Precomputed radiance transfer - pokračovanie

Matej Marko

20. apríla 2011

Úvod

V prvej časti dvojprednášky sme sa rozhodli riešiť problém výpočtu komplexného osvetlenia (tiene, odrazy, ...) v reálnom čase. Táto úloha je príliš zložitá, preto musíme použiť jeden z dvoch možných prístupov:

1. Zvoľniť požiadavky na to čo všetko chceme počítať. Tieto predpoklady môžme využiť pre rôzne zjednodušenia a triky.

2. Rozdeliť výpočet na predvýpočet a samotnú syntézu obrázkov. Keďže predvýpočet prebieha iba raz, musíme niektoré parametre zafixovať (materiály, geometria scény alebo aj polohu a smer pohľadu kamery).

PCA alebo SVD faktorizácia

Výsledkom predvýpočtu je matica, ktorá popisuje vzťah medzi osvetlením a výslednými farbami pixelov (resp. farbami vrcholov modelu). Využívame fakt, ktorý hovorí, že výsledné farby sú lineárnou kombináciou pixelov v mape prostredia.

Táto matica je však neúnosne veľká, preto sa ju snažíme rôzne skomprimovať. Jednou z možností je využiť SVD (Singular value decomposition). SVD je metóda, ktorá rozkladá maticu $\mathbf{I}$ rozmeru $p \times n$ na súčin 3 matíc $\mathbf{E}$ $(p \times p)$, $\mathbf{S}$ $(p \times n)$ a $\mathbf{C}$ $(n \times n)$.

$$\mathbf{I} = \mathbf{E} \times \mathbf{S} \times \mathbf{C}$$

Matice $\mathbf{E}$ a $\mathbf{C}$ sú ortonormálne. Matica $\mathbf{S}$ je diagonálna a na diagonále obsahuje tzv. singulárne hodnoty. Zvyčajne sú zotriedené podľa veľkosti. Počet nenulových singulárnych hodnôt je rovný hodnosti rozkladanej matice $\mathbf{I}$.

Ak najnižšiu singulárnu hodnotu zmeníme na 0, po prenásobení dostaneme maticu $\hat{\mathbf{I}}$, ktorá má nižšiu hodnosť ako $\mathbf{I}$, ale je k $\mathbf{I}$ najbližšie zo všetkých matíc s nižšou hodnosťou v zmysle L^2-normy. Takto môžeme vynulovať aj ďalšie singulárne hodnoty. Zároveň po vynulovaní hodnoty môžme vynulovať aj 1 stĺpec z $\mathbf{E}$ a 1 riadok z $\mathbf{S}$. Ich príspevok v súčine matíc by bol aj tak nulový. Často sa matice $\mathbf{S}$ a $\mathbf{C}$ vynásobia a uchováva sa iba výsledok.

PCA je metóda, ktorá pre vstupné dátové vetkory nájde novú bázu. Pre túto bázu platí, že počiatok je v priemere vstupných vektorov. Prvá os novej báze je orientovaná v smere, v ktorom majú dáta najvyššiu varianciu. Druhá os je na prvú kolmá a dáta pozdĺž nej majú druhú najvyššiu varianciu, atď. Pre 2D príklad ide o otočenie a posun bodov, tak aby boli najviac "roztiahnuté" pozdĺž osi X. PCA súvisí s SVD tak, že je možné vypočítať PCA pomocou SVD.

Predpočítaná matica prenosu radiancie má zvyčajne vysokú hodnosť a nedá sa dobre aproximovať pomocou výsledku z SVD. Môžeme využiť lokálnu koherenciu a redukovať hodnosti podblokov matice.

Local/Clustered PCA

Predpokladajme, že máme množinu bodov na povrchu telesa a každý obsahuje 3D informáciu (napr. normálový vektor). V prípade, že chceme túto informáciu komprimovať máme 2 základné možnosti:

1. Clustering - rezdelíme vektory do disjunktných množín podľa toho, ako sú si "podobné" a každú množinu reprezentujeme jedným spoločným vektorom.

2. PCA dekompozícia -povodné dáta reprezentujeme pomocou projekcie do lineárneho podprestoru ("zabudneme" najmenej významné osi PCA).

Spojením obidvoch postupov dostávame CPCA (Clustered PCA). Najprv vytvoríme clustery dátových vetkorov a každý z nich, nezávisle na ostatných, aproximujeme pomocou PCA. Tým, že v rámci clusterov sú dátové vektory blízko seba sa v následnej aproximácii dopustíme menšej chyby.

K-means algoritmus a Iterative PCA

K-means je často používaný clusterovací algoritmus. Predpokladá, že vieme spočítať vzdialenosť medzi dvoma vektormi a poznáme očakávaný počet clusterov - k. Jeho inicializácia spočíva v náhodnom výbere k vektorov, ktoré tvoria počiatočné centroidy. Následne opakujeme 2 kroky

1. Rozdelíme vetkory do clusterov podľa toho ku ktorému centroidu sú nabližšie.

2. Pre každý cluster vypočítame nový centroid (priemer hôdnot z clusteru).

Tieto kroky môžu iterovať pevný počet krát alebo dovtedy kým sa nové centroidy posunú dostatočne ďaleko od minulých. Výsledok je veľmi závislý na iniciálnej voľbe centroidov.

Vylepšenou variantov je K-means++. Prvý centroid sa vyberie náhodne a spočítajú sa vzdialenosti vektorov od neho. Tieto vzdialenosti sa použijú ako rozdelenie pravdepodobnosti pre náhodný výber druhého centroidu (čím ďalej da nachádza, tým je vetkor pravdepodobnejší). Vektory rozdelím podľa toho, ku ktorému centroidu sú bližšie a vypočítam vzdialenosti. Tie sa použijú pre výber tretieho centroidu, ... Opakovanie zastvíme po tom čo máme požadovaný počet clusterov. Nájdené centroidy slúžia ako inicializačný vstup pre K-means algoritmus.

Spojenie clusteringu a PCA popísané v predchádzajúcej časti nazývame Static PCA. Statické preto, že clustering beží iba raz, na začiatku. Jeho kritériom je vzdialenosť vektorov k centroidu. Uvedomomme si však, že aproximácia pomocou PCA je projekcia do podprestoru. Pri clusteringu by našim kritériom mala byť práve chyba tejto aproximácie. Tieto myšlinky využíva Iterative PCA, ktoré pužíva výpočet PCA vo vnútornom cykle clusteringu.

Wavelety

Použitie waveletovej transformácie pre reprezentáciu mapy prostredia využíva rovnaký princíp ako tranformačné metódy v kompresii obrázkov: vyjadríme dáta v inej báze, kde sú jednoduchšie oddeliteľné významné a menej významné zložky.

Waveletová transformácia rekurzívne zmenšuje obrázok (polovičné rozlíšenie = menej detailov) a ukladá waveletové koeficienty, ktoré vyjadrujú rozdiel medzi zmenšeným obrázkom a pôvodným. Tieto koeficienty vyjadrujú vysokofrekvenčnú informáciu. Podstatné je, že vačšina z nich je blízka 0 a teda ich ignorovaním sa veľmi neodlíšime od pôvodného obrázku. Ak teda použijeme waveletovú transformáciu na každú stenu mapy prostredia a “zabudneme” nevýznamné koeficienty, znížime veľkosť vektoru osvetlenia a tým aj veľkosť predpočítanej matice.

Použitie waveletov má v porovnaní so sférickými harmonickými výhodu v nižšej numerickej chybe pri rovnakom počte koeficientov. Na druhej strane pri prakticky použiteľných počtoch koeficientov je aproximácia sférickými harmonickými prirodzene rozmazaná, zatiaľ čo wavelety vytvárajú artefakty.

Direct-to-Indirect Transfer

Doteraz sme stále ako zdroj svetla používali mapu prostredia (Environment map). EM sú výborné pre realistické vnímanie materiálov, ale sú použiteľná iba pre exteriéry alebo malé objekty vo veľkých miestnostiach. DTIT je naopak vhodná pre interiéry osvetlené reálnymi svetelnými zdrojmi. Predopklady sú pevná geometria, materiály a kamera (osvetlenie sa môže meniť).

Táto metóda využíva tvrdenie: nepriame osvetlenie je lineárnou funkciou priameho osvetlenia. Ak dokážeme túto závislosť presunúť do predvýpočtu, stačí nám zistiť priame osvetlenie v niekoľkých vzorkovacích bodoch rozmiestnených na povrchu scény. A to dokážeme spočítať v reálnom čase. Maticu vyjadrujúcu závislosť nepriameho osvetlenia od priameho komprimujeme pomocou waveletovej transformácie. Práve nedostatočný počet waveletových koeficientov, ktorý je nutný pre praktické použitie, spôsobuje, v niektorých prípadoch, vytváranie artefaktov.