Modelli (o equazioni) di lighting

- ✓ Modello di lighting "storico" di OpenGL
  - ⇒l'unico utilizzabile nelle versioni precedenti di OpenGL
    - oggi, spetta allo sviluppatore dell'app implementare negli shader questo o altri modelli
  - ⇒ancora molto utilizzato
    - determina la terminologia standard di CG
  - ⇒non è physically based
    - equazioni che "a naso" danno un risultato ragionevole
    - no validazione, e neanche mantenimento energia!
    - (eccetto che per la componente diffusiva, che abbiamo già visto)
- ✓ Oggi persino i video-games, tendono ad adottare equazioni di rendering più physically based
  - ⇒e di conseguenza descrizione dei materiali physically based

50

I 3 addendi nel modello di Lighting di OpenGL

luce finale
=
ambiente
+
riflessione
+
emissione

per ogni addendo,
ho una componente R, G e B.

51

Componente *emissione*

- ✓LEDs, lampadine...
- ✓Non dipende dalle luci
  - ⇒solo dall'oggetto
- ✓E' solo una componente additiva
  - ⇒costante per R, G e B
- ✓Modella solo luce che va da: emitter a: occhio
  - ⇒Non: light-emitter → oggetto → occhio
  - ⇒Nota: non manda luce ad oggetti vicini
  - ⇒non e' illuminazione globale
  - ⇒per fare cio', devo settare un'altra luce

52

I 3 addendi nel modello di Lighting di OpenGL

luce finale

=

ambiente

+

riflessione

+

emissione

53

Componente *ambiente*

- ✓ Modella (ma in modo estremamente semplificato) la luce che arriva attraverso rifelssioni multiple
  - ⇒effetto globale
- ✓ Equivale all'assunzione: "un pò di luce raggiunge ogni punto di ogni superficie"
  - ⇒Per es. anche quelle in ombra
- ✓ Piccola costante additiva
  - ⇒non dipende dalla normale della superficie

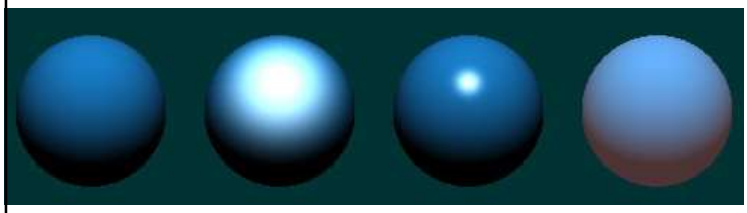
54

Componente *ambiente*

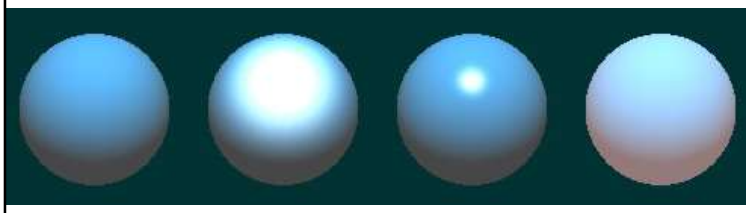
- ✓ prodotto fra:
  - ⇒"ambient color" del materiale (R_M G_M B_M)
  - ⇒colore della luce (R_L G_L B_L)
  - ⇒(nb: prodotto componente per componente)

55

Componente *ambiente*



senza



con

56

I 3 addendi nel modello di Lighting di OpenGL

luce finale

=

ambiente

+

riflessione

+

emissione

riflessione diffusa

+

riflessione speculare

solo componente *ambient*

57

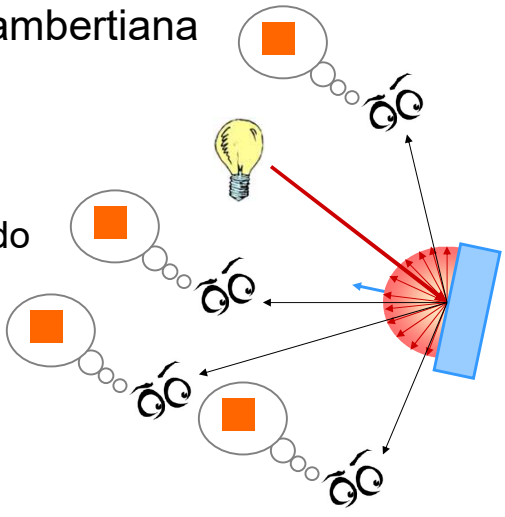
I 4 addendi nel modello di Lighting di OpenGL

luce finale
=
ambiente
+
riflessione diffusiva
+
riflessione speculare
+
emissione

58

Componente *riflessione diffusa*

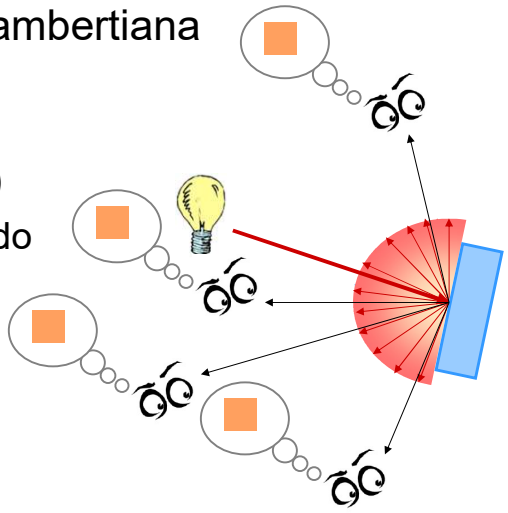
✓La luce che colpisce una superficie Lambertiana si riflette in tutte le direzioni (nella semisfera)
⇒nello stesso modo



59

Componente *riflessione diffusa*

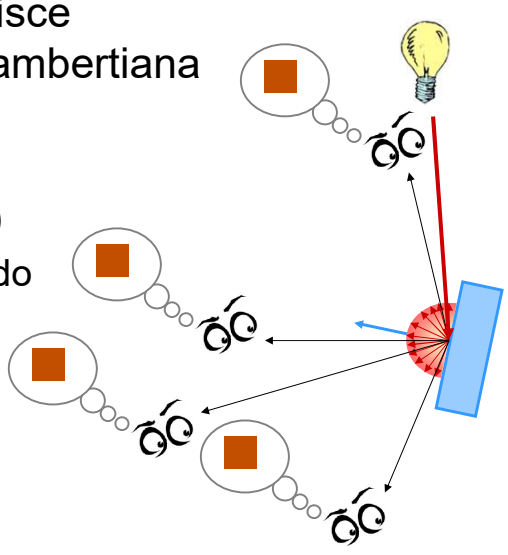
- ✓La luce che colpisce una superficie Lambertiana si riflette in tutte le direzioni (nella semisfera)
  - ⇒nello stesso modo



60

Componente *riflessione diffusa*

- ✓La luce che colpisce una superficie Lambertiana si riflette in tutte le direzioni (nella semisfera)
  - ⇒nello stesso modo



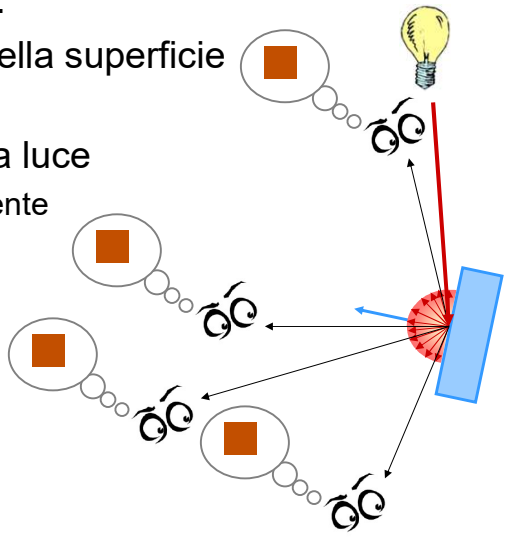
61

Componente *riflessione diffusa*

✓Dipende solo da:

⇒l'orientamento della superficie
- (la "normale")

⇒la direzione della luce
- del raggio incidente



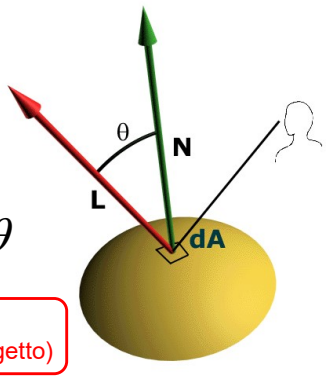
62

Componente *riflessione diffusa*

✓Dipende solo da:

⇒l'orientamento della superficie **N**
- (cioè la sua "normale")

⇒la direzione della luce **L**
- (cioé del raggio incidente)



$$I_{diff} = I_{luce\ diff} \cdot k_{materiale\ diff} \cdot \cos\theta$$

R, G, B
(di solito bianco: 1,1,1)

R, G, B
(il "colore base" dell'oggetto)

moltiplicazione componente per componente

63

Componente *riflessione diffusa*

✓Dipende solo da:

⇒l'orientamento della superficie **N**

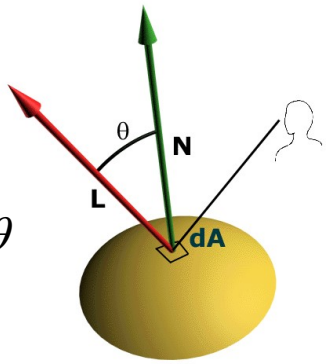
- (cioè la sua "normale")

⇒la direzione della luce **L**

- (cioé del raggio incidente)

$$I_{diff} = I_{luce\ diff} \cdot k_{materiale\ diff} \cdot \cos\theta$$

base color,
o diffuse color,
che fa parte del
"materiale"



64

Componente *riflessione diffusa*

✓Dipende solo da:

⇒l'orientamento della superficie **N**

- (cioè la sua "normale")

⇒la direzione della luce **L**

- (cioé del raggio incidente)

se angolo é compreso fra 0° e 90°,
else: 0,
(oggetto in ombra di se stesso)

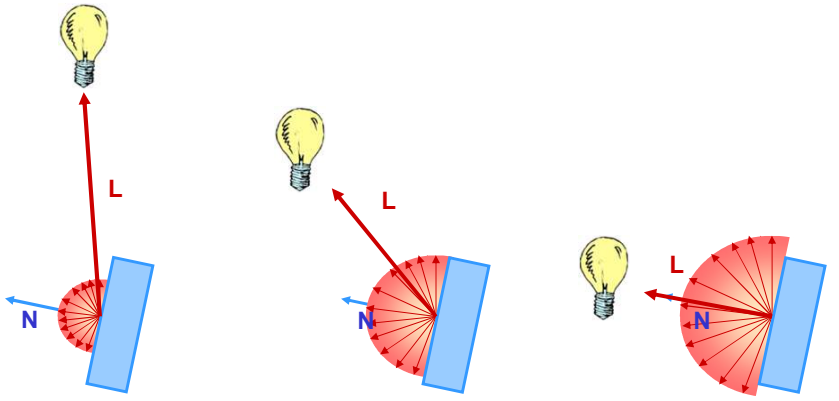
$$I_{diff} = I_{luce\ diff} \cdot k_{materiale\ diff} \cdot \cos\theta$$

$$= I_{luce\ diff} \cdot k_{materiale\ diff} \cdot (\hat{N} \cdot \hat{L})$$

nota: ciascuno dei "puntini" in questa equazione
rappresenta una operazione diversa!

65

Componente *riflessione diffusa*



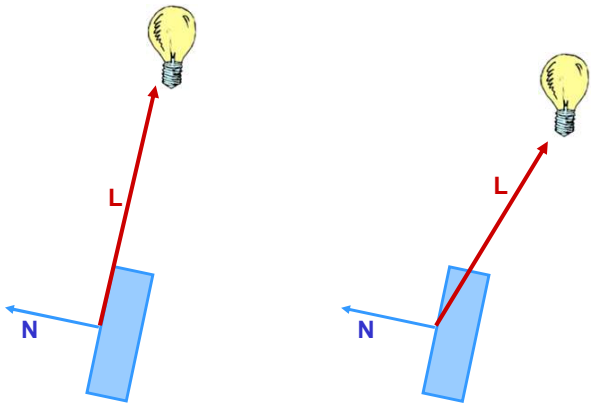
componente diffusa piccola
θ=70º

componente diffusa grande
θ=35º

componente diffusa massima
θ=0º

66

Componente *riflessione diffusa*



componente diffusa ZERO
θ=90º

componente diffusa ZERO
θ>90º

(la superficie è nella propria stessa ombra)

67

Componente *riflessione diffusa*

✓Proprietà

⇒modello fedele
delle BRDF
di alcuni materiali reali ☺

⇒ma non poi molti ☹

⇒modello fisicamente coerente ☺
- per es, conserva l'energia

⇒molto semplice da calcolare ☺

68

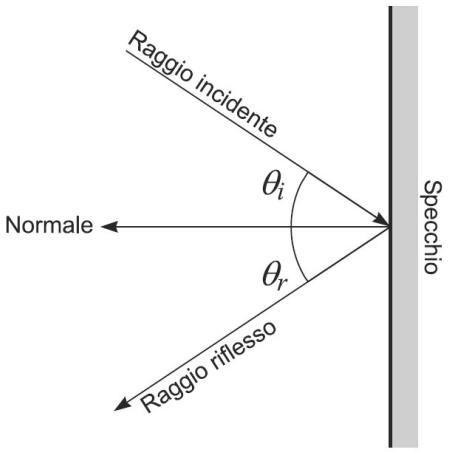
I 4 fattori che consideriamo

luce finale
=
ambiente
+
riflessione diffusa
+
riflessione speculare
+
emissione

69

Componente *riflessione speculare*

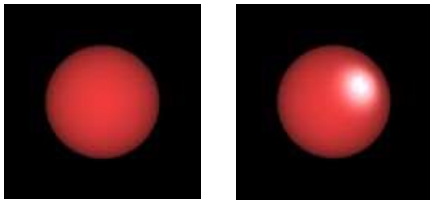
- ✓ Fenomeno molto semplice (concettualmente):
  fotoni che "rimbalzano"
  - ⇒ come se fossero palline!
    - NB: natura corpuscolare del foton
  - ⇒ il rimbalzo dipende dalla normale della superficie
    - in formula?
- ✓ **Microscopicamente:**
  la normale delle "*microfacets*" si discosta poco da quella della superficie macroscopica
  (meno se ne discosta, in media, maggiormente speculare apparirà la sup)



70

Componente *riflessione speculare*

- ✓"Specular" reflection
  - ⇒detta anche «phong» reflection
- ✓Per materiali lucidi
  - ⇒con riflessi brillanti
  - ⇒("highlights")

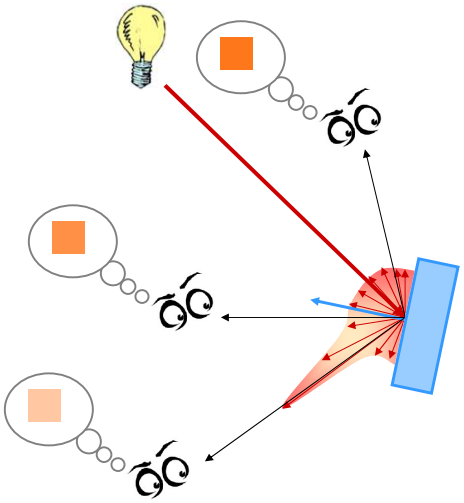


senza con

71

Componente *riflessione speculare*

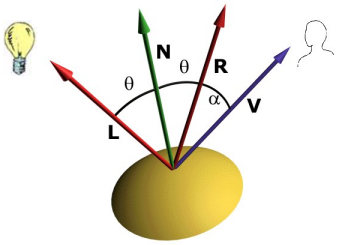
- ✓ Idea base:
  la luce non viene riflessa
  da materiali lucidi
  in maniera eguale
  in tutte le direzioni
- ✓ Riflessione
  view-dependent

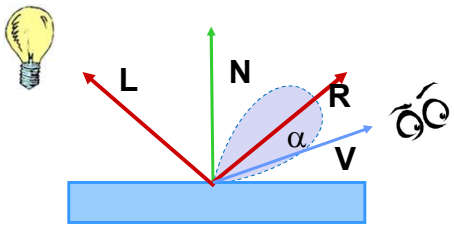


72

Componente *riflessione speculare*

L: raggio incidente
N: normale
R: raggio riflesso
V: dir. di vista



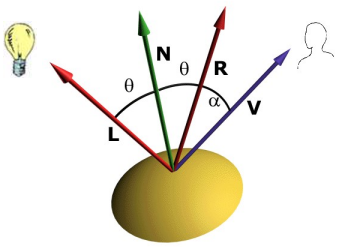


73

Componente *riflessione speculare*

✓Phong light model

⇒by Bui-Tuong Phong, 1975

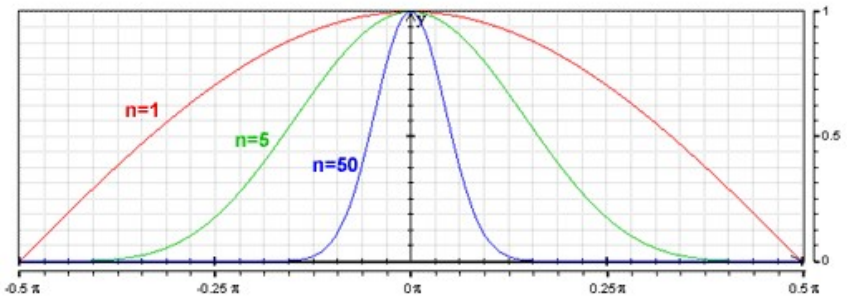


$$I_{spec} = I_{luce\ spec} \cdot k_{materiale\ spec} \cdot \cos\alpha$$

74

Componente *riflessione speculare*

✓Elevando il coseno ad una potenza >1,
si ottengono riflessi piu' piccoli e brillanti

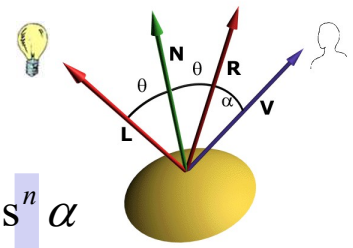


75

Componente *riflessione speculare*

✓Phong light model

⇒by Bui-Tuong Phong, 1975



$$I_{spec} = I_{luce\ spec} \cdot k_{materiale\ spec} \cdot \cos^{n} \alpha$$

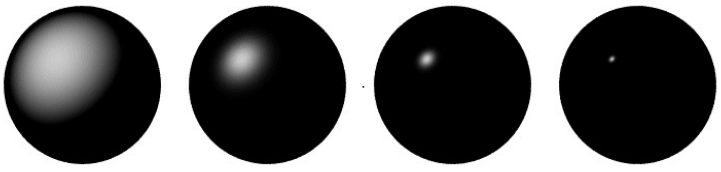
"shininess" (quanto sono *intensi* i riflessi)

"glossiness" (quanto sono *concentrati* i riflessi)

fanno parte del "materiale" (descriz. delle caratteristiche dell'oggetto)

76

Componente *riflessione speculare* Effetto della "glossiness"



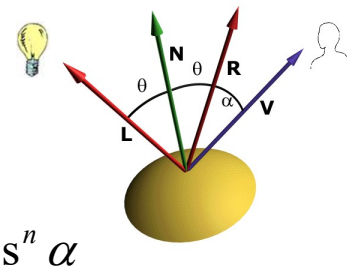
$n = 1$ $n = 5$ $n = 10$ $n = 100$

77

Componente *riflessione speculare*

✓Phong light model

⇒by Bui-Tuong Phong, 1975



$$I_{spec} = I_{luce\ spec} \cdot k_{materiale\ spec} \cdot \cos^n \alpha$$

$$= I_{luce\ spec} \cdot k_{materiale\ spec} \cdot (\hat{R} \cdot \hat{V})^n$$

direzione di luce *riflessa* specularmente.

$$\hat{R} = -\hat{L} + 2\left(\hat{L} \cdot \hat{N}\right)\hat{N}$$

78

Componente *riflessione speculare* variante di Blinn

- Blinn-Phong light model:
  - formulazione semplificata del light model di Phong
  - risultati simili, formula lievemente diversa
  - Intuizione: **V** è (quasi) in direzione di **R**

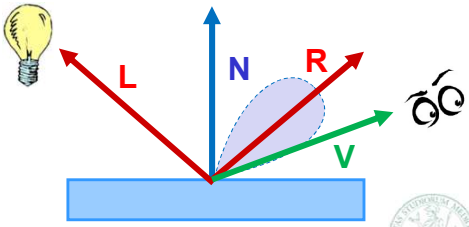
    sse

    N è (quasi) la media (rinormalizzata) fra **V** e **L**

    "half-way" vector
  - (e non devo neanche calcolarlo, **R**)

Jim Blinn

(pionere della CG)



79

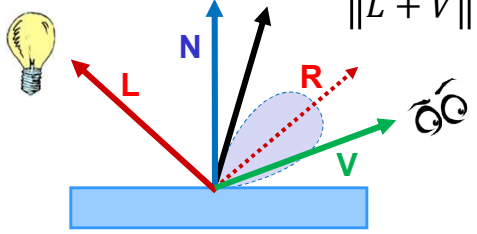
Componente *riflessione speculare* (due formule alternative)

phong: $I_{spec} = I_{luce\ spec} \cdot k_{materiale\ spec} \cdot (\hat{R} \cdot \hat{V})^n$

blinn-phong: $I_{spec} = I_{luce\ spec} \cdot k_{materiale\ spec} \cdot (\hat{H} \cdot \hat{N})^n$

"half-way" vector

$$\hat{H} = \frac{\hat{L} + \hat{V}}{\|\hat{L} + \hat{V}\|}$$



80

I 4 fattori che consideriamo

luce finale
=
ambiente
+
riflessione diffusa
+
riflessione speculare
+
emissione

81

Equazione di lighting "storica" di OpenGL in totale

$$I_{tot} = I_{luce\ ambient} \cdot k_{materiale\ ambient} + I_{luce\ diffuse} \cdot k_{materiale\ diffuse} \cdot (\hat{L} \cdot \hat{N}) + I_{luce\ spacular} \cdot k_{materiale\ spacular} \cdot (\hat{H} \cdot \hat{N})^n + k_{materiale\ emission}$$

$$\hat{H} = \frac{\hat{L} + \hat{V}}{\|\hat{L} + \hat{V}\|}$$

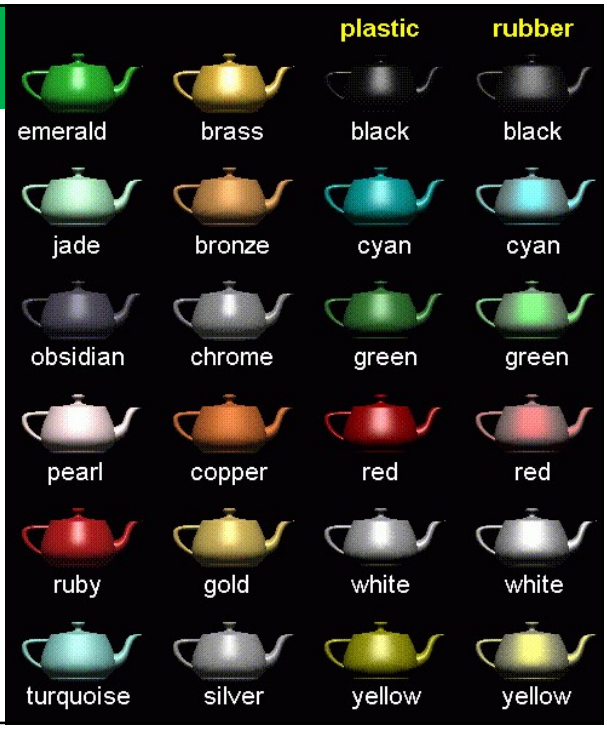
propretà del materiale

propretà della luce

82

Materiali...

Material	GL_AMBIENT	GL_DIFFUSE	GL_SPECULAR	GL_SHININESS
Emerald	0.0215 0.1745 0.0215 0.55	0.07568 0.61424 0.07568 0.55	0.633 0.727811 0.633 0.55	76.8
Jade	0.135 0.2225 0.1575 0.95	0.54 0.89 0.63 0.95	0.316228 0.316228 0.316228 0.95	12.8
Obsidian	0.05375 0.05 0.06625 0.82	0.18275 0.17 0.22525 0.82	0.332741 0.328634 0.346435 0.82	38.4
Pearl	0.25 0.20725 0.20725 0.922	1.0 0.829 0.829 0.922	0.296648 0.296648 0.296648 0.922	11.264
Ruby	0.1745 0.01175 0.01175 0.55	0.61424 0.04136 0.04136 0.55	0.727811 0.626959 0.626959 0.55	76.8
Turquoise	0.1 0.18725 0.1745 0.8	0.396 0.74151 0.69102 0.8	0.297254 0.30829 0.306678 0.8	12.8
Black Plastic	0.0 0.0 0.0 1.0	0.01 0.01 0.01 1.0	0.50 0.50 0.50 1.0	32
Black Rubber	0.02 0.02 0.02 1.0	0.01 0.01 0.01 1.0	0.4 0.4 0.4 1.0	10
Brass	0.329412 0.223529 0.027451 1.0	0.780392 0.568627 0.113725 1.0	0.992157 0.941176 0.807843 1.0	27.8974
Bronze	0.2125 0.1275 0.054 1.0	0.714 0.4284 0.18144 1.0	0.393548 0.271906 0.166721 1.0	25.6
Polished Bronze	0.25 0.148 0.06475 1.0	0.4 0.2368 0.1036 1.0	0.774597 0.458561 0.200621 1.0	76.8
Chrome	0.25 0.25 0.25 1.0	0.4 0.4 0.4 1.0	0.774597 0.774597 0.774597 1.0	76.8



83

Modellazione delle luci

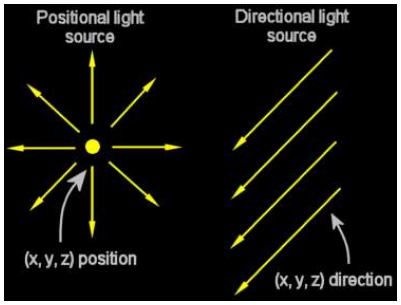
✓Quanto vale la direzione *L*?

⇒costante nella scena: fonti di luci "direzionali"

- modella fonti di luce molto distanti, e.g. il sole

⇒varia nella scena: fonti di luci "posizionali"

- modella fonti di luci vicine, e.g. lampadine



85

Luci posizionali e direzionali

✓Luce direzionale:

⇒la direzione $\hat{L}$ della luce è la stessa per tutti i punti

✓Luce posizionale:

⇒la posizione della luce L_{POS} è data

⇒la direzione della luce viene compuata per ogni punto di posizione P :

$$\hat{L} = \frac{(P - L_{POS})}{\|P - L_{POS}\|}$$

⇒Inoltre, posso introdurre fattori di attenuazione dipendenti dalla *distanza* della luce $d_L = \|P - L_{POS}\|$

86

Modellazione delle luci: luci posizionali

- ✓ Nelle luci posizionali, si può attenuare l'intensità in funzione della distanza
- ✓ In teoria (per la fisica) intensità = 1 / distanza^2

$$f_{attentuazione\ luce} = \left(\frac{1}{c \cdot d_{\mathrm{L}}^2} \right)$$

87

Modellazione delle luci: luci posizionali

- ✓ In pratica, questo porta ad attenuazioni della luce troppo repentine
- ✓ Spesso si usa un polinomio di sec grado generico:

$$f_{attentuazione\ luce} = \min\left(\frac{1}{c_1 + c_2 d_{\mathrm{L}} + c_3 d_{\mathrm{L}}^2}, 1 \right)$$

con $c_{0,1,2}$ costanti arbitrarie

88

Equazione di lighting

$$I_{tot} = \left(\begin{array}{l} I_{luce\ ambient} \cdot k_{materiale\ ambient} + \\ I_{luce\ diffuse} \cdot k_{materiale\ diffuse} \cdot (L \cdot N) + \\ I_{luce\ spacular} \cdot k_{materiale\ spacular} \cdot (H \cdot N)^n \end{array} \right) \cdot f_{attentuazione\ luce} + k_{materiale\ emission}$$

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

89

Equazione di Lighting storica di OpenGL

caratteristiche della luce
caratteristiche del materiale
dati dalla scena

$$I_{tot} = \left(\begin{array}{l} I_{luce\ ambient} \cdot k_{materiale\ ambient} + \\ I_{luce\ diffuse} \cdot k_{materiale\ diffuse} \cdot (\hat{L} \cdot \hat{N}) + \\ I_{luce\ spacular} \cdot k_{materiale\ spacular} \cdot (\hat{H} \cdot \hat{N})^n \end{array} \right) + k_{materiale\ emission}$$

sommato
per ogni luce,
ed attenuato
dal fattore di attenuazione
(per le luci posizionali)

93

Caratteristiche del materiale

- ✓Possono essere:
  - ⇒specificate per oggetto
    materiale costante
    - (allora saranno uniforms)
  - ⇒specificate per vertice
    materiale disomogeneo
    - (allora saranno *varying*)
  - ⇒specificate nelle tessiture (per «texel»)
    materiale ancora più disomogeneo
  - ⇒casi misti
    - (alcuni fattori per oggetto, altri per vertice, etc)

94

Circa il computo dei prodotti dot nel lighting

- ✓Es: nelle componenti diffuse e specular
- ✓Gli operandi devono essere vettori unitari
  - ⇒(altrimenti, non si ottiene il coseno!)
  - ⇒(se non lo sono: normalizzare prima dell'uso)
- ✓I due operandi possono essere espressi in qualisasi spazio
  - ⇒vista, oggetto, mondo …
- ✓…purchè sia lo stesso per entrambi!!!

95