

Lighting and Shading: il modello del Ray Tracing

Antonio Sirignano - Ciro Scognamiglio

Calcolo Scientifico per l'Innovazione Tecnologica - prof. Luisa D'Amore

Sommario—In questo progetto viene fatta un'introduzione generale al lighting e allo shading, con una presentazione dei modelli di riflessione e di shading che possono essere applicati alla grafica computazionale. Nel seguito viene presentato in maniera più approfondita il modello del Ray Tracing, con le sue applicazioni, in particolare al mondo del video game e il 3D video editing.

keywords—calcolo scientifico, lighting, shading, ray tracing.

Indice

| | | |
|-----|----------------------------|---|
| 1 | Introduzione | 1 |
| 2 | Le sorgenti di luce | 1 |
| 3 | Modelli di riflessione | 2 |
| 4 | Modelli di shading | 2 |
| 5 | Il modello del Ray Tracing | 2 |
| 6 | L'equazione di rendering | 3 |
| 7 | Scanline rendering | 3 |
| 7.1 | Z-buffering | 3 |
| 7.2 | Algoritmo del pittore | 3 |
| | Riferimenti | 4 |

1. Introduzione

La Computer Graphic Technology è l'abilità di produrre un effetto visivo realistico in un oggetto tridimensionale in un device di output bidimensionale, come un computer o un foglio stampato. Tutto ciò si ha grazie ai *metodi di rendering* nei quali è applicato lo *shading* per raggiungere il più possibile una rappresentazione di un oggetto vicina alla realtà.

Infatti lo shading computa quantità e colore della luce emessa da ogni punto della superficie.

Tali risultati dipendono dalle seguenti entità:

1. **La sorgente di luce.** Intensità, colore, forma, direzione e distanza della sorgente di luce devono essere prese in considerazione e inoltre possono essere sia puntiformi che di grandi dimensioni.
2. **La superficie dell'oggetto.** L'oggetto può essere lucido, liscio, ruvido, brillante o scuro. Può inoltre avere colori differenti quali opachi, trasparenti o traslucidi.
3. **L'ambiente.** Oggetti visti in uno spazio vuoto, senza un background che rifletta la luce su di essi, risultano duri (una navicella spaziale nello spazio profondo). Un modello realistico di shading deve tenere in considerazione la luce riflessa dagli altri oggetti (pareti vicine).

2. Le sorgenti di luce

Un oggetto illuminato dalla luce è colpito da raggi luminosi proiettati sulla sua superficie da un emittente chiamato *sorgente di luce*. La luce può descrivere diverse scene presentate di seguito.

1. **Luce ambientale.** Questa luce è una sorgente di luce non direzione la cui luce è emessa da ogni direzione. La sua intensità è indipendente da tutte le sue caratteristiche, come posizione e orientamento.

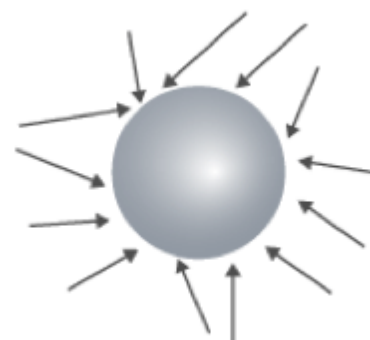


Figura 1. Luce ambientale

2. **Luce puntiforme.** È una sorgente di luce che non emette la stessa quantità di luce proveniente da tutte le direzioni, infatti un oggetto quanto è più vicino ad essa tanto è più luminoso. L'intensità della sorgente è quindi dipendente dalla distanza e dalla angolazione. È caratterizzata da colore, intensità, posizione e funzione di decadimento.

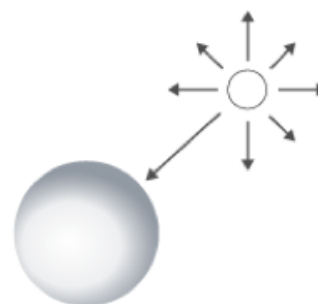


Figura 2. Luce puntiforme

3. **Luce direzionale.** Tale tipo è prodotta da una sorgente di luce da una distanza infinita dalla scena. Tutti i raggi di luce si espandono in una singola direzione e con la stessa intensità ovunque. È caratterizzata da colore, intensità e direzione.

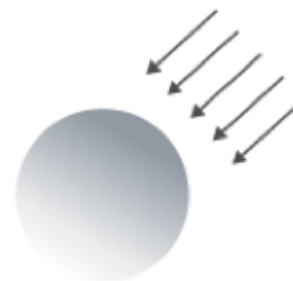


Figura 3. Luce direzionale

4. **Spotlight.** La luce si irradia in un cono con più luce al centro di esso. Questa luce è fissata all'asse primario di direzione con una restrizione su di essa. È caratterizzata come un punto di propagazione, un asse di direzione, un raggio intorno all'asse e la possibilità di una funzione di decadimento radiale.

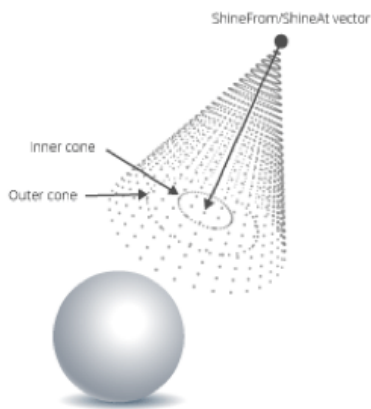


Figura 4. Spotlight

3. Modelli di riflessione

L'obiettivo principale dello shading è la produzione di un risultato accettabile quando la superficie dell'oggetto è affetta dai raggi di luce. I modelli di riflessione sono presentati di seguito.

1. **Riflessione diffusa.** Tale modello di riflessione diffonde la luce uniformemente in tutte le direzioni. In accordo con la legge di Lambert si ha che la diffusione del riflesso è proporzionale al coseno dell'angolo θ compreso tra la normale N e la direzione della sorgente L .

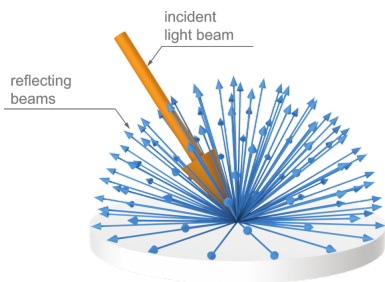


Figura 5. Riflessione diffusa

Da qui

$$\cos \theta = L \cdot N$$

E considerando il coefficiente di riflessione k_d , con $0 \leq k_d \leq 1$, si ha

$$R_d = k_d \cdot L \cdot N$$

Volendo considerare l'attenuazione della luce, si deve far riferimento alla distanza che essa percorre d . Quindi il termine di attenuazione quadratico è

$$R_d = \frac{k_d}{A + BD + C(D \cdot D)} (L \cdot N)$$

E quindi

$$I_d = \frac{k_d}{A + AD + A(D \cdot D)} (L \cdot N)$$

2. **Riflessione speculare.** Tale modello produce una riflessione luminosa sulla superficie dell'oggetto.

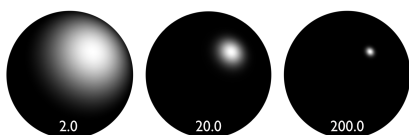


Figura 6. Riflessione speculare

L'angolo ϕ compreso tra la direzione della riflessione r e la direzione v dell'osservatore è affetto da una quantità di luce speculare. Il modello Phong stabilisce

$$R_s = k_s \cos^n \phi$$

per coefficiente k_s , con $0 \leq k_s \leq 1$. L'esponente n è il coefficiente di lucentezza. All'incrementare di n , la luce riflessa si concentra in una regione superficiale più piccola, centrata in r . Valori tra il 100 e il 500 corrispondono a superfici metalliche, valori più piccoli corrispondono a illuminazioni più ampie. Quindi

$$R_s = k_s (r \cdot v)^n$$

Una distanza viene moltiplicata similmente alle riflessioni diffuse.

La combinazione di tre tipi di riflessione danno

$$I = \frac{1}{A + BD + C(D \cdot D)} (K_d(L \cdot N)L_d + k_s(r \cdot v)nL_s) + K_aL_a$$

La direzione della riflessione r è semplice da n e I , assumendo che n sia un'unità di lunghezza, inoltre si vede che

$$\frac{L + r}{2} = (L \cdot n)n \implies r = 2(L \cdot n)n - L$$

4. Modelli di shading

I modelli di shading sono utilizzati per ottenere il modello di illuminazione desiderato. Modelli di shading efficienti per la superficie definita da un poligono si possono descrivere come seguono.

1. **Constant shading.** Questo modello è il più semplice modello di shading anche conosciuto come faceted shading o flat shading. Lo stesso colore è applicato su ogni intero poligono con un rendering veloce. L'equità della luce è usata una sola volta per poligono. Data una singola normale al piano, l'equazione della luce e le proprietà del materiale sono usate per generare un singolo colore. Il poligono avrà tale colore.
2. **Gouraud shading.** Questo modello è anche chiamato intensity interpolation shading o color interpolation shading. I colori sono interpolati attraverso il poligono e vi è la necessità di identificare ogni vertice. Il processo di rendering è più lento rispetto al modello flat. L'equità della luce è applicata ad ogni vertice e anche ogni colore è determinato dalla quantità di luce con le proprietà del materiale. La Scan-line interpolation è usata per assegnare un colore ad ogni punto di proiezione del poligono come la media pesata dei colori di dati vertici.
3. **Phong shading.** Tale modello è più realistico degli altri perché il suo algoritmo considera l'unione delle normali dei vertici ad ogni punto del poligono per avere una normale locale. Inoltre, il calcolo è applicato per avere un'illuminazione totale nel rendering.

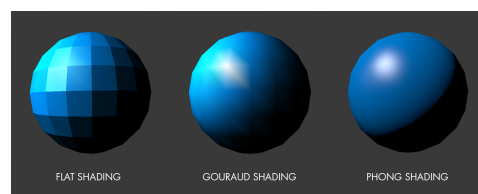


Figura 7. Flat, Gouraud e Phong shading

5. Il modello del Ray Tracing

L'ambizione degli artisti, dei designers, degli ingegneri è sempre stata la simulazione dell'immagine di una scena, una storia, un prodotto, un palazzo prima che sia effettivamente realizzati e costruiti - architetti devono costruire modelli in scala, produttori di film e

video-game devono utilizzare storyboards per provare e trasmettere le sensazioni di un film o un video-game. E gli inserzionisti vogliono creare interpretazioni perfette dei loro prodotti nella migliore luce possibile. Inoltre i designer di prodotti e macchinari vogliono trovare le debolezze di un progetto prima che esso venga effettivamente costruito - direttori di film vogliono vedere il risultato finale prima dell'arduo lavoro di post-produzione, i produttori vogliono mostrare a potenziali clienti il prodotto finale per stimolarne la domanda. Il test di tali idee è chiamato virtual prototyping nella manifattura e pre-visualization nell'industria cinematografica. Vi è quindi la necessità di disporre di queste immagini, o video, a basso costo e in poco tempo.

6. L'equazione di rendering

Nella computer graphics, l'equazione di rendering è un'equazione integrale nella quale la radianza di equilibrio che lascia un punto è la somma della radianza emessa più la radianza riflessa sotto un'approssimazione dell'ottica geometrica.

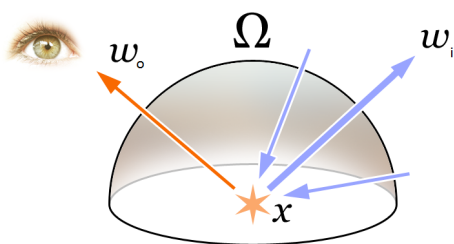


Figura 8. Equazione di rendering

Esistono svariate tecniche di rendering nella computer graphics che cercano di risolvere tale equazione. L'equazione di rendering descrive la quantità totale di luce emessa da un punto x lungo una particolare direzione di osservazione, data una funzione per la luce in entrata e una funzione di distribuzione bidirezionale di riflettanza (BRDF), dove,

- x è una posizione nello spazio
- w_o è la direzione della luce in uscita
- Ω è l'emisfero unitario centrato intorno $\{x\}$ contenente tutti i valori possibili di w_i
- w_i è il fattore di indebolimento dell'irraggiamento verso l'esterno a causa dell'angolo di incidenza, poiché il flusso luminoso si diffonde su una superficie la cui area è più grande dell'area proiettata perpendicolare al raggio

Risolvere l'equazione di rendering per ogni scena è la prima sfida per ottenere un rendering realistico.

Uno degli approcci è basato sui metodi degli elementi finiti, che ha portato all'algoritmo di radiosità. Un altro approccio utilizzando il metodo Monte Carlo ha portato a diversi algoritmi inclusi Path Tracing, photon mapping, il Metropolis light transport e molti altri. Monte Carlo ray tracing richiede come input la descrizione di una scena altamente dettagliata e basata sulla fisica delle cose. L'algoritmo applica le leggi fisica per simulare la propagazione della luce attraverso la scena, piuttosto che approssimazioni ad hoc per i fenomeni visivi. Questo tipo di simulazione richiede modelli geometrici estremamente dettagliati. Inoltre, rendering fotorealistici richiedono che le proprietà fisiche del materiale superficiale siano modellate correttamente, in modo da descrivere come la luce si disperde quando la colpisce.

7. Scanline rendering

Lo Scanline rendering è un algoritmo per la determinazione della superficie visibile che lavora riga per riga rispetto a un poligono per poligono o pixel per pixel.

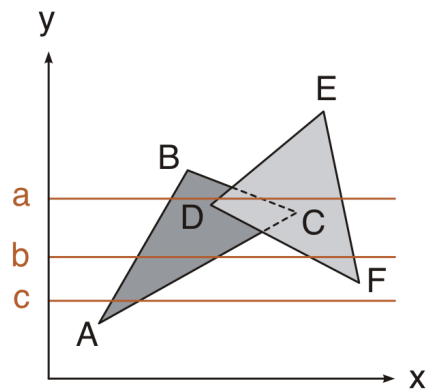


Figura 9. Algoritmo Scanline

Tutti i poligoni da renderizzare sono ordinate dall'alto sulla base delle ordinate in ordine di apparizione. Poi, per ogni riga o scanline dell'immagine è calcolato utilizzando l'intersezione di una scanline con i poligoni nella parte anteriore della lista ordinata, mentre quest'ultima viene aggiornata scartando i poligoni non più visibili man mano che la scanline attiva avanza verso il basso dell'immagine. Il vantaggio principale di questo metodo è che l'ordine dei vertici lungo la normale del piano di scansione riduce il numero di comparazioni tra i bordi. Un ulteriore vantaggio rispetto allo z-buffering è che il processo dei pixel visibili è mantenuto al minimo assoluto. Attraverso l'ordinamento front-to-back utilizzato nei moderni sistemi z-buffer, si possono realizzare benefici simili.

7.1. Z-buffering

Lo Z-buffering, anche noto come depth buffering, è la gestione delle coordinate di profondità delle immagini nella 3D graphics, fatta di solito a livello hardware. Quando si proietta un oggetto sullo schermo con un motore di 3D-rendering, la profondità (z-value) del pixel generato nell'immagine proiettata viene memorizzata in un buffer (z-buffer). Un z-value corrisponde alla misura della distanza perpendicolare da un pixel sul piano di proiezione alla sua corrispondente coordinata 3D su un poligono nello spazio. Lo Z-buffer ha la stessa struttura di un'immagine, ovvero un 2D-array, con la sola differenza che memorizza uno z-value per ogni pixel dello schermo invece che dei pixel dei dati. Ha le stesse dimensioni di uno screen buffer, eccetto quando sono utilizzati Z-buffer multipli.

Quando vediamo un'immagine contenente oggetti o superfici parzialmente o totalmente opachi, non è possibile vedere completamente questi oggetti sia che siano lontani dall'osservatore sia che siano dietro altri oggetti. L'identificazione e la rimozione di tali oggetti è chiamato problema delle superfici nascoste (hidden surface problem). Per migliorare il tempo di rendering, le superfici nascoste vengono rimosse prima che l'immagine sia memorizzata nel z-buffer, in modo che quest'ultimo calcoli lo z-value del pixel corrispondente al primo oggetto che compare e lo compara con lo z-value del pixel nella stessa posizione nel z-buffer corrispondente all'oggetto che è più vicino all'osservatore per trovare la sovrapposizione.

7.2. Algoritmo del pittore

L'algoritmo del pittore è una delle soluzioni più semplici al problema della visibilità nella 3D computer graphics. Quando proietti una scena 3D in un piano 2D, è necessario decidere quale tra i poligoni è visibile.

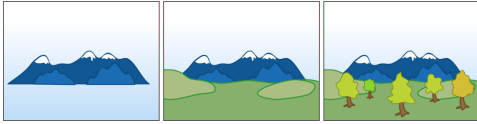


Figura 10. Algoritmo del pittore

L'algoritmo del pittore ordina tutti i poligoni nella scena partendo dalla profondità e poi li disegna in questo ordine, dal più lontano al più vicino. Si disegna sulle parti non visibili, al costo di avere oggetti già disegnati al nascosti. L'ordine usato dall'algoritmo è chiamato *depth order*: se un oggetto oscura la parte di un altro, allora il primo oggetto disegnato dopo il secondo viene oscurato.

Riferimenti bibliografici

- [1] B. Saleh, *Computer Graphics Fundamental: Lighting and shading*, mag. 2017. indirizzo: https://www.academia.edu/33137083/Computer_Graphics_Fundamental_Lighting_and_Shading.
- [2] J. Peddie, *Ray Tracing: A tool for all*. Springer, 2019.