

Generalità

8 giugno 2011

Informatica Grafica a.a. 2010-2011

- I raggi luminosi che provengono dalle sorgenti luminose:
 - subiscono riflessioni multiple tra gli oggetti
 - possono essere in parte assorbiti
 - incontrano superfici trasparenti che li trasmettono, filtrandoli.
- Nasce la necessità di modellare matematicamente il processo di riflessione della luce. A questo modello matematico si darà il nome di *equazione di illuminamento*.

Luce ambientale

8 giugno 2011

Informatica Grafica a.a. 2010-2011

- Si partirà con il caso acromatico.
- La luce ambientale è la luce ottenuta dalla continua inter-riflessione dei raggi luminosi tra le superfici: non si può più individuare una direzione di provenienza.

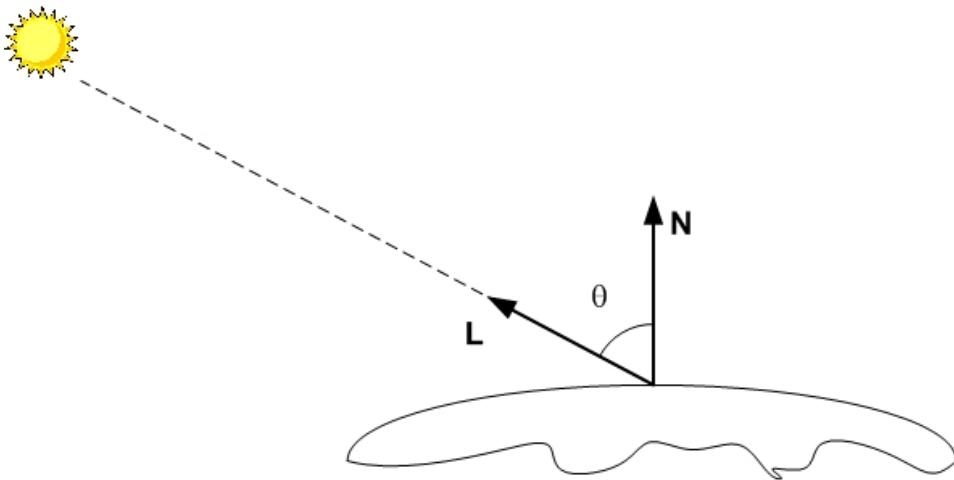
$$I = I_a k_a, \quad k_a \in [0,1]$$

- I_a è l'intensità di luce ambientale e k_a il coefficiente di riflessione da parte della superficie dell'oggetto.

Riflessione diffusa (lambertiana)

8 giugno 2011

Informatica Grafica a.a. 2010-2011



$$I = I_a k_a + I_p k_d \cos \theta = I_a k_a + I_p k_d \mathbf{N} \cdot \mathbf{L}$$

- Sorgente puntiforme ed all'infinito.
- Per una sorgente al finito **L** cambia da punto a punto della superficie.
- Riflessione dipendente solo dall'orientamento locale della superficie rispetto alla direzione di illuminazione.

Attenuazione con la distanza dalla sorgente

8 giugno 2011

Informatica Grafica a.a. 2010-2011

- L'energia radiante da una sorgente luminosa puntiforme decresce con una legge inversamente proporzionale alla distanza dalla sorgente stessa.
- Al crescere della distanza la potenza radiante si distribuisce su sfere di raggio crescente, quindi la potenza per unità di area decresce.
- Ciò si può modellare con un fattore di attenuazione che modula l'intensità luminosa della sorgente.

$$f_{att} = \min\left(\frac{1}{c_0 + c_1d + c_2d^2}, 1\right)$$

Estensione al caso cromatico

8 giugno 2011

Informatica Grafica a.a. 2010-2011

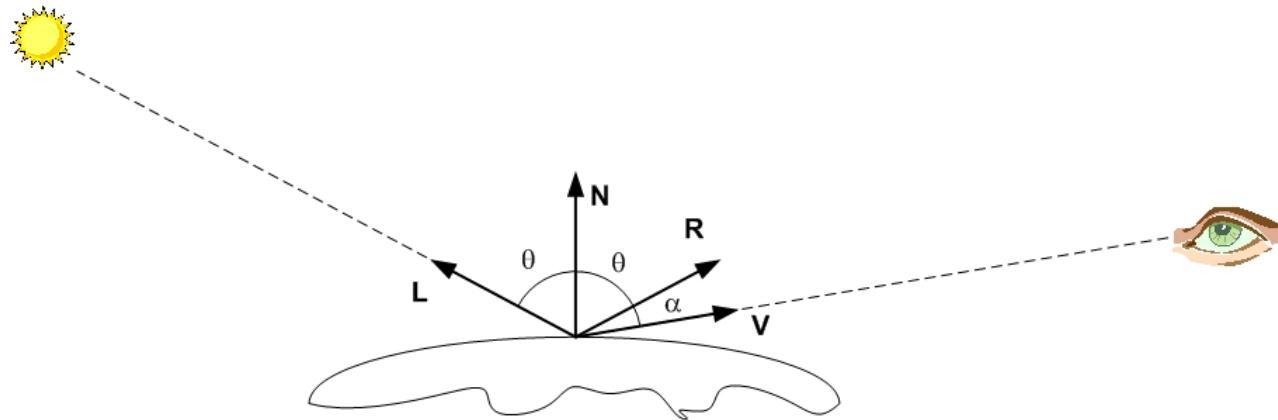
- Si dovrà tenere in considerazione una sorgente luminosa $I_p = \{I_{p\lambda}\}$, $\lambda = R, G, B$.
- Anche la superficie sarà fatta di un materiale con un suo comportamento di riflessione diffusa diverso per i tre canali:
 $O_d = \{O_{d\lambda}\}$, $\lambda = R, G, B$.
- Tali coefficienti moduleranno l'intensità luminosa, canale per canale.

$$I_\lambda = I_{a\lambda} k_a O_{d\lambda} + f_{att} I_{p\lambda} k_d O_{d\lambda} (\mathbf{N} \cdot \mathbf{L})$$

Riflessione speculare

8 giugno 2011

Informatica Grafica a.a. 2010-2011



- La riflessione speculare perfetta è quella addensa la luce riflessa nella direzione del vettore **R**, simmetrico di **L** rispetto alla normale **N**.
- Essa dipende dalla direzione di osservazione **V** e viene rappresentata utilizzando il modello di Phong.

$$I_{\lambda} = I_{a\lambda} k_a O_{d\lambda} + f_{att} I_{p\lambda} \left[k_d O_{d\lambda} (\mathbf{N} \cdot \mathbf{L}) + W(\theta) \cos^n \alpha \right] =$$

$$I_{a\lambda} k_a O_{d\lambda} + f_{att} I_{p\lambda} \left[k_d O_{d\lambda} (\mathbf{N} \cdot \mathbf{L}) + k_s O_{s\lambda} (\mathbf{R} \cdot \mathbf{V})^n \right]$$

Shading

8giugno 2011

Informatica Grafica a.a. 2010-2011

- Il processo di shading consiste nella definizione del colore di ogni singolo pixel appartenente ad un poligono che approssima una superficie.
- La scelta avviene sulla base del modello di illuminamento e del tipo di materiale prescelto per la superficie stessa.

Tipi di shading

8giugno 2011

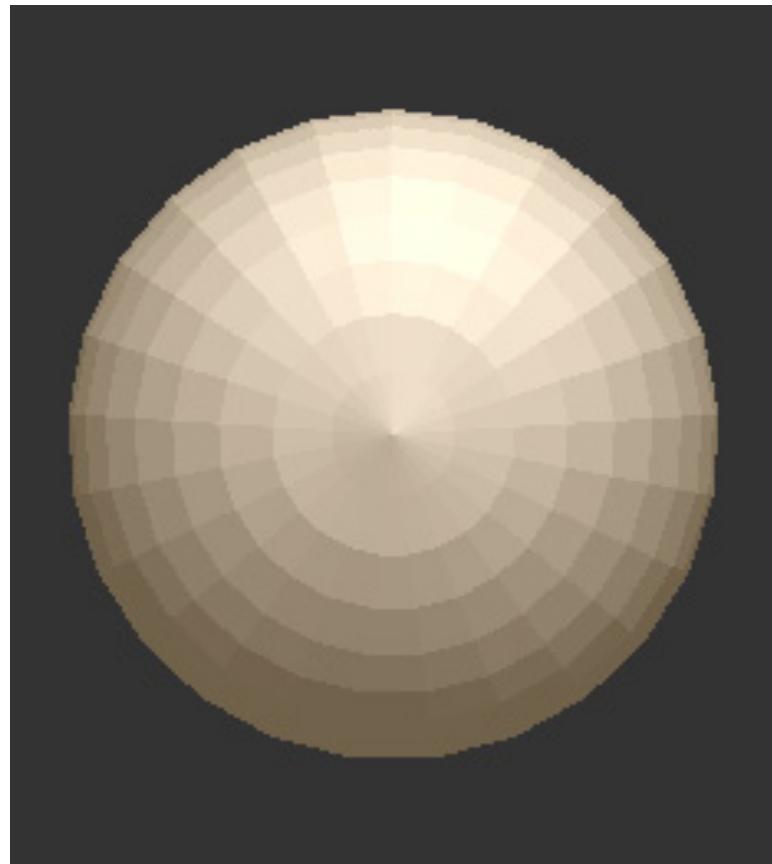
Informatica Grafica a.a. 2010-2011

- Il problema si riconduce al calcolo dell' illuminamento in maniera più o meno accurata, in dipendenza dalla conoscenza o meno delle normali locali alla superficie.
- Si hanno tre algoritmi principali:
 - Flat shading
 - Shading di Gouraud
 - Shading di Phong

Flat Shading

8giugno 2011

Informatica Grafica a.a. 2010-2011



- Nel flat shading ogni poligono viene approssimato tenendo conto di un'unica normale per tutti i suoi punti.
- Questa viene ricavata dall'equazione del piano che contiene il poligono.

Gouraud Shading

8 giugno 2011

Informatica Grafica a.a. 2010-2011

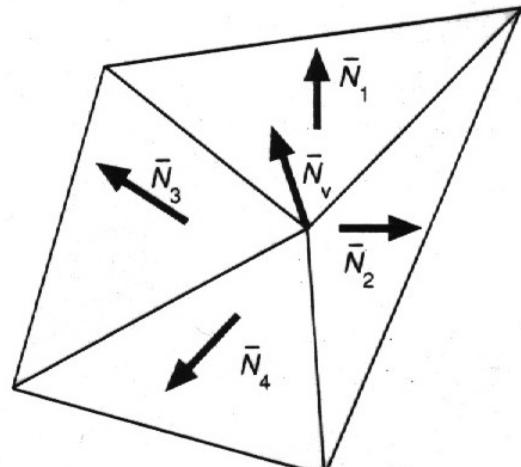


- Nel Gouraud shading, l'intensità luminosa di ogni pixel del poligono viene approssimata bilinealmente a partire dal calcolo esatto fatto in ogni vertice.
- Quando non si conosce la normale in un vertice si utilizza la media delle normali ai poligoni che lo condividono.

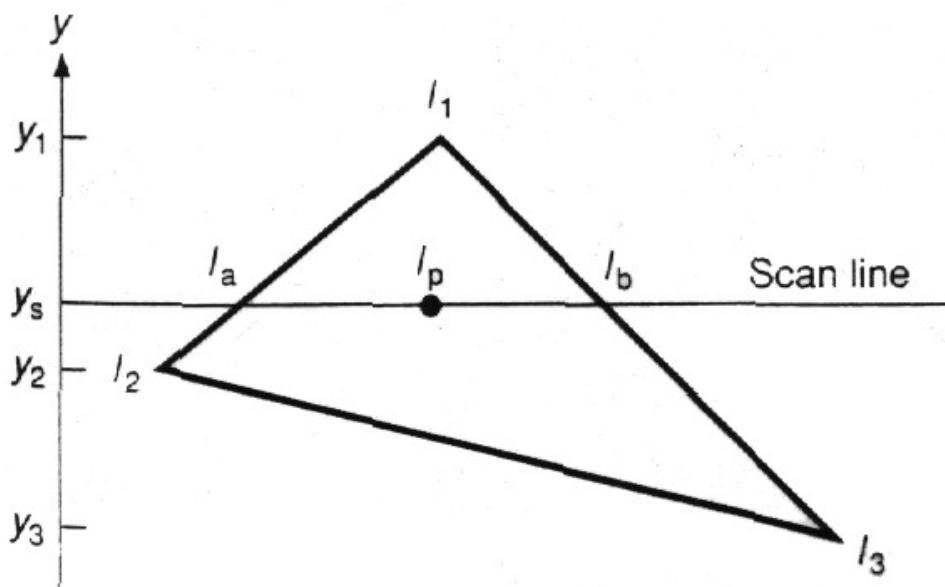
Gouraud Shading

8 giugno 2011

Informatica Grafica a.a. 2010-2011



$$N_v = \frac{\sum_{i=1}^n N_i}{\left| \sum_{i=1}^n N_i \right|}$$



$$I_a = I_1 - (I_1 - I_2) \frac{y_1 - y_s}{y_1 - y_2}$$

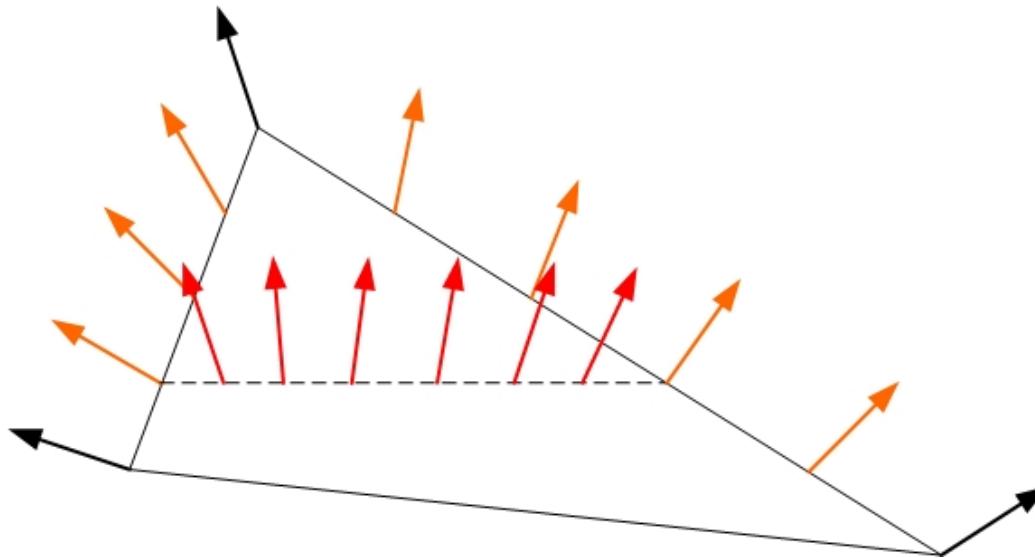
$$I_b = I_1 - (I_1 - I_3) \frac{y_1 - y_s}{y_1 - y_3}$$

$$I_p = I_b - (I_b - I_a) \frac{x_b - x_p}{x_b - x_a}$$

Phong Shading

8 giugno 2011

Informatica Grafica a.a. 2010-2011



- Nello shading di Phong, note le normali ai vertici, queste vengono interpolate bilinearmente su tutto il poligono e da ogni normale si calcola il valore di illuminamento.
- Consente di ottenere performance migliori per la riproduzione della riflessione speculare, ma è molto più oneroso dal punto di vista computazionale.

Estrazione di iso-superfici (marching cubes)

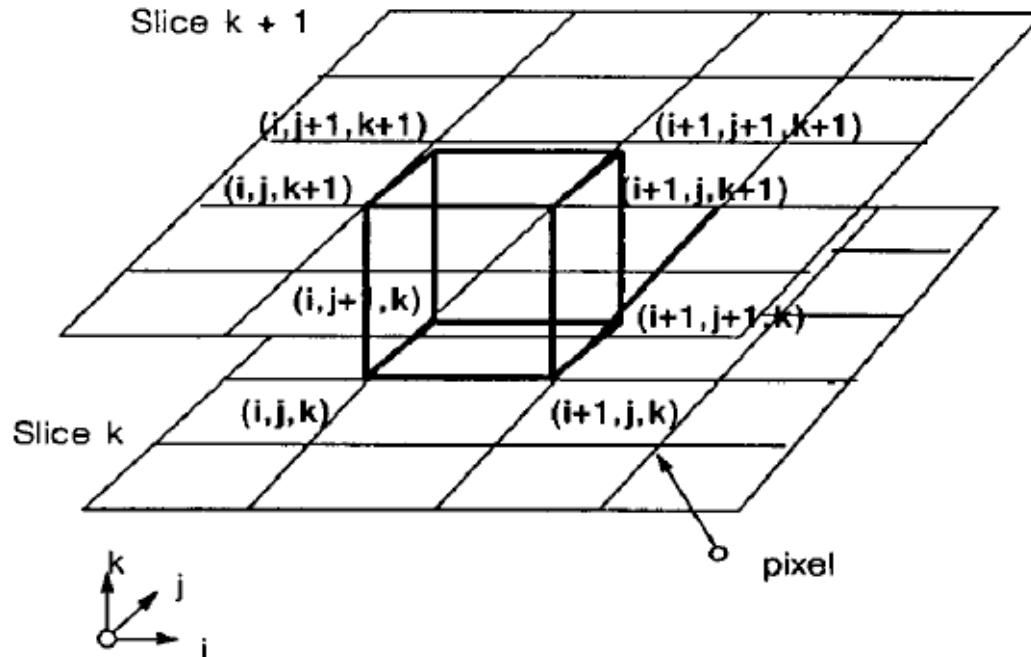
11 ottobre 2012

Informatica Grafica a.a. 2012-2013

- Lorenson e Cline (1987)
- Questo tipo di approccio è utilizzato soprattutto per visualizzazione o per costruire delle realtà virtuali di campi operatori e non per diagnostica.
- Si tratta di determinare come una iso-superficie $f(i,j,k)=c$, rappresentata mediante poligoni, attraversa il singolo voxel.

Marching cubes (2)

Informatica Grafica a.a. 2012-2013



- Ogni cubo è costruito connettendo otto pixel di due slice contigue.
- Per cui si conosce il valore di $f(i,j,k)$ ad ogni vertice del cubo.

Marching cubes (3)

11 ottobre 2012

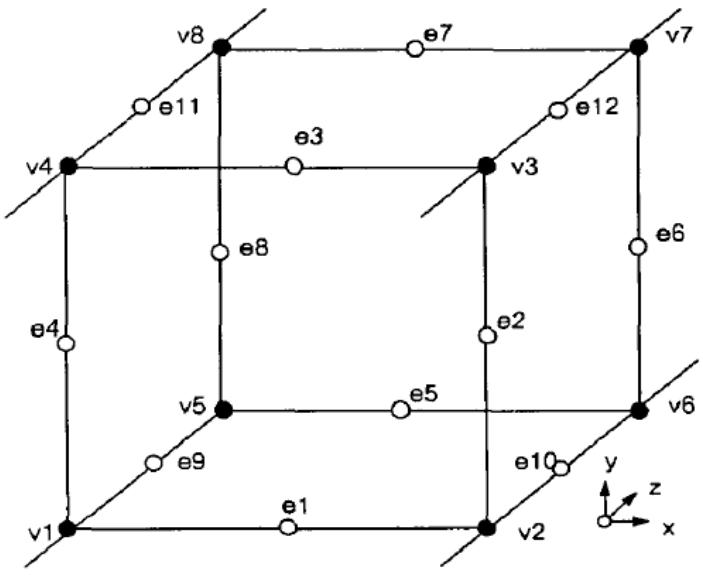
Informatica Grafica a.a. 2012-2013

- Ci sono 256 (2^8) possibilità di attraversamento
 - 8 vertici che possono essere dentro (1) o fuori (0) dalla superficie
- Mediante considerazioni di simmetria, le possibilità di attraversamento si riducono a 14 più il caso nullo con tutti i vertici esterni.
- Ad ogni vertice di ogni voxel è assegnato un valore in termini dell'informazione rappresentata dal volume: la iso-superficie rappresenta tutti i punti caratterizzati da un unico valore.
- L'algoritmo “marcia” da un voxel (cubo) all' altro e:
 - Individua la modalità di attraversamento (vertici del voxel interni ed esterni alla superficie)
 - Calcola l' effettiva posizione dei poligoni nel voxel per mezzo di una interpolazione lineare sugli edge intersecati.

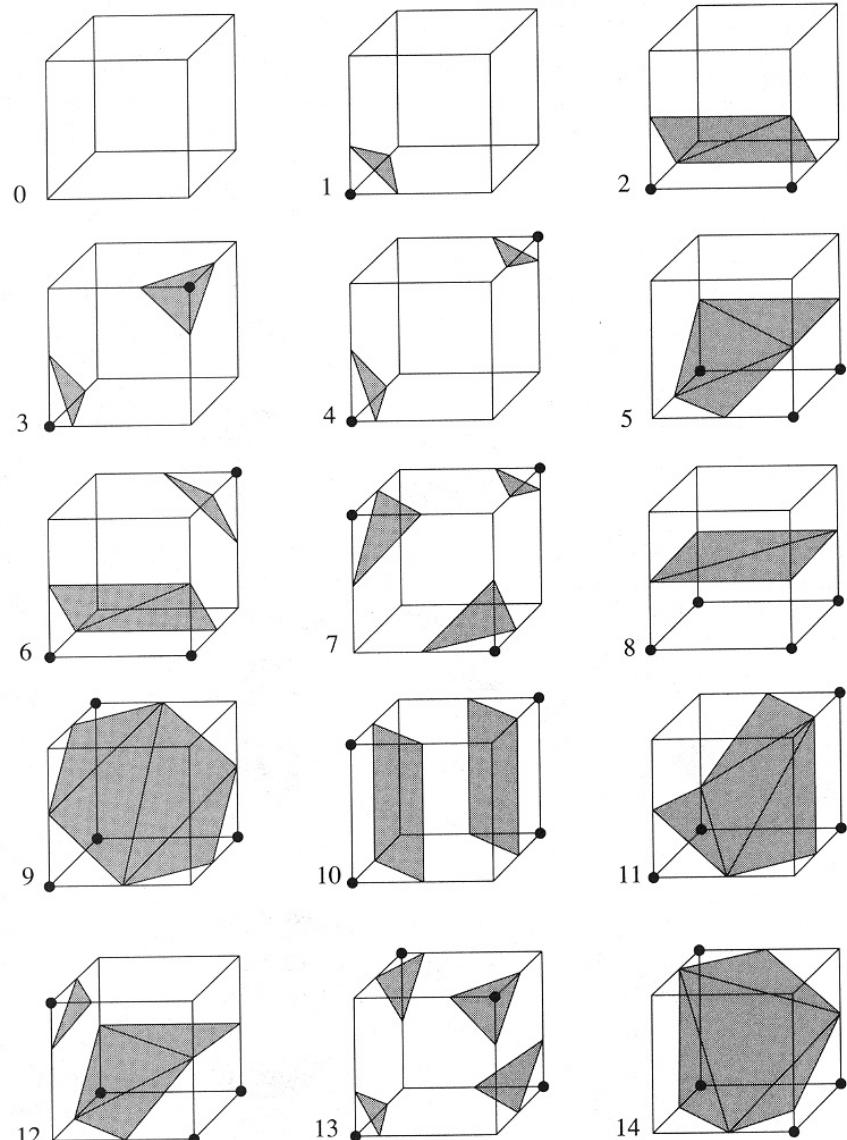
Marching cubes (4)

11 ottobre 2012

Informatica Grafica a.a. 2012-2013



Indice ad 8 bit che identifica
la tipologia di attraversamento e
punta alla tabella degli edge
numerati come in figura



Ray Casting

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

- Ogni voxel è determinato da un colore **C** e da una trasparenza α .
- **C** può essere un falso colore e rappresenta il contenuto informativo del voxel (ad es. il tipo di materiale, il particolare tessuto, etc.)
- α è legata alla densità dell' elemento di volume che, in questo contesto, non è mai interamente vuoto o pieno.

Ray Casting

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

- L' algoritmo di ray casting proietta dei raggi paralleli attraverso un volume, opportunamente ruotato e traslato secondo le esigenze di visualizzazione.
- I raggi attraversano il volume e raccolgono le informazioni di colore dei vari voxel, pesate con la loro trasparenza (*compositing*)
- Si può pensare di utilizzare dei criteri di raccolta delle informazione differenti
 - Valor massimo
 - Valor medio
 - Scostamento dal valore massimo o minimo

Ray Casting

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

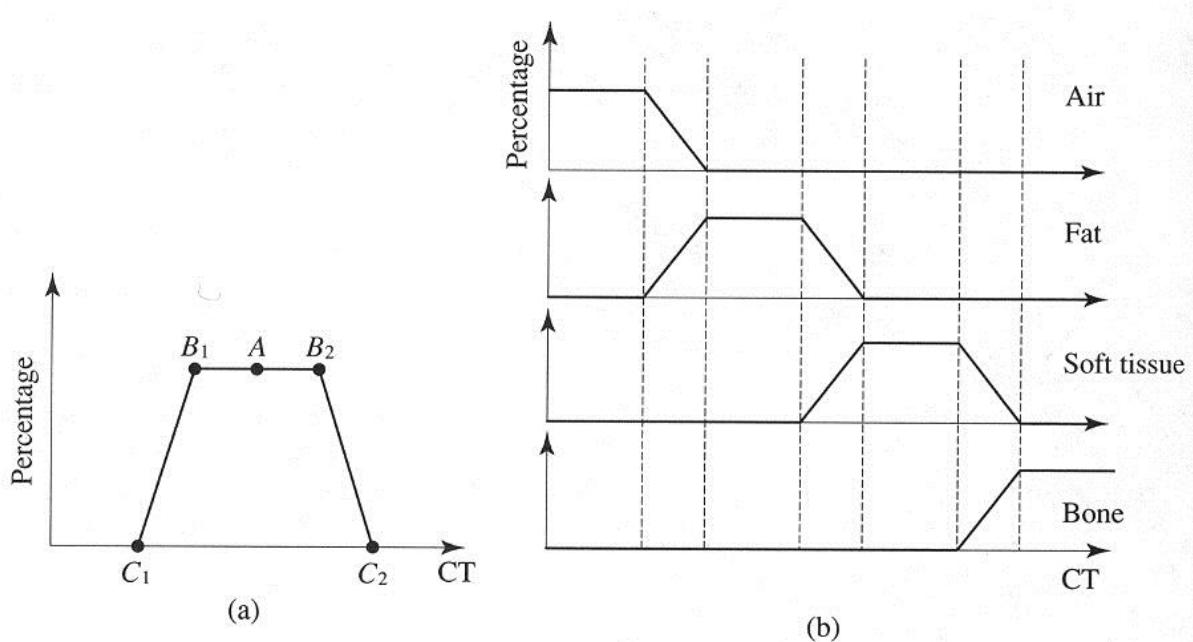
■ I passi dell' algoritmo sono

- Classificazione dei voxel per attribuire l' informazione di colore (ad es. un voxel ricade in una zona comune a più tessuti)
- Trasformazione geometrica del volume di vista
- Proiezione dei raggi e raccolta delle informazioni

Classificazione dei voxel

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014



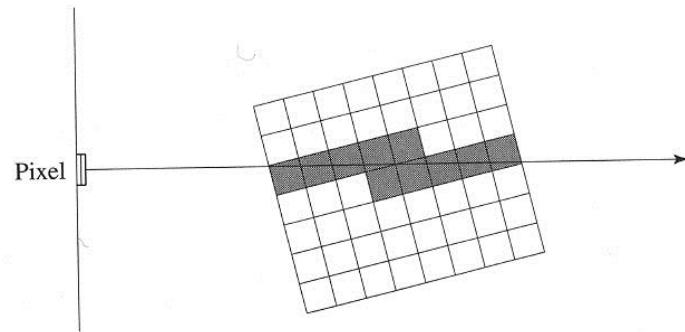
- Una tecnica (usata nel medical imaging) è quella di attribuire una probabilità di appartenenza di un voxel ad un dato tessuto (o materiale) a seconda del valore del segnale acquisito.
- Il colore e la trasparenza del voxel si compongono in rapporto alla percentuale di appartenenza ai vari tessuti.

Trasformazione lungo la direzione di vista

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

- Il problema della trasformazione del volume implica il fatto che un raggio non attraverserà una fila di voxel, ma seguirà un percorso attraverso voxel adiacenti
- Si ripropongono i problemi di aliasing e di interpolazione già visti nel caso di disegno di primitive 2D.



Compositing

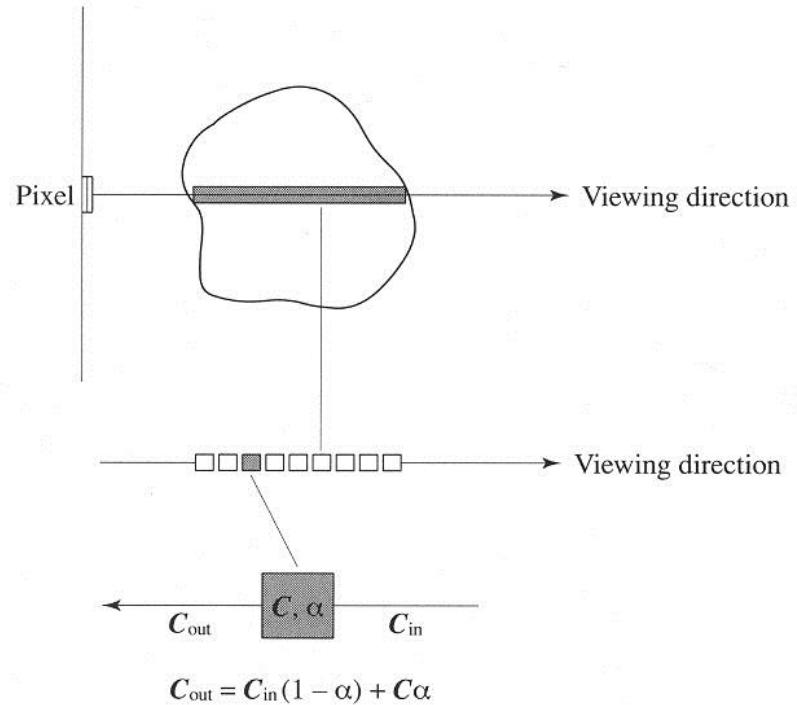
21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

- Per ogni voxel si applica una tecnica di blending secondo lo schema classico:

$$\mathbf{C}_{out} = \mathbf{C}_{in} (1 - a) + \mathbf{C} a$$

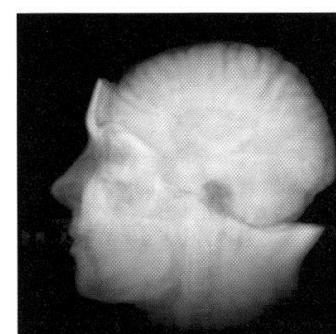
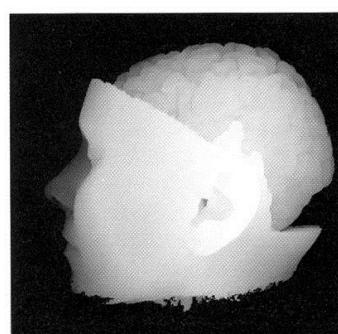
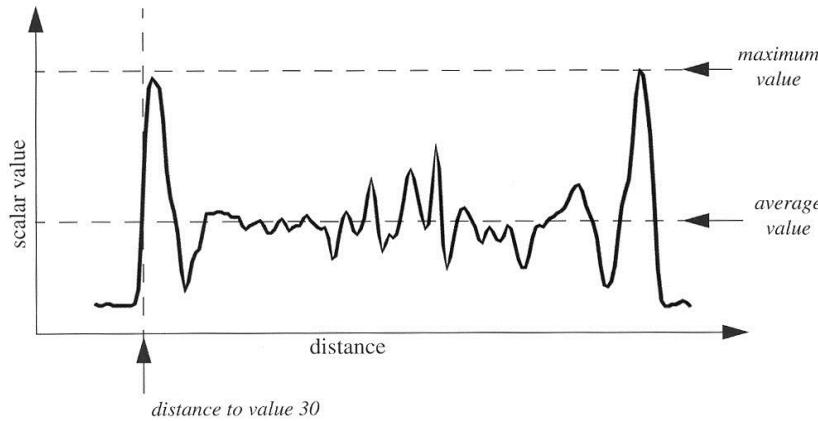
- Alla fine il colore del pixel sarà la “composizione” pesata dalle trasparenze dei colori dei voxel attraversati dal raggio (radiografia digitale)



Esempi di ray casting

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014



Compositing e shading

21 settembre 2013

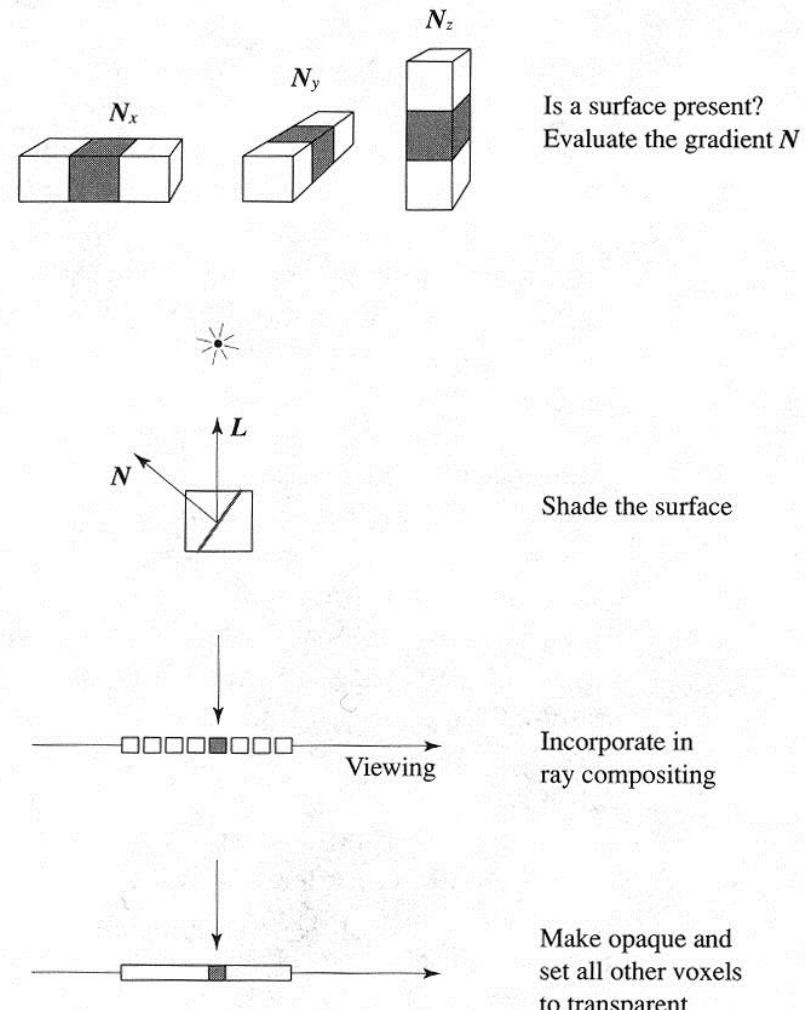
Corso di Informatica Grafica a.a. 2013-2014

- Si può pensare di calcolare una normale alla ipotetica superficie passante per un voxel a partire dal gradiente dei valori dei voxel in un intorno.

$$N_x = R(x+1, y, z) - R(x-1, y, z)$$

$$N_y = R(x, y+1, z) - R(x, y-1, z)$$

$$N_z = R(x, y, z+1) - R(x, y, z-1)$$



- Migliori stime si possono ottenere per intorni più ampi.

Differenze tra i vari tipi di ray casting

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014



Maximum intensity



Composite (unshaded)



Composite (shaded)

Discretizzazione di un raggio

21 settembre 2013

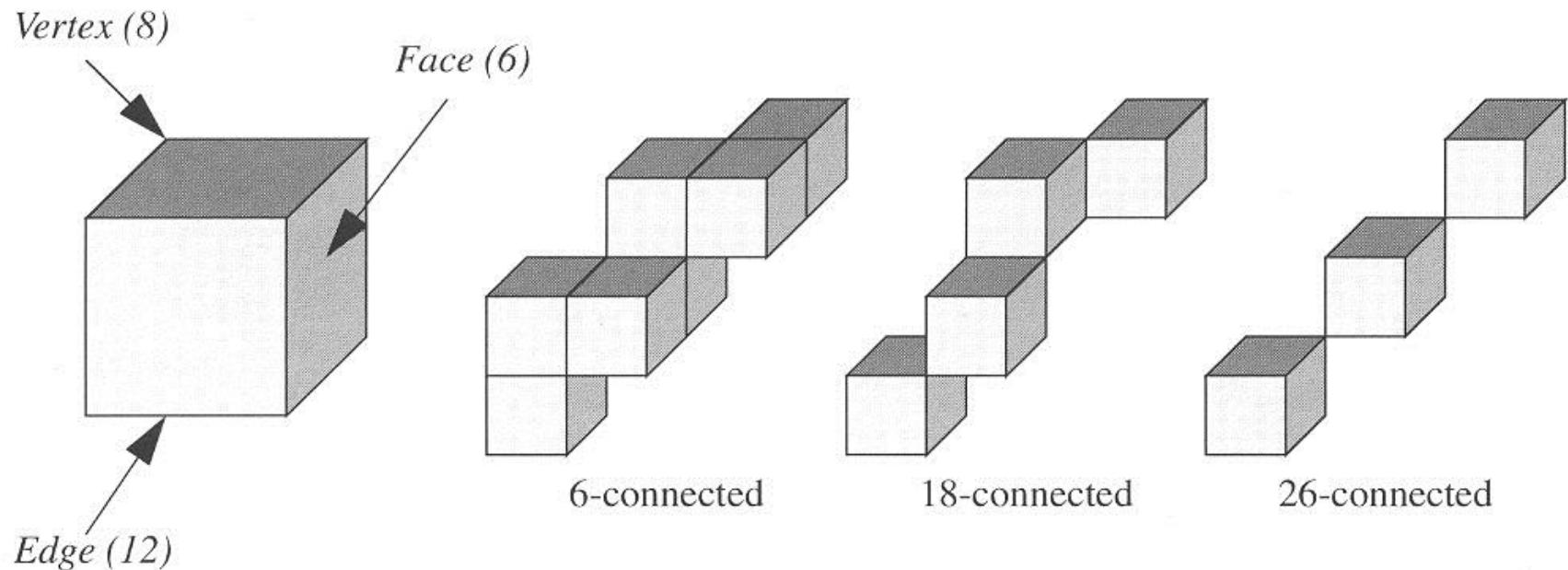
Corso di Informatica Grafica a.a. 2013-2014

- Il raggio viene discretizzato nella sequenza di voxel *adiacenti* che lo intersecano, inseguendone la traiettoria.
- Il calcolo dell' adiacenza dipende dal tipo di connessione utilizzata per i voxel
 - A livello di faccia
 - A livello di spigolo
 - A livello di vertice

Discretizzazione di un raggio

21 settembre 2013

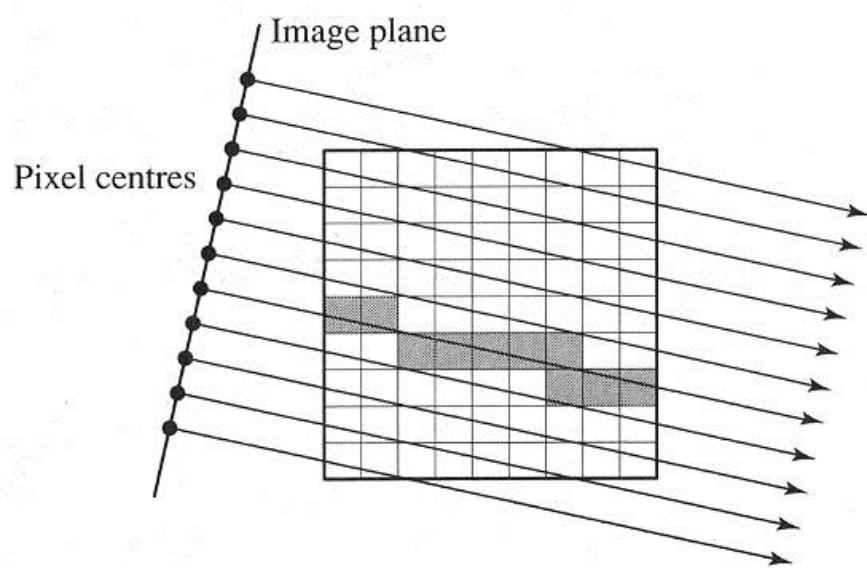
Corso di Informatica Grafica a.a. 2013-2014



Ray templates

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

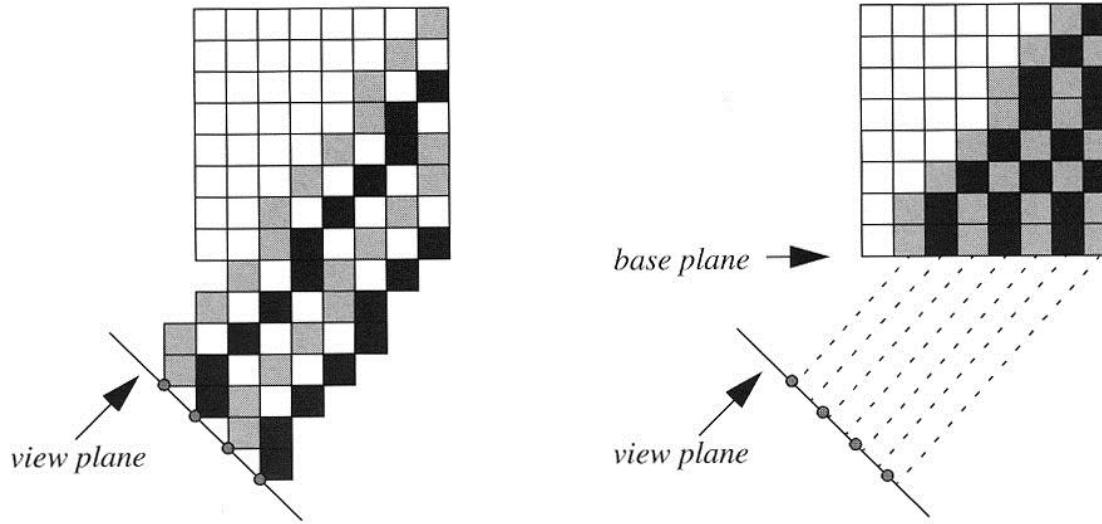


- L'approccio base è quello di proiettare ogni raggio dal centro di un pixel e definire i *ray templates* che ne costituiscono la discretizzazione
- Per ogni template si effettuerà il calcolo del valore del pixel, ad esempio mediante la tecnica compositiva.

Ray templates

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

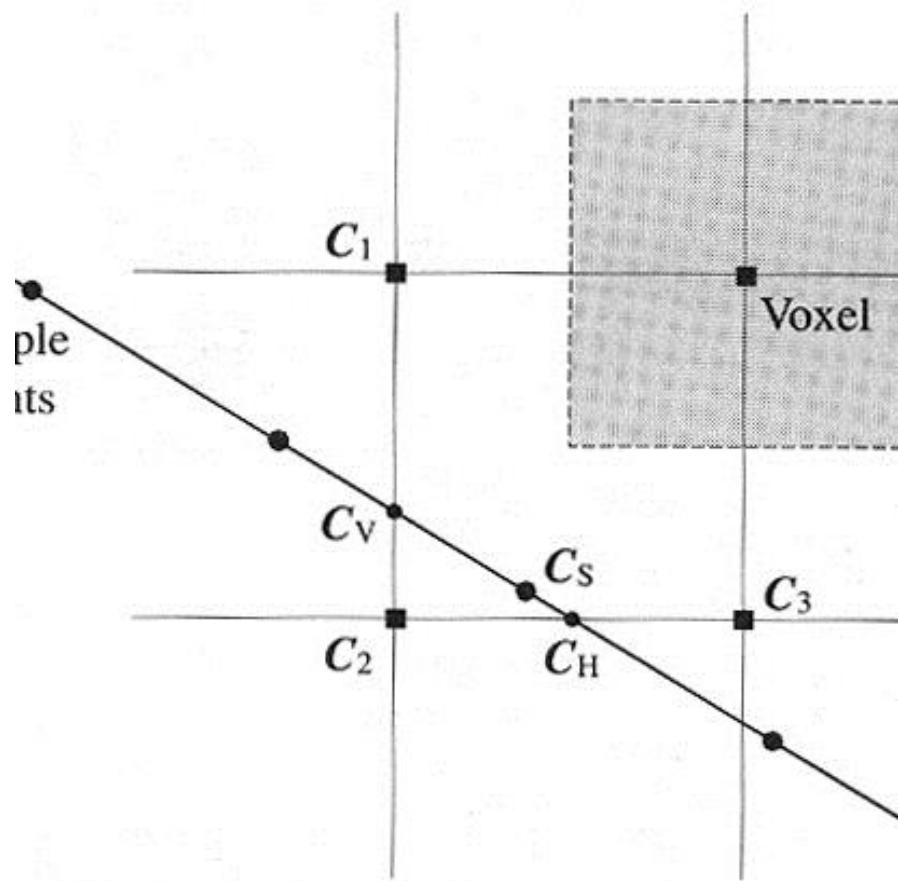


- A seconda di come si calcolano i templates si possono creare dei “buchi” cioè dei voxel che non vengono intersecati da alcun raggio e non danno contributo al rendering.
- Per evitare ciò si impone che ogni raggio passi dal centro di un voxel sulla faccia del volume che guarda verso la direzione di vista. Ovviamente bisognerà effettuare delle interpolazioni sul piano dell’immagine.

Interpolazione trilineare

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014

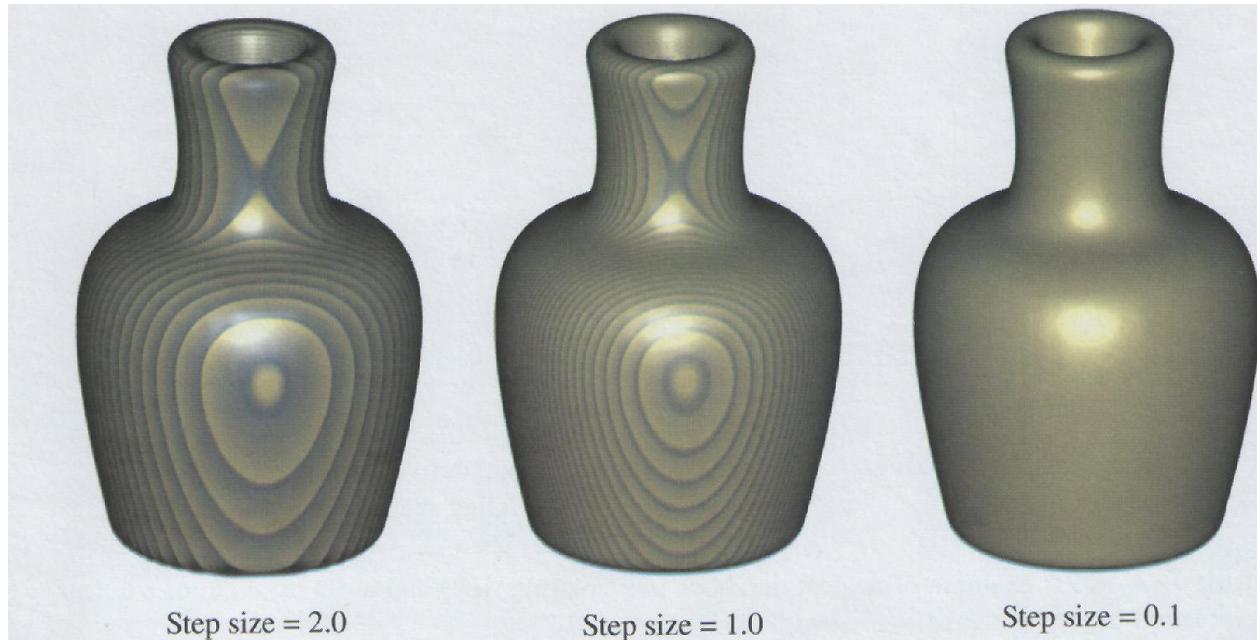


- Si definisce un passo di campionamento lungo il raggio.
- Ogni campione C_s è interpolato linearmente dai valori C_v e C_h .
- Questi sono ottenuti, dall'interpolazione bilineare lungo la faccia verticale e quella orizzontale di intersezione con il raggio, tenendo conto degli 8 valori scalari di colore determinati nei centri dei vicinato dei voxel attraversati.

Interpolazione trilineare

21 settembre 2013

Corso di Informatica Grafica a.a. 2013-2014



- La scelta errata del passo di campionamento del raggio può portare a fenomeni di aliasing.
- Appaiono degli pseudo contorni originati dall' eccessiva distanza tra due campioni successivi che non si riesce ad approssimare bene mediante l' interpolazione.

Marching cubes (5)

11 ottobre 2012

Informatica Grafica a.a. 2012-2013

Calcolo delle normali ai vertici dei triangoli

- Le normali alla superficie sono esprimibili in termini del gradiente G_f perché per $f(i,j,k)=c$ si ha che $G_f=0$
- Allora si calcolano i gradienti con le differenze finite ad ogni vertice del cubo e poi si interpolano linealmente lungo gli edge per trovarne il valore all'intersezione con $f(i,j,k)=c$.

$$G_x = \frac{f(i+1, j, k) - f(i-1, j, k)}{\Delta x}$$

$$G_y = \frac{f(i, j+1, k) - f(i, j-1, k)}{\Delta y}$$

$$G_z = \frac{f(i, j, k+1) - f(i, j, k-1)}{\Delta z}$$