

Časť z obsahového štandardu: Grafická informácia – *rastrová a vektorová grafika; animovaná grafika, video, kódovanie farieb; grafické formáty; aplikácie na spracovanú grafickej informácie.*

Pod grafickou informáciou chápeme obrázky, fotografie, schémy... Digitalizácia grafiky je vo všeobecnosti konverzia grafickej (analogovej) informácie do digitálneho tvaru (do zápisu pomocou čísel). Zosnímanému údaju sa priradí určitý počet bitov. Toto priradenie musí byť také, aby sa údaj z digitálnej podoby dal jednoznačne, alebo s určitým priblížením transformovať späť do analogového tvaru.

Rastrová grafika - všeobecne

Rastrové obrázky sa skladajú z množstva obrazových bodov – **pixelov** (picture element). **Počet týchto bodov závisí na rozlíšení obrázku. Každý z týchto bodov je popísaný niekoľkými bitmi a tieto body vytvárajú veľmi jemnú mriežku (raster alebo mapu),** preto sa im hovorí rastrové alebo bitmapové. Ak takýto obrázok **zmenšíme**, je nutné niektoré **body vypustiť** (body sa nemôžu zmenšovať). Tým **dôjde k strate detailu** a niekedy aj k skresleniu farieb. Ak bitmapový obrázok **zväčšujeme**, opäť nedochádza k zmene veľkosti bodov, ale začnú sa **od seba vzdďaľovať**. Aby obrázok nebol zrnitý, musí program chýbajúce body dopočítať (väčšinou vezme dva susedné body a medzi nich dá priemerný odtieň z týchto dvoch bodov). Teda aj tu dochádza **k zníženiu kvality obrázku**.

Vektorová grafika - všeobecne

Pri vektorovej grafike je **obrázok zložený z objektov skladajúcich sa z kriviek, kde jednotlivé objekty sú popísané matematicky. Objekt je popísaný počiatočným bodom, smerom, dĺžkou,... a jeho vlastnosťami ako farbou obrysovej čiary, farbou výplne, priehľadnosťou, tieňovaním, poradím, v akom bude vykreslený,...** Krivky spájajú jednotlivé kotvové body a môžu mať definovanú výplň (farebnú plochu alebo farebný prechod). **Základom vektorovej grafiky je matematika.** Súbor obsahujúci vektorový obrázok sa skladá z definície súradnicového systému, z rovníc (s konkrétnymi hodnotami parametrov) opisujúcich jednotlivé objekty a z vlastností objektov – obrysovej čiary, výplne, tieňa, priehľadnosti, poradia vykreslenia. **Vektorová grafika si teda nakreslené prvky pamätá ako geometrické útvary. Bod je základným stavebným kameňom** všetkých objektov vektorovej grafiky, je to elementárny prvok definovaný súradnicami. Nemá však nárok na samostatnú existenciu, t.j. že bod nie je možné vo vektorovej grafike nakresliť. **Krivka** je čiara – rovná, zakrivená, alebo zložená z viacerých kriviek spojených, alebo zoskupených. **Geometrický tvar** je uzavretý útvar ohraničený krivkou – štvorec, obdĺžnik, trojuholník, kruh, elipsa, hviezda, n-uholník...

Porovnanie rastrovej a vektorovej grafiky

Vektorová grafika má proti rastrovej grafike niektoré **výhody**:

- je možné ľubovoľne zmenšovať alebo zväčšovať obrázok bez straty kvality,
- je možné pracovať s každým objektom v obrázku oddelene,
- výsledná pamäťová náročnosť obrázka je obvykle omnoho menšia ako pri rastrovej grafike.

Nevýhody vektorovej grafiky

- oproti rastrovej grafike spravidla zložitejšie vytvorenie obrázka; v rastrovej grafike možno obrázok ľahko získať pomocou fotografie,
 - ak prekročí zložitost' grafického objektu určitú hranicu, začne byť vektorová grafika náročnejšia na pamäť, procesor, veľkosť disku ako grafika bitmapová, ...
-

Rastrová grafika – pojmy a širšia charakteristika – kódovanie farieb

Základné pojmy :

- **pixel** – picture element - **px** – základná jednotka rastrového obrázka – jeden bod určitej farby,
 - **rozmer obrázka** – je určený počtom pixelov na šírku a a počtom pixelov na výšku (napríklad **1024x768**, 800x600),
 - **rozlíšenie** určuje počet bodov na palec (udáva sa v jednotke **dpi**); (palec = 2,54 cm),
 - **farebná hĺbka** - počet bitov potrebných na zakódovanie farby (**bpp – bits per pixel**) (vyjadruje sa tým celkový počet reprodukovateľných farieb - 2 bpp - bližšia charakteristika - pozri tabuľku),
 - **farebný model** – spôsob výberu farby z konkrétneho farebného priestoru,
 - **EXIF údaje** – informácie o vzniku fotografie (rozmery v pixeloch, model fotoaparátu, expozičný čas, clonu, ISO, ohniskovú vzdialenosť, režim blesku, režim automatiky, spôsob merania, korekciu expozície a pravdaže čas fotografovania).
 - **grafický formát** - pravidlá, podľa ktorých sa obrázok ukladá do súboru.
-

farebná hĺbka (v bitoch)	počet reprodukovateľných farieb	Názov
32	$2^{24} = 16777216$ $2^8 = 256$ priehľadnosť	TrueColor + priehľadnosť
24	$2^{24} = 16777216$	TrueColor
16	$2^{16} = 65536$	HighColor
8	$2^8 = 256$	VGA
4	$2^4 = 16$	EGA
2	$2^2 = 4$	CGA
1	$2^1 = 2$	Monochromatické (B&W - ČB)
n	2^n	...

- **farebný model RGB** – Red, Green, Blue (aditívny model – „...čím viac farby, tým je svetlejšia...“)
- **farebný model CMYK** – Cyan – azúrová, Magenta – purpurová, Yellow – žltá, black – čierna (subtraktívny model – „...čím viac farby, tým je výsledok tmavší...“)
- **farebný model YUV** – pre TV vysielanie

Rastrové obrázky sú rozdelené na sieť „myslených štvorčekov“ - **pixelov**. Rozmer každého obrázka je pre počítač počet pixelov na šírku krát počet pixelov na výšku. Pre každý pixel (štvorček) je nutné okrem polohy (riadok a stĺpec) zakódovať aj farbu, resp. ďalšie parametre. Ak je **obrázok monochromatický (len čierna a biela farba)**, **kódovanie** je jednoduché (1 - rozsvietený (biely) bod, 0 - nerozsvietený (čierny) bod), napr.

```
100111001
011010110
101101101
110111011
111010111
111101111
```

A teda zápis v riadku vyzerá : 011000110 100101001 010010010 001000100
000101000 000010000, čo je 54 bitov.

Pri niektorých obrázkoch je však potrebné zaznamenať i odtiene, preto dve farby - čierna a biela - nestačia. Výhodné je zakódovať **odtienie sivej farby pomocou ôsmich bitov** tak, aby **informácia o každom bode zaberala 1 B**, t.j. jedno pamäťové miesto počítača. Čím bude bod svetlejší, tým väčšia hodnota sa do pamäte uloží. Preto bod **čiernej farby** bude uložený ako **0** a bod **bielej farby** ako maximálna možná hodnota - **255**. Obrázok teda bude kódovaný nasledovne:

```
192 0 0 255 255 192 0 0 255
0 255 192 0 192 0 255 192 0
192 0 255 192 0 255 192 0 255
255 192 0 255 255 192 0 255 255
255 255 192 0 192 0 255 255 255
255 255 255 192 0 255 255 255 255
```

Obrázok bude v pamäti
počítača zaberat' 54 Bajtov
(pamäťových miest).

Aby bol **obrázok farebný**, musíme **každému pixelu obrázka priradiť určitú farbu**. Spôsobov, akými sa to dá urobiť je opäť niekoľko. Na to, aby sme to vedeli urobiť, musíme **vedieť niečo o ľudskom oku** a spôsobe videnia farieb. Farby vnímame vďaka svetlu, ktoré vstupuje do nášho oka a dopadá na bunky - čapíky, ktoré sú citlivé na farbu. Svetlo je vlastne elektromagnetické vlnenie s rôznou vlnovou dĺžkou. **Ľudské oko dokáže vnímať iba určitý rozsah vlnových dĺžok svetla. Svetlo s najväčšou vlnovou dĺžkou, ktoré ľudské oko vníma, je červenej farby (780 nm) a svetlo s najnižšou vlnovou dĺžkou je svetlo fialovej farby (380nm)**. V tomto spektre farieb je ľudské oko schopné bez problémov rozlíšiť vyše 1,5 milióna rôznych farieb. Nie však každej farbe, ktorú rozlíšime pripadá iná vlnová dĺžka, pretože podľa intenzity dopadajúceho svetla, **ľudské oko rozlišuje sýtosť a jas daného odtieňa farby. Takže iná vlnová dĺžka prislúcha iba rôznym odtieňom farieb**.

Prvé riešenie problému kódovania farieb, ktoré sa ponúka, je zobrať celý rozsah vlnových dĺžok a rozdeliť ho na niekoľko častí a do pamäte počítača uložiť poradové číslo farby, ktorej prislúcha určitá vlnová dĺžka, ďalej jej sýtosť a jas. Kódovanie farieb takýmto spôsobom využívajú **farebné modely HSB, HLS**. Oba modely uchovávajú informáciu o odtieni (Hue) a sýtosti (Saturation). Tieto modely sa odlišujú iba v tom, že prvý model používa jas (Brightness) a druhý používa svetlosť (Lightness). V prvom modeli dostaneme čiernu farbu tým, že nastavíme jas na nulu a bielu tak, že jas nastavíme na maximálnu hodnotu (nezavisle od sýtosti). V druhom modeli dosiahneme čiernu, ak je svetlosť aj sýtosť 0 a bielu, ak je svetlosť aj sýtosť maximálna. Hlavným nedostatkom tohto problému je náročnosť výroby svetla so zadanou vlnovou dĺžkou. Našťastie existujú aj iné riešenia problému kódovania farieb.

Farebný model RGB je najčastejšie využívané kódovanie farby bodu obrázka. Empiricky sa zistilo, že takmer všetky farby sa dajú vytvoriť zmiešaním ľubovoľných troch nezávislých farieb (t.j. že pomocou zmiešania dvoch nedostaneme tretiu). Najvýhodnejšie pre výrobu svetelných lúčov bolo použitie farieb červená (Red), zelená (Green), modrá (Blue). Aby sme vedeli vytvoriť 1,5 milióna farieb stačí, ak každú z týchto farieb rozdelíme na 115 odtieňov, ktorých zmiešaním v rôznych pomeroch vzniknú všetky farby. Kvôli uchovaniu v pamäti počítača, je však výhodnejšie použiť až 256 odtieňov každej farby (8 bitov), čo nám umožní vytvoriť až 16 777 216 rôznych farieb. Pri takomto kódovaní je každá farba zakódovaná 24 bitmi, čo sú tri pamäťové miesta počítača. Pričom 255 0 0 je sýta červená farba, 0 255 0 je sýta zelená farba, 0 0 255 je sýta modrá farba, 0 0 0 je čierna farba a 255 255 255 je biela farba.

Farebný model CMY (CMYK) je model, ktorý používa doplnkové farby. V modeli RGB platilo, že ak zmiešame všetky tri základné farby s maximálnou sýtosťou, dostaneme bielu farbu. Takýto spôsob je výhodný pri obrazovkách monitorov, pretože tienidlo je čierne. Pri tlačiarňach sa však tlačí na biely papier, preto potrebujeme vziať také farby, pri ktorých, ak zmiešame ich najsýtejšie odtiene, dostaneme čiernu farbu. Takéto farby dostaneme, keď zoberieme doplnkové farby k farbám červená, zelená a modrá. Týmto farbami sú azúrová (**Cyan**), purpurová (**Magenta**) a žltá (**Yellow**). Kvôli tomu, že je lacnejšie vyrobiť čierny atrament ako ho miešať pomocou týchto troch farieb, sa k týmto farbám pridáva i samostatná čierna farba a tento model sa označuje tiež CMYK, kde posledné písmeno je odvodené od black - čierna. Výhodou tohto formátu je tá, že výsledná farba CMY sa dá veľmi jednoducho získať z modelu RGB pomocou vzorcov: $C = (255 - R)$; $M = (255 - G)$; $Y = (255 - B)$

Farebný model YUV slúži na zachovanie čiernobielej informácie pri televíznom vysielaní. Po vzniku farebného filmu nastal problém, ako zakódovať farbu tak, aby farebné filmy mohli pozerat' i ľudia s čiernobielymi prijímačmi. Bolo potrebné zachovať pôvodnú čiernobielu informáciu a doplniť ju tak, aby vznikol farebný obraz. Farba je kódovaná tak, že k čiernobielej zložke Y, ktorú tiež nazývame svietivosť (luminance), pridáme dve farebné zložky UV farebnosti (chrominance), pričom zložka U udáva odtieň medzi modrou a žltou a zložka V udáva odtieň medzi červenou a žltou farbou. Takže čiernobiely prijímač berie do úvahy iba zložku Y, farebné prijímače za pomoci zložiek YUV získajú RGB kód farby pomocou transformácie cez prepočet vzorcov.

Formáty rastrovej grafiky

Najbežnejšími formátmi rastrovej grafiky sú formáty **BMP, JPEG, GIF, TIFF alebo PNG**.

- Formát **BMP** (bitmap) je veľmi jednoduchý, **nekomprimovaný a bezstratový** formát. Záporom je veľká veľkosť obrázku.
- Formát **JPEG** je **najznámejší formát** na ukladanie digitálnych obrázkov, **používa stratovú kompresiu**. **Objem sa teda znižuje** skutočným skresávaním dátovej informácie. Odoberajú sa informácie, ktoré nerozlíši ľudské oko. Na nastavenie kompresie sa **používa faktor kvality** v rozmedzí 0 až 100 %. **Za ideálne nastavenie pre fotografie sa považujú hodnoty 70 až 85 %**. **Silnou stránkou tohto formátu sú obrázky s hladkými, ba až rozmazanými prechodmi jednotlivých hrán**. Ide hlavne o fotografie či realistické maľby.
- Formát **GIF** (Graphics Interchange Format) je **vhodný na obrázky s ostrými hranami**, medzi ktoré patria aj **grafické prvky internetových stránok**. Používa bezstratovú kompresiu. Jedna z jeho zbraní je farebná paleta s obsahom len 256 farieb – 8-bitové farby. Na jednej strane je to nevýhoda, pretože sa vôbec nehodí na realistické snímky, ale na druhej strane daná farebná schéma výrazne znižuje objem súboru. Medzi ďalšie výhody patrí prehľadnosť jednej z 256 farieb.
- Formát **PNG** (Portable Network Graphics - doslova prenosná sieťová grafika) je **nástupcom formátu GIF**, disponuje účinnnejším komprimačným algoritmom a plnohodnotným farebným modelom. Zachovala sa pritom funkcia prehľadnosti. **Jeho prioritným zameraním sú teda obrázky s ostrými hranami a prechodmi**. PNG ponúka podporu 24-bitovej farebnej hĺbky, nemá teda ako GIF obmedzenie na max. počet 256 farieb súčasne.
- Formát **TIFF** (Tag Image File Format) – **vytvára obrázky vysokej kvality, a preto sa často využíva pri publikovaní**. **Dokáže zobrazit' milióny farieb**. Obrazové súbory TIFF sú obvyčajne väčšie ako obrazové súbory GIF alebo JPEG. Tento formát má niekoľko verzií (s komprimáciou a bez nej).
- **množstvo ďalších ...**

Formáty vektorovej grafiky

- **PostScript** je programovací jazyk určený ku grafickému popisu vytlačených dokumentov vyvinutý v roku 1985 firmou Adobe Systems Incorporated. Jeho hlavnou výhodou je, že je nezávislý na zariadení, na ktorom sa má dokument tlačiť. Je považovaný za štandard pre drahšie tlačiarne. Vďaka svojim rozsiahlym množstvám sa však skoro **stal aj formátom k ukladaniu obrázkov**. Jeho štandardná prípona je **.ps alebo eps**.
- **PDF** (skratka anglického názvu **Portable Document Format – Prenosný formát dokumentov**) je súborový formát vyvinutý firmou Adobe pre ukladanie dokumentov. **Súbor typu PDF môže obsahovať text aj obrázky, pričom tento formát zabezpečuje, že sa ľubovoľný dokument na všetkých zariadeniach zobrazí rovnako**. Vytvárať PDF dokumenty ide ako v Acrobat Adobe, tak v ďalších programoch (často však iba ako export do PDF). Prehliadanie je jednoduchšie, existujú voľne dostupné prehliadače pre mnoho platforiem, najznámejšou je oficiálny prehliadač materskej firmy *Adobe Reader*. PDF súbory majú príponu **.pdf**.
- **AI - Adobe Illustrator** - Vektorový grafický formát, ktorý používa známi kresliči vektorových obrázkov Adobe Illustrator. Ide zrejme o najkvalitnejší formát svojho druhu so spoľahlivým PostScriptovým výstupom pre tlačiareň.
- **CDR- CorelDRAW** - súbor môže obsahovať vektorové informácie a informácie rastrového obrázka.
- **SVG Scalable Vector Graphics** - škálovateľná vektorová grafika je značkovací jazyk a formát súboru, ktorý popisuje dvojrozmernú vektorovú grafiku pomocou XML. Formát SVG by sa mal v budúcnosti stať základným otvoreným formátom pre vektorovú grafiku na Internete. SVG definuje tri základné typy grafických objektov: vektorové tvary (vector graphic shapes – obdĺžnik, kružnica, elipsa, úsečka, lomená čiara, mnohoúhelník a krivka), rastrové obrazy (raster images) a textové objekty.
- **ZMF - Zoner Callisto** - Súborový formát ZMF a formát tlačových zostáv ZMP sú proprietárnymi formátmi producenta programu. Súbory v tomto formáte ale nejde vytvoriť v inom editore
- **množstvo ďalších ...**

Programy pre prácu s obrázkami:

- na vytváranie vektorových obrázkov (napr. CorelDRAW, Inkscape, Zoner Callisto,),
- na vytváranie rastrových obrázkov (MyPaint, Skicár, LogoMotion, ...),
- na úpravu obrázkov, fotografií a koláží (Adobe Photoshop, XNVIEW, ...).

Rasterizácia / vektorizácia

Procesu zmeny vektorového obrázka na rastrový hovoríme rasterizácia, opačnému (z rastrového na vektorový) vektorizácia. Vektorizácia rastrového obrázka nedokáže úplne uspokojivo transformovať rastrové údaje do vektorovej grafiky, no v niektorých prípadoch je lepší nedokonalý vektorový ako dokonalý rastrový obrázok.

Príklady a úlohy

Príklad1: Čiernobiely obrázok veľkosti 640 x 480 (čiže s 307 200 bodmi) zaberie v počítači práve toľko bitov. Teda $640 \times 480 = 307\,200$ bodov = 307 200 bitov = 38 400 Byte = 37,5 KB. Čo sa týka farebnosti platí, že čím lepší a kvalitnejší obrázok, tým viac farieb je na ňom použitých. Ale čím viac farieb, tým viac bitov, pomocou ktorých paletu farieb môžeme zakódovať. Už vieme, že ak použijeme len 2 farby, veľkosť obrázka bude 37,5 KB. Čo ak použijeme iný počet farieb? Jasne, že sa zmení veľkosť obrázka. A to nasledovne (pri výpočtoch použijeme údaje z tabuľky):

Ak použijeme 16 farieb, tak : $640 \times 480 = 307\,200$ bodov $\times 4 = 1\,228\,800$ bitov : 8 = 153 600 Byte : 1024 = 150 kB.
Ak použijeme 256 farieb, tak : $640 \times 480 = 307\,200$ bodov $\times 8 = 2\,457\,600$ bitov : 8 = 307 200 Byte : 1024 = 300 kB.
Ak použijeme HighColour, tak : $640 \times 480 = 307\,200$ bodov $\times 16 = 4\,915\,200$ bitov : 8 = 614 400 Byte : 1024 = 600 kB.
Ak použijeme TrueColour, tak : $640 \times 480 = 307\,200$ bodov $\times 24 = 7\,372\,800$ bitov : 8 = 921 600 Byte : 1024 = 900 kB.

Príklad2: Na digitálnom fotoaparáte máme nastavené rozlíšenie, pri ktorom sa fotografie ukladajú ako bitmapové súbory s veľkosťou 600 KB s 24 bitovým farebným kódovaním. Ak prekonvertujeme takúto fotografiu do 256 farieb, akú bude mať približne veľkosť?

Najprv musíme zistiť koľko bodov sa nachádza v obrázku. Vieme, že jeden bod obrázka je kódovaný 24 b, zaberá teda tri pamäťové miesta teda 3 Bajty ($24:8=3$). Ďalej vieme, že 1 kB = 1024 B, takže obrázok bude zaberat':

- $600\text{ kB} \times 1024 = 614\,400\text{ B}$

Teraz zistíme, koľko bodov sa nachádza v obrázku:

- $614\,400\text{ B} : 3\text{ B} = 204\,800$ (veľkosť obrázka : veľkosť bodu = počet bodov)

Teraz potrebujeme zistiť, koľkými bitmi sa dá zakódovať jeden bod 256 farebného obrázka, a tiež koľko pamäte takýto bod zaberie. Ak náhodou nevieme z pamäti, že $256 = 2^8$ takže je to 8 bitov, a to zodpovedá 1 B pamäte, môžeme počet bitov vypočítať takto: $\log_2 256 = \ln 256 : \ln 2 = 8$

Veľkosť 256 farebného obrázka teda bude:

- $204\,800 \times 1\text{ B} = 204\,800\text{ B}$ (počet bodov $\times$ veľkosť bodu = veľkosť obrázka)

Teraz už len prevedieme výsledok na kB: $204\,800\text{ B} : 1024 = \mathbf{200\text{ kB}}$

Príklad3: Katka má fotografiu triedy uloženú v bitmapovom súbore a chce ju poslať MMS-kou na Máriin mobilný telefón. Fotka má rozmery 256 x 180 pixelov (obrazových bodov) a je v nej použitých 256 farieb. MMS-ka môže mať maximálne 5 KB. Najmenej koľko MMS musí Katka poslať, ak chce poslať celú fotografiu rozloženú do viacerých MMS?

Rozmery fotografie sú 256 x 180 pixelov, teda počet bodov v obrázku bude : $256 \times 180 = 46\,080$

Každý bod 256 farebného obrázka zaberie 1 B pamäte, takže obrázok bude zaberat':

- $46\,080 \times 1\text{ B} = 46\,080\text{ B}$ (počet bodov $\times$ veľkosť bodu = veľkosť obrázka)

Jedna MMS môže zaberat' 5 kB čo je v B: $5\text{ kB} \times 1024 = 5\,120\text{ B}$

Počet MMS vypočítame takto: $46\,080\text{ B} : 5\,120\text{ B} = \mathbf{9}$ (veľkosť obrázka : veľkosť MMS = počet MMS)

Príklad4: Obrázok v pamäti je zložený zo 600 x 800 farebných bodov. Najmenej koľko KB pamäte zaberie tento 6-farebný neskomprimovaný obrázok?

Obrázok pozostáva z 600 x 800 bodov. V obrázku je použitých 6 farieb, teda na zakódovanie jedného bodu budeme potrebovať nasledovný počet bitov:

$\log_2 6 = \ln 6 : \ln 2 = 2,58$ výsledok presiahol celú hodnotu, musíme teda použiť 3 bity

Ak chceme zistiť, koľko takýto obrázok zaberie pamäte, musíme deliť 8, pretože jedno pamäťové miesto (1 B) pozostáva s 8 bitov: $800 \times 600 \times 3 : 8\text{ B}$

Výsledok je potrebné ešte previesť na kB, teda vydeliť 1024:

$800 \times 600 \times 3 : 8 : 1024\text{ kB} = 800 \times 600 \times 3 : (8 \times 1024)\text{ kB}$