

13.5 Vyhlazování hran

Základním problémem zobrazování v rastru je jev, zvaný alias. Alias je pojem používaný v teorii informace. Označuje rekonstruovaný signál, který sestavíme na základě diskrétních vzorků původního signálu. Hustota a přesnost vzorkování určuje kvalitu rekonstrukce, tj. při málo přesném a řídkém vzorkování signálu dojde ke ztrátě informace, příliš husté vzorkování naopak je nákladné a zvýrazní šum.

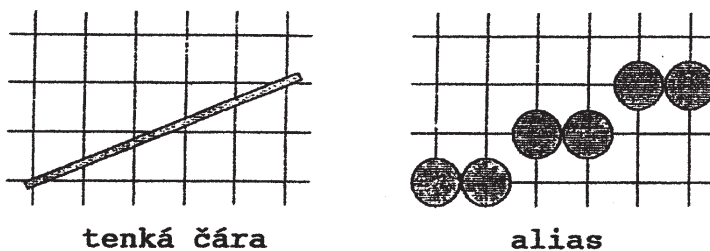
V počítačové grafice se tento jev projevuje geometrickým, barevným, případně pohybovým zkreslením. Hladké obrysy kreslených prvků a spojitě změny barvy jsou při kresbě v diskrétní rastrové síti nahrazeny diskrétním vzorkováním obecně spojitých průběhů. Přitom dochází ke ztrátě obrazové informace, nepřesnému vyjádření tvaru, potlačení významných detailů aj. Při sestavení obrazu z nedostatečných vzorků je vytvořen obraz, který je buď zřetelně zkreslený (čáry nejsou hladké, barvy se nemění spojitě), nebo na obraze nejsou detaily.

- Alias je počítačové grafice zřetelný nejméně ve třech různých případech. Nejběžnější 2.)
- 1) zkreslení se projevuje jako schodovitý průběh čar a hranic polygonálních oblastí. Druhý případ charakterizuje nepřirozené zvětšení a skoková změna polohy u malých objektů a sobě blízkých tenkých čar. Tenké čáry jsou zobrazeny jako nepravidelné posloupnosti bodů (nejsou opticky hladké).
 - 3) Třetí případ se týká zobrazení složitější scény s blízkými detaily. Tyto detaily jsou buď potlačeny (zakryty) nebo zkresleny tak, že nelze rozeznat jejich původní tvar.

Alias se výrazně projeví především u animace nebo během interaktivní práce např. při přemísťování a jiné transformaci objektů. Dokonce i obyčejná úsečka vykáže nepříjemné zkreslení při polohové transformaci, např. při otáčení. Malé objekty se na obrazovce střídavě objevují a mizí v závislosti na poloze.

Antialiasing, vyrovnávání, vyhlazování označuje postupy, které se používají v počítačové grafice pro odstranění nebo zmírnění uvedeného nežádoucího zkreslení kresby. Protože úplný rozběr vyhlazovacích technik používaných v různých případech přesahuje možnosti tohoto textu, uvedeme si pouze principy nejrozšířenějších postupů.

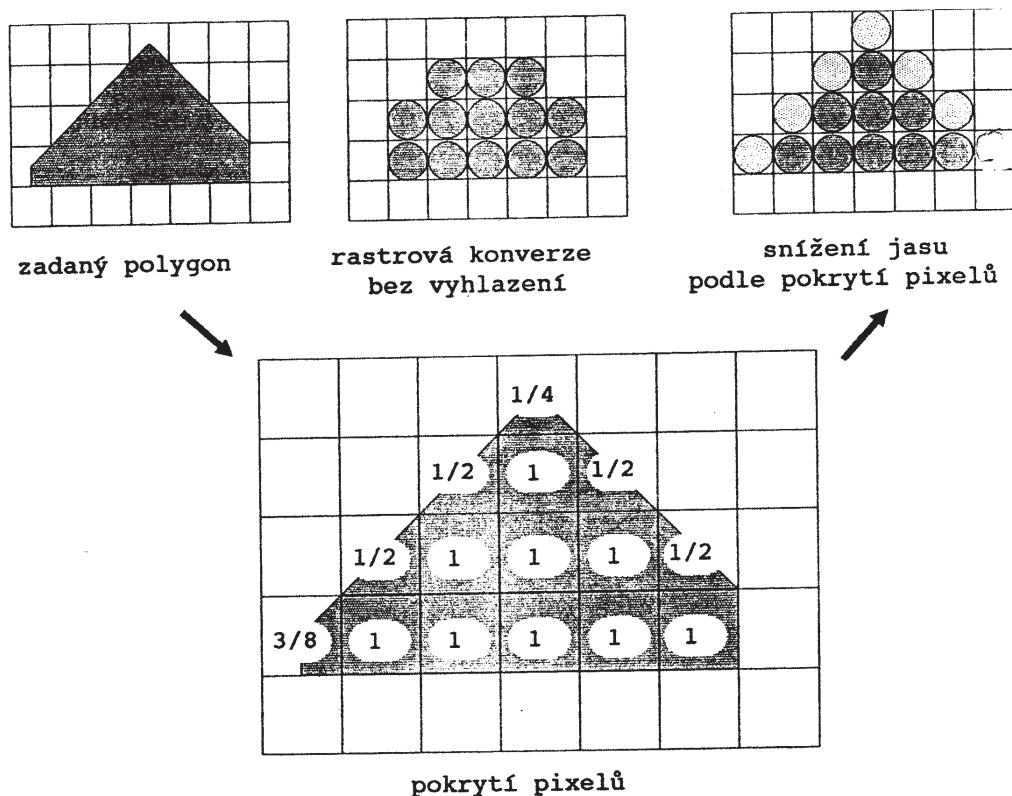
Základní myšlenka vyhlazování spočívá v tom, že pixel nepovažujeme za ideální bod, ale pracujeme s ním jako s plochou, na kterou promítáme část scény. Každý objekt ve scéně pak ovlivňuje hodnotu nejméně jednoho pixelu. Informaci, která se ztrácí při převodu do rastru, převědeme na jasovou nebo barevnou hodnotu pixelu. Podmínkou je samozřejmě skutečnost, že zařízení (např. displej) má "třetí" rozměr, tj. volitelný jas nebo barvu pro každý pixel.



Obr.13.15: Alias tenké čáry

Intenzitu (barvu) každého pixelu určíme jako součet (nebo jinou funkci) barevných příspěvků všech objektů, které jsou zobrazeny v daném pixelu. Na obr.13.15 je zobrazena úsečka, procházející pixelovou mřížkou. Čára je tenčí, než je šířka pixelu, a proto nemůže být věrně zobrazena např. Bresenhamovým algoritmem.

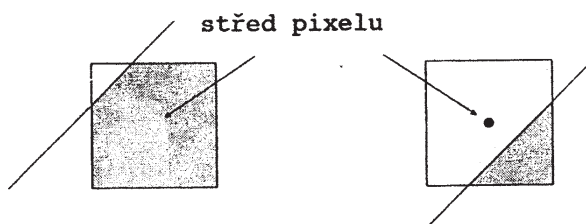
Věrnějšího zobrazení dosáhneme, když intenzitu každého pixelu odvodíme z velikosti plochy zakryté kresleným prvkem (obr.13.16).



Obr.13.16: Vyhlazení hranic vyplněného mnohoúhelníku

Okrajové pixely vyplňovaného polygonu jsou kresleny barevným odstínem, který je odvozen ze základní barvy polygonu, intenzity pixelů jsou násobeny koeficienty, které odpovídají poměru pokrytí plochy pixelů. Tyto koeficienty lze řešit v mnoha případech poměrně jednoduchým způsobem.

Jako příklad uvedeme použití Bresenhamova algoritmu pro vyhlazování. Protože chybový člen v Bresenhamově algoritmu má přímý vztah k velikosti plochy omezené úsečkou protínající pixel, lze uvedený algoritmus po úpravě použít pro vyhlazování okrajů polygonálních ploch. Jasovou informaci ztotožníme s velikostí plochy omezené skutečnou úsečkou v jednotkovém rasterovém čtverci. Rastrová síť je posunuta do středů čtvercových pixelů (obr.13.17).



Obr.13.17: Pokrytá plocha na čtvercovém pixelu

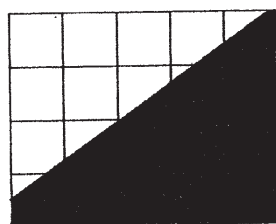
Alg: Vyhlazení hraniční úsečky polygonální oblasti

```
procedure Bresenham_Antialiasing(X1,Y1,X2,Y2,BARVA:integer);
{ platí pro 1.oktant, vnitřek oblasti vpravo od úsečky }
var E,X,Y,I:integer;
    DX,DY:real;
    Jas,Prirustek_jasu,Doplnek:real;
begin
    X:=X1;Y:=Y1;
    DX:=X2-X1;
    DY:=Y2-Y1;
    Prirustek_jasu:=DY/DX;
    Doplnek:=1-Prirustek_jasu;
    Jas:=0.5;
    for I:=1 to DX do
        begin
            BOD(X,Y,BARVA,Jas);
            if Jas>=Doplnek then
                begin
                    Y:=Y+1;
                    Jas:=Jas-Doplnek
                end
            else
                Jas:=Jas+Prirustek_jasu;
                X:=X+1
            end
        end
    end;
```

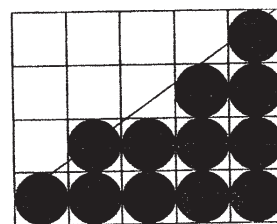
Proměnná Jas nabývá hodnot v intervalu (0,1) a je použita pro jasovou modulaci základních barevných složek proměnné BARVA.

13.6 Vyhlažování pomocí podrobnějšího vzorkování

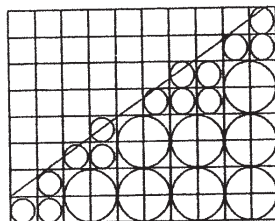
Přirozená vyhlazovací metoda spočívá ve vzorkování obrazu s vyšší frekvencí, než je rozlišení displeje. Kresba je vygenerována např. pomocí běžného interpolačního algoritmu s polovičním nebo menším krokem a takto jsou vypočteny souřadnice subpixelů, které aproximují kresbu. Běžným způsobem je určeno, zda subpixel je pro kresbu významný (např. zda více než polovina jeho plochy je pokryta kresbou). Výsledná intenzita pixelu určena jako vážený průměr intenzit sousedních subpixelů. Jako příklad si uvedeme opět zobrazení části vyplněné plochy omezené úsečkami. Na obr.13.18 je nakreslena část vyplňované plochy a její rastrová konverze bez vyhlazování. Při dělení 2x2 jsou určeny přesněji subpixely, které jsou blízko hraniční úsečky. U vnitřních pixelů není nutné použít hustější vzorkování.



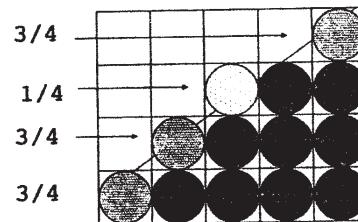
vyplněná oblast



vyplnění v základním rastru



dělení 2x2



průměrné intenzity

Obr.13.18: Vyhlažování při vzorkování 2x2 subpixely

Nevýhodou tohoto přístupu je skutečnost, že při řešení obrazu ve vyšší přesnosti se zvyšuje nelineárně výpočetní čas (dvojnásobná přesnost zvyšuje dobu interpolace čtyřikrát). Tuto dobu lze někdy na základě znalosti obrazu zkrátit, pokud použijeme podrobnější dělení jen v místech s vyšší hustotou informace, tj. na hranách jednotlivých objektů. Dále pak při zobrazení objektů a detailů, které jsou menší, než je velikost subpixelu, zvýšená vzorkovací frekvence stále neřeší problém jejich zobrazení. Výpočet prostého aritmetického průměru neodpovídá skutečnosti, že subpixely, které jsou blíže středu zobrazovaného pixelu, přispívají k jeho intenzitě větší měrou. Pro věrnější zobrazení je nutné použít vážený průměr pro výpočet intenzity. Příklady vah jsou uvedeny na obr.13.19.

$1/16 \times$

1	2	1
2	4	2
1	2	1

$1/81 \times$

1	2	3	2	1
2	4	6	4	2
3	6	9	6	3
2	4	6	4	2
1	2	3	2	1

Obr.13.19: Matice váhových koeficientů 3x3 a 5x5

13.7 Vyhlažování obrazu filtrováním

Filtrovací techniky jsou do jisté míry analogické předchozím technikám. Rozdíl je v tom, že intenzita pixelu může být obecně ovlivněna všemi objekty ve scéně. Míra vlivu je vyjádřena konvolucí:

$$G(x, y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} H(i, j) \cdot I(x+i, y+j) ,$$

$I(x, y)$ je intenzita v bodě $[x, y]$ před filtrací,

$G(x, y)$ je výsledná intenzita v bodě $[x, y]$ po filtraci,

$H(i, j)$ je filtrační funkce.

V praxi je H nenulová funkce v rozsahu menší obdélníkové oblasti, např.

$$H = \begin{bmatrix} H_{-1,-1} & H_{0,-1} & H_{1,-1} \\ H_{-1,0} & H_{0,0} & H_{1,0} \\ H_{-1,1} & H_{0,1} & H_{1,1} \end{bmatrix} = \begin{bmatrix} \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \\ \frac{2}{16} & \frac{4}{16} & \frac{2}{16} \\ \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \end{bmatrix} , H(i, j) = 0 , \text{ pro } |i|, |j| > 1$$

Filtrační algoritmus spočívá v realizaci uvedeného konvolučního vztahu pro vhodně zvolenou třídu filtračních funkcí. Vstupem filtrovacího algoritmu je obraz uložený v pixelové paměti $I(x, y)$ s intenzitami jednotlivých bodů vypočtenými např. pomocí interpolačních algoritmů. Výstupem je pixelová paměť $G(x, y)$ s intenzitami řešenými na základě výše uvedeného konvolučního vztahu. Protože v hraničních bodech zobrazovacího okénka se používají pro filtrování body ležící vně okénka, je vhodné obraz před filtrováním vygenerovat s přesahem odpovídajícím "záběru" konvoluční funkce. V opačném případě může být patrné zkreslení obrazu na hranici okénka. Filtrování můžeme aplikovat buď na obraz složený z pixelů, nebo na obraz vypočtený ve vyšší přesnosti a uložený do obrazové paměti subpixelů. V prvním případě obraz dodatečně filtrujeme na výstupu (post-filtering), výsledkem bude změkčení ostrých přechodů mezi jednotlivými objekty. Protože však při této úpravě obrazu nepoužíváme znalosti o skutečné scéně, nemůže aplikovaný filtr zvýšit informační obsah obrazu. Druhý případ se týká aplikace filtru na podrobněji vypočtený, hustěji vzorkovaný obraz. Zde je filtr použit jako vyhlazovací funkce, která převádí podrobnější informace ze subpixelů na intenzitu/barvu skutečného pixelu. Pro zvědavé čtenáře doplníme, že komplementární technikou k výstupnímu filtrování je vstupní filtrování (pre-filtering), které se používá pro odstranění nežádoucích složek obrazu. Jako příklad uvedeme aplikace různých vstupních filtrů na družicové obrazy. Pomocí filtrů lze z obrazu vyloučit nepodstatné obrazové prvky (např. řídkou mlhu) a dále pracovat s obrazem městské aglomerace nebo zalesněné krajiny.