

Costruzione di Interfacce Lezione 7 Shading

cignoni@iei.pi.cnr.it
<http://vcg.isti.cnr.it/~cignoni>

Il modello di Phong

- ❖ In termini matematici un modello di illuminazione viene rappresentato da un'**equazione di illuminazione** che descrive come ogni punto dell'oggetto è illuminato in funzione della sua posizione nello spazio
- ❖ Il processo di calcolo dell'equazione di illuminazione in uno o più punti dell'oggetto da illuminare si chiama *lighting*

Costruzione di Interfacce - Paolo Cignoni

2

Il modello di Phong

- ❖ Le formule che vedremo, elaborate da Phong intorno alla metà degli anni '70, si riferiscono a luce monocromatica (bianca)
- ❖ Questo perché è il caso più diffuso
- ❖ Si possono facilmente estendere a luci di colore qualsiasi esprimendo il colore della luce in termini delle sue componenti RGB e calcolando tre diverse equazioni di illuminazione per ognuna delle componenti

Costruzione di Interfacce - Paolo Cignoni

3

Il modello di Phong

- ❖ L'unico fenomeno fisico modellato è la riflessione diretta, non ci occupiamo della modellazione della rifrazione
- ❖ Le equazioni che vedremo, quindi, riescono a simulare solo il comportamento di materiali opachi e non di materiali trasparenti o semi-trasparenti

Costruzione di Interfacce - Paolo Cignoni

4

Componente emissiva

- ❖ Il modello di illuminazione più semplice, ma anche il meno realistico è quello che abbiamo, implicitamente, assunto fino ad adesso: ogni oggetto è dotato di una propria intensità luminosa, senza che vi siano fonti esterne di illuminazione

Costruzione di Interfacce - Paolo Cignoni

5

Componente emissiva

- ❖ Il risultato è quello di avere un mondo di sagome monocromatiche (a meno che i singoli poligoni di un poliedro non abbiano diversi colori)
- ❖ Questo modello elementare può essere descritto dall'equazione

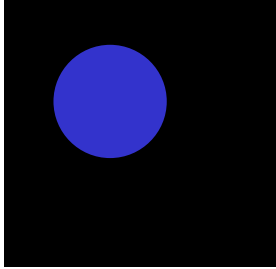
$$I = k_i$$

dove I è l'intensità risultante ed il coefficiente k_i è la luminosità intrinseca dell'oggetto

Costruzione di Interfacce - Paolo Cignoni

6

Componente emissiva



- ❖ Non essendoci termini dipendenti dalla posizione del punto si può calcolare I una sola volta per tutto l'oggetto

Riflessione diffusa

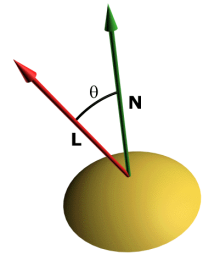
- ❖ Se supponiamo di posizionare nella nostra scena una sorgente luminosa puntiforme (*point light source*) i cui raggi sono emessi uniformemente in tutte le direzioni la luminosità di ogni singolo punto dell'oggetto dipende dalla sua distanza dalla sorgente luminosa e dalla direzione in cui i raggi incidono rispetto alla superficie

Riflessione diffusa

- ❖ Superfici opache come il gesso o il legno presentano un tipo di riflessione particolare: non modificano la loro apparenza al variare del punto di vista dato che riflettono la luce uniformemente in tutte le direzioni
- ❖ Questo tipo di comportamento si chiama **riflessione diffusa** (*diffuse reflection* o *Lambertian reflection*)

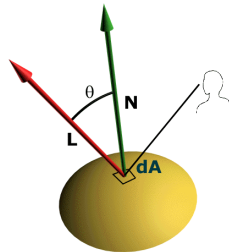
Riflessione diffusa

- ❖ In questo modello la luminosità dipende solo dall'angolo θ formato dalla direzione del raggio luminoso (L) e la normale alla superficie nel punto di incidenza (N)



Riflessione diffusa

- ❖ La quantità di luce che arriva all'osservatore è data (per superfici totalmente opache o lambertiane) dalla legge di Lambert

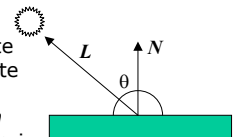


Riflessione diffusa

- ❖ L'equazione dell'illuminazione, per l'apporto dato dalla riflessione diffusa, è

$$I = I_p k_d \cos \theta$$

- ❖ I_p è l'intensità della sorgente luminosa e k_d è il coefficiente di riflessione diffusa del materiale (*diffuse reflection coefficient*) che assume valori tra 0 e 1



Riflessione diffusa

- ❖ L'angolo θ deve avere un valore compreso tra 0° e 90° per contribuire all'illuminazione del punto, in altre parole un punto della superficie non è illuminato da sorgenti luminose che stanno *dietro* di esso
- ❖ Se i vettori N e L sono normalizzati si può riscrivere l'equazione utilizzando il loro prodotto scalare

$$I = I_p k_d (\bar{N} \cdot \bar{L})$$

Riflessione diffusa

- ❖ Se la sorgente luminosa è posta a distanza infinita dall'oggetto che illumina, l'angolo tra i raggi di luce e tutte le superfici parallele è costante
- ❖ In questo caso la sorgente luminosa si dice *direzionale* (*directional light source*) e l'allineamento del vettore L è una costante della sorgente

Fattore di attenuazione

- ❖ Per tener conto anche dell'attenuazione dell'intensità dell'illuminazione all'aumentare della distanza si introduce anche un **fattore di attenuazione** (*attenuation factor*) f_{att} , inversamente proporzionale alla distanza della sorgente di luce dalla superficie

$$I = I_a k_a + f_{att} I_p k_d (\bar{N} \cdot \bar{L})$$

Fattore di attenuazione

- ❖ Questo fattore può essere essenziale, ad esempio, per distinguere nella scena due superfici parallele dello stesso materiale che nell'immagine si sovrappongono: senza bordo di separazione le due superfici sarebbero indistinguibili

Fattore di attenuazione

- ❖ Una tipica formulazione, abbastanza empirica, di f_{att} è

$$f_{att} = \min\left(\frac{1}{c_1 + c_2 d_L + c_3 d_L^2}, 1\right)$$

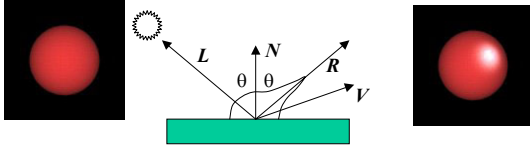
dove le costanti c_1 , c_2 e c_3 sono definite dall'utente e associate alla sorgente luminosa

Fattore di attenuazione

- ❖ Il termine fondamentale dipendente dal quadrato della distanza (d_L^2) è bilanciato dagli altri, soprattutto da c_1 , che, nel caso che la luce sia molto vicina, prevengono il denominatore dal divenire troppo piccolo
- ❖ L'espressione è comunque limitata ad 1 per evitare che diventi un fattore di amplificazione

Riflessione speculare

- ❖ Se la superficie di un oggetto non è totalmente opaca la luce non viene riflessa in maniera eguale in tutte le direzioni

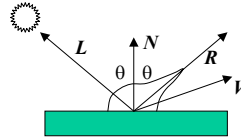


Costruzione di Interfacce - Paolo Cignoni

19

Riflessione speculare

- ❖ Da una superficie totalmente lucida, come uno specchio, la luce viene, infatti, riflessa solo nella direzione di riflessione R che, geometricamente, non è altro che L (direzione di incidenza) riflessa rispetto a N (normale alla superficie)

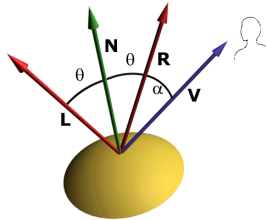


Costruzione di Interfacce - Paolo Cignoni

20

Riflessione speculare

- ❖ L'osservatore può vederla solo se la direzione di vista è allineata con la riflessione, cioè se l'angolo α è uguale a 0



Costruzione di Interfacce - Paolo Cignoni

21

Riflessione speculare

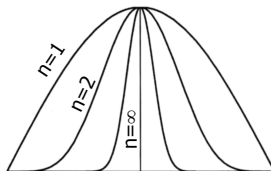
- ❖ Phong ha però sviluppato anche un modello di illuminazione per *riflettori non perfetti*, come ad esempio un oggetto di plastica o di cera
- ❖ Il modello assume che si abbia riflessione massima per $\alpha=0$ e che essa decada rapidamente all'aumentare di α

Costruzione di Interfacce - Paolo Cignoni

22

Riflessione speculare

- ❖ Un tale decadimento viene approssimato da $\cos^n \alpha$ dove n è detto esponente di riflessione speculare (*specular reflection exponent*) del materiale

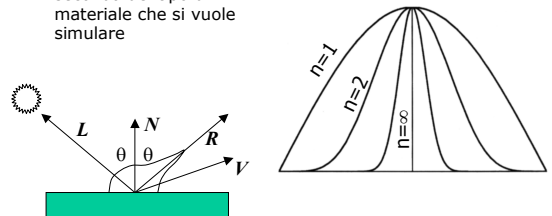


Costruzione di Interfacce - Paolo Cignoni

23

Riflessione speculare

- ❖ Il valore di n può variare tra 1 e valori molto alti (anche sopra a 100) a seconda del tipo di materiale che si vuole simulare

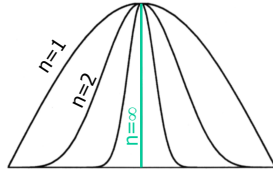


Costruzione di Interfacce - Paolo Cignoni

24

Riflessione speculare

- ❖ Una superficie perfettamente a specchio sarebbe teoricamente rappresentata da un n infinito



Riflessione speculare

- ❖ Aggiungendo la componente di riflessione speculare l'equazione di illuminazione diviene

$$I = f_{att} I_p [k_d \cos \theta + k_s \cos^n \alpha]$$

- ❖ dove k_s è il coefficiente di riflessione speculare (*specular reflection coefficient*) con intervallo di variazione $[0 \div 1]$

Riflessione speculare

- ❖ Se anche i vettori R e V sono normalizzati l'equazione si può riscrivere come

$$I = f_{att} I_p [k_d (\bar{N} \cdot \bar{L}) + k_s (\bar{R} \cdot \bar{V})^n]$$

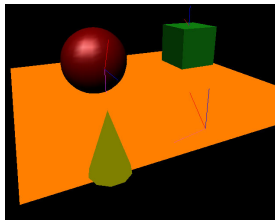
che rappresenta il nostro modello completo di equazione di illuminazione diretta

Componente ambientale

- ❖ Per aumentare il realismo della scena possiamo pensare che l'oggetto, anziché essere dotato di luminosità propria, sia illuminato da una sorgente di luce diffusa e non direzionale, quale il prodotto del riflesso della luce sulle molteplici superfici presenti nell'ambiente

Riflessione ambientale

- ❖ Oggetti illuminati con un modello di illuminazione comprendente solo il termine di illuminazione diffusa e speculare risultano non troppo realistici, come se fossero illuminati da una torcia in un ambiente altrimenti completamente oscuro



Componente ambientale

- ❖ Questo contributo si chiama **luce ambientale** (*ambient light*)
- ❖ Assumendo che il suo contributo non dipenda dall'orientamento delle superfici, l'equazione diviene

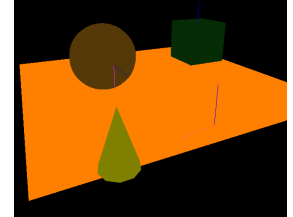
$$I = I_a k_a$$

Componente ambientale

- ❖ I_a è l'intensità della luce ambientale ed è costante per tutti gli oggetti
- ❖ L'ammontare di luce riflesso dalla superficie dell'oggetto è determinata dal coefficiente di riflessione ambientale k_a che varia nell'intervallo $[0÷1]$

Componente ambientale

- ❖ Gli oggetti illuminati da sola luce ambientale sono ancora uniformemente illuminati su tutta la loro superficie



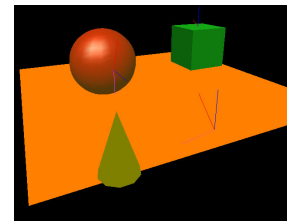
Riflessione diffusa e ambientale

- ❖ Dato che nella realtà si ha quasi sempre una certa luce di sottofondo (nell'ambiente) aggiungendo al modello anche il termine ambientale si ottiene un effetto più realistico

$$I = I_a k_a + I_p k_d (\bar{N} \cdot \bar{L})$$

Riflessione diffusa

- ❖ Alla scena precedente abbiamo aggiunto una luce ambiente



Lighting nella Pipeline

- ❖ L'equazione di illuminazione deve comunque essere valutata in coordinate di occhio perché sia le trasformazioni prospettiche che quelle di normalizzazione potrebbero portare N a non essere più perpendicolare alla superficie cui si riferisce.

Shading

- ❖ Utilizzando il modello di Phong si dovrebbe calcolare l'apparenza visuale (il colore) di ogni punto di ogni superficie nella scena in due passi:
 - ❖ calcolando la normale alla superficie nel punto
 - ❖ applicando il modello di illuminazione

Shading

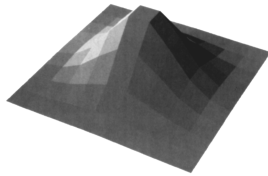
- ❖ Questo approccio è troppo dispendioso per essere adottato realmente e quindi si ricorre a metodi di *shading* (o determinazione dell'illuminazione) delle superfici approssimati ma molto meno costosi che operano sui poligoni come entità elementari anziché punto per punto

Flat shading

- ❖ Il modello più semplice di shading per un poligono è il *constant shading* (o *flat shading*)
- ❖ Consiste nell'applicare il modello di illuminazione scelto una sola volta per ogni poligono della scena e poi usare il valore determinato per l'intera superficie del poligono

Flat shading

- ❖ Una griglia di poligoni (in questo caso quadrilateri) rappresentata con flat shading



Flat shading

- ❖ Questo metodo aderisce al modello di Phong quando si verificano le seguenti condizioni:
 - ❖ La sorgente di luce è posizionata all'infinito (direzionale) cosicché $N \cdot L = k$ su tutta la superficie del poligono
 - ❖ L'osservatore è all'infinito (proiezioni parallele) cosicché $N \cdot V = k$ e $R \cdot V = k$ su tutta la superficie del poligono
 - ❖ Il poligono rappresenta effettivamente la superficie modellata e non è parte di un'approssimazione poligonale a tratti di una superficie curva

Flat shading

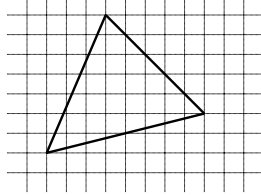
- ❖ La tecnica di flat shading è estremamente efficiente, poiché l'equazione di illuminazione va calcolata una sola volta per ogni poligono
- ❖ Tuttavia, il risultato visivo può non essere del tutto soddisfacente
- ❖ Se la *mesh* (griglia) di poligoni approssima una superficie più complessa lascia visibile la suddivisione in poligoni, senza rendere nell'immagine l'andamento della superficie approssimata

Shading interpolato

- ❖ In alternativa a valutare l'equazione di illuminazione in tutti i punti del poligono si può pensare di calcolarne i valori in ogni vertice ed interpolarli linearmente (all'interno dello spazio colore) per trovare i valori sui lati e nei punti interni
- ❖ Questa tecnica, dapprima proposta per triangoli, è stata in seguito estesa a poligoni generici

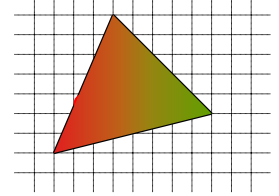
Shading interpolato

- ❖ All'interno di un algoritmo di rasterizzazione che determina i pixel che rappresentano la proiezione di un poligono 3D, applicare l'interpolazione anche dello shading comporta uno sforzo minimo



Shading interpolato

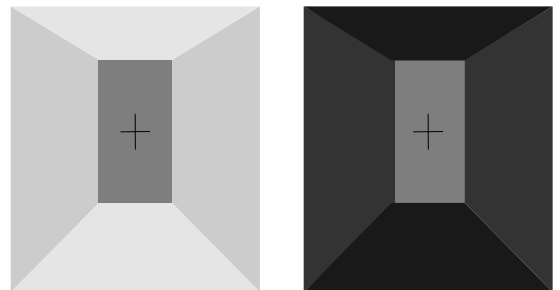
- ❖ Per ogni span il valore all'estremo può essere calcolato con un algoritmo incrementale, e, sempre incrementalmente, si possono calcolare i valori all'interno della span



Shading interpolato

- ❖ Questo non risolve però il problema di visualizzare correttamente una superficie curva con una mesh di poligoni
- ❖ Se lo shading (costante o interpolato) viene fatto indipendentemente su ogni poligono si ha comunque una netta visibilità, non voluta, dei bordi tra due poligoni adiacenti causata dalla brusca variazione della normale alla superficie

Effetto Mach banding

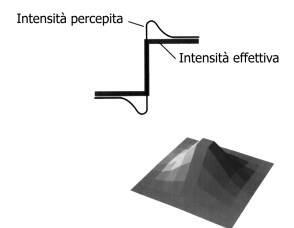


Effetto Mach banding

- ❖ A causa del cosiddetto effetto *Mach banding* (dal nome del suo scopritore) anche una maggiore finezza della griglia di poligoni non riduce le discontinuità di shading tra i poligoni adiacenti
- ❖ Questo effetto è quello per cui un oggetto messo vicino ad uno più chiaro risulta più scuro e messo vicino ad uno più scuro risulta più chiaro

Effetto Mach banding

- ❖ Questo è causato, fisiologicamente, dall'effetto inibitore laterale dei recettori dell'occhio: più luce un recettore riceve, più inibisce la risposta dei recettori vicini

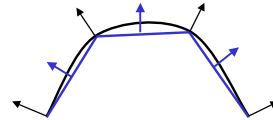


Gouraud shading

- ❖ Per ovviare a questo inconveniente si sono sviluppati dei modelli di shading che tengono conto delle informazioni date da poligoni adiacenti
- ❖ Il modello di Gouraud (*Gouraud shading*), è l'evoluzione diretta del metodo con interpolazione del colore su poligoni singoli

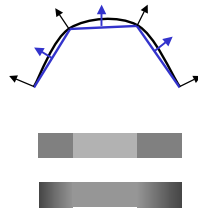
Gouraud shading

- ❖ Nel Gouraud shading si tiene conto della geometria effettiva che si sta visualizzando: se la griglia di poligoni rappresenta una superficie curva, per ogni vertice della griglia non si utilizza la normale al poligono ma la normale alla superficie che avresti voluto approssimare



Gouraud shading

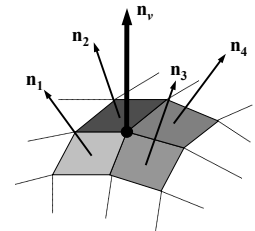
- ❖ In questa maniera il calcolo dello shading produce lo stesso valore su entrambi i lati di poligoni che hanno bordi in comune rendendo lo shading complessivo privo di *salti*



Gouraud shading

- ❖ Il metodo richiede che sia nota la normale alla superficie che si approssima in ogni vertice
- ❖ Se non è disponibile, si approssima con la media delle normali ai poligoni che condividono il vertice

$$\mathbf{n}_v = \frac{\mathbf{n}_1 + \mathbf{n}_2 + \mathbf{n}_3 + \mathbf{n}_4}{|\mathbf{n}_1 + \mathbf{n}_2 + \mathbf{n}_3 + \mathbf{n}_4|}$$



Gouraud shading

- ❖ Se uno spigolo deve essere visibile (ad esempio lo spigolo di raccordo tra la superficie dell'ala e della fusoliera di un aereo) si generano due insiemi di normali su ciascuno dei due lati dello spigolo

Gouraud shading

- ❖ Una volta calcolate (o ricavate analiticamente) le normali in ogni vertice si applica il modello di illuminazione per calcolare il valore di shading nel vertice e si interpola linearmente, all'interno dei poligoni, con lo stesso procedimento descritto in precedenza

Gouraud shading

- ❖ Dal punto di vista implementativo, il Gouraud shading è efficiente, poiché l'equazione di illuminazione va calcolata una sola volta per ogni vertice
- ❖ Per poter individuare i vettori normali necessari per calcolare la normale nei vertici, occorre tuttavia una struttura dati che rappresenti l'intera mesh di poligoni

Phong shading

- ❖ Il Gouraud shading è più che sufficiente per la maggior parte delle applicazioni, non risulta particolarmente realistico solo quando si vogliono rappresentare superfici dotate di un alto coefficiente di riflessione speculare
- ❖ Un modello molto più costoso è il modello di Phong (*Phong shading*) o shading con interpolazione delle normali

Phong shading

- ❖ Il Phong shading è più costoso in termini computazionali rispetto al Gouraud shading:
 - ❖ si interpolano vettori e non colori interi
 - ❖ l'equazione di illuminazione è calcolata per ogni pixel
- ❖ Mentre esistono implementazioni in hardware del Gouraud shading che consentono il suo uso in tempo reale, ciò non è normalmente vero per il Phong shading.

Alcune note

- ❖ Non confondere Phong model con Phong shading
- ❖ Phong shading è noto anche un sottoinsieme del più generico per-pixel shading

