

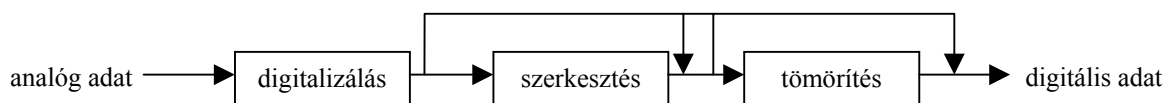
1. DIGITÁLIS ADATFELDOLGOZÁS

A médiumok szinte mindegyike előállítható már digitális formában. Ez az ún. „digitális közös nevező” lehetővé teszi az ilyen adatok egységes kezelését. Minél összetettebb egy médium, annál nagyobb tárigénnyel jelentkezik a digitalizálás. Ennek illusztrálására álljon itt néhány példa. Amíg szöveg médium esetében egy 8*8 képpontból álló raszter tárigénye 800*600-as felbontást és 2*8 bites színmélységet alapul véve:

$$\frac{800 * 600}{8 * 8} * 2 * 8 = 120.000bit = 14,6Kbyte,$$

addig egy 800*600-as felbontású, 8 bit színmélységű bittérképes képé 800*600*8 = 3.840.000 bit = 468,75 Kbyte, míg egy 800*600-as felbontású, 25 kép/s -os képváltsási frekvenciájú videojelé másodpercenként – képpontonként 3 byte (luminancia, krominancia) színmélységet feltételezve – 800*600*25*3 = 36.000.000 byte/s = 34,3 Mbyte/s! Mindezek egyértelműen bizonyítják, hogy ezeket a médiumokat célszerű tömörítve tárolni.

Digitális adatfeldolgozás alatt értjük azt a folyamatot, amikor az analóg adatokat tömörített digitális adatokká alakítjuk át. A digitális adatfeldolgozás három elég jól elkülöníthető részfolyamatból áll: az analóg adatok digitális adatokká történő átalakítása (*digitalizálás*), a digitális médiumok *szerkesztése*, *adattömörítés* (1. ábra). Ezek a részfolyamatok még további lépésekre bonthatók.



1. ábra

1.1. Digitalizálás

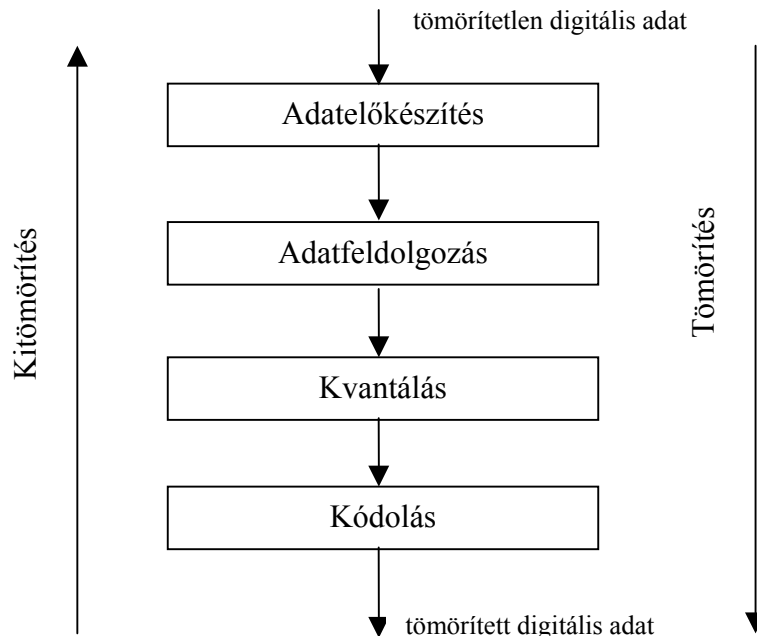
A digitalizálás célja, hogy a két- ill. háromdimenziós térben lévő objektumokról a számítógép által értelmezhető és feldolgozható adatokat állítson elő. A digitalizálás speciális eszközökkel (scanner, hangkártya, digitális fényképezőgép, digitális kamera, stb.) történik, az egyes lépések megvalósítása eszköztől függően különböző módon lehetséges. A digitalizálás folyamatát és jellegzetességeit a Multimédia hardvereszközei témakörben fogjuk tárgyalni.

1.2. Digitális médiumok szerkesztése

A digitalizált objektumok, típustól függően szerkeszthetők, amely műveletek végrehajtására a legkülönbözőbb programok állnak rendelkezésünkre. Ezen szerkesztő műveletek célja lehet az analóg-digitális átalakítás hibáinak kijavítása, az eredetitől különböző virtuális objektumok létrehozása, stb. Ezen témakörrel részben már volt szó [1], ill. még tárgyalásra fog kerülni az egyes időfüggő médiumoknál.

1.3. Tömörítés

Az adattömörítés okait – a fejezet elején – a nagy mennyiségű adattömegben jelöltem meg. A tömörítés reverzibilis folyamatát a 2. ábra mutatja, azaz kitömörítéskor a folyamat visszafelé ugyanilyen lépések sorozataként zajlik le.



2. ábra

Az *adatelőkészítés* során történik a médium, tömörítéshez szükséges előkészítése. Ez például állóképek esetében 8*8 képpontból álló egységekre bontást jelent. Ekkor határozzuk meg az egyes képpontokhoz rendelt bitek számát, ami annak minőségét is meghatározza.

Az *adatfeldolgozás* a tényleges tömörítés első fázisa. Ez lehet például egy transzformáció (pl. időtartományból a frekvenciatartományba – DCT – direkt koszinuszos transzformáció), amely elősegíti a tömörítés könnyebb végrehajtását.

A *kvantálás* az adatfeldolgozás során nyer együttthatók megadott szabályok szerinti kezelését, pl. súlyozását teszi lehetővé (JPEG eljárás).

A *kódolás* az előzőekben előállított diszkrét adatfolyam veszteségmentes tömörítését jelenti. Az adatfeldolgozás és a kvantálás során ún. veszteséges, míg a kódolás során ún. veszteségmentes tömörítési algoritmusokat alkalmaznak.

Veszteségmentesnek nevezzük a *tömörítést* (entrópiakódolás), amennyiben a tömörítés előtti ún. kódolandó adat megegyezik a kitömörítés után keletkezett ún. dekódolt adattal. Azt az eljárást pedig, amelynél a tömörítés során adatvesztés áll fenn, *veszteséges tömörítésnek* (forráskódolás) nevezzük. Az entrópiakódolás médiumfüggetlen, míg a forráskódolás médiumfüggő tömörítési eljárás. A médiumfüggőség azt jelenti, hogy a tömörítési algoritmusban kihasználják az adott médiumban rejlő sajátosságokat, például azt, hogy az álló képeknél a 2D DCT alapú kódolásnál az alacsony frekvenciák fontosabbak, mint a magas frekvenciák. A multimédia rendszerekben az ún. hibrid kódolást alkalmazzák, amely egy sajátos kombinációja a forrás- és entrópiakódolásnak. Mielőtt rátérnék az egyes kódolási eljárások bemutatására, vizsgáljuk meg röviden az adattömörítéssel szemben támasztott követelményeket:

- minél kisebb tömörítési aránytényező $[(\text{tömörített adatmennyiség}/\text{tömörítetlen adatmennyiség})]$, amely a tömörítési folyamat hatékonyságát jelöli,
- a tömörítési algoritmus nem jelenthet jelentős mértékű adatvesztést, amely már az emberi érzékelés minőségét lerontaná,
- a tömörítés nem tartalmazhat redundáns adatot. Azt az adatot nevezzük redundánsnak, amely nem a lehető legkisebb mennyiségben tárolja ugyanazt az információt. Redundáns lehet például az a kétszínű képi adat, amely csak fekete és fehér színt tartalmaz és 1 bit helyett 1 byte-on kódoljuk. $0 \rightarrow \text{fekete}$ és $1 \rightarrow \text{fehér}$ helyett $0 \rightarrow \text{fekete}$ és $255 \rightarrow \text{fehér}$, közben az 1-254 közötti tartomány kihasználatlan.
- általános érvényű algoritmus, amely a legszélesebb körű felhasználói alkalmazást lehetővé teszi. Így biztosított például, hogy az Internetes és multimédiás környezetben a legelterjedtebb képtömörítési eljárás a GIF, JPEG; mozgókép-tömörítési eljárás az AVI, MPEG; míg audiotömörítési eljárás a MP3.

A 3. ábra összefoglalva mutatja a legfontosabb tömörítési eljárásokat. [2]

Entrópiakódolás	Változó hosszúságú kódolás (VLC – Variable Length Coding)	Huffman kódolás
		Aritmetikai kódolás
	Bit-sík kódolás	
	Előrebecsléses (prediktív) kódolás	Veszteségmentes előrebecsléses kódolás
Forráskódolás	Veszteséges előrebecsléses kódolás	DPCM
		ADPCM
	Transzformációs kódolás	FFT (Gyors Fourier transzformáció)
		DCT (Diszkrét függvények koszinuszos Fourier transzformációja)
	Fontosság szerinti kódolás	Alulmintavételezéses kódolás
Hibridkódolás		Subband kódolás
	JPEG (Joint Photographic Experts Group)	
	MPEG (Moving Picture Expert Group)	

3. ábra

1.3.1. Veszteségmentes tömörítési eljárások

Állóképek tömörítésénél célszerűbb a képpont értékekhez rendelt kódolás helyett a képpont tulajdonságokban bekövetkezett változásokat lekódolni. A szomszéd képpontok luminanciája és krominanciája nagy valószínűséggel megegyezik, vagy kevésbé tér el egymástól. A ritkábban előforduló (nagy változások a luminanciában ill. a krominanciában) értékekhez hosszabb kódot, a gyakrabban előforduló (kis változások a luminanciában, krominanciában) értékekhez rövidebb kódot rendelünk hozzá. Ezt az eljárást *változó hosszúságú kódolásnak* nevezzük. Tovább növelhetjük az eljárás hatékonyságát, amennyiben először meghatározzuk a változás-értékek statisztikus eloszlását, majd a változó hosszúságú kódokat az eloszlás ismeretében rendeljük hozzá a változásokhoz. A változó hosszúságú kódolásnak esetét használják: Huffman kódolás, aritmetikai kódolás.

A Huffman (D. A. Huffman, 1952) kódolás szerint legyen k -féle bemenő, ismert előfordulási valószínűségű adatunk, majd minden egyes adathoz olyan bithosszúságú kódot rendeljünk hozzá, amelynek hossza fordítottan arányos annak előfordulási gyakoriságával. Ismerünk statikusan és adaptíven felépített Huffman kódtáblát. A statikusan felépülő kódtábla független az adott képtől, nagyobb mennyiségű kép elemzése által született kódtábla. Az adaptív kódtábla az adott kép hisztogramja alapján készül. Következik ebből, hogy először elemezni kell a képet (a hisztogram előállítás), majd csak az után következik a tömörítési algoritmus. Hátránya a lassúság, előnye a kis tömörítési aránytényező.

Az aritmetikai kódolásnál figyelembe vesszük az adatot megelőző adatot, vagy adatokat és a változást kódoljuk. Az így előállított kódolt adatfolyam nem diszkrét, elkülöníthető adatok sokasága, hanem csak előlről haladva, szekvenciálisan fejthető vissza tömörítetlen adatokká. Az aritmetikai és Huffman kódolást előszeretettel alkalmazzák a JPEG tömörítésnél is.

A *bit-sík kódolás*nál a képet képsíkokra bontjuk. Annyi képsíkot hozunk létre ahány bitek a képpontok színmélységei ($1 \text{ bit} = 2 \text{ szín} \rightarrow 1 \text{ bitsík}$; $4 \text{ bit} = 16 \text{ szín} \rightarrow 4 \text{ bitsík}$; $8 \text{ bit} = 256 \text{ szín} \rightarrow 8 \text{ bitsík}$; stb.). Egy kép tehát több (1, 2, 4, 8, 16, 24) egyenként egy bites képsíkokból áll. A magasabb bithelyiértékű képsíkok a lassan változó, míg a kisebb bithelyiértékű képsíkok a gyorsan változó képrészleteket tartalmazzák. A tömörítés a képsíkok mentén történik. A kép visszaállítása progresszíven történik, azaz először állítjuk vissza a magasabb bithelyiértékű síkokat és csak utána az alacsonyakat. A kép egyre finomabb felbontású lesz.

A *veszteségmentes előrebecsléses kódolás* (relatív vagy más néven differenciál kódolás) lényege, hogy az egymást követő adatok eltérése gyakran minimális vagy zérus, ezért például egy kép adott részlete alapján annak környezete jól becsülhető. Videofolyamnál is két egymást követő képtartalom alig, vagy egyáltalán nem tér el egymástól, stb. Így tehát elégséges letárolni azokat az adatokat, amelyek az előrebecslés tévedéseit kikompenzálják. Hangsúlyoznám, hogy ez veszteségmentes, tehát az eredeti képet pontosan visszkapjuk.

1.3.2. Veszteséges tömörítési eljárások

A veszteséges tömörítési eljárásoknak két fő fajtáját különböztetjük meg: veszteséges előrebecsléses kódolás, transzformációs kódolás. Az *előrebecsléses tömörítést* az előző fejezetben már érintettem, csak ott a veszteség nem volt megengedve. Veszteséges tömörítésnél az előrebecslés hibáját korrigáló ún. differenciaadat akkor kerül kódolásra, ha egy bizonyos küszöbértéket meghalad. Ez a küszöbérték határozza meg a kitömörítéssel nyert kép valóságűségét és persze ezzel összefüggésben a tömörítési aránytényezőt is. Mit is jelent ez a gyakorlatban? Egy állókép esetén differenciára ad lehetőséget az élek (kontúrok)-felületek átmenet, videofelvételnél időbeli differenciálási lehetőség van az egymást követő képkockák állandó és változó részletei között. Az előbb vázolt tulajdonsága miatt nevezik ezt az eljárást relatív kódolásnak is. Fontos lehet a differenciában az is, hogy milyen két egymást követő képrészlet közötti különbségről van szó. Azokat a tömörítési eljárásokat, amelyek lehetővé teszik a tömörítésnek a mindenkor tömörítendő adatokhoz való alkalmazkodását, adaptív előrebecslési eljárásnak nevezzük. Erre példa az audioteknikában alkalmazott DPCM (differenciális impulzus kód moduláció) és ADPCM (adaptív DPCM eljárás) eljárások közötti különbség. Ezt a későbbiek folyamán még részletesen vizsgálni fogom.

A *transzformációs kódolás*nál az adatokat egy másik matematikai térbe transzformáljuk, mert ott könnyebb őket tömöríteni. Követelmény, hogy ez a transzformáció reverzibilis

legyen (inverz transzformáció). Az adattömörítés szempontjából a leghatékonyabb az ún. diszkrét koszosinuszos transzformáció (DCT) és a gyors Fourier transzformáció (FFT).

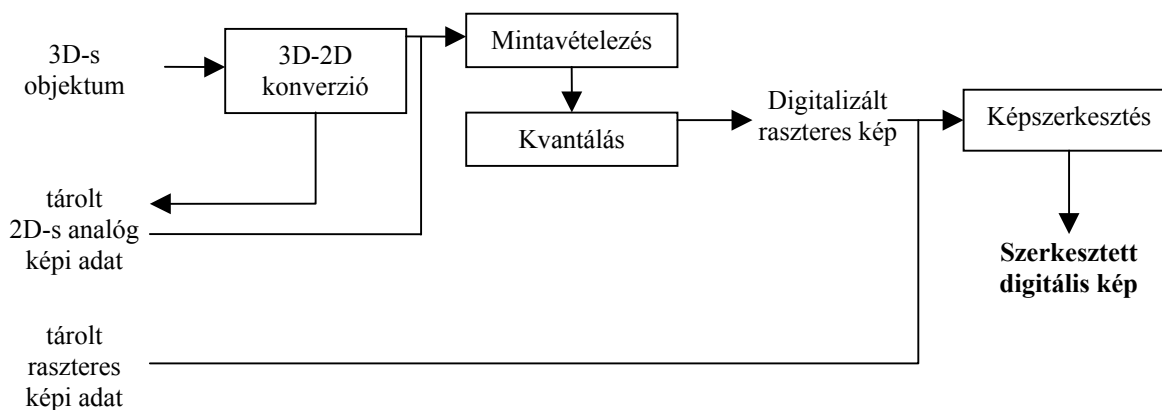
A *fontosság szerinti kódolás*nál az adatoknak valamilyen szabály szerint preferált csoportját kódolják. Például a Subband féle szelektív frekvencia-transzformációnál a jelnek csak egy előre meghatározott frekvenciasávba eső részét transzformáljuk.

2. KÉPFELDOLGOZÁS ÉS TÖMÖRÍTÉS

Az adatfeldolgozás folyamatának általános vizsgálata után nézzük meg azokat az egyes médiumoknál: állóképek; mozgóképek – animáció, videó; audio. A képfeldolgozás képszerkesztési műveleteivel ezen munka előző részében [1] már foglalkoztam, itt csak röviden térek ki rá.

2.1. Képdigitalizálás

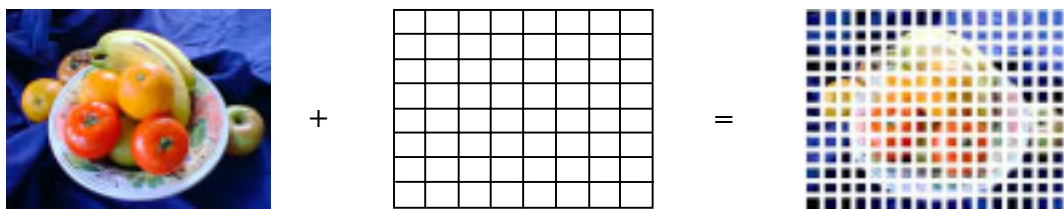
A valóságos világ objektumai háromdimenziósak, a rájuk eső fényt visszaverik, vagy pedig önállóan bocsátják ki. Amennyiben az így keletkezett fényt valamilyen érzékelő eszközzel rögzítjük, úgy kétdimenziós analóg képi információvá állítjuk elő, amely a tárolási módszertől függő képadatot eredményez. Ahhoz, hogy a kétdimenziós analóg képi információt (adatot) digitális információvá (adattá) alakítsuk át analóg-digitális (A/D) konverzióra van szükség. Ez egy kétlépéses művelet: mintavételezés, kvantálás (4. ábra).



4. ábra

A *mintavételezés* szemléltetésére álljon itt az alábbi példa (5. ábra). Az analóg képre helyezzünk egy kisméretű négyzetekből álló hálót, majd minden négyzetben található folytonos képi információt valamilyen módszerrel átlagoljuk, ezek után az egyes négyzetekben lévő analóg értékeket ezzel az átlagértékkel helyettesítjük.

Fontos megjegyezni, hogy ez az érték egy meghatározott intervallumban bármilyen értéket felvehet. Ez óriási adattömeget eredményezne, ezért ezt az intervallumot kisebb résztartományokra osztják fel, és ha az előbb kapott átlagérték egy ilyen résztartományba esik, akkor a fényinformációt annak bináris kódjával azonosítjuk. Ezt a folyamatot *kvantálás*nak nevezzük. Következik az előzőekből, hogy minél nagyobb az intervallum és minél kisebb résztartományokra osztjuk fel annál jobban fog a digitális kép az eredeti analóg képhez közelíteni, persze a képállomány mérete is annál nagyobb lesz. Az előzőekben kapott digitális kép feldolgozható a raszter (pontmátrix)grafikában (képszerkesztés) vagy már végállapotnak tekinthető.



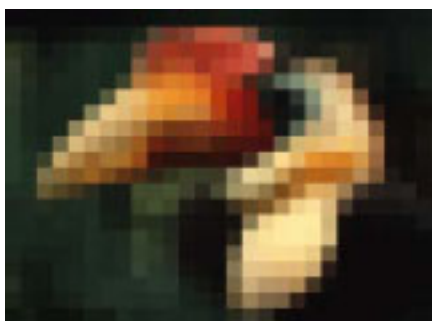
5. ábra

2.1.1. Leképezés

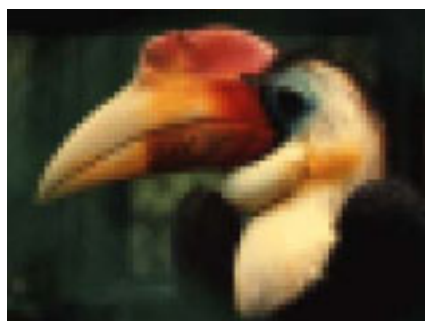
A leképezés során a háromdimenziós térben lévő objektumról ill. objektumból érkező elektromágneses ill. akusztikus hullámokat leíró függvényeket az emberi látás ill. hallás fiziológiai tulajdonságait figyelembe véve egy kétdimenziós függvénnyé transzformáljuk át: $f(x,y,z,t,\lambda,\underline{n}) \rightarrow g(x,y,t)$, amelyben az objektum egy pontját az x,y,z koordináták, időbeni függőségét a t jelöli. Időfüggetlen médiumoknál $t=0$, időfüggő médiumoknál $t \neq 0$. A szemlélőnek a helyzetét az objektumhoz képest az $\underline{n}$ vektor jellemzi, míg a fény ill. hang hullámtermészetét a λ hullámhossz írja le. A $g(x,y,t)$ függvény matematikai értelemben folytonos, nem negatív és korlátos továbbá a sík azon pontján van értelmezve, ahová leképeztük. Minden egyes szín-ill. hangjellemzőre létezik egy ilyen leképezési függvénykapcsolat, pl. a szín esetén az egy luminancia és a két krominancia függvény írható fel. Ez a leképezés a hardver eszközben zajlik le. A videokamera esetén a töltésképp, a fényképezőgép esetén a film ill. az előhívott fotó, hangfelvételnél a hangszalag tekinthető a $g(x,y,t)$ függvény fizikai megtestesítőjének. Fizikai értelemben azonban a részecskék véges méretét figyelembe véve legfeljebb diszkrét függvényekről beszélhetünk.

2.1.2. Mintavételezés

Az időfüggetlen médiumok esetén a $g(x,y)$ függvénnyel megadott sík egységnyi területről (kép) vett mintát, időfüggő médiumoknál megadott időközönként a $g(x,y,t)$ függvénnyel leírt jelből vett mintát mintavételezésnek nevezzük. Ennek következtében a folytonos $g(x,y)$ ill. $g(x,y,t)$ függvényből állítjuk elő a $g(x^*,y^*)$ ill. $g(x^*,y^*,t)$ diszkrét függvényeket.



6. ábra

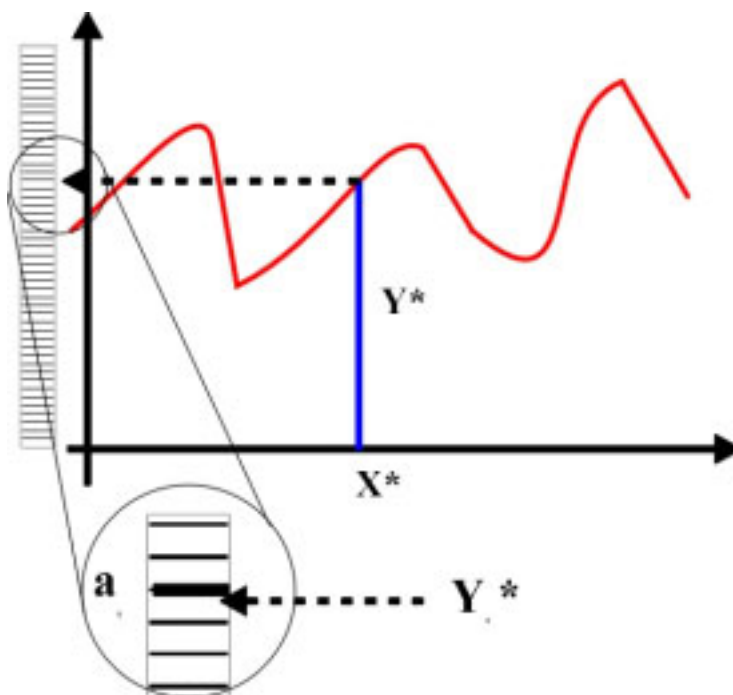


7. ábra

Ebből egyértelműen következik, hogy minél kisebb az egységnyi terület, amelyből mintát veszünk (6-7. ábra), ill. minél gyakrabban veszünk mintát (időfüggő médiumok), annál jobban közelít a $g(x,y)$ ill. $g(x,y,t)$ függvény a $h(x^*,y^*)$ ill. $h(x^*,y^*,t)$ függvényhez. Befolyásolja a $h(x^*,y^*)$ függvény minőségét a hozzárendelés módja, azaz, hogy az egységnyi területből hogyan válasszuk ki reprezentáló diszkrét értéket. A legegyszerűbb, de ugyanakkor legrosszabb eredményt adó középpontból vett színminta helyett az átlagos színminta értéket használjuk. Az egységnyi kis négyzet középpontjában lehet egy olyan kiugró színérték, amely jelentősen eltér a többitől és a hiba az egész négyzetre kiterjed. A mintavételezés tulajdonképpen egy integrálás, amely során kapott képpontérték – a $h(x^*,y^*)$ függvénynek a kép adott pontjába tartozó helyettesítő értéke – egy egységnyi terület meghatározott tulajdonságértékeit adja. A kép médium esetében ez egy luminancia és két krominancia érték. Az integrálás során a súlyfüggvény a mintavételezés helyétől távolodva csökken, azaz a mintavételezés a távolfekvő értékeket kevésbé veszi figyelembe. Az ideális mintavételezés esetén a $h(x^*,y^*)$ függvényből a teljes intervallumban visszaállítható a $g(x,y)$ eredeti függvény.

2.1.3. Kvantálás

A digitalizálás befejező művelete a kvantálás, amikor is a tetszőleges diszkrét értékű $h(x^*,y^*)$ függvényt egy meghatározott diszkrét értékű $j(a,b)$ függvénnyé transzformáljuk át. Ezek a meghatározott diszkrét értékek egy előre meghatározott minta értékei közé tartoznak. Ezek az értékek bitben, azaz 2 hatványaiként vannak megadva. Képeknél $2^8=256$ féle, $2^{16}=65.536$ féle; hangoknál $2^{16}=65.536$ féle elemű leggyakrabban ez a minta. Színes kép esetén a kvantálás összetevőnként (luminancia, krominancia), egymástól függetlenül történik. A minta lehet egyenlő ill. nem egyenlő osztásközű, attól függően, hogy melyik eredményez jobb képet.



8. ábra

A kvantálás előbb leírt folyamatát a 8. ábra szemlélteti, ahol az Y_1^* értékhez rendelt legközelebbi mintaelem az a_1 .

Mint a 9-10. ábra mutatja képeknél a minta elemeinek száma jelentősen befolyásolja a kvantálás minőségét. A 9. ábrán egy 2 bit (4 szín), míg az 10. ábrán egy 8 bit (256 szín) színmélységű képet láthatunk, a különbség önmagáért beszél.



9. ábra



10. ábra

2.2. Digitális képek szerkesztése

A digitális időfüggetlen médiumok szerkesztési műveleteiről a korábbiakban már volt szó [1], a most következő részben elmélyítjük ezen ismereteinket. A végcél szempontjából a képjavításnak kétféle módszerét különböztetjük meg: *képhelyreállítás*, képminőség-fokozás. Az előbbinek célja, hogy a digitalizálás során torzult képből előállítsuk az ideális képet, amely eltávolítja a képet zavaró hatásokat. A *képminőség-fokozás* célja ezzel ellentétes lehet, amikor is a kép bizonyos tulajdonságait kiemeljük, hangsúlyozzuk. A képjavítási eljárások elvégezhetők a kvantálás után kapott függvény(ek) felhasználásával - $j(a,b)$ – vagy pedig ezeket a függvényeket a frekvencia tartományba transzformáljuk és így hajtjuk végre a szerkesztési műveleteket. Ez utóbbi esetben Fourier transzformációt hajtunk végre, mert az - bizonyos feltételek teljesülése esetén - egyértelműen és invertálható módon sorbafejthetők. Ennek matematikai hátterét világítom meg röviden az elkövetkező fejezetben.

2.2.1. A képjavítás matematikai háttere – a Fourier transzformáció

a) Folytonos függvények Fourier transzformációja (FFT)

A Fourier transzformációval kapott $k(t)$ periódikus időfüggvény sorbafejtés után felbontható egy konstans tag, továbbá végtelen sok harmonikus összetevő összegére. A harmonikus összetevőket általában szinuszos és koszinuszos időfüggvényekkel írják le. A függvény T periódusidejének reciproka adja az alapharmonikus frekvenciáját, és a további összetevők frekvenciája ennek az alapharmoniciának az egész számú többszörösei (n):

$$k(t) = c_0 + \sum_{n=1}^{\infty} (a_n \cdot \cos(2\pi n \frac{t}{T}) + b_n \sin(2\pi n \frac{t}{T})) \quad (1)$$

,ahol a konstans összetevő: $c_0 = \frac{1}{T} \int_0^T k(t) dt$, míg a harmonikus összetevők amplitúdói:

$$a_n = \frac{2}{T} \int_0^T k(t) \cdot \cos(2\pi n \frac{t}{T}) dt \quad (2)$$

$$b_n = \frac{2}{T} \int_0^T k(t) \cdot \sin(2\pi n \frac{t}{T}) dt. \quad (3)$$

A sorbafejtés feltételei, hogy a $k(t)$ függvény a T tartományban legyen korlátos, integrálható és legalább szakaszonként differenciálható.

A trigonometrikus függvények komplex exponenciális függvényekkel is kifejezhetők, ennek alapja a következő két összefüggés:

$$\cos(2\pi n \frac{t}{T}) = \frac{1}{2} \cdot (e^{j2\pi n \frac{t}{T}} + e^{-j2\pi n \frac{t}{T}}) \quad (4)$$

$$\sin(2\pi n \frac{t}{T}) = \frac{1}{2j} \cdot (e^{j2\pi n \frac{t}{T}} - e^{-j2\pi n \frac{t}{T}}) \quad (5)$$

Ezek behelyettesítésével a $k(t)$ függvény a következőképpen írható fel (átcsoportosítva a szumma jel alatti tagokat):

$$k(t) = c_0 + \sum_{n=1}^{\infty} \left(\frac{a_n - jb_n}{2} \cdot e^{j2\pi n \frac{t}{T}} + \frac{a_n + jb_n}{2} \cdot e^{-j2\pi n \frac{t}{T}} \right) \quad (6)$$

Ha bevezetjük a harmonikus összetevőkre a komplex amplitúdót

$$C_n = \frac{a_n - jb_n}{2},$$

akkor a $k(t)$ még egyszerűbben felírható:

$$k(t) = \sum_{n=-\infty}^{\infty} C_n \cdot e^{j2\pi n \frac{t}{T}} \quad (7)$$

Az előbb definiált a_n és b_n amplitúdók felhasználásával rendezés után a C_n komplex amplitúdó a következőképpen határozható meg:

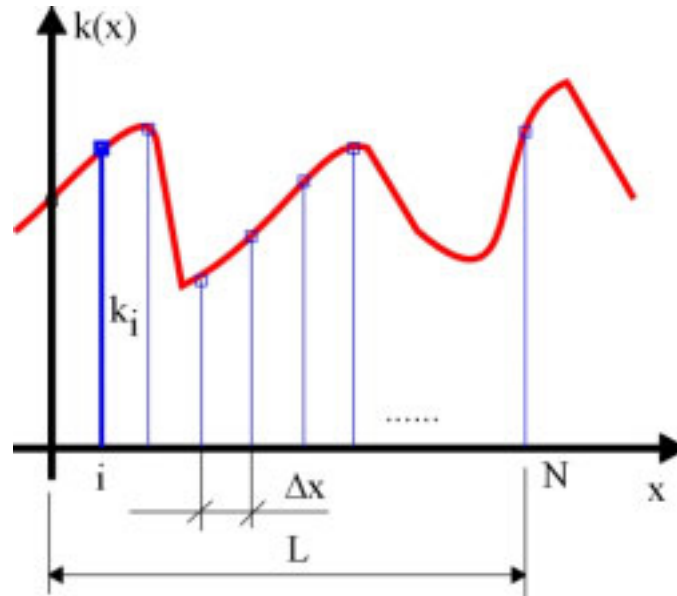
$$C_n = \frac{1}{T} \cdot \int_0^T k(t) \cdot e^{-j2\pi n \frac{t}{T}} dt \quad (8)$$

A Fourier sorfejtés nemcsak a periódikus időfüggvényekre, hanem tetszőleges $k(x)$ skalárfüggvényre igaz, ahol T periódus idő helyett az L periódushosszt bevezetve az előző két egyenlet a következőképpen alakul:

$$k(x) = \sum_{n=-\infty}^{\infty} C_n \cdot e^{j2\pi n \frac{x}{L}} \quad (9)$$

b) Diszkrét függvények Fourier transzformációja (DFT)

A mintavételezésnél már láttuk, hogy a függvény nem folytonos, hanem diszkrét mintáival áll rendelkezésre (11. ábra).



11. ábra

L periódushossz alatt N darab mintát feltételezve, az i . helyen k_i értéket mérve igyekszünk a sorbafejtést végrehajtani. Két mintavétel között a függvény nem mutathat olyan változásokat, ami információvesztéshez vezet. A mintavételi tétel szerint a mintavételnek olyan sűrűnek kell lennie, hogy a jelben előforduló legmagasabb frekvenciájú összetevő egy periódusára legalább két minta essék.

A $\Delta x = \frac{L}{N}$ mintatávolságot felhasználva a Fourier sor együtthatói csak közelítéssel határozhatók meg:

$$C_n \cong \frac{1}{L} \cdot \sum_{i=0}^{N-1} \Delta x \cdot k_i \cdot e^{-j2\pi n \frac{1}{L} \cdot i \cdot \Delta x} = \frac{1}{N} \cdot \sum_{i=0}^{N-1} k_i \cdot e^{-j \frac{2\pi}{N} i n} \quad (10)$$

Az N darab elemből álló k_i értéksorozat diszkrét Fourier transzformáltját az előző képlet végén látjuk és D_n -nel jelöljük:

$$D_n = \frac{1}{N} \cdot \sum_{i=0}^{N-1} k_i \cdot e^{-j \frac{2\pi}{N} i n}, \text{ ahol } 0 \leq n \leq N-1$$

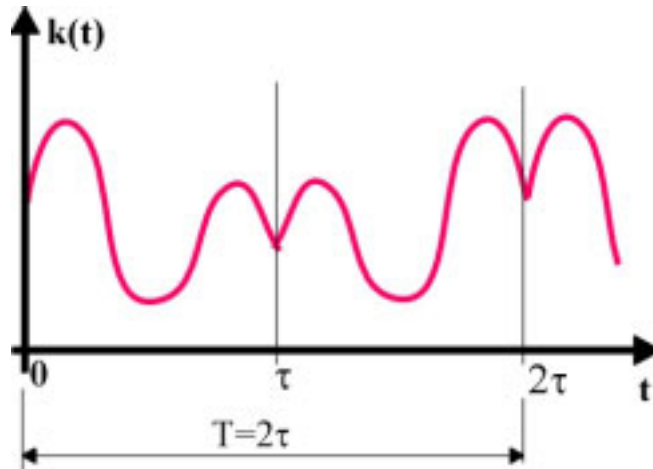
Az előző együtthatók felhasználásával a függvény az alábbi összefüggés felhasználásával transzformálható vissza:

$$k_p = \sum_{n=0}^{N-1} D_n e^{j \frac{2\pi}{N} p n}, \text{ ahol } 0 \leq p \leq N-1$$

c) Folytonos függvények koszinuszos Fourier transzformációja (FCT)

A gyakorlatban a Fourier transzformáció egy speciális esetét a koszinuszos Fourier transzformációt alkalmazzák. A (1)-(3) képletek felhasználásával gondoljuk végig a

sorbafejtést. A $k(t)$ függvényt sorral közelítjük ismét, de most csak a $[0;\tau]$ intervallumban, ui. az ezen kívüli tartományban nem cél a jó közelítés.



12. ábra

Sőt arra a tartományra nézve tetszőleges feltételezést is tehetünk a függvény lefutására vonatkozóan. Ezért az egyszerűség kedvéért a $k(t)$ függvény a $[\tau;2\tau]$ intervallumban legyen $t=\tau$ helyre szimmetrikus a $[0;\tau]$ -hez képest (12. ábra). A $k(t)$ függvény $T=2\tau$ szerint periodikus. Mindezek figyelembe vételével a két harmonikus amplitúdó így alakul:

$$a_n = \frac{2}{\tau} \int_0^{\tau} k(t) \cdot \cos(\pi n \frac{t}{\tau}) dt \quad (13)$$

$$b_n = 0 \quad (14)$$

A $k(t)$ függvény szimmetrikusságának az előnye a (14) egyenletben látható. A konstans együttható a (15) szerint írható fel.

$$c_0 = \frac{1}{\tau} \int_0^{\tau} k(t) dt \quad (15)$$

A $k(t)$ függvény most már koszinuszos tagokból álló sor összegeként állítható vissza, innen ered a koszinuszos sorbafejtés elnevezés:

$$k(t) = c_0 + \sum_{n=1}^{\infty} a_n \cdot \cos(\pi n \frac{t}{\tau}) \quad (16)$$

d) Diszkrét függvények koszinuszos Fourier transzformációja (DCT)

A 2.2.1.3. fejezetben megismert törvényszerűségeket a 2.2.1.2. fejezetben bevezetett (11) és (12) képletekre alkalmazva a D_n és a k_p az alábbiak szerint írhatók fel:

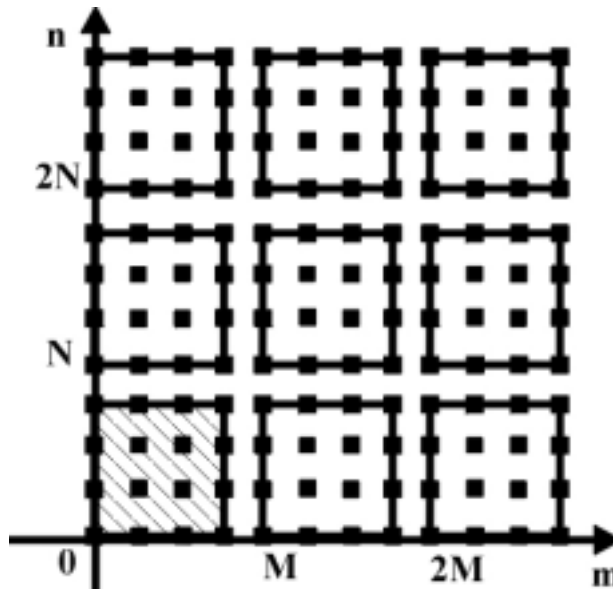
$$D_n = \sum_{i=0}^{N-1} k_i \cdot \cos\left(\frac{\pi}{N} n \left(i + \frac{1}{2}\right)\right) \quad 0 \leq n \leq N-1 \quad (17)$$

$$k_p = \frac{2}{N} \sum_{n=0}^{N-1} D_n \cdot \cos\left(\frac{\pi}{N} n \left(p + \frac{1}{2}\right)\right) - \frac{D_0}{N} \quad 0 \leq p \leq N-1 \quad (18)$$

Az összefüggések felírásakor felhasználtuk a (4 és 5) komplex exponenciális függvényeket is.

e) Kétváltozós diszkrét függvények Fourier transzformációja (2D DFT)

A kép leírásához kétváltozós függvényre ill. függvényekre van szükség, még pedig szám szerint egyre ill. háromra attól függően, hogy fekete-fehér vagy színes kép leírásáról van-e szó. Fekete-fehér képnél a $k(x,y)$ a luminancia, míg színes képnél ezen kívül az $l(x,y)$ ill. $m(x,y)$ függvények a krominancia leírására szolgálnak. Az egyszerűség kedvéért most vizsgáljuk meg a luminanciát leíró függvényt egy x,y síkon, pontosabban „diszkrétten” véve az m,n síkon (13. ábra).



13. ábra

A $k(x,y)$ folytonos függvény sorba fejtésével kapott együtthatómátrix értékeit az alábbi összefüggés adja a (11) felhasználásával:

$$D_{mn} = \frac{1}{M \cdot N} \sum_{q=0}^{N-1} \sum_{p=0}^{M-1} k_{pq} \cdot e^{-j2\pi \left(\frac{1}{M} pm + \frac{1}{N} qn \right)} \quad 0 \leq m \leq M-1 \quad \text{és} \quad 0 \leq n \leq N-1$$

A visszatranszformálást a (12) alapján végezzük:

$$k_{rs} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} D_{mn} \cdot e^{j2\pi \left(\frac{1}{M} mr + \frac{1}{N} ns \right)} \quad (20)$$

A diszkrét függvény az m,n síkon M és N szerint periodikus. A (20) szerinti inverztranszformációt felhasználva állapítsuk meg, hogy az m,n sík egyes pontjaihoz milyen

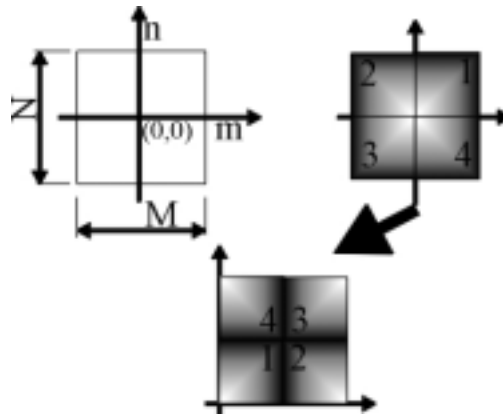
térharmonikusok tartoznak. Először az r és s diszkrét változóknak feleltessük meg az x és y folytonos változókat:

$$r = \frac{x}{\Delta x} \quad \text{és} \quad s = \frac{y}{\Delta y} \quad \text{ahol } \Delta x \text{ és } \Delta y \text{ a két mintavételi lépésköz.}$$

Ezt felhasználva a (20) egy tagja:

$$K(x, y) = D_{m, n} \cdot e^{j2\pi \left(\frac{1}{M} m \frac{x}{\Delta x} + \frac{1}{N} n \frac{y}{\Delta y} \right)} = D_{m, n} \cdot \left(\cos \left(2\pi \left(m \frac{x}{A} + n \frac{y}{B} \right) \right) + j \cdot \sin \left(2\pi \left(m \frac{x}{A} + n \frac{y}{B} \right) \right) \right) \quad (21)$$

Az $M \times N$ méretű téglalap esetén a $(0,0)$ koordinátájú pontban van a térfrekvenciáknak minimuma van. Fekete-fehér árnyaltos képet feltételezve a $(0,0)$ pontban a luminancia értéke minimális, ettől távolodva minden irányban folyamatosan növekszik (14. ábra).



14. ábra

Mindezek alapján megállapítható, hogy ha egy folytonos $k(x,y)$ képfüggvényt mintavételezünk, akkor először a magas térfrekvenciás harmonikusokat kell kiszűrni. A *kétdimenziós mintavételi tétel* kimondja, hogy ha a szűrés olyan, hogy az m irányban az $\frac{M}{2A} = \frac{1}{2\Delta x}$ feletti térfrekvenciákat, míg az n irányban az $\frac{N}{2B} = \frac{1}{2\Delta y}$ feletti térfrekvenciákat távolítjuk el, akkor a (21) szerinti tagokból álló transzformáció visszaadja az eredeti képet.

A fekete-fehér képfelvételi eljárásoknál ez tétel érvényesül, hiszen egy pixel nem a kép egyetlen pontjának fényességéből adódik, hanem egy kicsi környezet fényességének átlagából adódik. Természetesen színes képnél ez háromszor hajtódik végre.

Visszatérve a 14. ábrához és a (21) összefüggéshez, a D_{mn} mátrix kis frekvenciás elemei a transzformált mátrix sarokpontjaiba, míg a magas térfrekvenciás együtthatók a téglalap közepére kerülnek.

f) Kétféle változós diszkrét függvények koszinuszos Fourier transzformációja (2D DCT)

Az előző pontban megismertek alapján felírhatjuk a kétféle változós diszkrét koszinuszos függvényekre is a k_{rs} -t.

$$krs = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} D_{mn} \cdot \cos \left(2 \pi (2n+1) \cdot r \cdot \frac{1}{4N} \right) \cdot \cos \left(2 \pi (2m+1) \cdot s \cdot \frac{1}{4M} \right) \quad (22)$$

A levezetések mellőztem, de ez nem jelentheti annak kevésbé fontosságát, hiszen olyan képformátum is mint a JPEG is ezt használja. Ennek kiemelt fontosságára utalva, erről külön fejezetben szólok.

2.2.2 Képjavítás

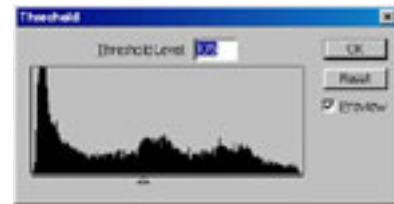
Általános érvényű megállapítás, hogy a képminőséget javító eljárások nem eredményezhetnek olyan információt, amely valamilyen formában ne lett volna rajta a kiindulási képen is. A képjavítási eljárásokat a felmerülő hibák típusa szerint lehet csoportosítani. Ezek szerint megkülönböztetünk: kontrasztfokozó, zajelnyomó, élkiemelő műveleteket. Az egyes műveletek végrehajtásához az Adobe Phoshop program 5.0-s verzióját hívjuk segítségül. Az egyes műveletek bemutatásához a kiinduló kép legyen 15. ábrán látható csend-élet. [3-CD mellékletén található] A legfontosabb képjavító műveletek érvényesíthetők a kép kijelölt részeire vagy az egész képre is.



15. ábra



16. ábra



17. ábra

2.2.2.1. Színkorrekció

A beszkenelt kép vagy digitális fotó élességének fokozását színkorrekciónak nevezzük. Ezzel a művelettel képünkön több és szebb szín látszatát tudjuk kelteni, holott valójában csökken a szerkesztett képünket alkotó színek száma. Két olyan pixel, amelyeknek a színe különböző volt, a színkorrekció hatására azonos színűekké válhatnak. Lehet, hogy a kép sokkal jobban fog mutatni, de kevésbé lesz színgazdag. Az ide tartozó műveletek az *Image/Adjust* menüben találhatók meg. A legfontosabb színkorrekciós műveleteket:

- a **színkiegyenlítés** (*Equalize*) az összes színcsatornát alapul véve a legvilágosabb szint fehérre a legsötétebbet pedig feketére változtatja, ezáltal a képpontok egyenletes eloszlásúak lesznek a luminancia (fényerősség, világosság) tartományban (16. ábra).
- a **határérték** (*Threshold*) az összes olyan szint, amelyek a beállított értéknél világosabbak fehérre, míg amelyek sötétebbek feketévé alakítja át. Ez a művelet fekete-fehér képek szerkesztésénél alkalmazható. A szerkesztési műveletet elősegíti a bejelentkező hisztogram, amely a kép egészére mutatja a luminancia értékek gyakoriságát (az adott luminancia értékből hány pixel fordul elő a képen). A csúszkát úgy célszerű beállítani, hogy balról ill. jobbról ugyanakkora területet fedjenek le a függőleges vonalak (17. ábra). Az eredmény a 18. ábrán látható.

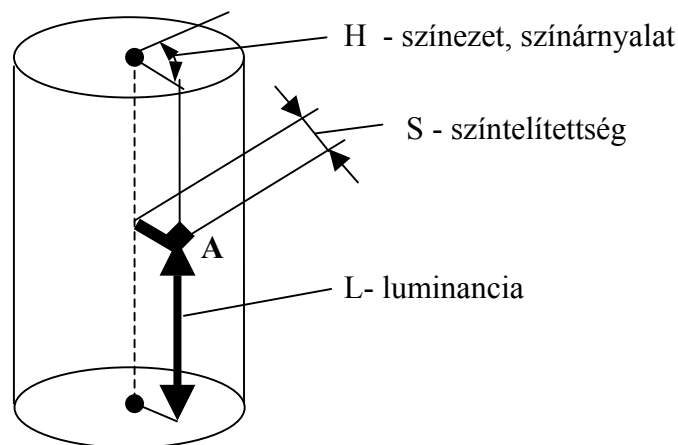
- a **keményítés** (*Posterize*) a teljes luminancia tartományt automatikusan felosztja meghatározott számú egyenlően növekvő részekre és a megadáskor azt az értéket állítjuk be, melyeket megakarunk tartani. Nagyobb értékek finomabb, míg kisebb értékek durvább színbeállítást eredményeznek. Ezt a műveletet nem csak fekete-fehér, hanem színes képekre is alkalmazhatjuk, de ez akkor igazán hatékony, ha magasan áteresztő szűrőkkel kombinálva alkalmazzuk.



18. ábra



19. ábra



20. ábra

A 20. ábrán látható ábra műveletei sorban a következők: rétegmásolat készítés – magasan áteresztő szűrés - keményítés a másolaton – a két réteg összefűzése – rétegátfedési módok megadása (Luminosity, Hard Light, Hue).

- a **színeskorrekció** (a színtulajdonságok széleskörű megváltoztatását eredményezi (színezet – H, világosság – L, színtelítettség – S). A 16. ábrán a HSL színmodell összetevőit látjuk, amelyek megváltoztatása jelenti a kép átszínezését. Ezek a műveletek elvégezhetők az egyes színcsatornákra, vagy pedig az összes színcsatornára egyidejűleg (23. ábra - Edit).

A színezet (Hue) jelenti a szivárvány tiszta színeit, amelyet egy -180° - $+180^\circ$ -os szögtartományban mérnek, 0° -nál van a piros, 60° -nál a sárga, 120° -nál a zöld, 180° -nál a ciánkék, 240° -nál a kék és 300° -nál pedig a bíbor szín található. A telítettség (Saturation) adja a szín tisztaságát, élénkségét. -100 jelenti a szürkét és $+100$ pedig az adott színezet legtisztább színeit adja. A világosság, fényesség (Lightness, Brightness, Luminancia) adja a szín helyét a fehér-fekete tartományban. -100 jelenti a feketét és $+100$ pedig a fehéret (23. ábra). A Pipetták szolgálnak szín hozzáadására ill. elvételére, míg a színtartomány szabályzók segítségével kitágíthatjuk ill. leszűkíthetjük azt a tartományt, amelyen a színeskorrekciót végrehajthatjuk.(csak ha a Master opción kívül vá-

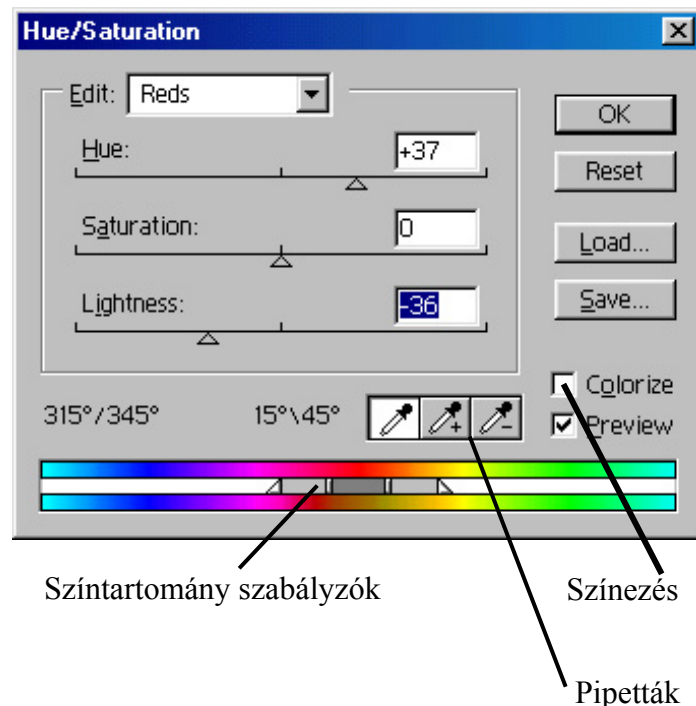
lasztunk valamit). A 21. ábrán a vörös csatornában a H, S, L értékek egyidejű állításával sikerült éretlen paradicsomok látszatát kelteni, míg a 23. ábrán a Brightness/Contrast állításával sikerült érettebb paradicsomok látszatát kelteni.



21. ábra



22. ábra



23. ábra

- az **átszínezésre** többféle művelet is szolgál, emeljük ki ezek közül *Replace Color* és a *Selective Color* opciókat. ezek lehetővé teszik, hogy kijelöljünk egy egymással összefüggő színű területet, majd beállítsuk a színezetét és a telítettségét (24. ábra).

A Pipetták alkalmazásával lehet a színeket kijelölni, bővíteni ill. csökkenteni. A Fuzziness a kijelölt színen túlmenően kiválasztandó színtartomány állítására szolgál. A 0 érték beállításakor csak a kijelölt színt választja ki, vagyis a tűrés értéke 0. A Fuzziness nagyon hasonlatos a Varázspálca kijelölés (Magic Wand) Tűrés (Tolerance) opciójához, azzal a különbséggel, hogy a Tűrés állításával a kijelölés pereme finom lesz (anti-aliased), míg a Fuzziness inkább egy áttűnő, lépcsőzetes peremet eredményez. A Transform blokkon belül látjuk azokat a tulajdonságokat, amelyek a kijelölt színre vonatkozóan megváltoztathatóak. A 25. ábrán látható a Selective Color parancs párbeszédablaka, ezzel elsősorban CMYK színrendszerben készült képek színezhetők át.

A Colors opciónál kiválasztható a szerkeszteni kívánt szín, majd a négy nyomdai szín szerkeszthetővé válik. Egy ilyen szerkesztés eredménye látható a 25. ábrán. Az előző műveleteknél fejlettebb színeskorrekciós parancs a Variations (26. ábra), amellyel mindazon műveletek elvégezhetőek, amelyekről eddig szó volt, csak minden egyidejűleg szemlélhető is. Az ablak közepén az RGB ill. CMY színek láthatók, ezzel lehet eltolni a képet valamelyik szín felé, ennek lépésköztét a Fine-Coarse (Finom-Durva) értéke adja. Az ablak tetején lévő rádiógombok segítségével szabályozhatók, hogy a kép mely színei legyenek befolyásolva. A Shadows a sötétebb, a Highlights a világosabb, míg a Midtones a kettő közötti színek változtatására szolgál. A Saturation a színtelítettség állítására szolgál. Az ablak jobb szélén lévő Lighter-Darken a fényesség, luminancia állítására szolgál.



24. ábra



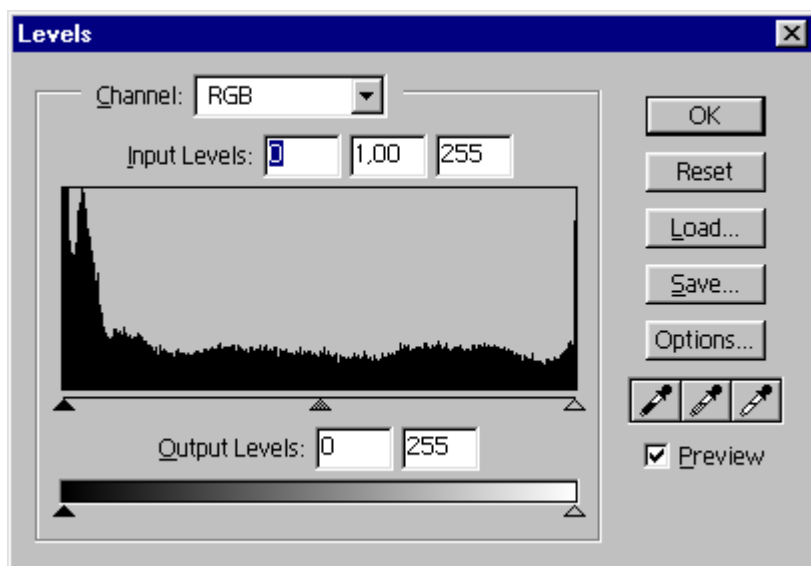
25. ábra



26. ábra

Az eddig megismert képszerkesztési műveletek kibővített tárháza található a *Levels* ill. a *Curves* parancsokban. A *Levels* parancs meghívásakor a 27. ábrán látható ablak jelenik meg, amelynek értelmezése a következő:

- Channel – színcsatorna, amelyre a módosítás vonatkozik,
- Input levels – bemeneti szintek (ezek összhangban vannak a hisztogram alatti három csúszkával) adják a kép kontrasztosságát. Az első (fekete) és utolsó (fehér) szám ill. csúszka 0 és 255 közötti értéket vehet fel. Az első rovatba írt 35 például azt jelenti, hogy az eredeti képen minden olyan szín, amelynek fényessége (világossága) 35-nél kisebb fekete lesz, ezzel a képet sötétítjük. Ha viszont az utolsó rovatban lévő értéket csökkentem például 220-ra, akkor minden olyan szín melynek fényessége (világossága) 220-nál több fehér lesz, ezzel a képet világosítjuk. A középső rovatban lévő Gamma érték (0.10-9.99) ill. a középső csúszka adja a középszürke világossági értékét. Minden Gamma-érték változtatás a kép kontrasztosságának csökkenését fogja eredményezni anélkül, hogy az árnyékok ill. az erős megvilágítások megváltoznának. A Gamma 1.00-re állítása adja a középszürkét, ennek csökkentése a középszürke (középtónus) sötétítését, növelése a annak világosítását eredményezi.
- Output levels – kimeneti szintek a világossági tartomány (luminancia) lerövidítésére szolgál. Ennek eredményeként a kép legsötétebb pontjai világosodnak, a világosak pedig sötétednek. A bal oldali rovatba írt 40 például azt jelenti, hogy képünkön nem lesz olyan szín, amelynek világossága, fényessége (luminanciája) 40-nél kevesebb, ezzel a képet világosabbá teszi. Ha viszont a jobb oldali rovatba például 190-at írunk, nem lesz a képünkön olyan szín, melynek világossága 190-nál több, ezzel a képet sötétebbé teszi.
- Eyedroppers – pipetták automatikus világosságállítást fog eredményezni. A fekete pipettával (bal oldali) egy pixelre kattintva az összes olyan színt, amelynek világossága a pixelénél kevesebb, feketíti. A fehér pipettával (jobb oldali) egy pixelre kattintva az összes olyan színt, amelynek világossága a pixelénél több, fehéríti. A középső pipetta a középszürke világosítására szolgál. A pipetták szerepe, tehát meg egyezik az Input levels-nél leírtakkal.



27. ábra

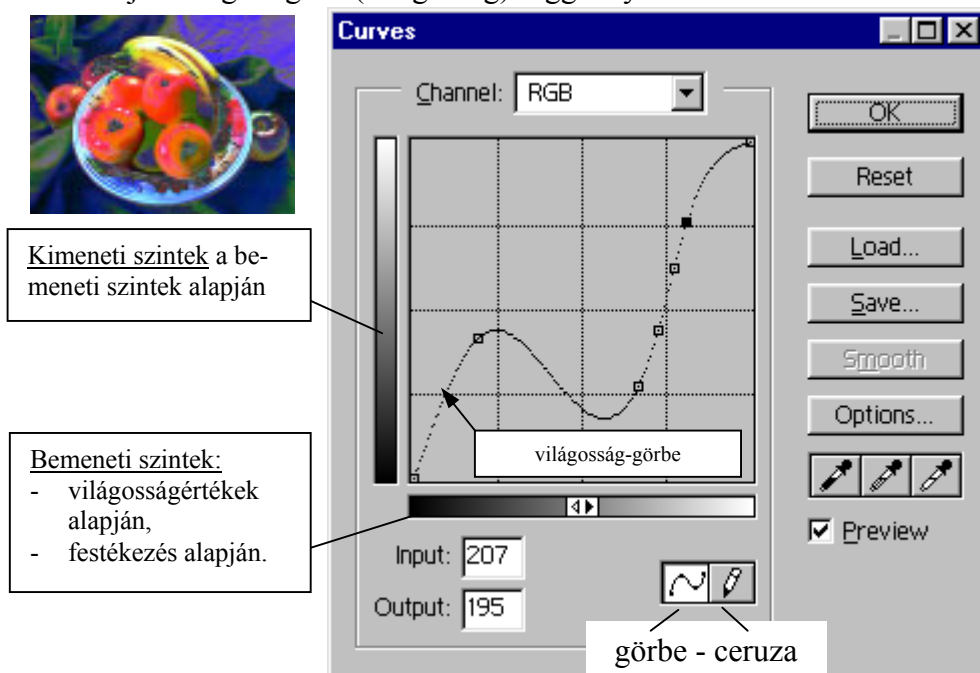
Az előbb leírt műveletek az egyes színcsatornákban (R, G, B; C, M, Y, K) is elvégezhetők. Mintaképek közlése helyett bátorítok mindenkit a következő értékek egymás utáni kipróbálására egy tetszőleges képen:

Input levels			Output levels	
0	1.00	255	0	255
100	1.00	255	0	255
0	1.00	170	0	255
0	0.60	255	0	255
0	2.00	255	0	255
0	1.00	255	40	255
0	1.00	255	0	150

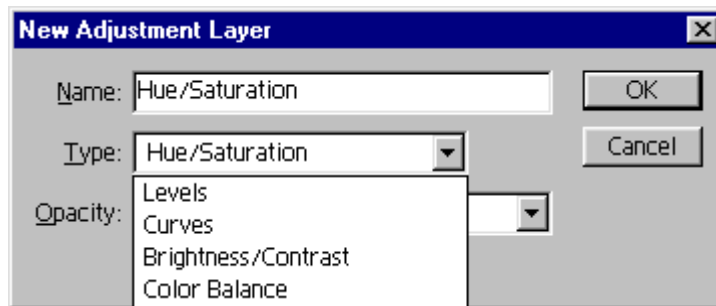
1. táblázat

A *Curves* parancs az előbb említett műveleteken túlmenően a Gradációs görbék segítségével lehetővé teszi, hogy akár színcsatornánként az összes világosságértéket átkonvertáljuk egy másik értékre. A fényerősségskala (vízszintes tengely) két részből áll (fekete-fehér ill. fehér-fekete) – a sorrendnek jelentősége van -, amelyek a színeket jellemzik. A fekete-fehér sorrend esetén az egyes színeket fényerősségük alapján adhatjuk meg, ha ilyenkor a gradációs görbét felfelé mozdítjuk el, akkor a képet világosítjuk, ha pedig lefelé, akkor sötétítjük. A fehér-fekete sorrend esetén az egyes színeket a szükséges festék szerinti osztásban adja. A gradációs görbe módosítható pontonként (görbe) vagy meg is rajzolható (ceruza).

Ha a digitális képünk életlen fényképről származik, akkor az egyes képpontokon kapott intenzitás és az adott pontra beesett fény mennyisége nincs lineáris arányban egymással. Alapértelmezésben a lineáris kapcsolatból indulunk ki és ezt transzformáljuk például a 28. ábrának megfelelően. A gradációs függvény megadja tehát az egyes képpontok intenzitását adja a megvilágítás (világosság) függvényében.



28. ábra



29. ábra

Fontos eleme lehet a színekorrakciónak az ún. korrekciós réteg (Layer/New/Adjustment Layer), amely olyan réteg ami a matematikai színekorrakciós információkat tartalmazza. Ez a réteg végrehajtja a korrekciót az összes alatta lévő rétegen, de a fölötté lévőeken nem. A parancs meghívása esetén a 29. ábrán látható ablak jelentkezik be, amely nem csak a rétegekre jellemző paramétereket teszi beállíthatóvá, hanem a Type opción keresztül támogatja a színekorrakciót is. A réteg-alapú korrekció előnyei:

- örökre szerkeszthetőség,
- támogatott rétegmaszkolás,
- a korrekciók sorrendiségének megváltoztathatósága,
- az összzhatásmód módosíthatósága.

2.2.2.2. Képjavítás szűréssel

A képet ért zavaró hatásokat *zaj*nak nevezzük. A zajt a képfeldolgozás előtt el kell távolítani a képről, vagy legalább is minimalizálni kell. Ezt a műveletet zajsűrésnek nevezzük. A legtöbb hibajavítás elvégezhető a zajsűréssel mind a frekvencia, mind pedig a Fourier transzformáció előtt, azaz a képtartományban.

A *frekvencia tartományban végrehajtott zajsűrés*kor nem egyből a D_{mn} együttható mátrixsal (19) hajtjuk végre a visszatranszformálást, hanem egy S_{mn} szűrőegyüttható mátrixsal beszorozzuk, majd ezek után végezzük el az inverz Fourier transzformációt. A visszatranszformáláskor kapott k_{rs} függvény (20) helyett a „szűrt” függvénnyel leírt képet kapjuk (k'_{rs}). A frekvenciatartományban végrehajtott szűrés alkalmas jól meghatározott frekvenciájú periodikus zajnak a képből történő kiszűrésére. Ez úgy történhet, hogy az S_{mn} bizonyos elemeit 0-nak vesszük, így a kép visszaállításakor ennek az együtthatónak megfelelő bázisfüggvény kiesik. A képet érő zajok egy jelentős része abban nyilvánul meg, hogy megváltozik bizonyos képpontok világosságkódja. A zaj a világosságkódokban a véletlenszerű, hirtelen bekövetkező változást jelenti, azok pedig az előbb említett magas frekvenciájú bázisfüggvények „lenullázásával” kiszűrhetők. Ennek azonban lehet egy hátrányos következménye is, ui. az élek is „elmosódhatnak”. A zajsűrő függvényeket *aluláteresztő szűrők*nek nevezzük. Nézzünk néhány aluláteresztő szűrőt:

- az **ideális szűrő** az $u \leq u_0$ sugarú kör belsejébe eső frekvenciákat átengedi, míg az ezen kívülieket kiszűri. A 14. ábra alapján tudhatjuk, hogy az origó környezetében vannak a kis frekvenciás együtthatók, míg onnan távolodva pedig a magasak találhatók. Így távolítja el a magas frekvenciákat a luminancia függvényből. Az ideális szűrő hátránya, hogy a magas frekvenciás összetevőket teljesen kiszűri, így az éleket is elsímítja, a képet homályosítja.

$S_{mn} = 1$, ha $u \leq u_0$; különben $S_{mn} = 0$.

Az $u = (r^2 + s^2)^{1/2}$ távolságfüggvény (kör) adja az adott frekvenciapár origótól mért távolságát, az u_0 pedig egy konstans.

- az **exponenciális szűrő** igyekszik kiküszöbölni az ideális szűrő hátrányait, kevésbé homályosít, az éleket jobban visszaadja.

$S_{mn} = \exp[-(u/u_0)^n]$, ahol az n érték változtatásával befolyásoljuk a szűrőfüggvény lefutását.

Ezért ennek a módszernek az ellentéte az *élkiemelés*, amelyhez azonban már nem aluláteresztő, hanem feluláteresztő szűrőt kell alkalmazni. Ezek a szűrők a frekvenciatarományban kiszűrik az alacsony frekvenciás Fourier-transzformáltakat, míg átengedi a magasakat. Ez utóbbi lehet éles kontúr, de lehet zaj is. Nézzünk néhány feluláteresztő szűrőt:

- az **ideális szűrő** az $u \geq u_0$ sugarú kör belsejébe eső frekvenciákat átengedi, míg az ezen belülieket kiszűri. A 14. ábra alapján tudhatjuk, hogy az origó környezetében vannak a kis frekvenciás együtthatók, míg onnan távolodva pedig a magasak találhatók. Így távolítja el az alacsony frekvenciákat a luminancia függvényből. Az éleket jól kiemeli, de a kép zajos marad.

$S_{mn} = 0$, ha $u \leq u_0$; különben $S_{mn} = 1$.

Az $u = (r^2 + s^2)^{1/2}$ távolságfüggvény (kör) adja az adott frekvenciapár origótól mért távolságát, az u_0 pedig egy konstans.

- az **exponenciális szűrő** igyekszik kiküszöbölni az ideális szűrő hátrányait, kevésbé zajos a kép és az éleket is elég jól visszaadja.

$S_{mn} = \exp[-(u_0/u)^n]$, ahol az n érték változtatásával befolyásoljuk a szűrőfüggvény lefutását.

A képtartományban végrehajtott szűrés alapja a *konvolúció*. Az eljárás lényege, hogy minden világosságmódot helyettesítünk a képpont környezetéből vett konvolúciós szűrőmátrix által definiált súlyozott értékkel. A konvolúciós szűrő, tehát egy mátrix, mérete meghatározza az adott képpont figyelembe vehető környezetét (3×3 , 5×5 , stb.), míg a mátrix elemei ún. súlyozott együtthatóként funkcionálnak, hiszen a hozzátartozó képpont luminancia értékét ezzel a szorzóval veszi figyelembe. A mátrix sorait, oszlopait célszerű páratlan számra választani, mert ilyenkor úgy helyezhető a képpontokra, hogy középen lesz a vizsgált képpont. A konvolúciós szűrés folyamata a következő:

- az $N \times N$ méretű konvolúciós mátrixot egy kitüntetett eleménél fogva az adott képpontra „helyezzük”,
- a mátrix „alatt” lévő képpontok mindegyikének luminanciaértékét megszorozzuk a mátrix megfelelő elemével,
- az így kapott szorzatokat összeadjuk,
- az összeg úgy normalizáljuk, hogy az a 0-255 intervallumba essék,
- az így kapott szám adja a bemenő kép $P(i,j)$ képpontjához tartozó luminancia értéket szűrt kép $P'(i,j)$ pontjában.

Mindezt képlettel az alábbiak szerint írhatjuk le:

$$j'(a,b) = \sum_{i=-m}^{+m} \sum_{j=-n}^{+n} j(a+i, b+j) \cdot t_{ij} \quad (23)$$

ahol t_{ij} a $\underline{T}$ konvolúciós mátrix egy eleme, $j(a,b)$ a kvantálás után kapott képfüggvény (2.2. fejezet), $j'(a,b)$ a szűrt eredménykép. Ennek bemutatására álljon itt egy példa. Legyen a $j(a,b)$ függvény egy részlete a következő luminancia-mátrix (30. ábra):

14	11	13	14	14
12	10	23	15	11
12	17	50	48	20
14	15	32	16	18
13	16	21	18	18

30. ábra

Vizsgáljuk meg három konvolúciós mátrix ($\underline{T}_1$, $\underline{T}_2$, $\underline{T}_3$) hatását a 30. ábrán bekeretezett luminancia együtthatókra.

$$\underline{T}_1 = \begin{vmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix} \quad \underline{T}_2 = \begin{vmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{vmatrix} \quad \underline{T}_3 = \begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{vmatrix}$$

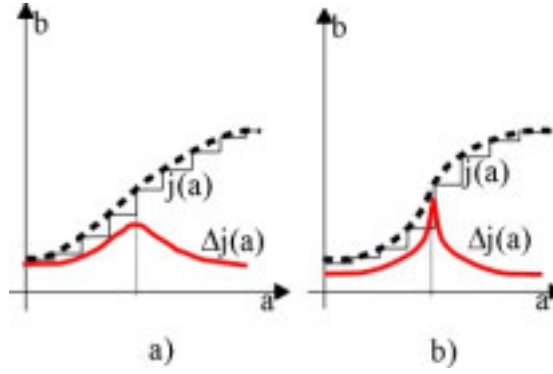
A bekeretezett luminancia-mátrix elemeit összeszorozva a konvolúciós (szűrő)-mátrix megfelelő elemeivel kapjuk a szűrt, új luminancia együtthatót, amely a középső (50 – kétpont-vonallal jelölve) értéket fogja helyettesíteni:

$$\begin{aligned} j'_1(a,b) &= 1/9*(1*10+1*23+1*15+1*17+1*50+1*48+1*15+1*32+1*16) = 25 \\ j'_2(a,b) &= 1/10*(1*10+1*23+1*15+1*17+2*50+1*48+1*15+1*32+1*16) = 28 \\ j'_3(a,b) &= 1/9*(1*10+2*23+1*15+2*17+4*50+2*48+1*15+2*32+1*16) = 31 \end{aligned}$$

Ezekből látható, hogy az 50-es luminancia érték jelentősen lecsökken, így a képnek ez a része elmosódottabbá válik. Látható, hogy ez a módszer a nagyon kiugró értékeket „lecsönkolja”, úm. tompítja a csúcsokat. Van ennek a módszernek egy nagy hátránya, az éleket is „elmossa”.

A képtartományban a zajszűrésnél említett súlyozott átlagolás helyett a differencia operátort alkalmazzuk, ezt pedig megfelelő konvolúciós szűrők alkalmazásával valósíthatjuk meg. A 31. ábra mutatja egy egyváltozós függvény ($j[a]$) differenciájának alakulását ($\Delta j[a]$) az élkiemelés előtt (a) és után (b). Egyváltozós függvény differenciája a következőképpen írható fel: $\Delta j(a) \equiv j(a + \Delta a) - j(a)$. A $\Delta j(a)$ -t differenciaoperátornak nevezzük. Élgyanús pontok ott vannak, ahol differencia-görbének ($\Delta j[a]$) szélsőértéke van.

A kétváltozós képfüggvények esetében az élek tetszőleges irányban elhelyezkedhetnek, ezért ha sorok mentén végeznénk az élkiemelést, akkor a vízszintes élek luminancia változását nem érzékelnénk. Ennek kiküszöbölésére kétváltozós differencia operátort alkalmazunk, amelynek leírására a gradienst használjuk. A gradiens ($\text{grad } j[a,b] = \frac{\partial j(a,b)}{\partial a} i + \frac{\partial j(a,b)}{\partial b} j$) mindig a függvény legnagyobb növekedésének irányába mutat és megadja az egységnyi távolságon bekövetkező változást a függvényértékekben (24).



31. ábra

$$\text{grad}[j(a, b)] = \sqrt{\left(\frac{\partial j}{\partial a}\right)^2 + \left(\frac{\partial j}{\partial b}\right)^2} \quad (24)$$

A parciális deriváltakat a differenciákkal helyettesítve a (25) összefüggést kapjuk.

$$\begin{aligned} \text{grad}[j(a, b)] &= \sqrt{[j(a, b) - j(a+1, b)]^2 + [j(a, b) - j(a, b+1)]^2} \cong \\ &\cong |j(a, b) - j(a+1, b)| + |j(a, b) - j(a, b+1)| \end{aligned} \quad (25)$$

Alkalmazzuk ezek után a (25) összefüggés második részét a 30. ábrán látható luminancia értékekre, például a $j(2,2)$ – szaggatott körrel jelölve – és a $j(2,3)$ – folytonos körrel jelölve – koordinátájú képpont-értékekre:

$$\text{grad}[j(2,2)] = |10-23| + |10-17| = 20; \quad \text{grad}[j(2,3)] = |17-50| + |17-15| = 35.$$

Mint látható, ahol a luminancia értékekben jelentős változás van ott a differencia operátor is nagyobb értéket ad, ez pedig élgyanús pontokat jelent. Azért csak élgyanús pontokat és nem éleket, mert lehet, hogy a gradiensben bekövetkezett változás eredhet zajból is.

Ennek a problémának a megoldására szolgálhat a Laplace operátor és más módszerek is. Az előbbi azon alapszik, hogy az éleknél a másodrendű parciális differenciálhányadosnak helyi szélsőértéke van. A Laplace operátort Laplace deltának is nevezik, és Δ -val jelölik (26).

$$\Delta[j(a, b)] = \frac{\partial^2 j(a, b)}{\partial a^2} + \frac{\partial^2 j(a, b)}{\partial b^2} \quad (26)$$

A differenciálhányados differenciahányadossal történő közelítése a digitális képfeldolgozásban a (27) összefüggéssel lehetséges.

$$\Delta[j(a, b)] = j(a+1, b) + j(a, b+1) + j(a-1, b) + j(a, b-1) - 4*j(a, b) \quad (27)$$

A (27) összefüggés mátrix formában is felírható. Vizsgáljuk meg három konvolúciós mátrix ($\underline{T}_1$, $\underline{T}_2$, $\underline{T}_3$) hatását a 30. ábrán bekeretezett luminancia együtthatókra.

$$\underline{T}_1 = \begin{vmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{vmatrix} \quad \underline{T}_2 = \begin{vmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{vmatrix} \quad \underline{T}_3 = \begin{vmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{vmatrix}$$

Alkalmazzuk ezt az összefüggést a 30. ábrán bekeretezéssel jelölt luminancia értékekre.

$$\begin{aligned} j'(2,2) &= |23+17+12+11-4*10| = 23 \\ j'(2,3) &= |50+15+12+10-4*17| = 19 \\ j'(3,3) &= |48+32+17+23-4*50| = 80 \end{aligned}$$

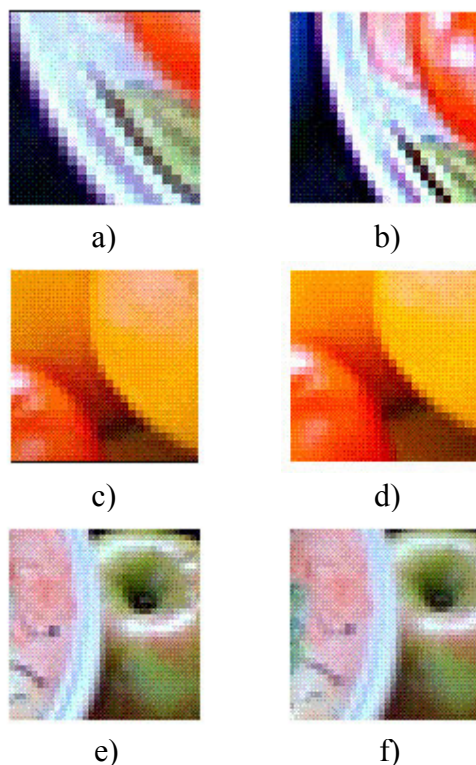
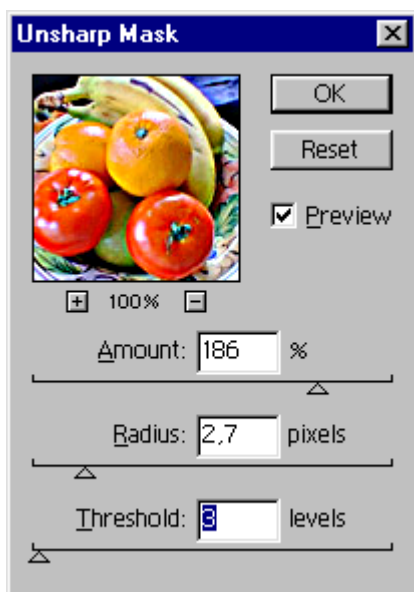
Ezekből látható, hogy az 50-es luminancia érték jelentősen megváltozott, így a képnek ez a része még kiemelkedettebbé válik. Látható, hogy ez a módszer a kevésbé kiugró értékeket „csúcsosítja”, ún. kiemeli a csúcsokat.

Mindezek után nézzük át a Photoshop 5.0 legfontosabb szűrőit, megjegyezve azonban, hogy csak azokat a szűrőket vizsgáljuk meg, amelyek szerepet játszanak a képjavításban. Fontos megjegyezni, hogy a Photoshop nagyon sok szűrőt tartalmaz (Filter menüpont), amelyek a képjavítás mellett képtorzításra is alkalmasak. Akinek nem elégségesek ezek a szűrők, bővíthetők is, ezeket külső plug-in-nek nevezzük (csak a Kai's emelném ki). A Photoshop-ban található képjavító szűrők a következők:

- **élesség- és kontrasztnövelő szűrők** (*Filter/Sharpen* almenü) a szomszédos képpontok közötti kontrasztot növelik, azaz a képet „élesítik”. A jobb megértés végett kezdjük először az *Unsharp Mask* opcióval, amely magában hordozza a *Sharpen*, *Sharpen More* és a *Sharpen Edges* szűrők tulajdonságait, azaz vagy a kép egészére hat, vagy pedig csak az éleknél. A 32. ábrán látható az a párbeszédablak, amely az *Unsharp Mask* parancs meghívásakor bejelentkezik. Az Amount szolgál a kép élesítésére, értéke általában 25-50% körüli érték, maximuma 500%, amit nagyon életlen képeknél tudunk csak kihasználni. A Radius szolgál az élek vastagságának a beállítására (0.1-250 pixel), a kis érték vékony, míg a nagy érték vastag éleket jelent. Szkennelt, majd képernyős megjelenítésre szánt képet 0.4-2.0 pixel közötti sugárértékkel kell megadni. A Threshold szolgál az élek felismertetésére, ui. az itt megadott érték (0-255) megadja két egymás melletti képpont közötti luminancia-különbséget. Ha például két egymás melletti képpont luminancia értéke 50 és 72, a Threshold pedig 11-en áll, akkor ezt élként fogja értelmezni, hiszen $72-50 = 22$ és az pedig nagyobb mint 11. Ellenben, ha a Threshold értékét 24-re állítom, akkor azt már nem érzékeli élként.

Nehéz általánosan jó értékeket megadni, ezért célszerű kísérletezni. Először az egyes paramétereket (Amount, Radius, Threshold) változtatni, majd azokat kombinálva. A 33. ábra a paraméterek egyedi változtatását mutatja be a 2. táblázat beállításainak megfelelően. Lehet ezeket a műveleteket külön-külön színcsatornákra is vonatkoztatni, ezáltal az élek mentén az árnyékokat lehet szabályozni.

A magasan áteresztő szűrő (*Filter/Other/High Pass*) – mint korábban azt láttuk – alkalmas az alacsony kontrasztú területeket elkülöníteni a magasaktól és ezzel szintén elősegíti az élkiemelést. Az itt megadott alacsony értékű Radius hatására jelentős különbséget tesz az alacsony és a magas kontrasztú képrészek között, míg a magas Radius érték hatása pont ezzel ellentétes.



32. ábra

33. ábra

Amount [%]	Radius [pixels]	Threshold [levels]	Ábraszám
15	1	0	a)
150	1	0	b)
15	1	0	c)
15	200	0	d)
40	1	0	e)
40	1	130	f)

2. táblázat

A kis Radius érték tehát minden nagy kontrasztú területet sötétszürkévé és minden kis kontrasztú területet pedig világos szürkévé változtat. A nagy Radius érték hatása ezzel pont ellentétes. Az így kapott képünkben azonban nem sok örömünk telik (34. ábra), hiszen mint láttuk ez a szűrő úm. kilopja a képből a színeket. Ezt a hátrányt kiküszöbölhetjük az által, ha csak bizonyos színesatornákon (itt most az R csatornát szűrtem) hajtjuk végre *High Pass* szűrést (35. ábra).

Az éllel kapcsolatosan néhány beépített automatikus művelet is létezik, például a szegélyek megkeresésére a *Find Edges* (Filter/Stylize menü) (36. ábra), a kontúrok „kihúzására” a *Trace Contour* (Filter/Stylize) (37. ábra).

- **élesség- és kontrasztsökkentő szűrők** (Filter/Blur almenü) az előző művelettel pont ellentétes hatást fejt ki, hiszen csökkentik a luminancia különbséget két egymásmelletti

képpont között, ezáltal csökkentik a kontrasztot. Szerepük a képszerkesztésben kevésbé jelentős, ezért néhányat említenék csak meg:

- élesség csökkentése Gauss eloszlási görbe mentén (*Gaussian Blur*),
- „életlenítés” (*Blur*), még több életlenítés (*Blur More*),
- megadott irányokba végrehajtott életlenítés: „lineáris mozgás” (*Motion Blur*), „körkörös” mozgás (*Radial Blur*),
- „életlenítés” a határérték megváltoztatásával (*Smart Blur*) csökkenthető a beszkenelt képünk szemcséssége, foltossága (hasonlóképpen működik mint azt a 2. táblázatban láttuk – *Threshold*),
- képünk átméretezésével is csökkenthetjük a kontrasztosságot, ha az *Image/Image Size* parancs meghívása után a mérettartás (*Resample Image*) ill. a kép újraalkotási módjánál a *Bicubic* opciót választjuk ki.



34. ábra



35. ábra



36. ábra



37. ábra

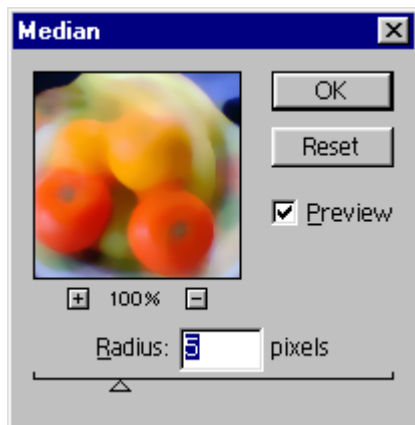
- **zajkezelő szűrők** (*Filter/Noise* menü) általában az oda nem illő színek eltávolítására szolgál, de van amikor meg zaj adunk hozzá a képhez, hogy valamilyen érdekes kompozíciót állítsunk elő. A zaj hozzáadására a *Add Noise* parancs szolgál. Zaj eltávolítására szolgál a *Despeckle*, *Dust&Scratches* és a *Median* parancsok szolgálnak. A *Despeckle* parancs képünket homályosítja, miközben megtartja annak éleit. A *Sharpen Edges* szűrő ellentétéként hat és úgy működése a következő:

- élek keresése *Unsharp Mask* –nál leírtak szerint (*Threshold* = 5),
- az éleket figyelmen kívül hagyva életlenítés *Blur More* –nál leírtak szerint.

A *Median* szűrő úgy zajtalanít, hogy minden képpont színét a szomszédos képpontok színeinek átlagolásával határozza meg. A *Radius* érték megadásával határozzuk meg azt a kört, amelyben lévő pontokat vonja be az átlagolásba, kihagyva a kiugró szélsőségeket. A medián a *Radius* sugarú körön belül adja a középpont színét (38. ábra). A *Dust&Scratches* szűrő szolgál szkennelt képekről a karcolások, hajszálak, porszemcsék, moiré, stb. eltávolítására. Ezek közül egyedül a moiré szolgál magyarázatra. Amennyiben könyvből vagy újságból szkennelünk a folyamatos tónusú képek rácspontokként jelennek meg, amelyek interferálnak a felbontással, ezáltal létrehoznak egy ismétlődő zavaró mintát. Ezt nevezzük moirénak, amelynél a *Threshold* értéket célszerű 60 alá választani (39. ábra). Lehet, hogy a moiré nem minden színcsatornát érint egyformán, ilyenkor célszerű színcsatornánként szűrni.

- **roncsoló, torzító szűrők** különleges hatások keltésére alkalmas, a teljesség igénye nélkül álljon itt néhány:
 - mozaikosítás, pixelesítés (*Pixelate*),
 - él- ill. kontúrkiemelés (*Stylize*),
 - képpontok szín- és alaktulajdonságainak megváltoztatása, eltolása (*Distort*),
 - fényhatások létrehozása (*Render*),
 - művészeti hatások létrehozása (*Artistic*),
 - ecsetvonásszerű hatások létrehozása (*Brush Strokes*),

- mintázat készítése (*Texture*),

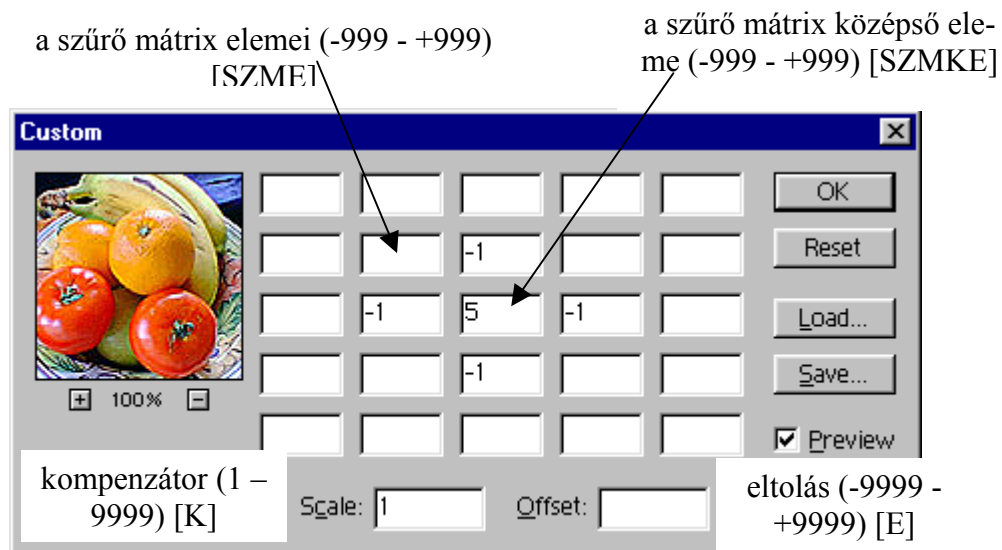


38. ábra



39. ábra

- **a felhasználók által definiált szűrők** (*Other/Custom*) működése a konvolúciós szűrés elvén alapszik (30. ábra alapján). A felhasználó maximum 5*5-ös szűrőmátrixot definiálhat, amelyet a 40. ábra mutat.



40. ábra

Jelöljük a vizsgált képpont luminancia értékét VPL –lel, a szomszédos képpontok luminanciáját SZPL –lel, míg a vizsgált képpont módosított luminancia értékét VPL’ –vel, akkor az a következő összefüggés alapján határozható meg:

$$VPL' = [(VPL * SZMKE) + (SZME1 * SZPL1) + (SZME2 * SZPL2) + \dots] / K + E \quad (28)$$

A kompenzátor [K] képes a szűrőmátrix helyi értékeinek általános érvényű módosítására, míg az eltolás a luminancia értékek lineáris megváltoztatására szolgál. A pozitív értékű eltolás világosít, míg a negatív sötétít. A szűrő mátrix lehet szimmetrikus (víz-

szintesen és függőlegesen is szimmetrikus) és aszimmetrikus. Nézzünk néhány speciális esetet:

- élesítés – SZMKE>0 és SZME<0,
- lágyítás – SZMKE>0 és SZME>0,
- lágyítás élkiemeléssel – SZMKE<0 és SZME>0.

Ha a szűrőmátrix értékeinek összegét csökkentjük anélkül, hogy SZMKE értékével azt kompenzálnánk a kép sötétedik, míg ellenkező esetben világosodik. Ezek a hatások a Scale értékével kompenzálhatók.

2.3. Képtömörítési algoritmusok

A digitális képek tömörítése a redundancián alapszik. A redundancia fogalmának értelmezéséhez szükségünk van az alábbi fogalmak tisztázására: információtartalom, entrópia.

Egy közleményben előforduló jelsorozat előfordulási valószínűsége különböző lehet. Például álljon jelsorozatunk a következő elemekből: $A_1, A_2, A_3, \dots, A_n$, amelyek előfordulási valószínűsége rendre: $P_1, P_2, P_3, \dots, P_n$, ezek segítségével megadható az A_i jelsorozat *egyedi információtartalma*:

$$I(A_i) = -\log_2 P_i \quad (29)$$

A (29) összefüggés értelmében minél ritkábban fordul elő egy elem (kisebb az előfordulási valószínűsége), annál nagyobb az általa hordozott információ tartalma. Egy jelsorozat átlagos információtartamát *entrópiának* nevezzük:

$$E = \sum_{i=1}^n P_i \cdot I(A_i) \quad (30)$$

Az entrópia lehetővé teszi, hogy egy adott jelsorozat információtartalmát összehasonlítsuk egy másik jelsorozatával. Az entrópia értéke akkor a maximális, ha a jelsorozatot alkotó összes elem előfordulási valószínűsége megegyezik ($P_1 = P_2 = P_3 = \dots = P_n$).

Jelölje E egy tetszőleges n elemű jelsorozat átlagos entrópiáját és $E_{\max}$ az n elemű jelsorozat entrópiáinak legnagyobb értékét. Ezek figyelembe vételével a *redundancia* (R) az alábbi összefüggés szerint határozható meg:

$$R = \frac{E_{\max} - E}{E_{\max}} \quad (31)$$

A (31) alapján megállapíthatjuk, hogy közlemény akkor redundáns („terjengős”), ha az értelmezéséhez szükséges elemeken túl sok felesleges elemet is tartalmaz. A képfeldolgozásban előforduló leggyakoribb redundanciák a következők lehetnek: [4]

- kódolási redundancia
Például egy fekete-fehér képpont kódolásához 1 bit helyett 8 bitet használunk.
- képi redundancia
Például egy azonos színű, nagy méretű objektum minden egyes képpontját tároljuk, annak geometriai jellemzői helyett. Ez elvezethet a raszteres és a vektoros képtípusok különbözőségéhez. Ezzel a lehetőséggel fogunk majd találkozni a digitális animációknál és a digitális videóknál, amikor az egymást követő képek csak kis mértékben kü-

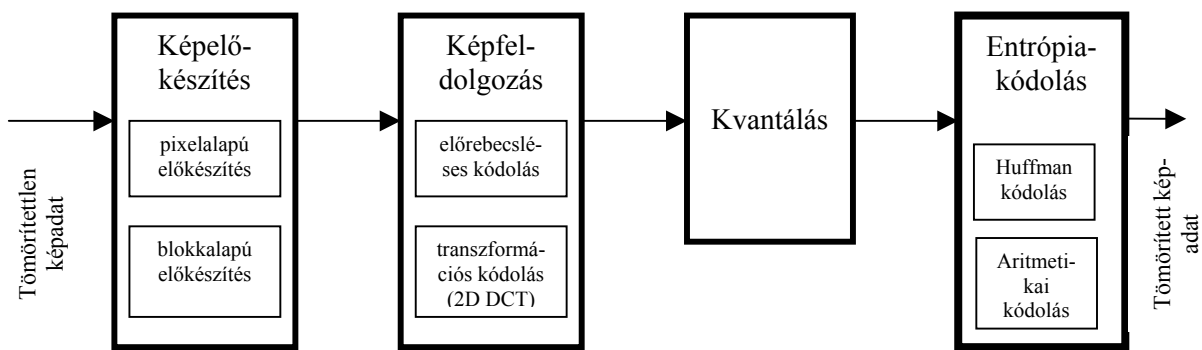
lönböznek egymástól és nem fogunk minden képkockát külön tárolni, hanem figyelembe vesszük a közöttük lévő eltéréseket.

- pszichovizuális redundancia
Például a kép azon minőségbeli eltéréseit, amelyet az emberi szem már érzékelni nem tud nem tároljuk le.

Az első két eljárást az ún. GIF képeknél, míg a harmadikat a JPEG képeknél alkalmazzák. Nézzük át röviden ezeket az algoritmusokat.

2.3.1. A JPEG eljárás

1987-ben az ISO (International Standards Organization) és a CCITT (Committee on International Telephon and Telegraph) szervezetek egy olyan munkacsoportot hoztak létre, amelynek egy hatékony, általános érvényű állókép tömörítési szabványt kellett kidolgoznia. Ezt az eljárást 1992-ben az ISO szabványosította JPEG néven, amely olyan népszerűvé vált, hogy kifejlesztettek mozgóképek tömörítésére alkalmas eljárást is (MJPEG). Előnye a JPEG-nek, hogy az algoritmus független a színmélységtől és a kép nagyságától. Az eljárás folyamatát a 41. ábra mutatja.



41. ábra

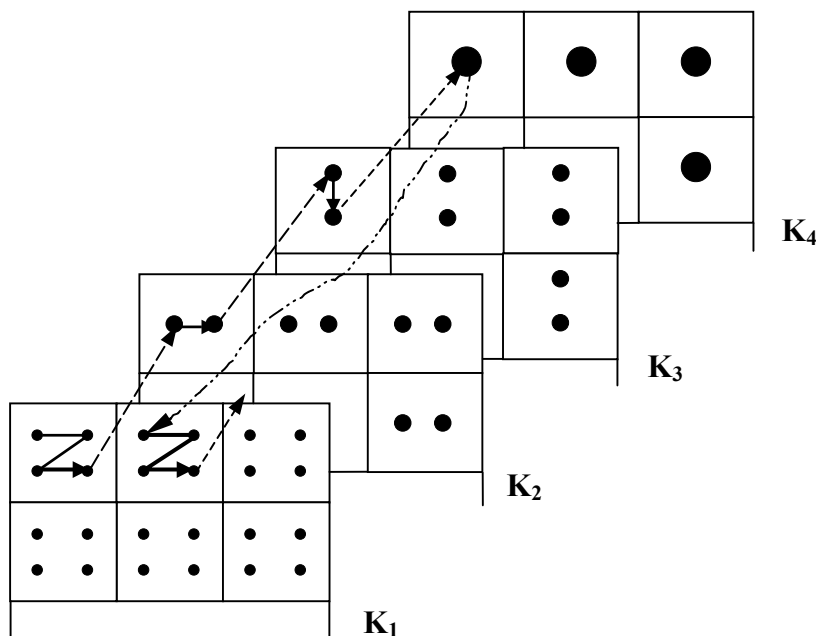
A JPEG szabvány a színösszetevőket egymástól függetlenül kezeli, ezért a tömörítést mind a három összetevőre végre kell hajtani. A látható színek legnagyobb része előállítható RGB színrendszerben, de nem igazán hatékony, mert a komponensek kódolására azonos sáv szélesség szükséges. Közismert, hogy az emberi látás szempontjából a zöld fontosabb szín mint a kék, ezért célszerű a színösszetevőket fontosabb és kevésbé fontos összetevőkre bontani. Így született az YUV színrendszer, amelyben az Y jelenti a luminanciát, az U ill. V a krominancia együtthatókat (kék ill. vörös színkülönbségi jel). Az RGB rendszerből az YUV színrendszerbe történő konvertálás az alábbi képletek alapján történik:

$$\begin{aligned}
 Y &= 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \\
 U &= (B - Y) \cdot 0.493 \\
 V &= (R - Y) \cdot 0.877
 \end{aligned}
 \tag{32}$$

Az Y összefüggésben az együtthatók összege 1. Mivel az U, V komponenseknek nincs luminancia tartalma, ezért csak az adott jel színéről hordoznak információt. Vizsgáljuk meg ezek után a tömörítés folyamatát részletesen:

a) képelőkészítés

A képelőkészítés egyrészt az RGB $\rightarrow$ YUV színrendszer transzformációt, másrészt pedig a tömörítendő kép adategységekre bontását jelenti. Az előbbit az előbb már érintettem, az utóbbi kétféleképpen oldható meg. A pixelalapú kódolás pixelenként kódol, a blokk alapú pedig a képpont sereget 8×8 -as egységekre osztja, és ezen hajtja végre a 2D DCT eljárást. A 8×8 képpontból álló adategységet blokknak nevezzük. A 42. ábrán egy pont egy adategységet jelöl. A transzformációs kódolás előtt a képet képsíkokra bontják, majd ezeken hajtják végre az adategységekre, blokkokra bontást. A JPEG támogatja a kép maximum 255 képsíkra bontását, azonban általában csak három (YUV ill. RGB) vagy négy képsíkot használnak (YRGB). Az adategységek sorrendisége kétféle lehet: nem egymásba ágyazott, egymásba ágyazott. A nem egymásba ágyazott felépítés esetén az adategységek szervezése képsíkonként (először az R, majd G, ill. B következik) történik. A képelőkészítés egy képsíkon a bal felső sarokból soronként indítva balról jobbra haladva történik. A JPEG inkább az egymásba ágyazott képelőkészítést alkalmazza, amely a négy képsík „összszeláncolását” jelenti (42. ábra). Egy ilyen adategységet Minimum Coded Unit –oknak (MCU) nevezzük. Az egyes MCU –k maximum 10 adategységet tartalmazhatnak. Az egyes képsíkok felbontása lehet azonos vagy különböző, ez utóbbit az emberi látás színszátartósságai indokolják (29. táblázat). A JPEG ez utóbbit használja. Az egyes képpontokat (mintavételi értékeket) 8 biten kódoljuk, ezért azok 0 - 255 közötti értéket vehet fel.



42. ábra

b) képfeldolgozás

A képfeldolgozás a 2D DCT szerinti transzformációs kódolást jelent. Ennek elméleti háttérét a 2.2.1. fejezetben részletesen áttekintettük. A 8×8 -as adategységek transzformációja a (3.22) alapján a következő összefüggés szerint történik,

$$krs = \frac{1}{4} \cdot \alpha_r \cdot \beta_s \cdot \sum_{n=0}^7 \sum_{m=0}^7 Dmn \cdot \cos\left(\pi(2n+1) \cdot r \cdot \frac{1}{16}\right) \cdot \cos\left(\pi(2m+1) \cdot s \cdot \frac{1}{16}\right) \quad (33)$$

amelyben $N = M = 8$; az $\alpha_r = \beta_s = \frac{1}{\sqrt{2}}$,

ha $r = s = 0$ és az $\alpha_r = \beta_s = 1$, ha $r \neq 0$ valamint $s \neq 0$:

Az egyes képpontok értéktartománya a transzformálás következtében a $0 - 255$ helyett $-128 - +127$ lesz. Egy 8×8 -as adategységre tehát ezt a transzformációt 64-szer kell elvégezni, ami 64 db k_{rs} együtthatót eredményez.

A kép későbbi rekonstruálása érdekében alkalmazzuk az inverz 2D DCT eljárást (34),

$$D_{mn} = \frac{1}{4} \cdot \sum_{s=0}^7 \sum_{r=0}^7 \alpha_r \cdot \beta_s \cdot k_{rs} \cdot \cos\left(\pi(2n+1) \cdot r \cdot \frac{1}{16}\right) \cdot \cos\left(\pi(2m+1) \cdot s \cdot \frac{1}{16}\right) \quad (34)$$

amelyben az $\alpha_r = \beta_s = \frac{1}{\sqrt{2}}$, ha $r = s = 0$ és az $\alpha_r = \beta_s = 1$, ha $r \neq 0$ valamint $s \neq 0$.

Visszatérve a 2D DCT transzformációra megjegyzem még, hogy a (33) összefüggéssel nem könnyű számolni, ezért helyette az ún. gyors Fourier transzformációt (FFT) alkalmaz-
zák, amely már a programkészítés alapját is képezheti.

Vizsgáljuk meg, most egy adategység (blokk) szerkezetét, amelyben szereplő k_{rs} együt-
hatók elhelyezkedését a 43. ábra mutatja.

k_{00}	k_{01}	k_{02}	...			...	k_{07}
k_{10}	k_{11}	k_{12}	...			...	k_{17}
k_{20}	...					...	k_{27}
k_{30}	...					...	k_{37}
k_{40}	...					...	k_{47}
k_{50}	...					...	k_{57}
k_{60}	...					...	k_{67}
k_{70}	...					...	k_{77}

43. ábra

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

44. ábra

Megvizsgálva ezeket az együtthatókat az alábbi megállapításokat tehetjük:

- a k_{00} a zérus frekvenciának felel meg, amely az adott adategység szín alaptónusát adja meg. Nevezik „*egyenáramú*” komponensnek is (DC). A többi együttható zérustól különböző frekvenciákat adja, ezért „*váltakozóáramú*” komponensnek is nevezik (AC).
 - a k_{70} jelenti a függőleges irányban fellépő legnagyobb frekvenciát,
 - a k_{07} adja a vízszintes irányban fellépő legnagyobb frekvenciát,
 - a k_{77} jelenti mind vízszintes, mind pedig függőleges irányban a frekvencia maximumát.
- A legtöbb kép kevés éles átmenetet, kontúrt tartalmaz, ezért sok AC együttható zérus, vagy zérus közeli értéket képvisel. Ez a tulajdonság fogja majd az entrópiakódolás alapját képezni.

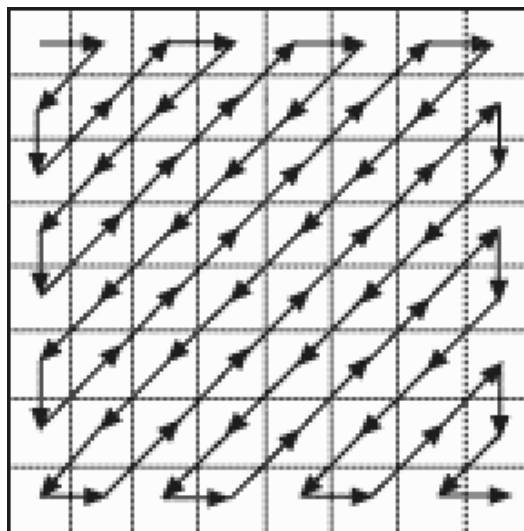
c) kvantálás

A kvantálás a transzformáció után kapott együtthatók leképezését jelenti megadott diszkrét értékek alapján (2.2.1. fejezet). Mielőtt ezt végrehajtanánk, a k_{rs} együtthatókat fontosságuknak megfelelően súlyozzuk. Ennek okát már többször is az emberi látás sajátosságai-
ban jelöltük meg. A súlyozást egy 64 elemű mátrixszal történő osztás ill. az így kapott eredmény kerekítése jelenti. A „súlyozó” mátrix megfelelő elemével osztjuk a k_{rs} mátrix megfelelő elemét. Minél nagyobb a „súlyozó” mátrix egy eleme, annál kisebb féle

eredményt kaphatunk kvantálás után. Ne felejtjük el, hogy a k_{rs} mátrix értékkészlete -128 és $+127$ közé esik. A 44-45. ábra a luminanciához ill. a krominanciához használatos „súlyozó” mátrixot mutatja. A két táblázatból az is következik, hogy a krominancia komponensekre sokkal nagyobb tömörítés érhető el (kisebb a tömörítési aránytényező). A kvantáláshoz ill. a visszakvantáláshoz ugyanaz a mátrix használatos.

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

45. ábra



46. ábra

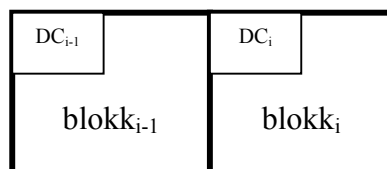
A kvantálással kapott mátrixot megvizsgálva az alábbi következtetéseket vonhatjuk le:

- az együtthatók túlnyomó többsége zérus, vagy ehhez közeli szám,
- a „súlyozó” mátrix ügyes megválasztásával a nem zérus együtthatók a mátrix bal felső sarkában helyezkednek el,
- a jobb alsó sarok felé haladva egyre több együttható adódhat zérus értékűre.

Mindezen tulajdonságok jó alapot jelentenek az entrópiakódoláshoz.

d) kódolás

A kódolás két rész műveletre bontható: AC együtthatók kódolása, DC együtthatók kódolása. Az AC együtthatók kódolása „fontossági sorrendben” (először a kisfrekvenciás, majd a nagyfrekvenciás együtthatók) történik. A kisfrekvenciás együtthatók általában nagy értékek, míg a nagyfrekvenciások általában zérusok. Ezt a cikk-cakk vonalat mutatja a 46. ábra. A DC együtthatók kódolása a szomszédos adategységek (blokkok) közötti különbség ($\text{Eltérés} = DC_i - DC_{i-1}$) alapján történik (47. ábra). Ezáltal is jó hatékonyságú tömörítés érhető el.



47. ábra

Az entrópiakódolás a 1.3.1. fejezetben vázolt Huffman és aritmetikai kódolás szerint történik.

A képfelépítés szempontjából kétféle kódolást különböztetünk meg: szekvenciális, progresszív. A *szekvenciális eljárás* során az egész képet egyetlen átfutás során, felülről lefelé haladva soronként kódolják, ill. dekódolják. A szekvenciális kódolás mintavételi értéke 8, 12 bit/képpont.

A *progresszív eljárás* esetén először egy életlen, durva képet, majd egyre finomabb struktúrájú képrész kerül kódolásra, ill. dekódolásra. A progresszivitás kétféleképpen érhető el: spektrálisan, szukcesszíven. A spektrális progresszív eljárás során először a kis, majd pedig a nagy frekvenciájú DCT együtthatókat kódolják, ill. dekódolják. A szukcesszív eljárás során az összes együttható átvitelre kerül egy átmeneti tárolóba, majd a feldolgozás sorrendje a magas bithelyiértékektől kezdve az alacsonyok felé haladva történik. A kódolás mintavételi értéke itt is 8, 12 bit/képpont.

Mint arról korábban már esett szó a JPEG eljárásnál a pszichovizuális redundanciát használják fel a tömörítés alapelveül. A tömörítés mértéke aszerint változhat, hogy a képminőségnek milyen követelményeket kell kielégítenie:

- 1:20 – 1:30 tömörítési arány: az emberi érzékeléssel ne lehessen észre venni a minőségromlást.,
- 1:200 tömörítési arány: a kép lényeges részei felismerhetőek legyenek.

Ezt az értéket a legtöbb képszerkesztő programban be lehet állítani, ezt mutatja a 48. ábra a Photoshop program kapcsán.



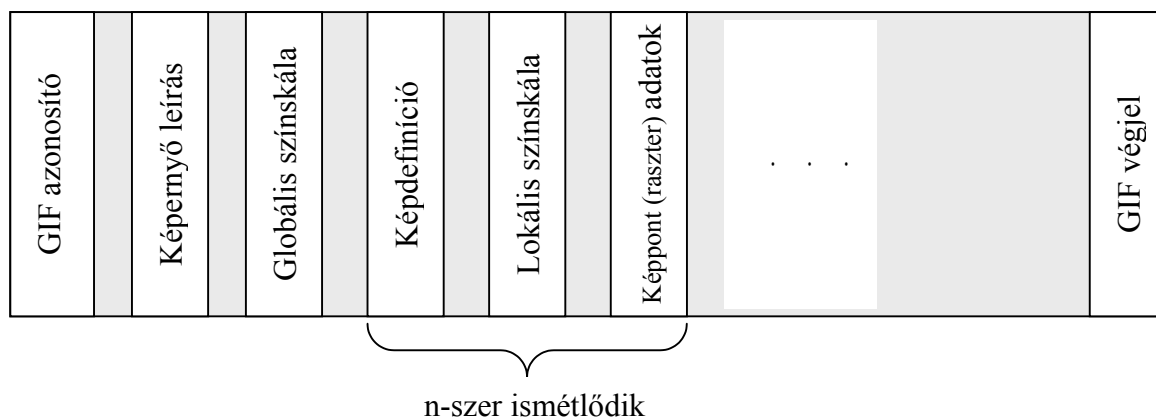
48. ábra

A kép tömörítés mértékét a *Quality* opcióval állíthatjuk be. Az alacsonyabb érték kisebb állomány méretet, de nagyobb képtorzulást eredményez. Ajánlatos 5-ös, ill. 6-os értéket választani. A Web-re szánt képeknél fontos ismerni a *Format Options* szerepét:

- Az ún. alapvonalas általános (*Baseline „Standard”*) opció a soros megjelenítést támogatja, azaz a képet sorról-sorra rajzolja ki a monitorra. Ezt támogatja az előbb említett ún. progresszív JPEG eljárás.
- Az ún. alapvonalas optimalizált (*Baseline Optimized*) az előző jellemzőkön túlmenőleg még javított Huffman kódolást is alkalmazunk.
- A progresszivitás mértékét a *Progressive* opciónál az átfutások számával határozzuk meg. Magasabb érték a kezdeti kép gyors megjelenítését eredményezi, de a sorozatos frissítés miatt a teljes kép kirajzolása végül több időt vesz igénybe. Javasolt a *Scans* értékét 3-4 közé választani.

2.3.2. A GIF eljárás

A raszteres képek Interneten elterjedt veszteségmentes képtömörítési eljárása a GIF (Graphics Interchange Format) képformátum, amelyet a Compuserve fejlesztett ki (1987). A GIF algoritmusú kódolás alapja a korábban említett kódolási és képi redundancia. A képen azonos jellemzőkkel bíró ún. homogén foltokat kódolhatjuk a geometriai adatokkal és 1 pixel jellemzőivel, ahelyett hogy az összes pixelről tárolnánk az összes információt. Az első GIF algoritmus csak a 8 bites színkódolást tette lehetővé, most már ezt 24 bitesre egészítették ki. Minden pixelt a színértéke jellemez, amelyet az RGB alapszínek 0-tól 255-ig terjedő skálán ábrázolt mértéke határoz meg. Az egyes képpontok helyzetét nem kell megadni, mivel a leírásban egymás után következnek, soronként balról jobbra haladva. Ez az eljárás hasonlatos a bittérképes tároláshoz (bmp). A GIF állomány szerkezetét a 49. ábra mutatja.



49. ábra

A GIF formátum egyik előnye, hogy a megjelenítést a grafikus kártya színmélységéhez és felbontásához lehet igazítani. Ez például a színskálák definíciói révén valósítható meg, vagyis a hardver által meghatározott optimális színmélységet kell a grafikához alapul venni. Ezzel a módszerrel a különböző grafikus kártyák nagyobb színhűséggel jelenítik meg a nagyfelbontású színes képeket. A 49. ábrán látható képernyő leírás megadja a képernyő magasságát, szélességét, a képernyő háttérszínét, a színmélységet, a globális színskála meglétét. A *képernyő leírás*ban megadott logikai képméret (magasság * szélesség) nagyobb is lehet, mint a valóságos képméret. Az ilyen hardver függő képábrázolás támogatja a kép ún. görgetős (scroolling) megjelenítését.

Az olyan képeknél, amelyeket egy bizonyos grafikus kártyával állítottak elő, de egy másikon szeretnék megjeleníteni, fontos szerepe lehet a valódi színhűség visszaadásában a *globális színskálának*. A globális színskála bejegyzéseinek száma $2^{\text{bitek száma/pixel}}$. Minden bejegyzés három bájtos, egy-egy bájt a színén alapszínek (R, G, B) arányát jelzi egy 0-255 skálán. Például a fehér színt a 255, 255, 255, míg a feketét a 0, 0, 0 számhármasság jelöli. A képernyőn megjelenő hardvertől függő kép minden pixele csak olyan színt vehet fel ami ebben a globális színskálában definiálva van.

A GIF állomány következő része a *képdfiníció* határozza meg a kép méreteit pixelben: kép szélessége, magassága, a kép távolsága a bal oldali képkerettől, a kép távolsága a felső képkerettől, globális színskála léte. A GIF állomány egyidejűleg több képet is tartalmazhat, amelyeket a „” ASCII kódja választ el.

A *lokális színskála* szerkezetileg megegyezik a globálissal, azzal a különbséggel, hogy bejegyzéseinek számát a bitek pixelenkénti száma adja. A lokális színskála a következő képelválasztó jelig érvényes, a következő képre vagy a globális színskála lesz érvényes, vagy pedig újabb lokális színskálának kell következnie.

A GIF következő része a *képpont (raszter) adatok*, amelyek tömörített adatokat tartalmazó blokkok sorozatából állnak. A tömörítés az ún. LZW (Lempel-Ziv & Welch) algoritmus szerint történik, amelyet 1984-ben fejlesztettek ki. A tömörítés folyamata a következő:

- A kép átalakítása pixelszín-értékek sorozatából karakterek sorozatává (ASCII).
- A kódolási hossz rögzítése bitben.
- LZW tömörítés. A karaktersorozat elemeinek hozzárendelése az ún. tömörített kódokhoz (LZW kódtábla alapján).
- Az előző pontban kapott LZW kódsorozat 1 bájt méretűvé szegmentálása.
- A bájtok blokkokba „csomagolása”.

3. MOZGÓKÉPFELDOLGOZÁS ÉS TÖMÖRÍTÉS

3.1. Videotechnikai alapfogalmak

Mielőtt rátérnénk a képdigitalizálás folyamatára, vizsgáljuk meg néhány, az analóg videotechnikában is használatos alapfogalmat: színes tv-jelek, színes videojel-rendszerek, képfelépítési rendszerek.

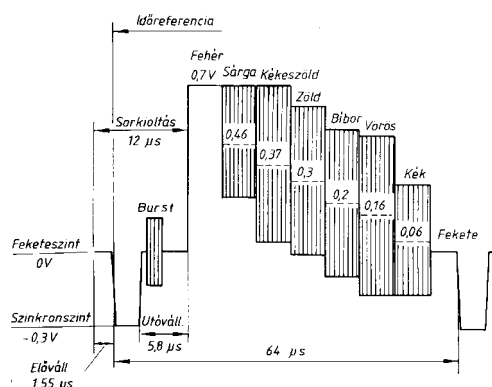
3.1.1. Színes tv-jelek

Amennyiben a színes tv-technika az RGB színrendszeren alapulna, úgy a tv-nek három képcsöve lenne. Igaz ugyan, hogy az összes tv-rendszer ezt a három színösszetevőt használja fel a képi információ előállításához, továbbításához, de a színinformáció előállítása, továbbítása racionálisabban, kódolt formában történik.

A videokamera az alapszíneknek megfelelő (R, G, B) csatornákkal rendelkezik. A megvilágított tárgy képét reprezentáló fényinformáció egy színszűrőn (színfelbontó rendszeren) keresztül a három felvevőcsőbe kerül, amely vörös, zöld, kék színekre bontja azt fel. A felvevőcsövekben a fényinformáció elektromos jelekké alakul át. A kamera objektívjét egy fehér vagy szürke (akromatikus) felületre irányítva a vörös, a zöld és a kék információnak megfelelő elektromos jelek az ún. gammakorrekció és erősítés után azonos amplitúdójúak lesznek. Ezt az állapotot a három színcsatorna erősítésének beállításával idézzük elő, mivel a felvevőcsövek kimeneti feszültsége kismértékben különbözik egymástól. A szintek kiegyenlítését fehéregyensúly-állításnak nevezzük. A fehéregyensúly beállítás az emberi szem azon tulajdonságán alapul, miszerint a tárgyak színét nem a valósághű érzékelésének tartjuk természetesnek. Színüket a reprodukálás során is olyannak szeretnénk látni, mintha a tárgyakat természetes nappali fény világítaná meg. Ily módon a helyes színvisszaadás csak akkor egyezik meg a valósághű képpel, ha a megvilágítás az ún. referenciafehérrel történt. Ezért a megvilágítások során a 3200 K színhőmérsékletű lámpákat kell alkalmaznunk. Megváltozott megvilágítási viszonyok során a fehéregyensúlyt is után kell állítani.

A színcsatornákból kapott három azonos szintű jelet a szem spektrális érzékenysége ill. az állandó fényssűrűség elve alapján a következő súlyozással összegezzük: 30% vörös, 59% zöld, 11% kék (37). Ily módon összegezett komponensek adják a világosságjelet (luminanciajelet-Y). A színes képi információ átviteléhez három adatra van szükség ezért két ún. színkülönbségi jelet visznek még át (R-Y, B-Y).

A világosságjel és a világosságjelre szuperponált, színkülönbségi jellel modulált színsegédvívők alkotják az összetett színes videojelet. Ezt kiegészítik a sor- és képszinkronjelekkel, amelyek lehetővé teszik, hogy a monitor képernyőjén futó elektronsugár szinkronban mozogjon a kamera felvevőcsövét letapogató elektronsugárral. E kötött szinkronkapcsolat biztosítja, hogy a kamera képpontjai egységes képet alkossanak a képernyőn. A horizontális és vertikális szinkronjeleken kívül - amelyek tehát az elektronsugár eltérítését szinkronizálják - az összetett videojel tartalmaz ún. színszinkronizáló jeleket (burst) is. Ennek szerkezetét a PAL szekvenciára vonatkoztatva a 50. ábra mutatja. [5]



50. ábra

3.1.2. Színes videojel-rendszerek

a.) NTSC rendszer

A világon először bevezetett színes tv-rendszert először az USA-ban dolgozták ki (National Television System Committee). Az 1953-ban bevezetett rendszerben a színinformáció átvitelét az ún. I és Q jeleket viszik át. Az I és Q jelek bevezetését megelőzően vizsgálatokat végeztek az emberi szem színfelbontó képességének vizsgálatára. Ezek értéke a következő:

$$\begin{aligned} I &= 0.74(R-Y) - 0.27(B-Y) \\ Q &= 0.48(R-Y) + 0.41(B-Y) \end{aligned} \quad (35)$$

A színinformáció átvitelére egyetlen ún. színsegédátvívőt alkalmaznak, amely a sorfrekvencia 228-szorosa. Az NTSC rendszer a színinformáció átvitelére az ún. kvadratúramodulációt használja, amely szerint az egyetlen színsegédátvívőt amplitúdóban modulálják a két színkülönbségi jellel (R-Y, B-Y) oly módon, hogy a vörös és a kék színkülönbségi jellel modulált segédvívő részek között pontosan 90°-os kötött fáziskapcsolat létesüljön. Ez a nem túl egyszerű eljárás biztosítja, hogy a két színkülönbségi jel nem "zavarja" egymást a kódolás és dekódolás során. Az NTSC rendszer képfelbontása 525 sor, míg képváltási sebessége 29.97 képkocka/sec.

b.) SECAM rendszer

A Henry de France által 1957-ben kidolgozott eljárás gyökeresen különbözik az amerikaiak által alkalmazott NTSC rendszertől, hiszen frekvenciamodulációval közvetíti a szín-

különbségi jeleket és nem egy hanem két színsegédvívót (vörös - 4.40625 MHz, kék - 4.25 MHz) használ. A világosság ill. a színkülönbségi jelek értékei a következők:

$$\begin{aligned} Y &= 0.3R + 0.59G + 0.11B \\ D_R &= -1.9 \cdot (R - Y) \\ D_B &= 1.5 \cdot (B - Y) \end{aligned} \quad (36)$$

A SECAM rendszer képfelbontása 625 sor (Franciaországban alkalmaznak 819 sorost is), míg képváltási sebessége 25 képkocka/sec.

c.) PAL rendszer

Az 1962-ben bevezetett PAL (Phase Alternation Line) rendszer az NTSC rendszer módosított változata, miszerint a vörössel modulált segédvívó fázisát soronként 180°-kal változik. Mindez hozzájárul a kép minőségi jellemzőinek javulásához.

A világosság ill. a színkülönbségi jelek értékei a következők:

$$\begin{aligned} Y &= 0.3R + 0.59G + 0.11B \\ U &= 0.877 \cdot (R - Y) \\ V &= 0.493 \cdot (B - Y) \end{aligned} \quad (37)$$

A Magyarországon is alkalmazott eljárás sorváltási frekvenciája 15.625 kHz, képváltási frekvenciája 25 Hz, míg felbontása 625 sor. A 3:4 arányú téglalap alapú képnél soronként 834 képpont keletkezik, így egy kép átviteléhez $625 \cdot 834 = 521500$ képpont tartozik (a 625 sor felrajzolása 40 ms-ig tart). Figyelembe véve a 25 kép/sec.-os képváltási frekvenciát ez másodpercenként közel 13 millió képpontot jelent.

3.1.3. Képfelépítési módok

A mozizásra és videózásra a szem recehártyájának és idegsejtjeinek viszonylagos lassúsága miatt van lehetőség. A szem recehártyájára vetített kép az idegsejteket impulzusok generálására készíti, amelyeket azután az agy kiértékel. Amennyiben a kép hirtelen megváltozik, úgy eltart egy kis ideig, amíg az idegsejtek feldolgozzák az információt. Amennyiben a képváltások kellő sebességgel követik egymást, úgy a látósejtek tehetetlensége miatt az egyes képek egymásba folynak, tehát a szem mozgást lát, holott csak állóképek követik egymást. A televíziózásnál alkalmazott (PAL) 25 kép/sec.-os képváltási (képfelépítési) sebességnél úgy javítják a néző "mozgás illúzióját", hogy 50 félképet alkalmaznak másodpercenként. Elérkeztünk tehát oda, hogy a kétféle képfelépítési rendszer terjedt el, egyrészt a váltott soros (interlaced), másrészt pedig a folyamatos (non-interlaced) képfelépítési módszer.

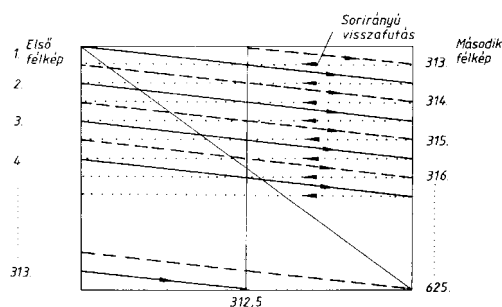
- váltott soros (interlaced) képfelépítési rendszer (NI) (51. ábra)
Az elektronsugár a képnek csak minden második sorát tapogatja le. Az eljárás lehetővé teszi a nagyobb felbontás megvalósítását. Mivel ilyenkor a foszfor fényének csökkenése minden egyes letapogatási ciklus során észrevehető mértékű villódzást érzékelünk.
- folyamatos (non-interlaced) képfelépítési rendszer (I)
Az elektronsugár a képernyőn a bal felső saroktól elkezdve végigszalad és egy sor "felépít". Az így felépített egy sort raszternek nevezzük. A sor végén az elektronsugarat kioltják, miközben a sugár visszatérítik a következő sor bal szélére (sorváltási frekvencia). A képcsőbe épített mágneses eltérítő egység folyamatosan változtatja az elektron-

sugarak hajlásszögét, így a sugár a bal felső saroktól a jobb alsó sarokig az egész képernyőt "végigpásztazza". A képernyő egyszeri letapogatását képkockának nevezzük.

A számítástechnikában alkalmazott monitoroknál mind a interlaced, mind pedig a non-interlaced képépítést alkalmazzák csak a jellemzők különböznek a televízió technikától. Alapul véve egy SVGA (Super-Video Graphics Array) videokártyát a következő jellemzők gyakoriak egy jól működő rendszerben:

- képváltási frekvencia: 56, 60, 72 Hz (NI); 87 Hz (I)
- sorváltási frekvencia: 31.5 - 58 kHz
- grafikus felbontás: 640*480, 800*600, 1024*768, 1280*1024
- színmélység: 16 szín (4 bit), 256 szín (8 bit), 65536 szín (16 bit), 16.77 millió szín (24 bit), 4.3 milliárd szín (32 bit)
- videómémória igény: 512 kB-12 MB

Ez utóbbi három jellemző egyértelműen bizonyítja, hogy tömörítésre márpedig szükség van, hiszen egy 25 kép/sec.-os képváltási frekvenciával megépített Pal szabványú videojel luminancia (Y) összetevőjét 13.5 MHz-en, míg krominancia jeleit (R-Y, B-Y) 6.75 MHz-en mintavételezve (lásd később), egyöntetű 8 bites kódolást feltételezve a mintavételezési sebesség: $(13.5 \text{ MHz} + 6.75 \text{ MHz} + 6.75 \text{ MHz}) \cdot 8 \text{ bit} = 216 \cdot 10^6 \text{ bit/sec}$; adatátviteli sebesség $640 \cdot 480 \cdot 25 \cdot 3 \text{ byte/sec} = 23040000 \text{ byte/sec}$; míg a szükséges tárhelykapacitás közel 22 MB másodpercenként.



51. ábra

3.2. A videomédium feldolgozásának lehetőségei

A videomédiumot tartalmazó multimédia alkalmazás elkészítése kétféle képpen lehetséges:

a) Videodigitalizálás

Az analóg videojelet *videodigitalizálókártya* segítségével bittérképek sorozatává alakítjuk és az így kapott állományokat háttértárban tároljuk. Megjegyzendő, hogy ezen állományok a nagy méret miatt veszteségmentes tömörítésen is átesnek (pl. MPEG). Visszajátzás során az egymásután tárolt tömörített képek "kifejtődnek" és a visszajátzási sebességgel lejátszódnak. A szemünk a 3.1.3. fejezetben vázolt okok miatt az egymást követő képkockákat mozgóképként érzékeli.

Lehetőség van a videojellel szinkronizált audiojel digitalizálására ill. visszajátzására is.

Ahhoz, hogy video- és/vagy audioinformációt olvassunk be a számítógépbe az alábbi eszközökre van szükség:

- videomagnó vagy videokamera vagy DVD lejátszó, amely tárolja, rögzíti az analóg vagy digitális videojelet. Felmerülhet az a kérdés, hogy miért kell egy digitális videojelet tovább digitalizálni. Az ok elég prózai: más rendszerű digitális, mint amit a számítógép fogadni tud.
- video-digitalizálókártya, amely végrehajtja a digitalizálás folyamatát. Vannak olyan kártyák, amelyekkel a videomagnó is vezérelhető. A vezérlés az SMPTE (Society of Motion Picture and Television Engineers) timecode szerint történik, erről később még bővebben esik szó.
- nagy kapacitású háttértár, amely a nagy méretű digitalizált videoállomány tárolására hivatott.

A videoadatmennyiség mérete annál nagyobb, minél jobb minőségű felvételt (finom felbontás, nagy színmélység) szeretnénk készíteni. Ez az adatmennyiség, amelyet ráadásul valós időben kell átvinni a merevlemezre, háromféleképpen csökkenthető:

- a képek hardveres vagy szoftveres tömörítése,
- a színelbontás, képfelbontás csökkentése,
- a képváltási arány csökkentése.

A rögzített videojel minősége a videodigitalizáló kártyán kívül a videorendszertől is függ: D1, D2, D3. A televízió technikában már szabvány a Beta, a DigitBeta, vagy a Dv, DVD is. Ezeknek a különféle videorendszereknek vannak azonos és egyéb speciális lehetőségeik is. Például a DV rendszerben a digitális audio-video jelet D/A átalakítás nélkül is rögzíthetjük, vagy ha van *composite* és S-Video kimenetünk, akkor az előbbit célszerű választani a jobb minőség miatt.

A videó-adatbevitel minőségét az alábbi tényezők határozzák meg:

- a videodigitalizálókártya adatátviteli sebessége és tömörítési sebessége,
- a merevlemez adatátviteli sebessége,
- a CPU sebessége (MHz, MIPS),
- az adatátviteli buszrendszer sebessége,
- a rendelkezésre álló RAM mérete.

b) videó overlay

Az overlay technika alkalmazásával lehetőségünk van arra is, hogy az analóg videojelet tárolás nélkül az overlay-kártya azonnal digitalizálja és így rögtön a multimédia alkalmazásba ültethető.

3.3. A digitalizált videomédium jellemzői

- **képváltási sebesség (arány) – frame rate**

A képváltási arány megadja, hogy hány állóképet dolgozunk fel egységnyi idő alatt (1s). FPS (Frame-Per-Second)-ben méri, azaz képkocka/másodpercben. Azt tudjuk, hogy az emberi szem a 24 kép/másodperc gyorsaságú megjelenítést már mozgóképnek érzékeli ezért pl. a televízióban ezt használják. Az MPEG-nél ez az arány akár 30 kép/s

is lehet. Lényeges különbség azonban a képkockák letapogatásában van. Az analóg technikákban félképekből (mezőkből) rakják össze az állóképet, először a páratlan sorokat, majd a páros sorokat. Ezt hívják átlapolt (interlaced) képmegjelenítésnek. A számítógép, azaz a digitális megjelenítőknél nincs szükség mezőkre, egy menetben rakja ki a képet, ez a nem átlapolt (non-interlaced) képmegjelenítés (3.1.3. fejezet).

- **színmélység (színfelbontás)**

A színfelbontás megadja, hogy hány színt jelenítünk meg a képernyőn egyidejűleg. Természetesen többfajta színrendszer létezik. Az számítógép RGB színrendszert használ, a videoformátumok többfélét, leginkább az YUV felosztást használják. Használják ezenkívül a színmélység kifejezést is a színfelbontásra, tulajdonképpen itt a bitmélységre utalnak, hány bitre van szükség egy képelem (képpont-pixel) tárolásához. Pl.: RGB színrendszerrel 8 bit/képpont = 256 szín vagy 24 bit/képpont = 16,7 millió szín. YUV színrendszerrel általában 7 vagy 8 bit/képpont mélységet használnak.

- **képfelbontás**

Ez a kifejezés a kép méretére vonatkozik X,Y irányban. Ezeket a méreteket is szabványok rögzítik. A tengerentúl az NTSC 352x240, itt Európában a PAL 352x288 képpontos felbontást ír elő.

- **képméret**

A multimédia alkalmazások nagy része a képernyő egy különálló ablakában jeleníti meg, ill. játssza le. Ez korábban 4:3 oldalarány volt, manapság ettől már el lehet térni.

Az előző három összetevő együttese határozza meg a megjelenített képek minőségét, amely általában a legfontosabb szempont a kódolás során, ezért alaposan kell megválasztani a videó alkalmazásának függvényében. Pl.: a felbontás legyen 352x288 (PAL), 30 kép/sec frame rate és 24 bites (16,7 millió) színmélység, így kapjuk az adott videónk minőségét, jelen esetben egy igen jó minőséget hozunk létre.

Jelölje T a digitalizált videoanyag méretét, S a színmélységet bitben, K a képfelbontást pixelben és F a képváltási sebességet.

$$T = \frac{K * S * F}{8 * 1024 * 1024} \quad (38)$$

$$T = \frac{352 * 288 * 24 * 30}{8 * 1024 * 1024} = 8.7 \text{ MB/sec}$$

Egy másodpercenyi tömörítetlen videoanyag mérete közel 9 MB, indokolt a tömörítés.

3.4. Digitálisvideo szerkesztése

A digitális videószerkesztés legáltalánosabban elterjedt programja az Adobe Premiere. A program alapegysége a *projekt*, amely magában foglalja az importált médiumok sorozatát, az ezeken végrehajtott trükköket, vágásokat. A projektben szereplő médiumok környezeti beállításait *preset*-nek nevezzük. Ennek megadására akkor van mód és lehetőség amikor egy új projektet megnyitunk. Itt visszaköszönnek mindazok az alapfogalmak, amelyeket a korábbi fejezetekben megtanultunk. A projekt indításakor kétféle környezeti beállítást eszközölhetünk: