1 Základy

Metody vykreslování:

- *object order* – sekvenčně po objektech, bez interakce → bez stínů
- *image order* – sekvenčně po pixelech, bez globální interakce → tvrdé stíny
- *globální* – celá scéna najednou → měkké stíny

2 OpenGL

OpenGL – standard pro počítačovou 3D grafiku, knihovny pro různý HW a OS

Grafická primitiva jsou bod, úsečka, polygon, pixmapa, bitmapa. Pomocí těchto se vše vykresluje.

Stavový stroj je základním principem vykreslování OpenGL, nastavení parametrů mění stav stroje a je dále používáno až do další změny.

Framebuffer je oblast paměti, do které se vykresluje výsledný obrázek, skládá se z několika bufferů (stencil, depth, color, ...) a může jich být více (stereografika, double buffering).

Vykreslovací řetězec je způsob průchodu dat subsystémem OpenGL od vstupních objektů až po výsledný obrázek.
Obsahuje cestu pro vektorová i rastrová data, rovněž je možné data číst z framebufferu. Nejdůležitější části – evaluátory, per-vertex operace, pixel operace, texture a primitive assembly. Per-fragment operace ovlivňují již rasterizovaná data, která vznikla z polygonů.

Prostorová data jsou uložena v podobě *vrcholů* (body, úsečky, polygony) a doplněna o rastrové informace (bitmapy, pixmapy). Vykreslovány jsou podle stavu stroje (určuje, zda vrcholy jsou body, nebo body úseček, vrcholy uzavřeného polygonu, ...).

Interpolace barev je automaticky prováděna při rozdílných barvách vrcholů polygonu.

Display list – příkazy pro OpenGL lze uložit pro pozdější vyvolání

3 OpenGL – grafická primitiva

Vykreslování probíhá pomocí příkazů `glBegin()` a `glEnd()`, parametrem je typ primitiva `GL_POINTS`,`GL_LINES`,`GL_TRIANGLES`,`GL_QUADS`...

Vlastnosti bodů: `glPointSize(size)`, `glEnable(GL_POINT_SMOOTH)` (antialiasing)

Vlastnosti úseček: `glLineWidth(width)`, `glLineStipple(factor, pattern)`, `glEnable(GL_LINE_SMOOTH|GL_LINE_STIPPLE)`

Vlastnosti polygonů: polygony mají 2 strany (back, front), lze ořezat vykreslování jednotlivých stran, dále nastavit vzorek...

Normálové vektory udávají orientaci primitiv (která strana je front a která back, dále se využívají ke stínování).

Stínování: flat (normálový vektor je přiřazen ploše), Goraudovo (přiřazen vrcholům)

Při spojování primitiv je třeba hlídat odpovídající orientaci polygonu a T-průsečíky.

Je možné využít také efektů *mlhy* (několik typů) a *blendingu*.

4 OpenGL – nastavení kamery

Nastavení kamery (i scény a pozic objektů) je prováděno pomocí *transformačních matic* 4×4.

Matice ovlivňují translaci, rotaci, typ promítání (rovnoběžné nebo středové).

Použitím matic za sebou (násobením) skládáme pohyb a nastavení kamery i scény.

Viewing transformation – umístění a orientace kamery

Modeling transformation – rozmístění a natočení objektů

Projection transformation – výběr parametrů promítání (typ, FOV)

Viewport transformation – umístění výřezu výsledného snímku

OpenGL obsahuje stavové proměnné pro jednotlivé typy matic, je možné dodat i vlastní ořezávací plochy.

5 OpenGL – materiály, osvětlení

Osvětlování znamená určování odstínu pixelu na základě normálového vektoru, pozice kamery a světelného zdroje.

Phongův model má tři složky: ambient (světlost prostředí), diffuse (rozptýlené světlo materiálem) a specular (odlesk)

Materiál je charakterizován odpovídajícími složkami – ambient, diffuse, specular, emmisive (není považován za zdroj, pouze ovlivňuje barvu), shininess (lesklost materiálu)

Světelné zdroje v OpenGL se mohou lišit podle implementace, základ je 8 světel. Lze nastavit pozici a směr, dále kužel, rozptyl a zaostření, složky světelného modelu ...

6 OpenGL – texturování, MIP mapping

Textury v OpenGL mohou být 1D, 2D i 3D. Pixely textury se nazývají *texely*, tyto se aplikují na povrch objektu. Je zapotřebí zadat *texturovací souřadnice* pro správné nanesení textur.

Je možné využít opakování nebo roztažení v daných směrech. Při texturování dochází k aliasingu.

MIP textury, tedy Multum in Parvo (více v jednom). Několik textur (jedna ve více rozlišeních) je uložena v jedné textuře, vybere se podle vzdálenosti od kamery (LoD). Mimo jiné se tak zabraňuje aliasingu.

Interpolace (bilinear, trilinear, anisotropic filtering) MIP textur je důležitá pro neznatelné přechody mezi jednotlivými úrovněmi detailu.

Multitexturování umožňuje nanést více textur najednou (např. textura a světelná mapa). Jedná se o rozšíření standardu.

Speciální textury – bump maps, environment maps, ...

7 OpenGL – frame buffer, stencil buffer

Framebuffer se zkládá z několika bufferů: color, depth (Z), stencil, accumulation

Color buffer se vyskytuje několikrát (je-li zapnutá stereovize nebo double buffering).

Depth buffer určuje, zda se fragment na daném místě má vykreslit (podle hloubky – vzdálenosti od kamery). Hodnoty nebývají brány lineárně.

Stencil buffer slouží k maskování části jiných bufferů, aby se do nich nekreslilo. Často využíván pro stínová tělesa a CSG. Nastavuje se u něj testovací funkce.

Accumulation buffer se používá pro sloučení více obrázků do jednoho. Použit pro antialising a motion-blur.

Selection buffer se využívá k výběru objektu ve scéně.

8 Globální viditelnost

Globální viditelnost řeší problém výběru částí scény, které se mají vykreslit. Dochází tak k urychlení vykreslování (neviditelné věci se nekreslí), je možné uplatnit LoD. Také řeší pořadí vykreslování (např. u průhledných objektů) a stíny.

Urychlení se provádí ořezáním (near a far rovina, zorné pole) a nekreslením zakrytých částí (jiným objektem).

OpenGL řeší pomocí *Z-bufferu*

Hierarchické ořezání tělesem záběru je řešení bez Z-bufferu. Z objektů scény se sestaví strom, mřížka nebo jejich kombinace. Struktury pak obsahují obalová tělesa (spheres, boxes, ...) a nebo je prostor rozdělen pomocí BSP (Binary Space Partitioning). Dalším řešením jsou mřížky (uniformní nebo adaptivní).

Portály jsou technikou využívanou hlavně v budovách. Jednotlivé buňky scény jsou spojeny portály (dveře, okna), které určují viditelnost do ostatních buněk.

Je možné využít hierarchického Z-bufferu pro různé úrovně detailu.

9 Úroveň detailu

Vzdálenější objekty jsou menší, tedy není potřeba vykreslovat zbytečné detaily → výkon, spotřeba paměti.

Zjednodušování modelů je základní technikou. Při snížení LoD se některé vrcholy modelu spojí a změnší se tak počet trojúhelníků (edge-collapse), při zvýšení naopak (vertex-split).

Geomorphing je metoda přecházení mezi předpočítanými modely (resp. jeden model s různými detaily) pomocí interpolace → plynulý přechod.

Billboard v grafice je jednoduchý objekt s texturou, který se vždy natočí přímo ke kameře, není potřeba třetí rozměr pro zachování detailu.

Výšková mapa je předpis, podle kterého se modeluje terén a na něj se pak mapuje textura terénu. Je možné generovat i dynamicky.

Algoritmus ROAM zajišťuje optimalizaci uložení výškových dat. Mřížka není uniformí, ale dělí se v případě složitějšího terénu. Je opět možné využít split/merge při změně LoD.

Chunked LoD je technika spojování více výškových map v jednu. Pokud nedoléhají správně, využívají se tzv. zástěrky.

Visual masking je technika výběru takové textury, která zamaskuje nedostatečný počet detailu modelu. Velká úspora výkonu.

10 Textury

Procedurální textury neexistují předem, ale hodnota bodu na objektu je spočítána při dopadu paprsku. Výsledná hodnota je tedy funkcí, kde parametrem je úhel dopadu a místo dopadu. Takovéto textury mohou rovněž modifikovat normály (bump mapping). Procedurální textury mají stále stejné rozlišení, jsou nekonečně velké a mohou být i 3D.

Generování výsledku procedurálních textur je založeno na skládání jednoduchých funkcí a použití šumu (value noise, perlin noise, ...).

Modelování plynů je pokročilou technikou procedurálního texturování. Jedná se o 3D textury s definovanou turbulencí, šumem, hustotou, ...

Hypertextury jsou rozšířené 3D textury. Definují geometrii modelu, využívají fuzzy logiku. Daly by se označit za pokročilý bump mapping.

11 Sledování paprsku

Sledování paprsku (ray tracing) je jednou z metod realistického zobrazování. Exituje také backward-ray-tracing, kdy se paprsek vrhá ze zdroje světla a sleduje se, zda dopadne do oka (zbytečná režie – do oka jich dorazí málo).

Postup: Pro každý pixel obrazu se vyšle paprsek (směřován z oka – nastavení kamery) a podle odrazů, průchodů a dopadu se spočítá výsledná barva.

Ray-casting nesleduje podružné paprsky, ale pouze do prvního dopadu. Používalo se v minulosti pro generování textur (Wolfenstein 3D).

Sledování paprsku je matematická metoda. Jednoduché objekty zde nejsou trojúhelníky ale matematicky lehce popsatelné objekty (koule, elipsoidy, ...).

Stíny lze velmi jednoduše zobrazit, jelikož při dopadu paprsku se paprsek sleduje, zda dokáže dorazit ke zdroji světla. Vznikají tak pouze tvrdé stíny. Pro výpočet jsou opět důležité normály plochy v daném bodě (ovlivnitelné bump mappingem).

Sekundární paprsky jsou odražené nebo lámané paprsky. Ray-tracing výborně zobrazuje lesklé a průhledné objekty.

Adaptive subsampling je metoda antialiasingu. Paprsek se vyšle pouze rohovými subpixely. Pokud se tyto liší, vrhají se paprsky všemi subpixely.

12 Realistické osvětlení

Distributed ray tracing využívá více paprsků, ze kterých pomocí distribuční funkce (fce rozložení) vypočte výslednou hodnotu. Využívá se pro vznik měkkých stínů, odrazů na matných plochách, refrakce světla, motion-blur (více paprsků v časovém intervalu) a supersampling AA. Také je možné jej využít pro simulaci čočky (depth of field).

Radiozita pracuje se všemi povrchy jako s plošnými zdroji světla. Začíná opravdovými světly, ale při aplikaci jednoho průchodu se osvětlené plochy stávají novými zdroji. Iteračně se dojde k výsledku. Interakční koeficienty je třeba předpočítat, může se také přidávat ambientní složka světla.

Kombinace předchozích je v praxi také používána.

Photon mapping je metoda vystřelování fotonů ze zdroje světla a zachytáváním do textur. Pak je možné použít i OpenGL nebo zpětné sledování paprsku.

Path tracing je matematicky sofistikovanější ray tracing. Přes pixel se vyšle několik paprsků, ale při dopadu se výsledek nepočítá jako součet pohlceného a odraženého sekundárního paprsku, ale podle pravděpodobnosti se určí, zda je pohlcen nebo odražen. Výsledek je pak vypočítán ze všech takto vyslaných paprsků z jednoho pixelu.

Dvousměrné metody jsou kombinací ray tracingu a backward ray tracingu. Výpočty se potkají uprostřed.

13 Zobrazování objemových dat

Objemová data – zdroj vstupních dat bývají různá měření nebo simulace. Pomocí několika projekcí je potřeba opět sestavit 3D obraz.

Je problém s uložením více dat, data jsou 3D. Adaptivní dělení mřížky, RLE kódování, marching cubes.

Ray tracing je poměrně přesná metoda, která je však dost náročná, princip stejný jako ve 2D.

Volume rendering rozděluje těleso na objemové jednotky v mřížce (voxely). Tyto voxely se pak "rozplácnou" o projekční plochu podobně jako sněhová koule. Voxel se tedy promítne do kruhu, gaussovské rozložení (uprostřed nejvíce, ke krajím méně). Sousední voxely se pak sčítají. Metoda je rychlejší než ray tracing, ale nemá tak kvalitní výsledky.

Shear-warp reconstruction je poměrně nová metoda, otočí model tak, aby došlo k co nejmenšímu zkreslení promítáním (snaha o rovnoběžnost osy promítání a modelu). Výsledek se pak záměrně pozmění (warp) tak, aby zobrazení odpovídalo původnímu natočení modelu.

14 2D vektorový a rastrový morfing

Morphing = přeměna tvaru pomocí digitální technologie (z jednoho na jiný objekt)

Warping = deformace obrazu do libovolného tvaru (nezbytná součást morphingu)

Využití je nejen v přechodu mezi diametrálně odlišnými objekty, ale i v plynulých přechodech mezi pozicemi objektu nebo fázemi animace. Warping se také používá při aplikaci textur.

Korespondence modelů je důležitá, např. lineární interpolace není vhodná, pokud má pohyb být po křivce.

Object-space morphing se snaží o co největší korespondenci objektu pomocí rozdělení na malé části, aby deformace byla minimální.

15 Částicové systémy

Využití: sníh, déšť, kouř, oheň, exploze, prach, tekutiny, ...

Částice je základní fyzikální jednotkou pro modelování s *nulovým objemem*, která *reaguje na síly*. Je možná interace částic, generování nových apod.

Hlavní problém je výpočet sil, které působí na jednotlivé částice (gravitace, elektrický náboj, částicové vazby, ...).

Hybridní systém obsahuje jak vypočítané vazby, tak generované při simulaci.

Vizualizace pomocí raytracingu, surface splattingu, point-based renderingu.

16 Body jako obrazová primitiva

Body mohou být použity jako obrazová primitiva namísto trojúhelníků. Výhodou je velice snadná regulace úrovně detailu a je možné jednoduše přerušit/pozastavit vykreslování a pokračovat později (není zde návaznost).

Nevýhody zahrnují hlavně nesouvislost, špatné texturování a těžké modelování rovných, barevně monotónních ploch.

Hierarchické zanořování bodů se většinou provádí pomocí stromů a seznamů.

Zobrazování využívá podobných metod jako částicové systémy (ray tracing, splatting, ...).

Skenování objektů je jednodušší, ukládají se pouze naskenované body.

Ray tracing je velice pomalý.

Splatting má výhodu v jednoduchém antialiasingu (samotný bod je rozostřen).

Rychlejší vykreslování je možné jednoduchým vykreslením bodů pomocí Z-bufferu a dodatečným vyplněním děr.