

Arvutigraafika

Teksti järel on multimeediumis info edastamisel tähtsuset järgmine komponent graafika (fotod, joonistused, diagrammid, skeemid jne.).

Graafika on artefakt, mis reprodutseerib mingi subjektiga (tavaliselt füüsiline objekt, isik) sarnase pildi.

Termini "arvutigraafika" pakkus 1960ndal aastal välja William Fetter, et kirjeldada oma tööd firmas Boeing.

Arvutigraafika (*computer graphics*) all mõistetakse üldiselt objektide geomeetriliste mudelite ja kujutiste konstrueerimist, muutmist, kuvamist ja töötlemist arvutis.

Kujutis (*image*) viitab lihtsalt pikselite (*pixel - picture element*) kogumile. Geomeetrilised mudelid (*geometric models*) kujutavad endast objektide kirjeldusi omavahel seotud ja hästi määratletavate geomeetriliste elementide kaudu.

Räägitakse 2D (*2 Dimensional*) ehk tasapinnalisest graafikast ja 3D (*3 Dimensional*) ehk ruumilisest graafikast. Arvutigraafika alla kuulub lihtsate piltide joonistamine ja muutmine, projekteerimine ja modelleerimine arvutil, animatsioonid, virtuaalne tegelikkus (*virtual reality*) jne.

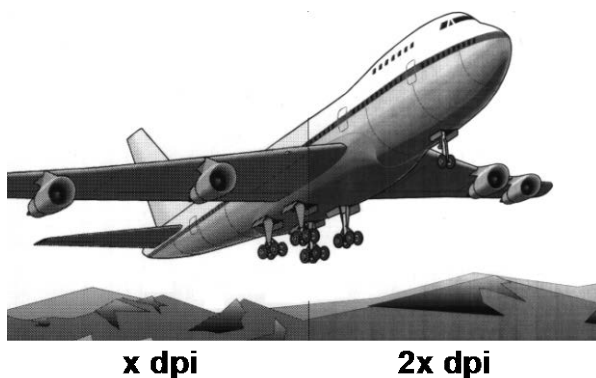
Kuigi arvutiekraanil koosnevad kõik kujutised ühtemoodi pikselitest, eristatakse kahte, põhimõtetelt täiesti erinevat graafikaliiki: rastergraafikat ning vektorgraafikat.

Rastergraafika

Rastergraafikas (inglise keeles *raster graphics* või vahel ka *pixel graphics*) koosneb pilt üksikutest punktides ehk pikselitest (*pixel*), milledest igaüks omab oma värvi ja on eraldi töödeldav. Rastergraafikat kasutatakse digiteeritud fotode ja realistlike joonistuste puhul kuna saab kasutada väga sujuvaid värviüleminekuid ning objektidel ei pea olema selged piirjooned.

Rastergraafika pildidel on mitu olulist parameetrit:

- **Resolutsioon** (*resolution*) ehk punktutihedus, mida mõõdetakse pikselite arvuga tolli kohta ning mida tähistatakse lühendiga **dpi** (*dots per inch*), näiteks on levinumad väärtused 72; 75; 96; 100; 150; 200; 300; 400; 600; 1200 ja 2400 dpi. Arvutiekraanil kasutamiseks mõeldud pildidel peaks punktutihedus olema 72 – 100 dpi, sest see sarnaneb ekraani enda tihedusele, printimiseks mõeldud piltide punktutihedus võiks olla 300 dpi või isegi suurem.
- **Suurus pikselite arvuna** (näiteks 1600X1200). Suuruse meile harjumuspärasemates mõõtühikutes - sentimeetrites saame pikselite arvu jagamisel punktutihedusega ja siis veel 2,54-ga korrutamisel (tollid ja sentimeetri suhe, näiteks pilt mõõtmetega 1600X1200 pikselit ja punktutihedusega 200 dpi on paberile printituna 8X6 tolli ehk 20,32X15,24 cm).



- **Värvussügavus** (*color resolution, color depth, color mode*). Värvussügavus määrab, kui palju andmeid iga pildi pikseli kohta salvestatakse, ehk mitut erinevat värvi antud pildil kujutada saab. Kõik värvid pildil tuleb kirjeldada nii mitme nulli ja ühe kombinatsioonina (binaarkood), kui mitme bitise värvussügavusega on tegemist (näiteks 1 bitise korral saame ainult variandid 0 ja 1, ehk vaid kaks võimalikku värvi, 4 biti korral saame variandid 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111, ehk 16 erinevat värvi). Kasutatakse järgmisi standardeid:
 - 1 bitine värvussügavus - 2 värvi (must ja valge)
 - 4 bitine värvussügavus - 16 värvi
 - 8 bitine värvussügavus - 256 värvi
 - 16 bitine värvussügavus - 65 536 värvi, on kasutusel variant, kus iga RGB komponendi jaoks on 5 bitti ja lisaks 1 bitt *Alpha* kanali jaoks. See värvussügavus kannab nimetust *high color*.
 - 24 bitine värvussügavus - 16 777 216 värvi, 8 bitti iga RGB komponendi jaoks. Kuna see on kõige lähemal meid elus ümbritsevate värvide hulgatele, siis käib selle standardiga kaasas nimetus *true color*
 - 32 bitine värvussügavus - *true color*, tavaliselt on iga RGB komponendi jaoks 8 bitti ja lisaks 8 bitti *Alpha* kanali jaoks (pole üldse värviga seotud).

Lisaks on olemas mitmed standardid mustvalgetele fotodele, nn *grayscale*, näiteks:

- 4 bit *grayscale* 16 halltooni
- 8 bit *grayscale* 256 halltooni
- 16 bit *grayscale* 65536 halltooni
- 24 bit *grayscale* 16777216 halltooni



1 bit ehk 2 värvi



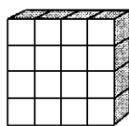
4 bitti ehk 16 värvi



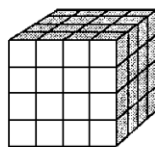
8 bitti ehk 256 värvi

Muidugi määrab värvussügavus ka faili suuruse, näiteks samade mõõtmetega 24 bitise värvussügavusega pilt võtab kõvekettal 3 korda rohkem ruumi kui 8 bitise värvussügavusega.

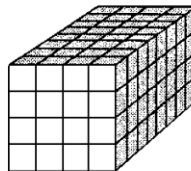
Rastergraafika faili mahu sõltuvus värvussügavusest



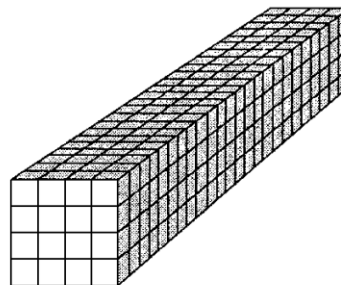
1 bit
2 värvi



4 bitti
16 värvi



8 bitti
256 värvi



24 bitti
16 777 216 värvi

Rastergraafika failid on arusaadavatel põhjustel tavaliselt suure mahuga, mis seab nende töötlemisel arvutile mõningaid nõudeid (palju operatiivmälu, suur kõvaketas, hea graafikakaart ja kvaliteetne monitor).

Virvtoonimine (*dithering*)

Tihti peale võimaldab ekraan näha vähem värve kui pildi värvisügavus seda lubaks. Sellisel juhul kasutatakse tehnoloogiat, mida nimetatakse virvtoonimiseks (*dithering*) ja mis sarnaneb difusioonile. See seisneb selles, et kättesaadavate värvide saamiseks näidatakse kahe kättesaadava värvi punkte segamini.



Kuna lähestikku asuvate eri värvi pikselite korral inimaju tajub neid hoopis kolmanda, nende pikselite värvide segamisel saadava värvina. Näiteks musta ja valge värviga pikselite segamini näitamise saavutatakse erinevaid halltoone, punase ja sinisega saavutatakse lilla jne.

Virvtoonimist kasutatakse ka objektide astmiktäitmisel (täitmisel üleminekuga ühelt värvilt teisele ehk *gradient fill*), et üleminekud paistaksid sujuvad.

Meetod on kasutusel ka piltide teisendamisel väiksema värvigammaga piltideks, et kättesaadavate värvide n.ö. võltsida (eriti aktuaalne näiteks 8 bitise ehk 256 värviga värvigamma korral).

Soovitusi rastergraafika töötlemiseks

Esimeseks soovitusena on töö käigus erinevate kujutiste salvestamine eraldi failidesse. Kui kujutised on kord ühele rastergraafika pildile kokku monteeritud, on neid sealt eraldada praktiliselt võimatu.

Rastergraafika pildid on arvutiekraanil alati pisut sakilised, kuna ekraani resolutsioon (punktihedus) pole kuigi suur. On tähtis, et pildi enda resolutsioon poleks väiksem, kui ekraanil.

Peamised reeglid töös rastergraafika piltidega puudutavad mastaapimist (suurendamist/vähendamist, *scaling*) ja pööramist (*rotation*).

Mastaapimine

Ideaalne on, kui rastergraafika elemendid on loodud sellises suuruses, nagu neid kasutada soovitakse. Vahetevahel on suuruse muutmine (*scaling*) siiski vajalik. Kehtib reegel:

- **Kui võimalik väldi rastergraafika kujutiste suurendamist/vähendamist!!!**

Kui rastergraafika pilti suurendada näiteks 200%, siis keskmine programm asendab iga pikseli kahe pikseliga, mis muudab pildi palju sakilisemaks. Vastupidisel juhul, kui pilti vähendada 50%, siis tuleb osa pikseleid “ära kaotada”, millel on ka oma pahad kõrvalmõjud.

Juuresoleval pildil on keskel näha originaalsuuruses rastergraafika pilt, resolutsiooniga 70 dpi, millest kahel pool on muudetud suurusega koopiad, paremal 50% ja vasakul 200%.

Vähendamisel on parimad variandid need, mis saadakse 100% jagamisel paarisarvuga, näiteks 50% ja 25%, millede korral “visatakse ära” iga teine või iga neljas piksel. Valides 77% anname me arvutile ülesande, mis toob kindlasti kaasa tehisliskuse probleemi (kujutise loomulikkus kaob).



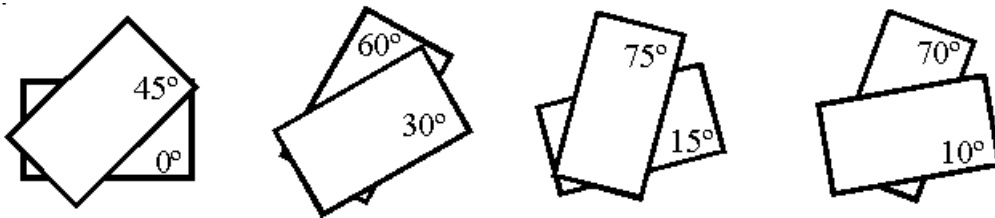
Suurendamisel saab tekkivaid sakilisi ääri ise käsitsi kohendada lisades ja kustutades sobivates kohtades vajalikke pikseleid. On olemas ka programme, mis kasutavad suurendamisel/vähendamisel keerukamat interpolatsioonimeetodit, mis annab juba tunduvalt siledama tulemuse.

Pööramine

Enamus rastergraafikat töötlevaid programme lubavad jooni ja teisi geomeetrilisi kujundeid joonistada ja pöörata (*rotate*) mistahes nurgal. Probleem on selles, et kujundite servad, mis on ilusad horisontaal- ning vertikaalsuunal muutuvad praktiliselt igasuguse nurga all sakiliseks. Reegliks on:

- **Kasuta nurki, mille puhul tehiskuse efekt on kõige väiksem!**

Parim valik on 45° , kuna sellisel juhul on kõik pikselid ikkagi enam-vähem ühel joonel. Paremusjärjestuses järgmised on nurgad 30° ja 60° ning kolmandana 15° ja 75° .



Silumine

Nii suuruse muutmisel (eriti suurendamisel) kui ka pööramisel tekkib tavaliselt sakilisuseefekt (*aliased*). Et selle efekti mõju vähendada, võib kasutada silumistehnoloogiat, mille nimetus on *antialiasing* (eesti keeles kasutatakse ka terminit sakitõrje) ja mis muudab objekti serva pikselite värvuse selliseks, mis saadakse objekti ja tausta värvi segamisel. Näiteks punasel taustal oleva sinise objekti servad teisendatakse lillaks. Tänu sellele tajub inimsilm objekti servi siledamatena, luuakse illusioon siledast servast.

aliased
anti-aliased

Selle tehnoloogia rakendamisel peab olema ettevaatlik, kuna peenemad jooned võivad üldse kaduma minna.

Sama tehnoloogiat kasutatakse ka piltide kuvamisel arvutiekraanil.

Pildi punktiheduse ja värvussügavuse muutmine

Kui Teil on tarvis rastergraafika pildil vähendada nii resolutsiooni (punktihedus) kui ka värvussügavust, siis sooritage alati enne resolutsiooni muutmine ning alles seejärel värvissügavuse muutmine. Põhjendus seisneb selles, et resolutsiooni vähendamisega (aga ka pildi vähendamisel) läheb osa pikseleid kaduma ning sellega tavaliselt kaob ka mingisugune osa värve. Kui alles nüüd värvigammat vähendada, jääb meil lõppkokkuvõttes rohkem erinevaid värve järele ning kujutis loomulikum. Samasugune põhimõte kehtib ka pildist mingi osa väljalõikamisel (*crop*): enne lõige, siis värvigamma muutus.

Samal põhjusel tuleks ka JPEG kompressiooni rakendada alati pärast resolutsiooni muutmist, mitte enne.

Rastergraafika allikad

Skaneerimine

Skaneerimine on graafiliste kujutiste (fotod, joonised jms.) **digiteerimine** spetsiaalse seadme: skanneri (*scanner*) abil. Toimub omamoodi printimisega vastupidine protsess: paberil olev kujutis sisestatakse arvutisse.

Skaneeritava kujutise **kvaliteet** sõltub eelkõige muidugi algmaterjalist, kuid tähtsusetl teisel kohal on skanneri (*scanner*) kvaliteet.

Skaneerimistarkvara on põhimõtteliselt kahte tüüpi:

- spetsiaalselt selleks väljatöötatud programmid;
- lisamoodulid graafikaprogrammidele.

Mõlemal juhul pakuvad programmid kõigepealt eelskaneerimist (*pre-scan* või mõnel juhul ka *preview*), mille käigus tuuakse ekraanile kiirendatud korras skaneeritud ebakvaliteetne nn eelvaate kujutis. Eelvaate kujutisel saab kasutaja raamiga (vahetevahel ka vabakäejoonega) märkida pildi osa, mida soovitakse skaneerida. See hoiab kokku nii edaspidist salvestatavate andmete mahtu, kui ka tööks vajaminevat aega.

Skaneerimisel on võimalik alati määrata ka skaneeritava pildi resolutsiooni (*resolution*), heledust (*brightness*), kontrasti (*contrast*) ja värvigammat (kasutatavate värvide hulka).

NB! Kuigi rastergraafika failid on mahukad, tuleks skaneerimisel valida siiski maksimaalselt kvaliteetne skaneerimiviis (maksimaalsed võimalikult vajaminevad värvid, tihedus, võimalikult paras suurus, seega ka küllaltki suur andmehulk). Põhjus selleks on lihtne: värve ja tihedust saab hiljem vähendada, kuid juurde saamine on väga raske.

NB! Mustvalgeid skeeme ja joonistusi, mida hiljem plaanite mustvalgetena ka välja trükkida, võiks skaneerida siiski 256 halltooniga (mustvalgete fotodena), siis jäävad paljud jooned, mis muidu oleksid skanneri jaoks liiga peenikesed, siiski nähtavaks ja jooned, mis muidu tuleksid sakilised, jäävad siiski siledad.

Digitaalfotografeerimine

Digitaalfotoaparaadi abil pildistatakse nagu hariliku fotoaparaadiga, kuid pilt jäädvustatakse kohe digitaalkujul ning teda saab praktiliselt kohe arvutis töödelda ja kasutada.

Ekraanilt “pildistamine”

Mõningatel juhtudel on meil tarvis kujutist otse ekraanilt (näiteks mingi programmi tutvustust koostades).

Üheks võimaluseks Windows keskkonnas on muidugi terve ekraanipildi “pildistamine”, vajutades klahvi **Prt Scr** (*Print Screen*). Seepeale paigutatakse kogu ekraanipildi jäljend mälu puhvrise (*clipboard*), kust seda saab vajalikku kohta kleepida (*paste*), graafikaprogrammi abi kasutades saab temast ka üksikuid vajalikke osi “lõigata”. Siinkohal on kasulik teada, et **Alt + Prt Scr** teeb jäljendi vaid aktiivsest aknast!

On olemas ka hulk spetsiaalseid programme (*screen capture utilities*), mis terve ekraanipildi või mingi osa mälu puhvrise "pildistavad" ja/või faili salvestavad.

Selliste hulka kuuluvad mitmed spetsiaalsed graafikaprogrammidega kaasasolevad vahendid aga näiteks ka Windows 7 töövahendite hulka kuuluv **Snipping Tool**.

Vektorgraafikafailide pööramine rastergraafikasse

See operatsioon on õige lihtne ja sellega tulevad toime kõik vektorgraafikat töötlevad programmid. Tuleb vaid anda käsk **Export** ja valida sobiv formaat.

Vektorgraafika

Vektorgraafika (*vector graphics*) ehk ka objekt-orienteeritud (*object oriented*) või struktureeritud (*structured graphics*).

Objektide kirjeldus koosneb tüüpiliselt ankrupunktidest (*anchor points*), objekti suurusel (*size*), nurkadest (*angle*), positsioonist (*position*), joone jämedusest (*line width*) ning tüübist (*line style*), täitevärvist (*fill color*) ja -meetodist (näiteks ringjoon on kirjeldatud keskpunkti ja raadiuse abil). Kui mõnda objekti muuta, siis muutuvad vaid mõned teda kirjeldavatest parameetritest ja ülejäänud osa jääb samaks. Nii on lihtsam ning kiirem. Vektorgraafika joonistused näevad välja nagu tavalised joonistused.

Objekte kirjeldavate parameetrite järgi väljastatakse ekraanile või printerile rastergraafika pilt, mille resolutsioon vastab väljundseadme omale. Seega on vektorgraafika väljundseadmetest sõltumatu (*device independent*).

Vektorgraafika allikad

Kõige tavalisem viis vektorgraafika saamiseks on ise midagi joonistada. Muidugi on kasutada ka hulk lõikepildikogusid (*clipart*), nagu näiteks on kaasas Corel Draw paketiga.

Pildi vektoriseerimine

Pildi vektoriseerimine (*tracing*) on tegevus, millega saab hakkama terve hulk erinevaid programme, milledest osa on spetsiaalselt selleks otstarbeks loodud (näiteks CorelTrace) või programmid, mis muu hulgas suudavad kasutada raster- ja vektorgraafikat (näiteks Flash).

Toome siinkohal erinevate võimaluste loetelu Corel'i graafikapaketti kuuluva CorelTRACE põhjal, mis suudab vektoriseerida mitmesugustes failivormingutes pildifaile. Väljundiks antakse EPS (*Encapsulated PostScript*), mida loeb Corel Draw. Keerulisi pilte originaaliga sarnanevaiks vektorpiltideks trasseerida ei saa, kuid lihtsamate piltide, jooniste ja skeemidega õnnestub see kenasti.

Vektoriseerimiseks saab määrata mitmeid võimalusi. Kõigepealt tuleb otsustada, kas vektoriseerida jooni või pindasid:

- *Normal Centerline* kasutatakse, kui joonis koosneb peentest detailidest ja kitsastest joontest, kui on tähtis kätte saada joonte enam-vähem õige kuju. Vektoriseerimisel luuakse vektorid üksikute sirgjoonte ja kõveratena (*curve*).
- *Normal Outline* programm otsustab, et iga pikselikogum moodustab pinna, need piiratakse, värvitakse ja asetatakse sobival viisil üksteise peale.

Kasulik võimalus on *Remove Noise*, mis laseb määrata, kui väikseid pikselikogumeid võib ignoreerida.

Saab määrata ka kõverjoonte sujuvust jms. Tähele tasuks panna ka võimalust *Invert Bitmap First*, mis laseb rastergraafika pildi enne trasseerimist negatiiviks muuta.

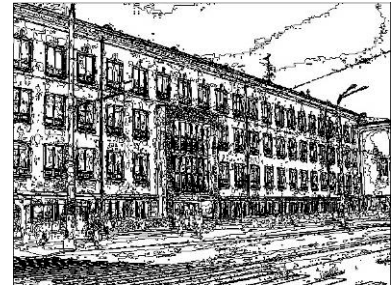
Loomulikult esinevad vektoriseeritud pildil originaaliga võrreldes mitmed erinevused (puudujäägid), mida keerukam on algmaterjal, seda rohkem detaile läheb kaduma.



Originaalne rastergraafika pilt



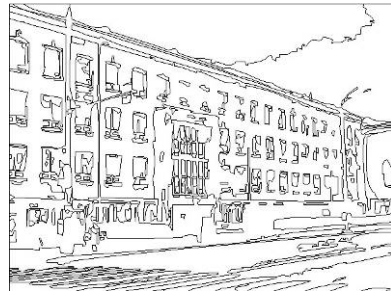
Vektoriseeritud pilt (kõrgkvaliteediga)



Sama pilt traatvõrena (*wireframe*)



Vektoriseeritud pilt (madala kvaliteediga)



Sama pilt traatvõrena (*wireframe*)

Graafika failivorminguid:

Erinevaid graafika failivorminguid on küllaltki suur hulk, Osa neist on seotud mingi kindla tarkvaraga (näiteks PSD ehk Adobe PhotoShop fail või CPT ehk Corel PhotoPaint Image), osa on muutunud standardiks. Iga vormingu väljatöötamiseks on olnud oma põhjus, igäihte neist kasutatakse omal kohal.

- **PCX** on eksisteerinud juba pikka aega. Aja jooksul on talle värve lisatud ja resolutsiooni tõstetud. Hoolimata kõigist uuendustest, muutub see järjest vähemoluliseks. Paljud Windows'i programmid ei toeta seda formaati.
- **TIFF** (*Tagged Image File Format*) on samuti küllaltki vana formaat, kuid tänu uuendustele veel täiesti elujõuline. TIFF on üsna tavaline kirjastamistöös ja praktiliselt kõik tarkvarapaketid toetavad seda. Viimased versioonid lubavad ka faili kompressiooni (kasutades RLE, LZW, CCITT ning JPEG algoritme), kuid kahjuks ei suuda kõik programmid kompressitud TIFF faile lugeda. TIFF failide laiendiks on TIF.
- **BMP** (*Bitmap*) ilmus alles koos Windows 3.0-ga ja on toetatud kõigi Windows'i programmide poolt (mõni programm ainult seda tunnistabki). Kuna **BMP** ei kasuta kompressiooni, siis kasutatakse seda suurte failide (näiteks kõrge resolutsiooniga värvigraafika) jaoks harva.
- **DIB** (*Device Independent Bitmap*) on teine formaat, mis on tekkinud koos Windows'iga. **DIB** on väga sarnane **BMP** formaadiga. Tema eeliseks on see, et teda suudavad kasutada praktiliselt kõik tarkvara paketid, teda saab kergesti kuvada monitoril ja trükkida printeril. **DIB** formaati pole paljudes programmides isegi valikuna antud, tema lugemiseks piisab kui valite **BMP** formaadi. Arusaadavatel põhjustel on see formaat laialt kasutusel programmeerijate seltskonnas.
- **GIF** (*Graphics Interchange Format*) on kompressiooni kasutatav formaat, mis loodi 1987. aastal Unisys'i poolt CompuServe'is kasutamiseks. Suur osa piltmaterjali, mida Internetist leida võib, on salvestatud just selles formaadis. Praegu toetab **GIF** formaati suurem osa olemasolevast tarkvarast.

See formaat toetab vaid 8 bitise värvisügavusega pilte. GIF on kadudeta (*lossless*) kompressiooniga, LZW-algoritmi kasutades saavutatakse maksimaalne kompressiooniaste 5:1.

GIF formaadi eripäradeks on võime salvestada mitmeid kujutisi, nende hulgas ka pisikesi pilte, mida saab üldiste pildimõõtude suhtes vabalt paigutada ja võimaldada osa pildi pinnast läbipaistvaks muuta (GIF89a). See eripära taasavastati WWW disainerite poolt, kes selle abil loovad kõikvõimalikke animatsioone.

Internetis kasutamiseks saab GIF faili salvestada *interlaced* kujul, mis laseb pilti laadida progresseeruvalt ridade kaupa.

Kahjuks on Unisys alates 1994. aastast hakanud seni tasuta formaadi eest litsentsitasu nõudma, mis võib sellele formaadile saatuslikuks saada.

- **PNG** (*Portable Network Graphics*) (hääldatakse "ping") on GIF formaadi edasiarendus ja täiesti tasuta.

PNG formaat sisaldab mitmeid suurepäraseid võimalusi nagu näiteks varieeritav läbipaistvus.

PNG kompressiooniaste on palju parem kui GIF formaadil ja mahub vahemikku 10 kuni 30.

Nagu GIF, nii ka PNG on kadudeta kompressiooniga formaat kuid ta pole piiratud 256 värvi kasutusvõimalusega ja lubab kasutada ka 24 bitist RGB graafikat. Kuna ta on kadudeta kompressiooniga formaat, siis on 24 bitised tihendatud pildid loomulikult suurema mahuga kui allpool kirjeldatud JPEG formaadi puhul.

Sarnaselt GIF piltidele saab PNG pilte salvestada *interlaced* kujul.

- **EPS** (*Encapsulated PostScript File*) on alguse saanud kirjastamismaailmast ning on saanud tavaliseks kindlat tüüpi graafika juures. See formaat salvestab graafilisi andmeid kasutades PostScript koodi. Pole sugugi lihtne või isegi praktiline teisendada graafikat teistest formaatidest EPS formaati, mõistlik on vastupidine konverteerimine. EPS failid on tegelikult mõeldud PostScripti kasutamiseks printerile saatmiseks. Tihti võib EPS fail sisaldada ka madala resolutsiooniga **TIFF** graafikat, et oleks võimalik pildi eelvaatlemist teha.
- **WMF** (*Windows Metafile Format*) on taas üks Windows'iga seostatud formaat, mida ei kasutata just kuigi palju, väljaspool Windows'i praktiliselt üldse mitte. Suurem osa Windows'i programme siiski toetab seda formaati. Tegemist on vektorgraafika formaadiga, kuid nagu nimetus *metafile* vihjab, suudab see kombineerida nii vektorgraafikat kui ka rastergraafikat.
- **TGA** (*Targa*) on esimene populaarne formaat kõrge resolutsiooniga (24 bitise värvipaletiga) kujutiste jaoks. Formaati sai nime *TARGA* videokaardilt, esimeselt täisvärviga videosalvestuskaardilt. Enamus praegusi videosalvestuskaarte toetavad seda formaati.
- **JPEG** (*Joint Photographic Experts Group*) on kujutiste maksimaalseks kompressiooniks kasutatav formaat.
- **JP2** (**JPEG 2000**) on uus formaat, JPEG edasiarendus. Seda toetavad praegu väga vähesed programmid.
- **SVG** (*Scalable Vector Graphics*) ehk mastabeeritav vektorgraafika on uus revolutsiooniline graafika failiformaat, mis on välja töötatud W3C (*World Wide Web Consortium*) poolt. SVG põhineb XML (*Extensible Markup Language*) keelel (XML-i arvatakse peatselt HTML-i asendavat). SVG pildid on vektorpõhised, mis tähendab, et neid saab suurendada/vähendada (*zoom*) ilma detailsust kaotamata. Failide suurus on väiksem kui JPEG, PNG või GIF piltidel. Lisaks kiirele allalaadimisele teeb SVG võimalikuks ka kõrge resolutsiooniga printimise, eduka

suurendamise/vähendamise ja kerimise (*pan*) ning isegi teiste piltide lisamise SVG pildi sisse.

JPEG-kompressioon

JPEG kujutab endast standardiseeritud pildikompressioonimehhanismi ja on oma nime saanud ISO (*International Standard Organization*) alluvuses töötavalt rühmalt *Joint Photographic Experts Group*.

Kuna see töötati välja kõikvõimalike piltide kompressiooniks, siis selle kasutamine teksti jaoks on võimatu, vaid joontest koosnevate, must-valgete ja lihtsate piltide korral aga pole nii efektiivne kui fotode ja täisvärviliste piltide puhul.

See mehhanism kasutab üht küllaltki uut kompressioonimeetodit "*lossy compression*", mis tegelikult "kaotab" osa kujutise taasesitamiseks vajalikku infot arvestades, et inimsilm ei avasta kadumaläinud osa. Seega pole saadud pilt enam absoluutselt identne esialgsega. Kasutatakse asjaolu, et muudatused värvides (eriti sinise toonides) on vähem märgatavad, kui muudatused heleduses. Sellega on seletatav ka kõlbmatus teksti jms. puhul kasutamiseks.

JPEG kompressioonimehhanismi üheks laialt kasutatavaks omaduseks on võimalus seada kompressiooniastme ja pildi kvaliteedi (kadude suuruse) vahekorda. JPEG kompressiooniga on võimalik saavutada kompressiooniaste kuni 200:1, kuid kvaliteet säilib parimana siiski kompressiooniastmetel 10:1 kuni 25:1. Kompressiooniastmetel 30:1 ja 50:1 on algpildiga võrreldes vähesed või keskmised silmaga nähtavad moonutused.

Pildi kvaliteet määratakse faili salvestamisel. Lisaks faili suuruse ja kvaliteedi suhte (*quality ... large file*) määramisele saab valida ka kahe formaadi vahel:

- *Baseline ("Standard")*
- *Baseline Optimized*, mis pakub parandatud värvikvaliteeti ja pisut väiksemat faili.

Halltoonides pilte ei saa sama suure kompressiooniastmega kokku pakkida kui värvipilte. Kuigi nad kasutavad vaid 8 bitti iga punkti esitamiseks, on nende JPEG failid vaid 10%-15% väiksemad samaväärsetest täisvärvilistest. Halltoonides piltide kompressioonil hakkavad moonutused sageli paistma juba kompressiooniastmel 5:1.

Teine huvitav omadus seisneb ka lahtipakkimise kiiruse ja kvaliteedi suhte seadmisvõimaluses. Kiiret lahtipakkimist kasutatakse peamiselt piltide eelvaate (*preview*) jaoks.

Internetis kasutamiseks saab JPEG faili salvestamisel määrata ka veel progressioon (*progressive*), mis laseb pilti laadida mitmes osas ja on WWW lehel kasutamiseks sobivam.

Eelis GIF vormingu ees on kõigepealt täieliku värviinformatsiooni sisaldamine (24 bitti), samas kui GIF võimaldab maksimaalselt vaid 8 bitti, seega "kaotab" JPEG kõvasti vähem infot. Teiseks võrreldamatult suurem kompressiooniaste, mis GIF-il pole kunagi suurem kui 5:1.

JPEG kompressioon toimub järgmiste sammudena:

- RGB (*Red Green Blue*) kujutis viiakse kujule, mis enam vastab inimsilma heleduse ja värvuse tundlikkusele (*luminance* ja *chrominance*).
- Kujutis jagatakse 8 korda 8 pikselist koosnevateks tükkideks ja iga tüki jaoks arvutatakse keskmine valgustatus (värv, heledus) (*luminance*) kasutades protsessi DCT (*Direct Cosine Transform*).
- Kõrgemad lainepikkused (värvid), mis on inimsilmale tajumatud, teisendatakse väärtusele null.

- Peened sarnaste värvide üleminekud eespool nimetatud tükkides, kvanditakse selgemateks üleminekuteks vähemate sammudega.
- edasi rakendatakse standardseid kompressioonimeetodeid.

Kompressiooniga “kokkupakitud” kujutiste vaatamiseks ja töötlemiseks on failile vaja rakendada dekompressiooni, mille korral läbitakse kõik loetletud sammud vastupidises järjekorras. “Lahtipakitud” kujutis vajab tunduvalt vähem mälu ruumi kui originaalkujutis, kuna kaotatud värvid ja värviüleminekud jäävadki kadunuks.

JPEG 2000

JPEG 2000 ehk ametlikult "ISO 15444 JPEG 2000: *Image Coding System*" kuulutati avalikult välja 1999 aasta augustis, kasutusele võeti see standard 2000 aastal.

Alates 1998. aasta augustist on DIG 2000 meeskond (*Digital Imaging Group*) teinud koostööd ISO JPEG 2000 töögrupiga, eesmärgiks oli uue standardiseeritud failiformaadi loomine, mis laiendaks meie võimalusi kujutiste säilitamisel ja edastamisel ilma märgatava kvaliteedikao. Uus formaat pakub rohkem tihendamisvalikuid (*compression options*), paremat kvaliteeti, rohkem funktsioone ning säilitab seejuures faili väikese mahu.

JPEG 2000 tuumaks on *Wavelet* tehnoloogia. Tavalisel JPEG-l oli selleks otsene koosinusteisendus ehk DCT (*Direct Cosine Transformation*), mis pakkis kujutise 8X8 pikseli suurustesse plokkidesse, paigutades nad ühte faili üksteise järele. See algoritm vähendab kompressiooni saavutamiseks sagedusinfot (*frequency information*), mida rohkem faili tihendatakse, seda enam infot eemaldatakse. Radikaalse kompressiooni korral jäetakse alles vaid kõige olulisem info, mis annab edasi vaid kujutise põhiosad, peenemad detailid jäävad kadunuks.

Wavelet tehnoloogia teisendab kujutise lainekeste (*wavelet*) seeriaks, mida saab efektiivsemalt salvestada kui pikselite plokkide. Seejuures tagab lainekeste ebahürtlane serv parema kvaliteediga kujutise, kuna välditakse ruudukujuliste plokkide tekitatavaid moonutusi.

Wavelet tehnoloogia võimaldab sarnaselt FlashPix standardile reguleerida laetava pildi resolutsiooni (eelvaate jaoks madal jne).

JPEG 2000 pakub veel huvitava võimalusena kadudeta salvestamist (*lossless*). Seega pole enam näiteks kujutiste kvaliteetselt arhiveerimiseks vaja kasutada erinevaid failiformaate (TIFF jt).

JPEG 2000 spetsifikatsioon näeb ette kuni 256 andmekanalit. See võimaldab kasutada nii CMYK värvimudelit kui ka ICC (*International Color Consortium*) värviprofiile.

Innovatiivne on ka uue standardi sRGB kasutamine vaikimisi värvimudelina. Senises JPEG standardis polnud vaikimisi värvimudelit ette nähtud ja see põhjustas ebajärjekindlust värvide taasesitamisel. Kõik määratlemata RGB värvid loetakse automaatselt sRGB värvideks.

Määratletud värvid (näiteks ICC profiilid) kaaluvad sRGB üle (*override*). Igal juhul on värvide taasesitamine täpsem, sest kõik on alati standardi poolt defineeritud.

