

Zkouška 31. 5. 2011

Phongův světelný model

- empirický světelný model – nemodeluje pomocí fyzikálních vztahů, pouze se je snaží napodobit
- scéna
 - vektor paprsku (dopadu) L (z bodu dopadu do zdroje)
 - normála N
 - vektor odrazu R (od bodu dopadu do prostoru)
 - vektor pozorovatele V (od bodu dopadu k pozorovateli)
 - barva materiálu C_M
 - barva světelného zdroje C_S
- osvětlení má 3 složky: ambientní (globální), difuzní (mat) a lesklá (specular)
 - ambientní
 - všesměrové konstantní světlo, eliminuje zcela černá místa, na která přímo nesvítí nějaký zdroj
 - simuluje sekundární odražené světlo
 - $E_A = C_M \cdot k_A$
 - difuzní
 - odpovídá osvětlení ideálně matného tělesa
 - $E_D = I_i \cdot C_M \cdot k_D \cdot \cos \alpha$ kde I_i je intenzita světelného zdroje a α je úhel dopadu paprsku (mezi normálou a paprskem)
 - $\cos \alpha = L \cdot N$ kde L i N jsou normované
 - lesklá
 - simuluje odlesk na povrchu lesklých těles
 - $E_S = I_i \cdot C_S \cdot k_S \cdot \cos^h \beta$ kde β je úhel pozorovatele vůči odraženému paprsku, pokud je nulový, pozorovatel se dívá přímo proti odráženým paprskům a odlesk bude nejsilnější
 - $\cos \beta = R \cdot V$ kde R i V jsou normované
 - $R = 2N(N \cdot L) - L$ (výpočet vektoru odrazu z normálového vektoru a vektoru dopadu)
 - $h \in (1, \infty)$ určuje jak ostrý bude odraz, čím větší, tím ostřejší odraz bude, předmět bude vypadat lesklejší, obvykle $h \in (5, 500)$
- každá složka má váhu k která určuje jak se projeví na výsledku
 - součet vah je roven jedné (ochrana proti přetečení): $k_A + k_D + k_S$
- osvětlení od jednoho zdroje: $E = E_A + E_D + E_S$

- více světelných zdrojů: $E = E_A + \sum_i (E_D + E_S)$, čím více světla je ve scéně tím méně je třeba ambientního světla
- oprava na vzdálenost: intenzita světla klesá se čtvercem vzdálenosti $1/d^2$, pak by byl příliš veliký rozsah hodnot (moc malé daleko a moc velké blízko), to monitor nezobrazí, používá se tedy $1/(c_0 + c_1 d + c_2 d^2)$, pak $E = E_A + \sum_i (E_D + E_S)/(c_0 + c_1 d + c_2 d^2)$

Ray-tracing

- ... neboli rekurzivní sledování paprsku
- řešení hloubky rekurze
 - statické – omezené konstantou
 - nevhodné pro scény obsahující zrcadla i nelesknoucí-se plochy, pro zrcadla je potřeba velká hloubka, pro matná tělesa malá
 - dynamické – dle významu paprsku (procentuální podíl právě sledovaného paprsku na výsledné barvě pixelu)
 - kombinované – omezení hloubky i významu
- výpočty průsečíků jsou nejnáročnější operace (až 90% času)
 - pokud se chce zrychlovat, tady je místo
 - obvykle probíhá dvoufázově, v první fázi se zjistí s jakým tělesem se protíná paprsek jako první a pak se už ke konkrétnímu průsečíku dopočítají všechny potřebné údaje jako normála, texturovací souřadnice, atd...

Adaptivní jittering

- jittering (roztřesení) je metoda pro vzorkování vykreslovaného obrazu za účelem zabránění zobrazení různých nežádoucích artefaktů
- jiné metody jsou např. pravidelné vzorkování, náhodné vzorkování, nezávislý jittering (N věží)
- $K \times K$ nezávislých náhodných vzorků v $K \times K$ shodných subintervalech pokrývajících původní interval beze zbytku
 - tj např. mřížka $K \times K$ a v každé buňce náhodný vzorek
- oproti náhodnému vzorkování omezuje velké shluky (a díry) vzorků, rovnoměrnější pokrytí intervalu, ale trochu “roztřesené” aby netrpělo na nežádoucí artefakty jako pravidelné vzorkování
- adaptivní znamená, že se přizpůsobuje aktuální situaci v obraze, tedy pokud je mnoho stejných vzorků, spokojí se s tím co má, pokud se sousední vzorky liší, v tom místě zvyšuje počet a hustotu vzorků
 - algoritmus se přizpůsobuje aktuální situaci (kterou neznal dopředu) – adaptibilita
- Poznámka: nestačí vysvětlit jittering a adaptivní supersampling zvlášť, je potřeba vysvětlit jak je zkombinovat

Mřížka pro akceleraci průsečíků

- rozdělení scény do uniformní mřížky, v každé vypočítat jaká tělesa do ní zasahují
- pak pro vržený paprsek stačí jít postupně od jeho začátku po buňkách mřížky do kterých zasahuje a u každé počítat průsečíky těles v ní obsažené
- obvykle stačí průsečík s prvním objektem, pak se prohledávání zastaví
- vhodné pro scény, kde jsou objekty celkem rovnoměrně rozděleny
- možné vylepšení
 - neuniformní mřížka – z počátku se zvolí mřížka velmi hrubá a buňky, kde je moc objektů se podrozdělí
 - u každé buňky si pamatovat jak daleko do okolí je volno, pak při testu této buňky lze skočit o tuto hodnotu dopředu (ne pouze o jednu)

Zkouška 3. 6. 2010

Nastiňte užití depth-bufferu pro vržené stíny, jaké má nedostatky a jak by se to dalo vylepšit

- nejprve proběhne výpočet z-bufferu pro pohled světla
- pak se při renderování pro každý pixel zjistí jeho vzdálenost od zdroje a porovná se s odpovídající hodnotou v z-bufferu světla a pokud je větší, je pixel zastíněn
- problémy
 - pokud je světelný zdroj daleko od objektu vrhající stín a kamera naopak blízko, jeden pixel v z-bufferu světla může být obrovský na kameře, stín je pak zubatý a zvláště zdeformovaný projekcí stínové mapy na terén
 - pro každý zdroj se musí renderovat scéna
- možné vylepšení
 - pokud je scéna statická, nebo alespoň její většina a světla se nepohybují, stínová mapa se může předpočítat na začátku renderování
 - pro malý počet dynamických objektů se pak můžou opakovaně “dorenderovávat” do předpočítané mapy jen dynamické objekty

Navrhněte reprezentaci paprsku a kužele a spočítejte průsečík paprsku s kuželem.

- paprsek: $P(t) = P_0 + t P_1$, $P_0 = [x_0, y_0, z_0]$, $P_1 = [x_1, y_1, z_1]$
- kužel (nekonečný): $x^2 + y^2 - z^2 = 0$
- průsečík (stačí dosadit paprsek do rovnice kužele):

- $(x_0 + t x_1)^2 + (y_0 + t y_1)^2 - (z_0 + t z_1)^2 = 0$
- dále umocnit a vyjádřit t

Navrhněte strukturu mramoru jako 3D textury a popište funkce, které na to použijete

- $mramor(x, y, z) = rampa(x + c \cdot snoise(x, y, z, p, n))$
- $rampa(x)$ je funkce, která vytváří podklad, který se pak šumem “kazí” tak, aby vypadal jako mramor, jelikož je mramor vrstvený materiál, vrstvy nejsou ideálně rovné, ale rozrušené, rampa by měla dělat nějaké ideální vrstvy, třeba $rampa(x) = \frac{1}{2}(1 + \sin(x))$ (to dělá sinus mezi 0 a 1)
- c je koeficient šumu, může být i záporný, tím se nastavuje jak moc má být struktura (rampa) rozrušena šumem
- $snoise(x, y, z, p, n) = \sum_{i=0}^{n-1} a_i \cdot noise(f_i x, f_i y, f_i z)$ je součet šumových funkcí
 - októva ne nazývá šumová funkce se změnou frekvence a amplitudy, tedy októva je 1 člen sumy
 - $p \in (0, 1)$ je persistence, která určuje rychlost klesání vlivu každého dalšího šumu (októvy) a platí $a_i = p^i$
 - f_i i-té októvy je $f_i = 2^i$
 - funguje to díky tomu, že perlin při volání s dvojnásobným parametrem $perlin(2x, 2y, 2z)$ vrací výsledek, který má dvojnásobnou frekvenci šumu
 - první člen součtu tedy vygeneruje nějaký základní hrubý šum, další se k němu přičítají, ale potlačeny o koeficient a_i , takže přidá jemnější šum s menší amplitudou

Popište hierarchická obalová tělesa, jak se kontrolují a jak se v nich hledá

- obalové těleso slouží pro urychlení hledání průsečíku, obvykle mají velmi jednoduchý tvar pro rychlý test na průsečík, pokud paprsek neprotíná ani obalové těleso je jasné, že neprotne ani původní
- více obalových těles se může zabalit do většího obalového tělesa a tak vzniká hierarchie
- takto lze postupně balit objekty ve scéně až zbude pouze jedno obalové těleso, které bude obalovat všechny objekty ve scéně
- hierarchie tvoří strom a při hledání průsečíku se postupuje od kořene k listům
- v ideálním případě snižuje asymptotickou složitost na $O(\log N)$ (to nastává když strom je vyvážený)

- vyplácí se pokud je scéna dobře strukturovaná a obalová tělesa se příliš nepřekrývají
- hierarchie je efektivní, pokud $K \cdot B + \sum_{i=1}^K p_i l_i < \sum_{i=1}^K l_i$ kde
 - K je počet nezabalených objektů (kořenů stromů)
 - B je čas výpočtu průsečíku s obalovým tělesem
 - p_i je pravděpodobnost zásahu i-tého obalového tělesa
 - l_i je čas výpočtu pro objekty v i-tém obalovém tělese

Zkouška 27. 5. 2010 a 8. 6. 2017

Jaké nabízí vylepšení Blinn-Phong oproti Phongově světelnému modelu

- značení navazuje na otázku ohledně phongova modelu
- zjednodušení výpočtů: $(\cos(\beta))$ pro lesklou složku
 - světelné zdroje v nekonečnu – v celé scéně L_i konstantní
 - pozorovatel v nekonečnu (rovnoběžná projekce) – v celé scéně konstantní vektor V
 - pak lze místo $(R_i \cdot V)^h$ psát $(H_i \cdot N)^{4h}$ kde $H_i = (L_i + V)/|L_i + V|$ je půlící vektor (konstantní v celé scéně), tento vztah je pouze aproximace, nicméně člověk téměř nepozná rozdíl

Máme scénu mnoha trojúhelníky, navrhnete a popište kroky algoritmu na urychlení detekce prvního průsečíku paprsek - trojúhelník [obalová tělesa se myslela]

- úkolem bylo neformálně popsat konstrukci datové struktury + jak probíhá vyhledávání (stačí pro jednu datovou strukturu)
- jedna možnost je (jak autor otázky uvádí) použít obalová tělesa
 - to se vyplatí v tom případě, když scénu tvoří oddělitelné shluky trojúhelníků
 - pak se vytvoří hierarchie obalových těles a paprsek se nejprve bude testovat oproti obalům a když je protne, pak bude testovat obsah (menší obaly nebo trojúhelníky)
- další možnost je rozdělit prostor do uniformních buněk a vypočítat jaké trojúhelníky zasahují do jaké buňky

- to se vyplatí pokud budou trojúhelníky celkem rovnoměrně rozmístěné po scéně a nebudou tvořit velké shluky
- při hledání postupně procházet buňky protnuté paprskem, v každé buňce projít seznam trojúhelníků a spočítat průsečíky
 - zastavit lze po kontrole průsečíků paprsku a všech trojúhelníků v buňce, pokud je nalezen průsečík, který leží uvnitř aktuální buňky
- rychlost prohledávání se dá ještě zlepšit o to, že nemusím jít buňku po buňce, ale u každé buňky si budu pamatovat jak velké okolí je určité prázdné a pak můžu skákat po větším počtu vpřed po paprsku
- také by se dalo prostor dělit neuniformně, např. pomocí oktantového stromu
 - algoritmus vyhledávání bude podobný jako v uniformím případě

Popište texturu dřeva a vzorce

- $wood(x, y, z) = perlin(x, y, z) - [perlin(x, y, z)]$ kde zápisem $[x]$ myslím dolní celou část
- vhodnou volbou harmonických složek ovlivníme vzhled struktury (perlin při volání s dvojnásobným parametrem $perlin(2x, 2y, 2z)$ vrací výsledek, který má dvojnásobnou frekvenci šumu)
- postup výpočtu perlina
 - rozdělení definičního oboru do pravidelné mřížky, vrcholy ozn. $[i, j, k]$
 - v každém vrcholu je definovaná pseudonáhodná funkce (vlnka), ta má tzv. poloměr za kterým je nulová, zároveň prochází počátkem
 - pro výpočet hodnoty v bodě nejprve určíme ve které buňce se souřadnice nachází
 - pro každý z osmi sousedů vypočítám hodnotu vlnky
 - hodnoty vlnky se sečtou
- alternativní řešení
 - $b(d(x + Noise1(x, y, z), y + Noise2(x, y, z), z + Noise3(x, y, z)))$, kde $b(x, y, z)$ je výsledná barva, $d(x, y, z)$ jsou souřadnice bodu vzhledem k ideální ose kmene a $NoiseX(x, y, z)$ jsou spojitě šumové funkce
 - (není funkce b jen $R^1 \rightarrow barva$ a d funkce vzdáleností od středu kmene?)

Popište ostrost objektivu a jak se implementuje

- pravděpodobně jde o hloubku ostrosti
- běžné objektivy zaostří na nějakou rovinu (kolmou na osu objektivu) a na obě strany od ní (tj. blíže a dále) se obraz stává čím dál tím více neostrý
- výpočet pomocí distribuovaného ray-castigu
 - pro každý pixel se vyšle více (nějak distribuovaných) paprsků
 - jednotlivé paprsky se vysílají ze zvětšené oblasti (o nějaký poloměr r) a nevysílají rovnoběžně, ale procházejí společným bodem na rovině zaostření (svazek všech

- paprsků tvoří jakoby kužel, který se zužuje k rovině zaostření, a druhý kužel, který na rovině zaostření začíná a pak se rozšiřuje)
 - velikost poloměru udává hloubku ostrosti, pro $r = 0$ je obraz úplně ostrý všude (posílá se n totožných paprsků)
 - pro bod v rovině zaostření žádný rozdíl, pro body blíže budou paprsky více rozptýlené než kdyby šly rovnoběžně (díky zvětšené oblasti startu) a body dál obdobně
- 2017: součástí otázky bylo také nutné navrhnout jak zkombinovat hloubku ostrosti s anti-aliasingem
 - místo posílání pouze n paprsků z různých směrů do jednoho bodu (v rovině zaostření) lze poslat navíc pro každý z nich m paprsků různými směry (v rámci jednoho pixelu)
 - efektivněji: vygenerovat počátek paprsku jako náhodný bod z okolí pozice kamery (jak je popsáno výše), pro něj vygenerovat náhodný směr (směr k aktuálnímu pixelu + náhoda, např. jako u jitteringu)
 - takový paprsek vzorkuje zároveň hloubku ostrosti i anti-aliasing

Skuska 14. 6. 2007

Co robít, keď nechceme flat-shading? [Gouraud, Phong]

- problém s konstantním stínováním je v ostrých přechodech, které ještě lidské oko zvýrazní (tzv. Machův efekt)
- Gouraudovo stínování
 - v umělých vrcholech tělesa spočítám normálové vektory a z nich osvětlení
 - uvnitř stěn počítám pomocí bilineární interpolací z vrcholů
 - výpočet normál v bodech z normál stěn
 - aproximací (průměr)
 - analyticky
 - skutečné vs pomocné hrany – v bodech na skutečných hranách se musí normála počítat opatrně, aby se těleso uměle “nezaoblilo” (viz válec a body na hraně podstavy)
 - problém interpolace barvy
 - špatně vystihuje maximum odrazu (zrcadlový odraz), pokud bude maximum uprostřed stěny, téměř se neprojeví, pokud bude na vrcholu, projeví se moc
 - není invariantní k otočení (souvisí s předchozím)
 - problémy s výpočtem normál (VVV)
- Phongovo stínování

- v umělých vrcholech tělesa spočítám normálové vektory
- uvnitř stěn dopočítám normálu v každém pixelu bilineární interpolací
 - k vypočítanému normálovému vektoru dopočítám osvětlení
- mnohem větší výpočetní náročnost
 - interpolace normály je složitější, normála musí být jednotková, počítání odmocniny
 - existuje aproximace bez nutnosti odmocňovat
 - výpočet světelného modelu je v každém pixelu

Priesečník paprsek-nekonečný válec, vymyslet reprezentaci a napsat vzorce.

- paprsek: $P(t) = P_0 + t P_1$, $P_0 = [x_0, y_0, z_0]$, $P_1 = [x_1, y_1, z_1]$
- válec: $x^2 + y^2 = 1$ (jedná se o rovnici kružnice ve 2D, ve 3D je to nekonečný válec podél osy z)
- průsečík (stačí dosadit paprsek do rovnice válce):
 - $(x_0 + t x_1)^2 + (y_0 + t y_1)^2 = 1$
 - dále umocnit a vyjádřit t

Bump mapa, co to je, ako sa zadava, nejake vzorce napísat.

- modulace normály
- napodobení nerovností na povrchu tělesa
- funkce $B(u, v)$ je funkce lokálního posunutí povrchu (+ ven z tělesa, - dovnitř)
- původní normála $N = U \times V$
- nová povrchová funkce (bere povrchové souřadnice a vrací normálu)

$$P'(u, v) = P(u, v) + \frac{B(u, v) \cdot N}{|N|}$$
 (pouhé posunutí o $B(u, v)$ ve směru normály)
- modifikované normály se vypočítají jako parciální derivace nové povrchové funkce
- bump mapa však může být zadána i ve formě textury, jen na místo RGB bude va pixelu $[x, y, z]$ normály

Mekke stíny & vzorkovani, navrhnut, popisat, nejake vzorce napísat.

- měkké stíny tvoří plošné světelné zdroje (bodové světelné zdroje dělají jen tvrdé (říká se "ostré", ale nechávám "tvrdé", páč mě to fakt pobavilo :-D) stíny)
- body, které vůbec nevidí na světelný zdroj jsou v úplném stínu (umbra), body, které vidí svět. zdroj jen částečně jsou v polostínu (penumbra)

- výpočet měkkých stínů se dělá pomocí distribuovaného ray-castingu
- každého pixelu se vyšle paprsek a při nárazu do tělesa se vyšle n nějak distribuovaných paprsků do obzorkovaného plošného zdroje (n vzorky, rovnoměrně)
- příspěvek nehomogenního plošného zdroje $I(A) = \int_{\text{plocha zdroje}} \int I_L(u, v) \cdot \text{vis}(A, u, v) du dv$
kde
 - $\text{vis}(A, u, v)$ je funkce viditelnosti, nabývá hodnoty 1 pokud $S(u, v)$ je vidět z bodu A, 0 jinak

Co by ještě mohlo být u zkoušky a není uvedeno v otázkách

Straussův světelný model

- používá intuitivní parametry, které jsou vcelku nezávislé (na rozdíl od Phongu, kde některé kombinace parametrů nedávají smysl)
- je fyzikálně věrnější (aproximuje Fresnelův vztah pro odraz světla)
- parametry materiálu
 - barva $C \in [RGB]$ (color)
 - hladkost $s \in [0, 1]$ (smoothness)
 - kovovost $m \in [0, 1]$ (metalness)
 - průhlednost $t \in [0, 1]$ (transparency) – 0 neprůhledný, 1 zcela průhledný
 - index lomu $n \geq 1$
- scéna
 - vektor paprsku (dopadu) L (z bodu dopadu do zdroje)
 - normála N
 - vektor odrazu R (od bodu dopadu do prostoru)
 - vektor pozorovatele V (od bodu dopadu k pozorovateli)
 - barva světelného zdroje C_s
- příspěvek intenzity směrového zdroje $I_{Di} = I_i \cdot (Q_d + Q_s)$ kde
 - I_i je intenzita světelného zdroje (skalár)
 - Q_d je rozptýlený (difuzní) odraz
 - $Q_d = (1 - m \cdot s) \cdot r_d \cdot C \cdot \cos \alpha$ kde $r_d = (1 - s^3) \cdot (1 - t)$ a α je úhel dopadu paprsku (mezi normálou a paprskem)
 - $\cos \alpha = L_i \cdot N$ kde L i N jsou normované
 - Q_s je lesklý odraz (specular)

- $Q_s = r_s \cdot C_s$ kde $r_s = r_j \cdot \cos^h \beta$, $h = 3/(1 - s)$ a r_j si ani nepřejte vědět přesně (viz slidy)

- celkové osvětlení bodu $I = I_A + \sum_i I_{Di}$ kde $I_A = I_i \cdot r_d \cdot C$ (I_i je intenzita) je ambientní složka (jako u phongu)

• Zdroje

- <http://cgg.mff.cuni.cz/~pepca>
- <http://forum.matfyz.info/viewforum.php?f=219>
- knížka: Moderní počítačová grafika
- http://wiki.gamedev.net/index.php/D3DBook:%28Lighting%29_Strauss — prechadza postupne od Phongu cez Cook-Torrance az k Straussovi (logickejsi postup nez v slajdoch, imho) a po ceste vysvetluje jednotlivé principy a členy rovní