

Franny's Adobe

Il blog di
Francesco
Carucci

<< Prince Of Persia Classic su Xbox Live | Home | 070707 >>

3D For Dummies: Bump Mapping e Normal Mapping

Chissa' se daro' seguito a questa idea del 3D For Dummies; al novanta per cento no, ma questa e' la prima (e molto probabilmente ultima) puntata.

Leggendo recensioni e speciali sui videogiochi si leggono spesso nomi di esotiche tecniche come Bump Mapping, Normal Mapping, Parallax Mapping. Ma che cosa sono? Concetti al di la' della comprensione umana, dedicati a pochi guru, di solito provvisti di enormi occhiali? No, non lo sono, io porto le lenti a contatto.

Bump Mapping.

Una bumpmap e' un una texture dove per ogni texel (pixel della texture) e' memorizzata un'altezza in una qualche scala, invece di un colore. Oppure sono memorizzati due valori che indicano uno scostamento sull'asse x e sull'asse y (sono rappresentazioni equivalenti).

Per ogni pixel del poligono, si legge l'altezza associata dalla bumpmap e la si usa in "un qualche modo" per ottenere un qualche tipo di perturbazione del colore di un poligono che altrimenti varierebbe uniformemente, perche' il poligono per definizione e' "piatto". Ad esempio si puo' usare la bumpmap per modificare leggermente la posizione dalla quale si recupera il colore in una mappa di riflessione (la scena riflessa) per simulare l'acqua. Modificando la bumpmap ad ogni frame secondo un qualche parametro dipendente dal tempo si ottiene l'impressione delle onde. E' la tecnica usata per l'acqua di Halo 2 o Black&White 2 ad esempio (e tanti altri).

Nell'Environment Bump Mapping, si usa l'altezza presa dalla bumpmap per perturbare la posizione nell'environment map dal quale si recupera il colore (tecnica usata tantissimo in passato, ma soprattutto in Rogue Leader dei Factor 5 sul GameCube).

Normal Mapping.

Per capire la differenza fra Bump Mapping e Normal Mapping bisogna capiri prima che cos'e' una normale. La normale ad una superficie e' una freccia che per ogni punto della superficie punta dritto verso l'esterno. Rappresenta per certi versi la curvatura. La normale di un poligono e' per ogni punto una freccia che punta sempre nella stessa direzione: questo e' ovvio perche' un poligono e' "piatto". La normale di una sfera per ogni punto e' una freccia che punta all'esterno in una direzione che congiungi il centro della sfera con il punto in questione. Se disegnate un cerchio su un foglio e unite ogni punto del cerchio con il centro avete la direzione di ogni normale.

La normale si rappresenta in tre dimensioni come un vettore di tre numeri, ed ha sempre una lunghezza pari ad uno (quindi

 Search

Recent Comments

- THANGYOU
http://www.trsohbeti.net
http://www.sohbe...
by oyular
- thanks
by sohbete
- I am really curious about
what is behind Ctrl + Sc...
by xmc
- ciao.. non riguarda il post.
Poco tempo fa c'è sta...
by Michele
- Belle cose essere taggati se
vuoi ti taggo io
by Pompini Paolo

Archives

- dicembre 2007 (1)
- novembre 2007 (1)
- ottobre 2007 (3)
- settembre 2007 (6)
- agosto 2007 (4)
- luglio 2007 (2)
- giugno 2007 (2)
- maggio 2007 (1)
- aprile 2007 (1)
- marzo 2007 (1)
- febbraio 2007 (1)
- gennaio 2007 (6)
- dicembre 2006 (1)
- novembre 2006 (3)
- ottobre 2006 (6)
- settembre 2006 (2)
- giugno 2006 (4)
- maggio 2006 (3)
- aprile 2006 (6)
- marzo 2006 (8)
- gennaio 2006 (2)
- dicembre 2005 (3)
- novembre 2005 (7)
- ottobre 2005 (6)
- settembre 2005 (4)

Post Categories

- Programming
- General
- Black&White 2
- Off Topic

Syndication:

- RSS
- ATOM

Hosted by



bastano in realta' due numeri e un verso per rappresentare una normale).

La normale e' usata per calcolare quanta luce riceve un punto della superficie data la direzione della luce: se la normale in punto guarda direttamente verso la luce, questo punto e' illuminato al massimo, se guarda nella direzione opposta questo punto non e' illuminato. E' un modello di illuminazione molto semplificato usato nel 99.99% dei giochi. (Ma non in Perfect Dark Zero ad esempio). Varie complicazioni del modello di illuminazione dipendono sempre e comunque dalla normale. Ora, se un poligono ha la stessa normale in ogni suo punto perche' e' piatto, ne deriva che ogni punto del poligono e' illuminato allo stesso modo da una luce direzionale (il Sole ad esempio).

Ma se io per ogni punto del poligono prendessi da una texture map tre valori che mi rappresentano un vettore da "aggiungere" in qualche modo alla normale del poligono per far finta che il poligono non sia piu' piatto? Questo e' il Normal Mapping ed e' concettualmente piu' vicino alla tecnica che Blinn nel '76 (mi sembra) chiamo' Bump Mapping.

Quindi, quando si fa Normal Mapping, la texture map contiene tre valori (due in realta') che rappresentano una normale che varia per ogni punto del poligono che, una volta inserita nel modello di illuminazione, fa' si' che il poligono non appaia piu' piatto ma presenti tutta una serie di dettagli ad alta frequenza che spariscono nel momento in cui si toglie la luce (vedi Motorstorm).

Parallax Mapping.

Come visto prima, i dettagli ad alta frequenza nel Normal Mapping dipendono solo dalla luce e dalla sua posizione rispetto al poligono, ma non dipendono dalla posizione dell'osservatore rispetto al poligono stesso.

Nel Parallax Mapping si hanno in ingresso due texture, una che codifica la normale per ogni punto del poligono ed una che fornisce l'altezza di ogni punto del poligono (in realta' sarebbe teoricamente possibile ricavare la prima dalla seconda). Per ogni punto si legge la sua altezza dalla base del poligono, e poi si modifica la sua "posizione" nel mondo in base a questa altezza e alla direzione della telecamera rispetto al punto e si recupera il colore e la normale dalla diffuse map e dalla normal map in base a questo calcolo. Questo vuol dire che per lo stesso punto del poligono la sua normale e il suo colore saranno diversi a seconda della direzione dalla quale lo si guarda: simulando l'effetto di paralasse che si avverte guardando un muro, ad esempio, da due punti di vista differenti. Nota: diffidate quando qualcuno grida al parallax mapping guardando uno screen shot, perche' ci si accorge di questo effetto solo muovendo la telecamera ed e' impossibile distinguerlo da un banale normal mapping attraverso un solo screenshot.

Displacement Mapping.

Le tecniche precedenti lavorano tutte a livello di pixel e sono calcoli effettuati per ogni pixel della scena. Il Displacement Mapping e' una tecnica che si implementa a livello di vertice e consiste nell'associare ad un modello una texture map che per ogni suo texel codifica un'altezza; ogni vertice legge dalla height map la sua altezza e si "muove" lungo la sua normale di una quantita' pari a quel valore. Il Displacement Mapping non e' altro che una tecnica di compressione della geometria e serve a rappresentare un modello con certe caratteristiche con meno memoria. E' comunemente usato per rappresentare il terreno o l'acqua, molto piu' raramente per rappresentare modelli umanoidi o oggetti.

Ci sara' una seconda puntata? Mostratemi entusiasmo e scene

di giubilo e cerco di vincere la mia pigrizia.

posted @ sabato 16 giugno 2007 11.47

