y}_{R_H} = \frac{1}{2^{15}} [(\mathbf{TR}_H \mathbf{T}^T) * \mathbf{V}] = \begin{bmatrix} -1 & -5 & 3 & -6 \\ 0 & 6 & -4 & -8 \\ 0 & 0 & 3 & 3 \\ 0 & 6 & -1 & 1 \end{bmatrix}$$

A blokk átlaga kerekítve 60, így a blokk és a DC átlag eltérése a DC mód predikciós hibája:

$$\mathbf{R}_{DC} = \begin{bmatrix} 58 & 64 & 51 & 58 \\ 52 & 64 & 56 & 66 \\ 62 & 63 & 61 & 64 \\ 59 & 51 & 63 & 69 \end{bmatrix} - \begin{bmatrix} 60 & 60 & 60 & 60 \\ 60 & 60 & 60 & 60 \\ 60 & 60 & 60 & 60 \\ 60 & 60 & 60 & 60 \end{bmatrix} = \begin{bmatrix} -2 & 4 & -9 & -2 \\ -8 & 4 & -4 & 6 \\ 2 & 3 & 1 & 4 \\ -1 & -9 & 3 & 9 \end{bmatrix}$$

Hasonlóan a korábbiakhoz

$$\mathbf{Y}_{R_{DC}} = \frac{1}{2^{15}} [(\mathbf{TR}_{DC} \mathbf{T}^T) * \mathbf{V}] = \begin{bmatrix} 0 & -5 & 3 & -6 \\ -4 & 6 & -4 & -8 \\ -3 & 0 & 3 & 3 \\ 2 & 6 & -1 & 1 \end{bmatrix}$$

Látható, hogy a horizontális intra predikcióhoz tartozó újrakvantált együtthatómátrix (\mathbf{Y}_{R_H}) tartalmazza a legtöbb nulla együtthatót, így nagy valószínűséggel a horizontális predikcióhoz tartozó hibablokk kódolásának bitigénye lesz a legkisebb.

6. feladat Írjuk fel a 100 % intenzitású színsávakra színeihez tartozó YCbCr komponenseket ITU-601-es Y együtthatók ($Y_{601} = 0.299 R + 0.587 G + 0.114 B$), ITU-709-nek megfelelő gamma-korrekció

$$V' = \begin{cases} 4.500V & V < 0.018 \\ 1.099V^{0.45} - 0.099 & V \geq 0.018 \end{cases}$$

és 10 bites stúdió kódtartományú komponens ábrázolás esetére.

6. feladat, megoldás Mivel $Y' = 0.299 R' + 0.587 G' + 0.114 B'$, és a 100% intenzitású színsávakra minden RGB értéke 0, vagy 1 értékű, ezért $V'_0 = 4.500 \times 0 = 0$, illetve $V'_1 = 1.099 \times 1^{0.45} - 0.099 = 1$.

Az $R' - Y'$, és $B' - Y'$ értékek a $Y'_{601} = 0.299 R' + 0.587 G' + 0.114 B'$ kifejezésből közvetlenül számíthatók.

A C'_r és C'_b komponensek képzéséhez először az $R' - Y'$, és $B' - Y'$ jeleket $-0.5 \dots +0.5$ tartományba kell skálázni. Ehhez $C'_r = \frac{0.5}{0.7} (R' - Y')$ és $C'_b = \frac{0.5}{0.89} (B' - Y')$. Az $R' - Y'$, és $B' - Y'$ jelek $-0.7 \dots +0.7$, illetve $-0.89 \dots +0.89$ tartománya egyébként a színsávakra ($R=1, G=0, B=0$), illetve ($R=0, G=0, B=1$) értékei esetén látható.

A 8 bites komponensek a $Y'_{8bit} = 219 Y' + 16$, a $C'_{r8bit} = 224 C'_r + 128$ és a $C'_{b8bit} = 224 C'_b + 128$ kifejezésekből kaphatók kerekítéssel. (A 8 bites kódok kifejezése a kódtartományok ismeretében (Y:

16...235, Cb/Cr: 128-112=16 ... 128+112=240, ahol 128 a 0 Cb/Cr értékhez tartozó offset) megfelelő $f(x) = mx + b$ alakú lineáris függvény felállításával megkapható.)

A 10 bites komponenseket a 8 bites komponens értékek 4-való szorzásával kaphatjuk meg legegyszerűbben.

R'	G'	B'	Y'	$(R' - Y')$	$(B' - Y')$	C'_r	C'_b	Y'_{8bit}	C'_{r8bit}	C'_{b8bit}	Y'_{10bit}	C'_{r10bit}	C'_{b10bit}
1	1	1	1	0	0	0	0	235	128	128	940	512	512
1	1	0	0.89	0.11	-0.89	0.08	-0.50	210	146	17	840	585	66
0	1	1	0.70	-0.70	0.30	-0.50	0.17	170	16	166	678	63	663
0	1	0	0.59	-0.59	-0.59	-0.42	-0.33	144	34	54	578	137	217
1	0	1	0.41	0.59	0.59	0.42	0.33	106	222	202	426	888	807
1	0	0	0.30	0.70	-0.30	0.50	-0.17	81	240	90	326	961	362
0	0	1	0.11	-0.11	0.89	-0.08	0.50	41	110	239	164	439	958
0	0	0	0	0	0	0	0	16	128	128	64	512	512