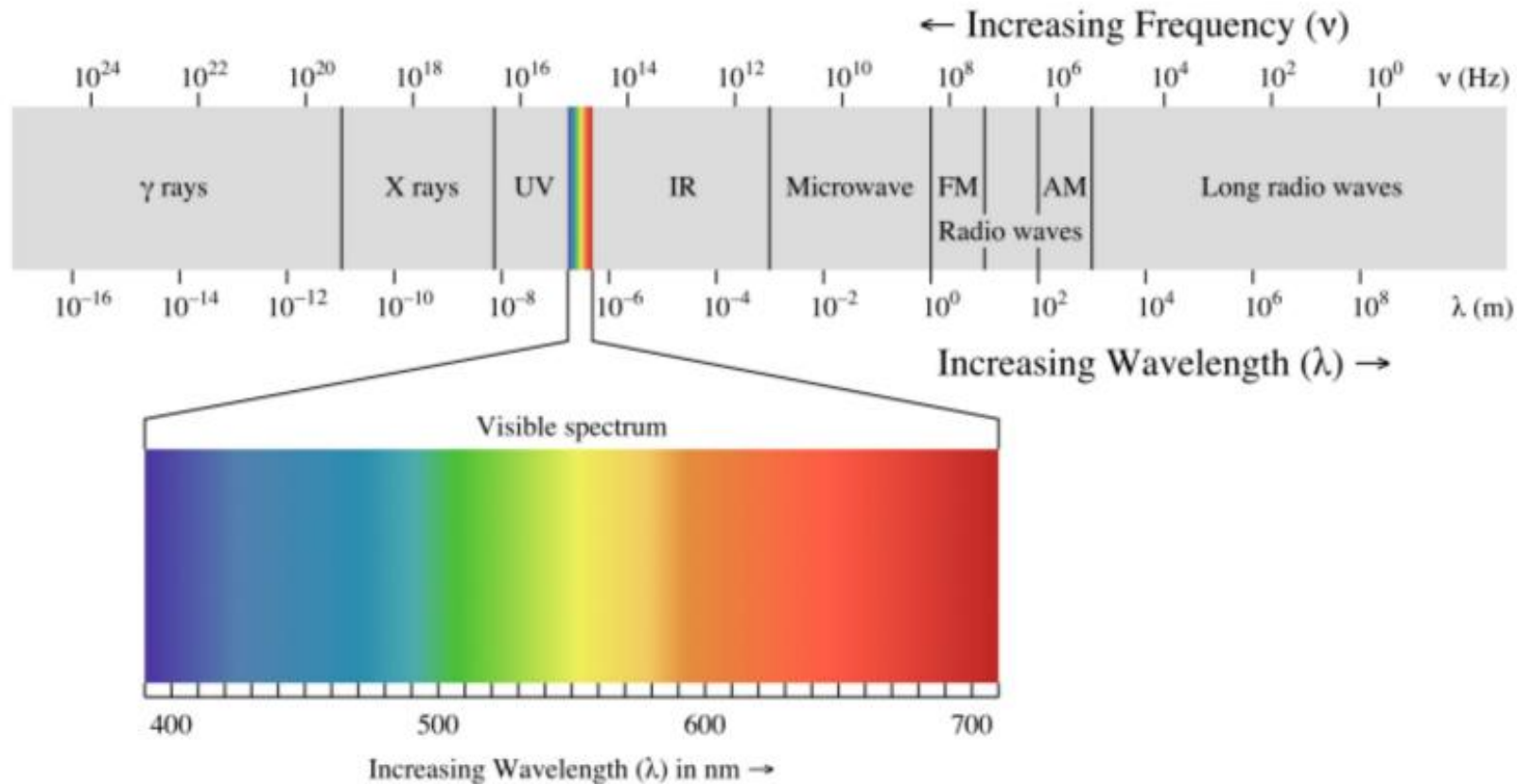


# Luz e Cores

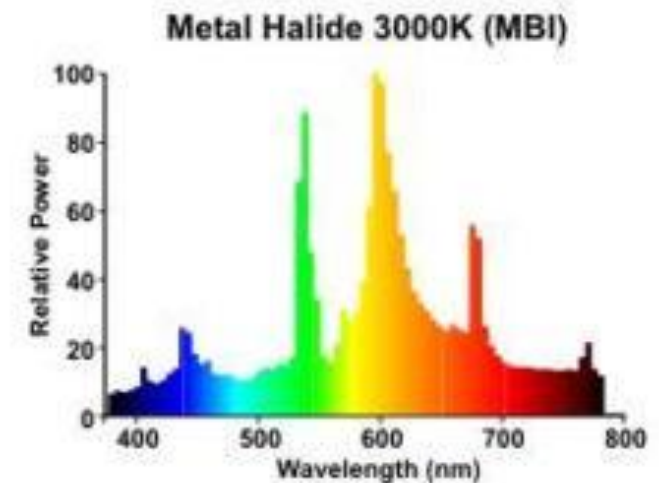
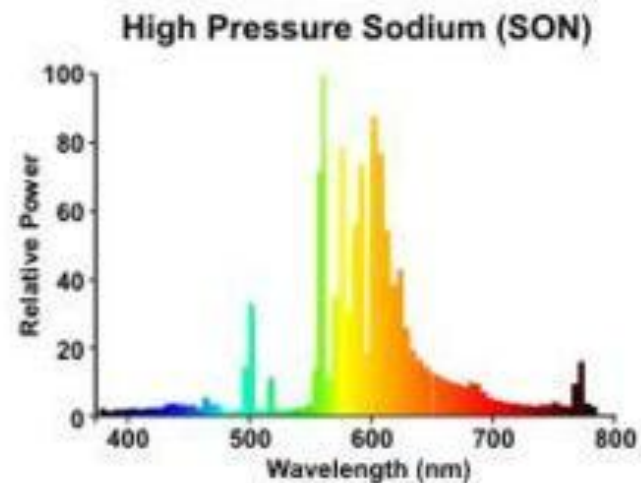
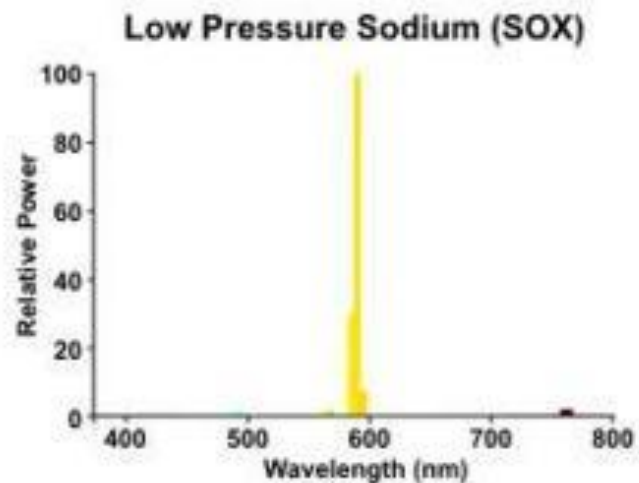
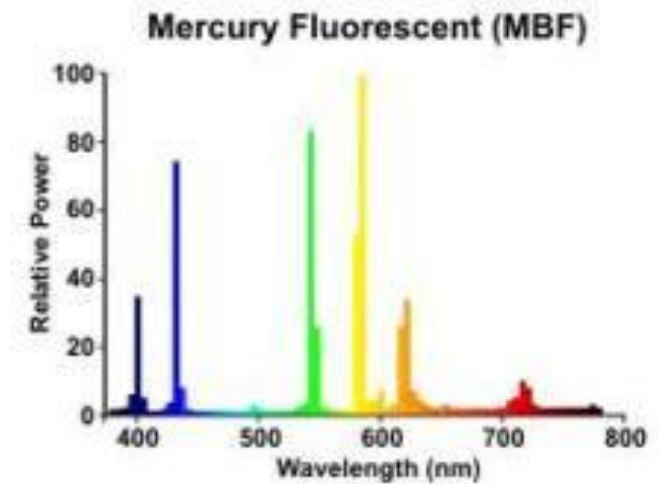
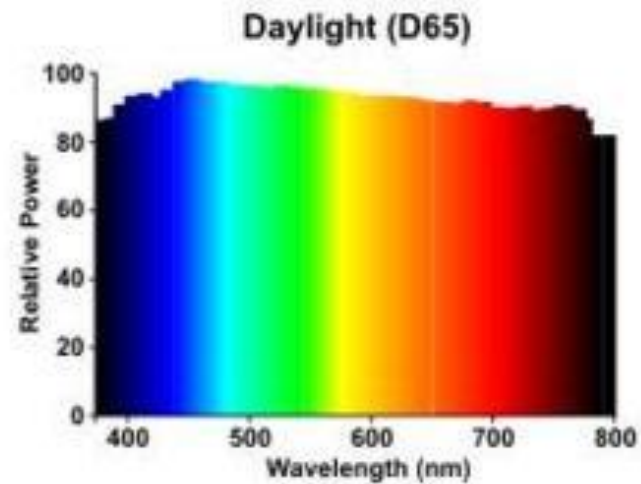
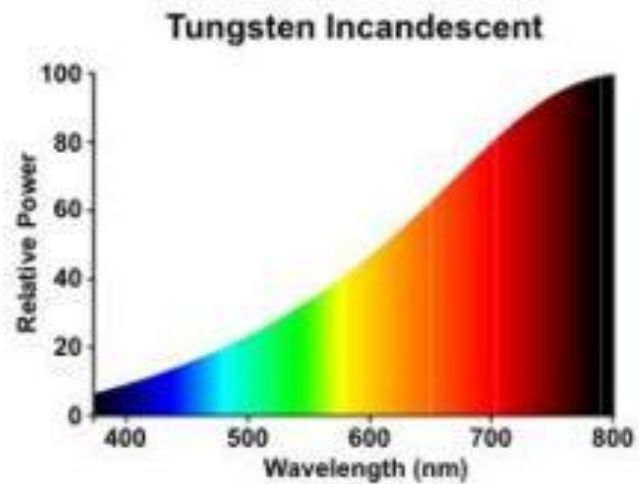
---

RAFAEL TEIXEIRA

# Espectro Eletromagnético

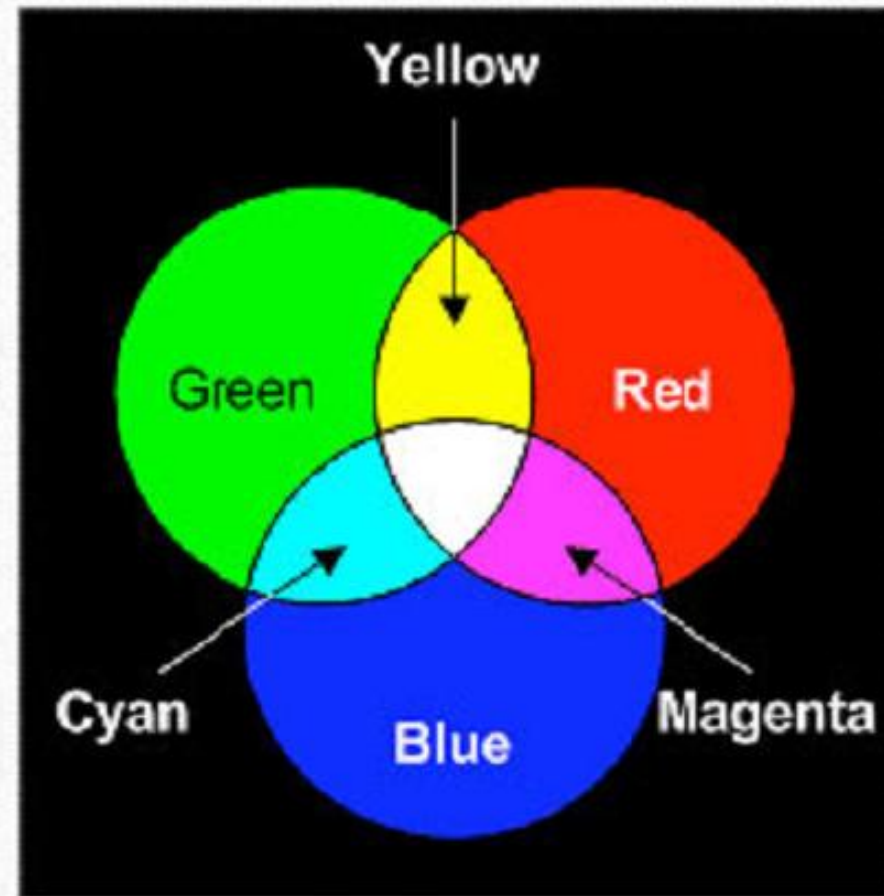


# Distribuição espectral de fontes luminosas

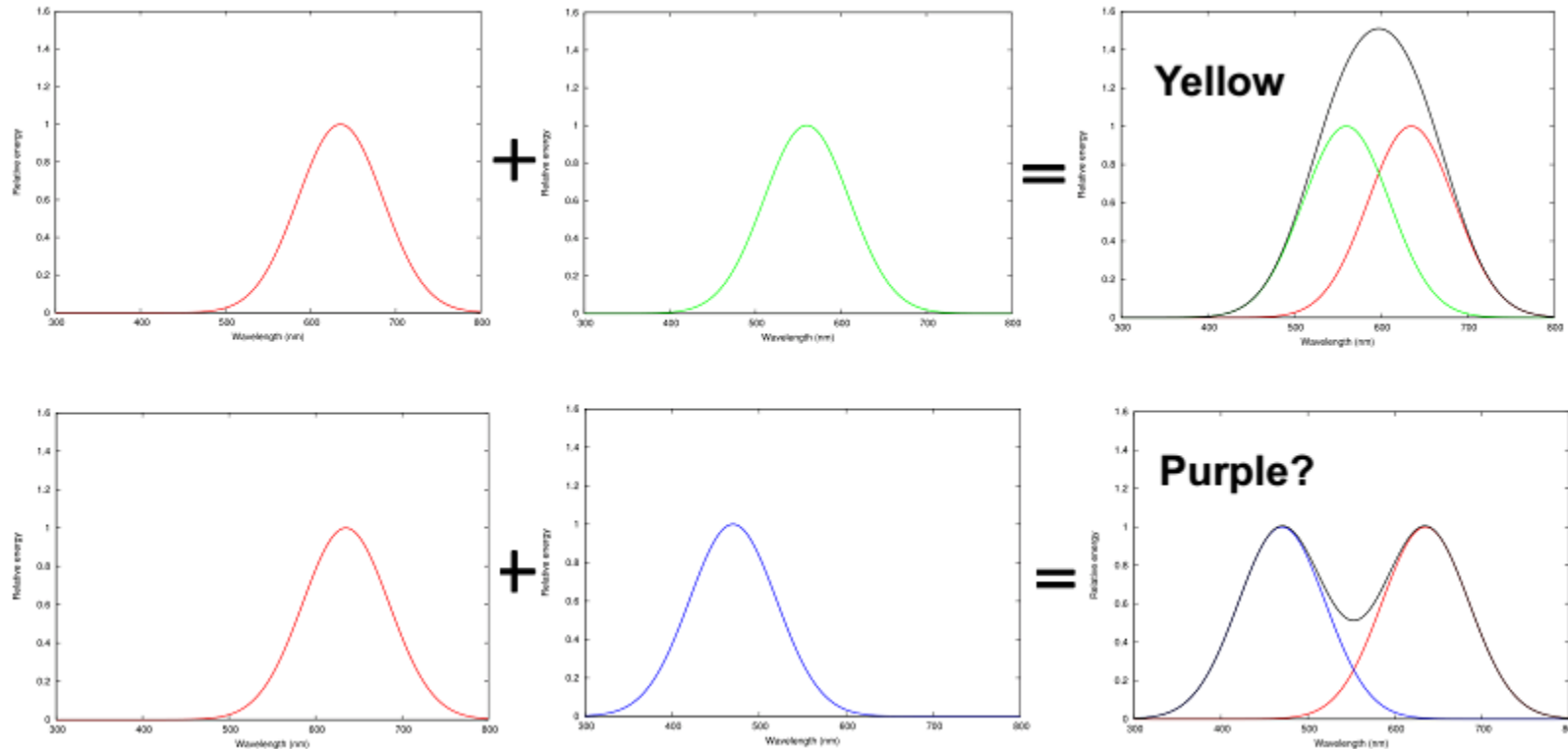


# Somando energias

---

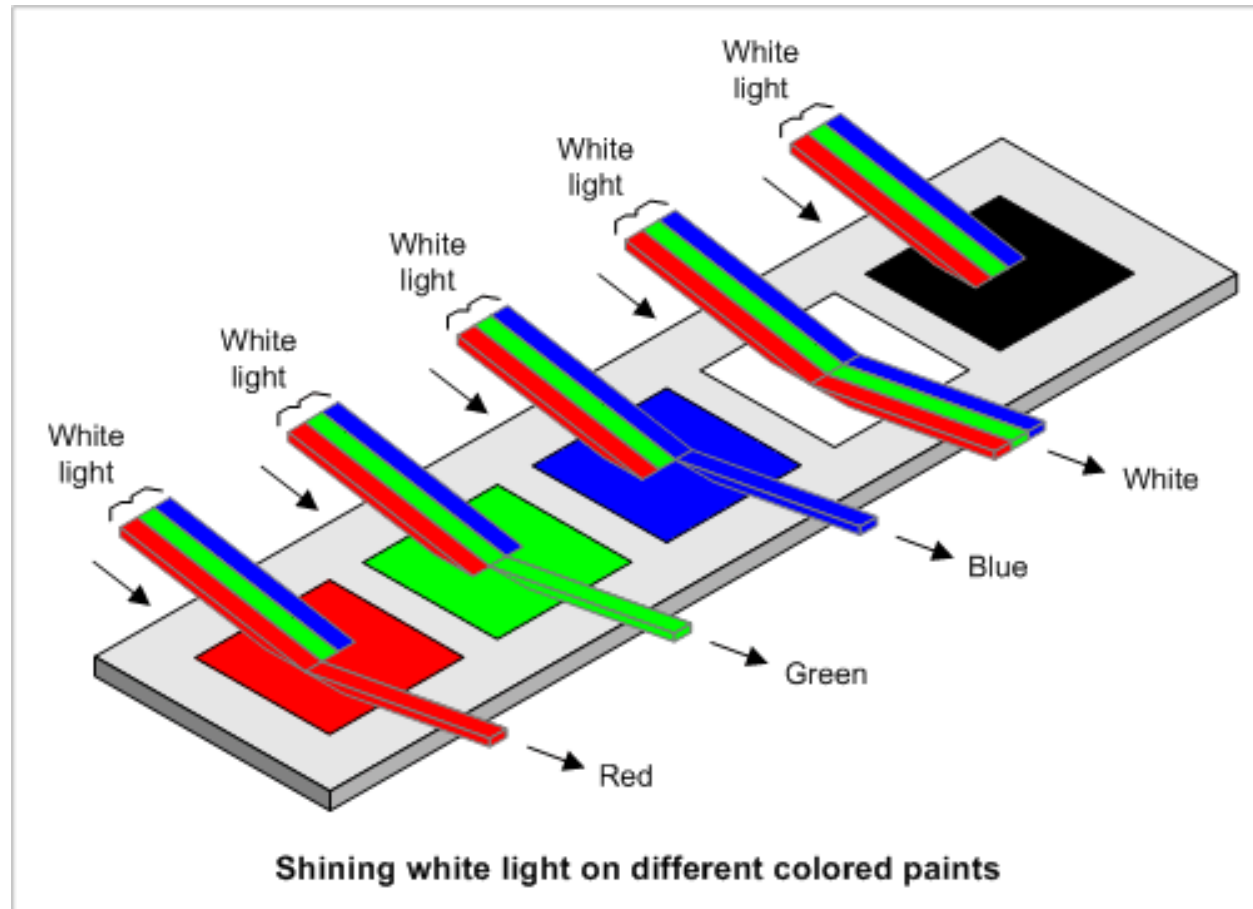


# Somando energias

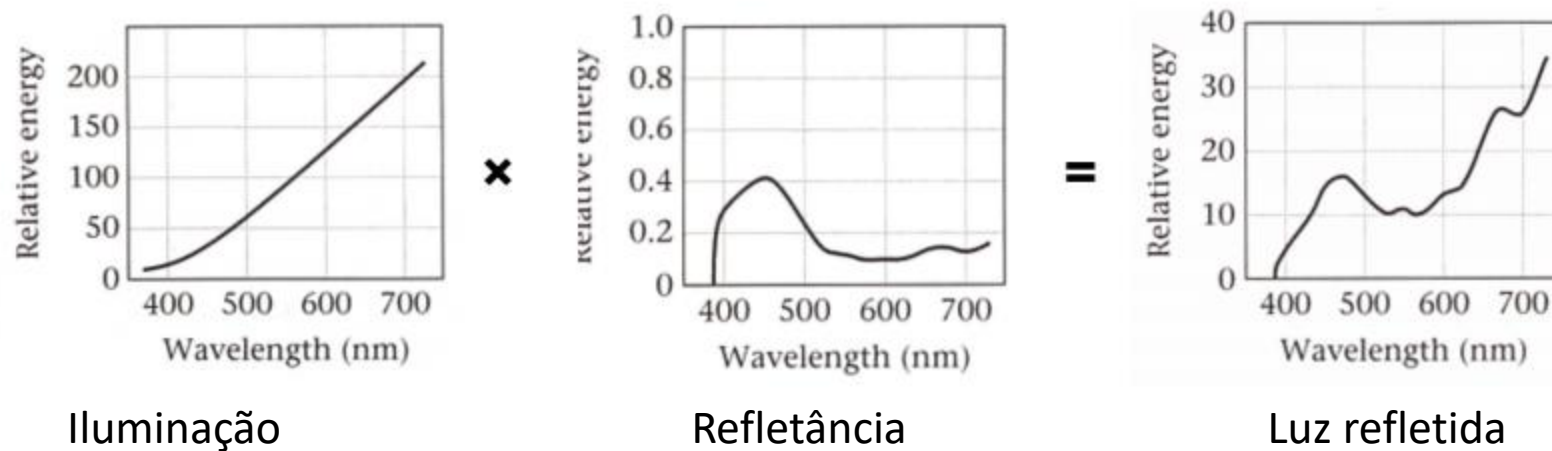


$$E(\lambda) = E_1(\lambda) + E_2(\lambda)$$

# Reflexão/absorção de luz



# Reflexão/absorção de luz



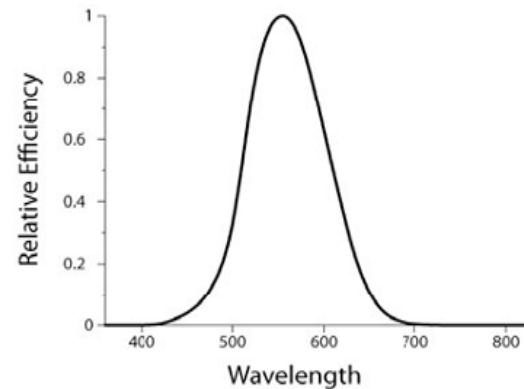
$$R(\lambda) = r(\lambda)E(\lambda) = (1 - a(\lambda))E(\lambda)$$

$$a(\lambda) + r(\lambda) = 1, \quad 0 \leq a(\lambda), r(\lambda) \leq 1$$

# Sensores de absorção

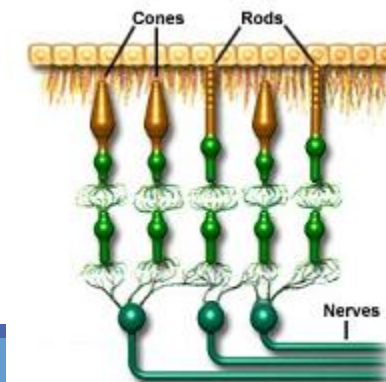
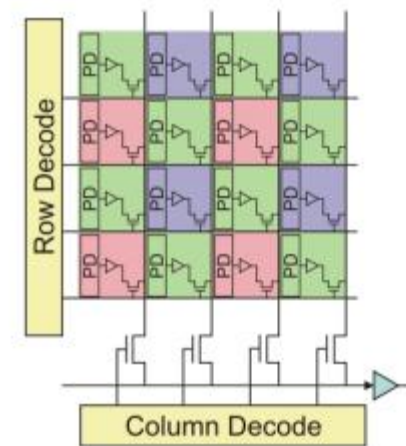
Sensores, cones, bastonetes, etc.

Função de resposta



Força do sinal

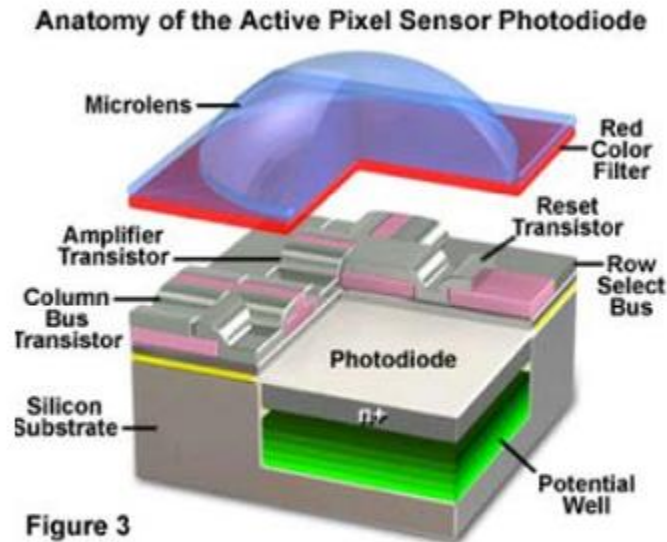
$$A = \int A(\lambda)E(\lambda)d\lambda$$



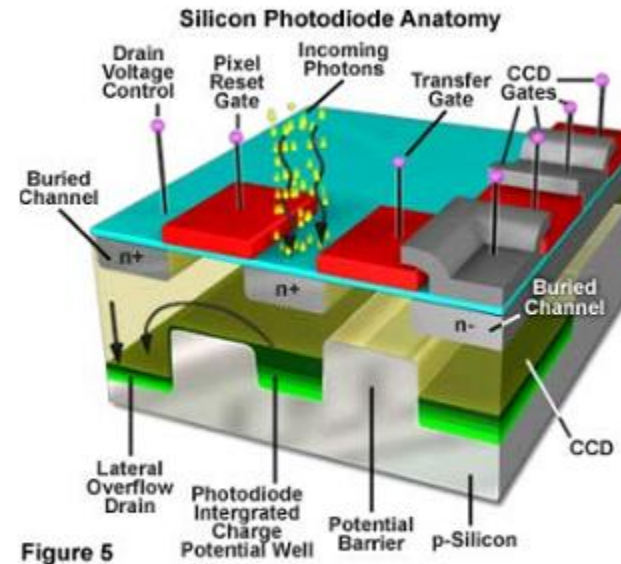


# Sensores eletrônicos

Sensor fotoelétrico: materiais que geram elétrons quando atingidos por um fóton.



**Complimentary Metal-Oxide Semiconductor (CMOS)**

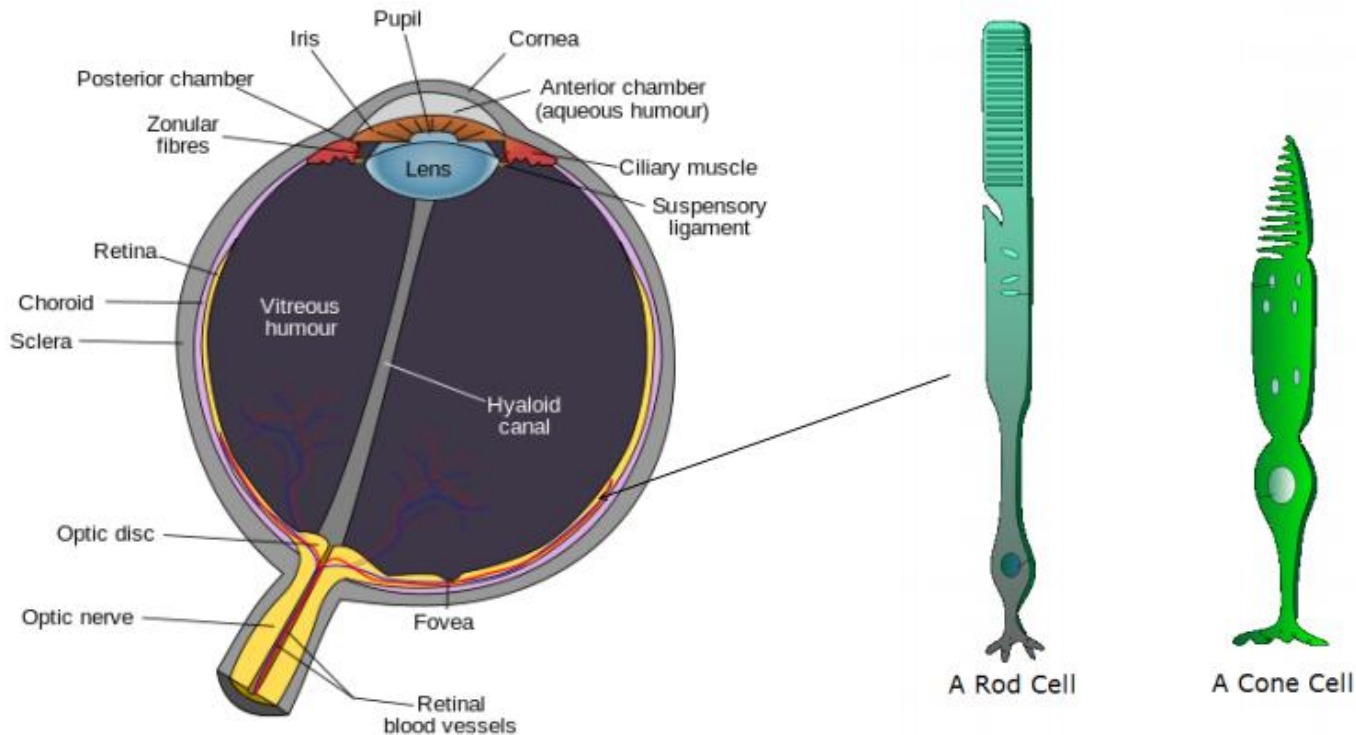


**Charge-Coupled Device (CCD)**

# Olho humano

A retina possui dois tipos de sensores

- Bastonetes reconhecem a intensidade da luz
- Cones enxergam três cores (vermelho, verde e azul)



# Cones e bastonetes

Quando o olho se fixa em um objeto a luz atinge o centro da retina (fóvea)

Os cones estão em grande número na fóvea, possibilitando uma visão aguçada

Durante o dia os sinais dos cones são saturados e fornecem nossa visão. Durante a noite usamos também os bastonetes.

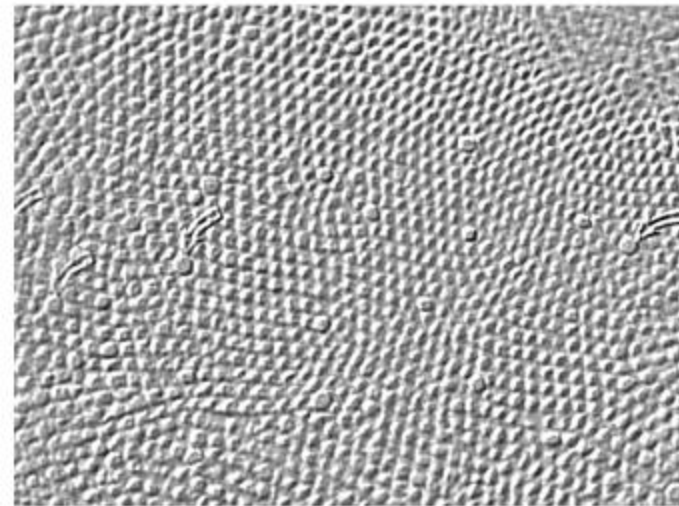
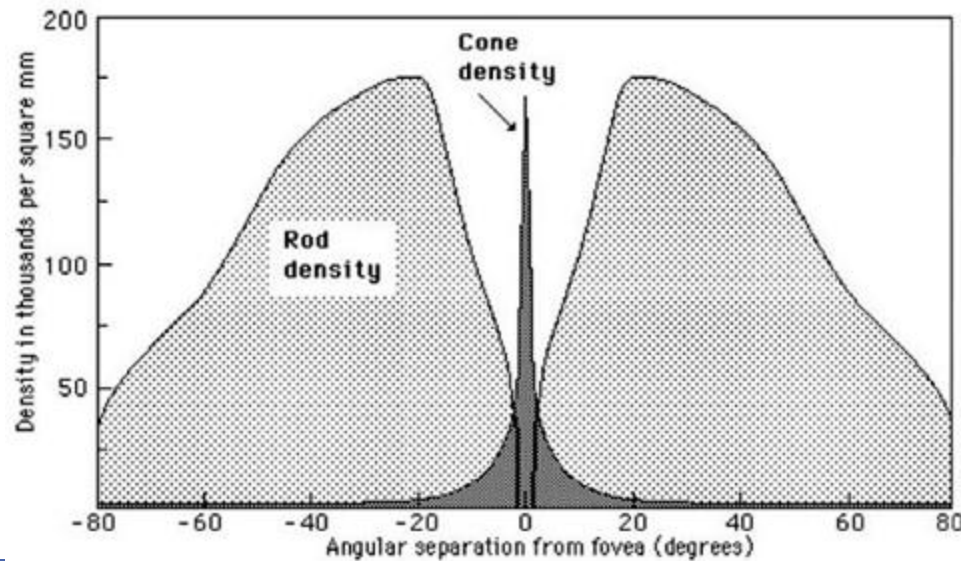
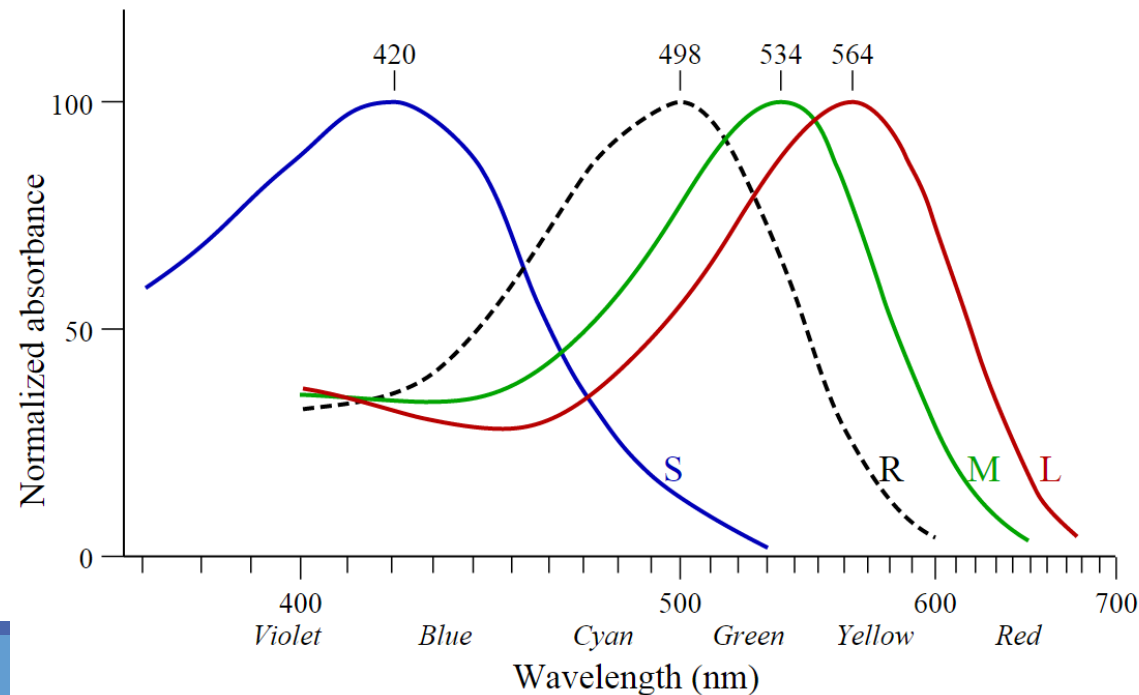


Fig. 13. Tangential section through the human fovea. Larger cones (arrows) are blue cones.

# Funções de resposta dos cones/bastonetes

Proteínas nas células dos cones/bastonetes alteram o potencial de absorção.

O olho absorve 4 tipos de sinais (vermelho, verde, azul e bastonetes).



# Teoria tricromática

---

Como o olho humano funciona com três sinais, nós também iremos trabalhar com 3 sinais em imagens, impressoras e monitores.

Imagens são armazenada com valores para os canais R, G e B;

- Os valores são entre 0 e 255
- Cada canal define uma cor e o valor a intensidade
- Similar aos cones

Os monitores podem aumentar ou reduzir a intensidade da imagem (brilho)

- Similar aos bastonetes

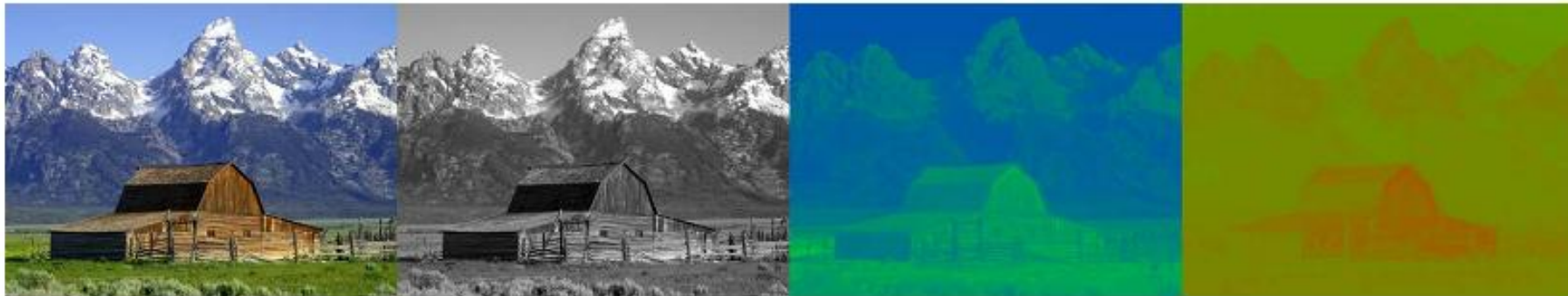
# Brilho (Luminescência)

---

O olho humano é mais sensível a variações espaciais no brilho (escala de cinza) do que nas cores.

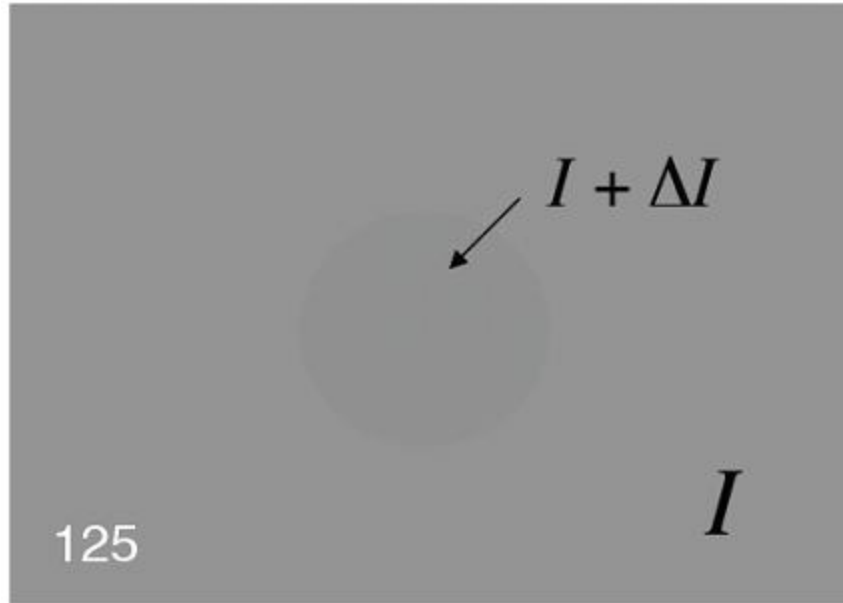
As três imagens somadas formam a da esquerda

Qual das três possui mais detalhes espaciais?



# Experimento da discriminação de brilho

---



$I$  é a luminescência  
medida em  $\text{cd/m}^2$

$$\frac{\Delta I}{I} \approx K_{Weber} \approx 1 \dots 2\% \text{ Lei de Weber}$$



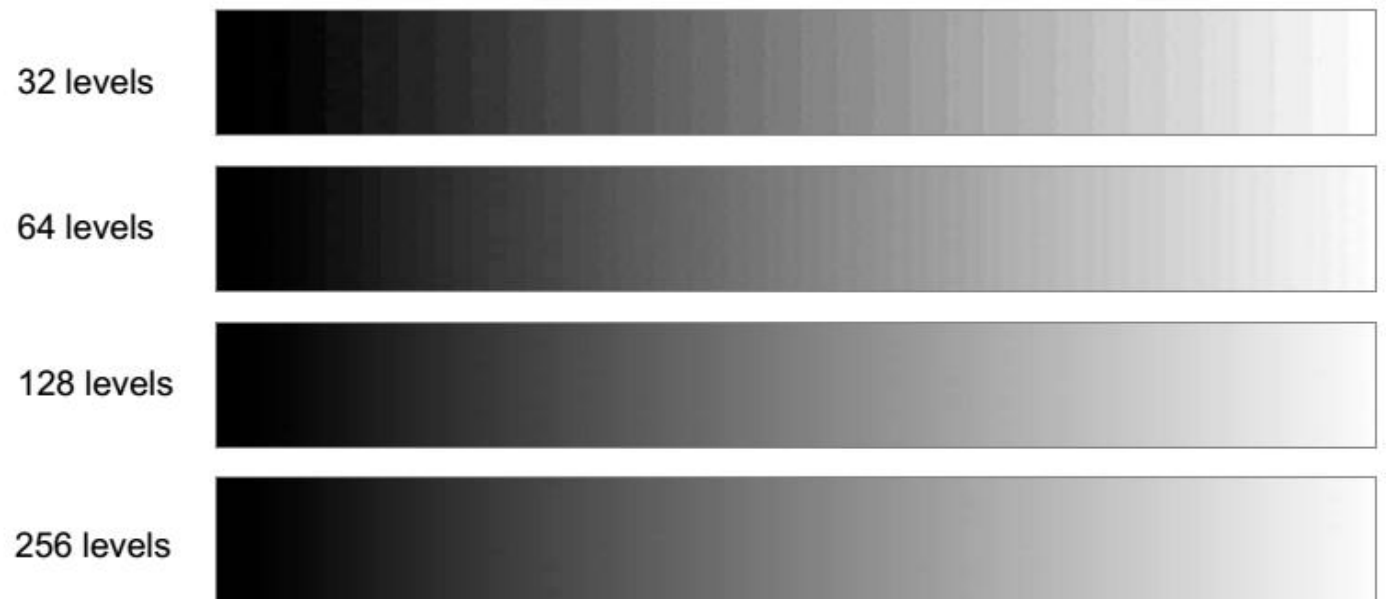


# Quantos níveis de cinza são necessários?

---

Como o olho humano enxerga pequenas diferenças no brilho, precisamos de muitos bits.

Caso contrário as mudanças no brilho criariam uma imagem com aparência descontínua.





# Alcance dinâmico (Max/Min)

## Mundo:

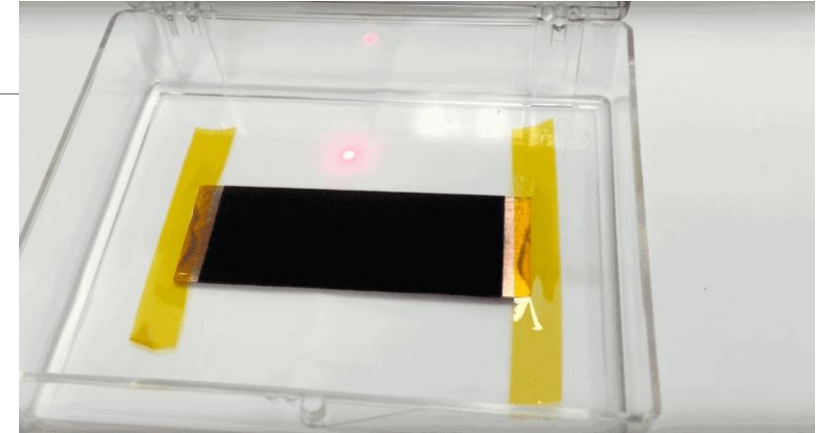
- Possível: 100.000.000.000:1  
(Do sol ao preto absoluto)
- Cenas típicas: 100.000:1

## Olho humano:

- Estático: 100:1
- Dinâmico: 1.000.000:1
  - Como o olho se move, ele se adapta ajustando a exposição com a mudança de tamanho da pupila

## Mídia

- Jornal: 10:1
- Impressão: 60:1
- Monitor:
  - Dell UP2715K: 1000:1 (estático), 8.000.000:1 (dinâmico)
  - Contraste estático é a diferença de luminescência entre um ponto mais claro e mais escuro em uma única imagem.
  - Contraste dinâmico é a diferença entre a imagem mais clara possível e a mais escura



# Mundo real

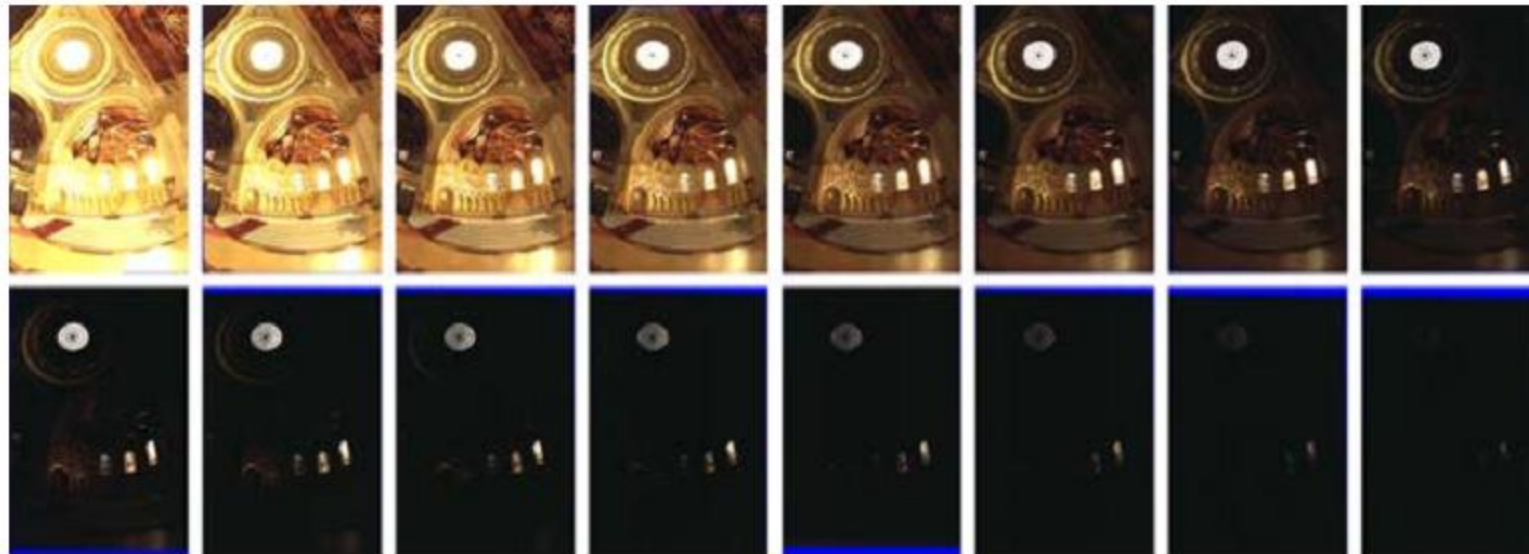
---



# Alcance dinâmico do brilho

---

16 fotografias foram tiradas com diferentes tempos de exposição, de 30 s a 1/1000 s



# Mapeamento de tons

---

Problema: Imagens armazenam um brilho com mais alcance do que os monitores conseguem exibir.

Solução 1: Mapeamento linear

- Pequenas diferenças de intensidade serão quantizadas e detalhes perdidos.

Solução 2: Mapeamento Logarítmico

- Similar a percepção humana

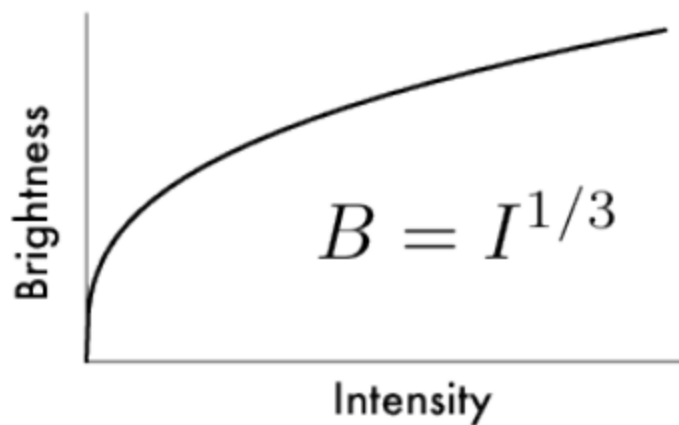
Solução 3: Operadores locais, etc.

# Percepção humana de intensidades

---

$$S = I^p$$

Sensação	Expoente
Brilho	0,33
Som	0,60
Comprimento	1,00
Peso	1,45



O mapeamento logarítmico usa a maior parte do alcance de exibição do brilho para as baixas intensidades da imagem.

# Mapeamento de tons

---

Linear



Logarítmico

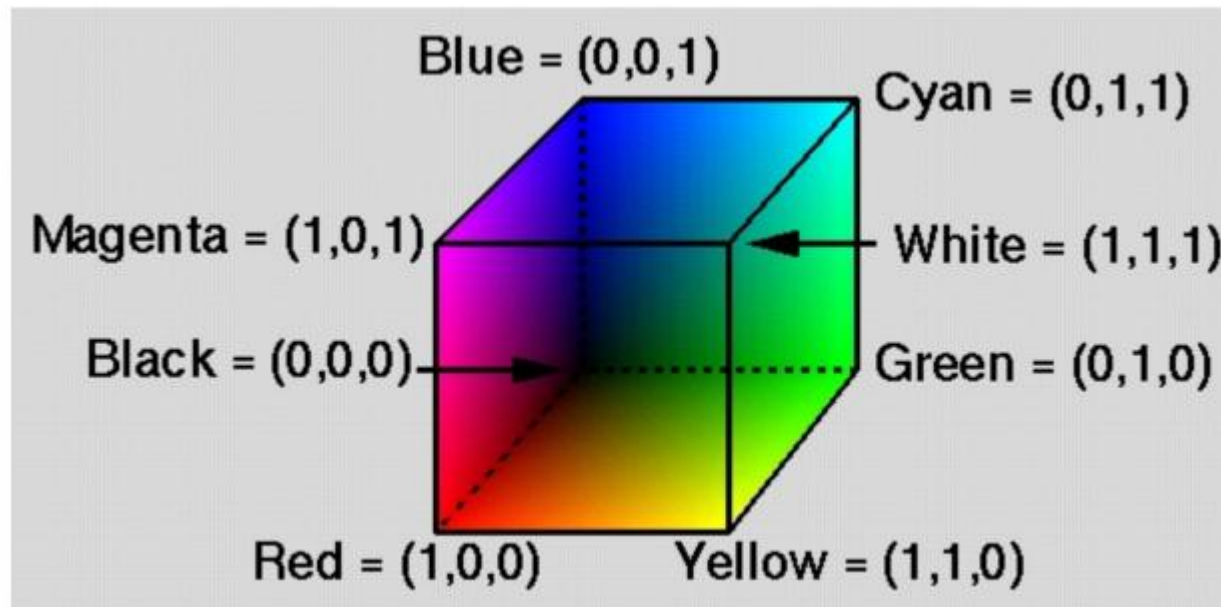


# Cubo RGB

---

Mapeia cada cor primária no espaço RGB para uma distância nos eixos x, y e z.

Preto no (0,0,0) e branco no (1,1,1)



# Modelos de Iluminação em CG

---

Tipicamente, luz é amostrada em um número discreto de primárias (cor)

## Modelos locais

- Apenas caminhos do tipo *fonte luminosa* → *superfície* → *olho* são tratados
- Simples
- Ex.: Pipeline clássico do OpenGL

## Modelos globais

- Muitos caminhos (*ray tracing*, path tracing)
- Complexos



# Iluminação em OpenGL

---

Assume fontes pontuais de luz

- Omnidirecionais
- *Spot*

Interações de luz com superfície modeladas em componentes (modelo de *Phong*):

- Emissão
- Ambiente
- Difusa
- Especular

# Iluminação em OpenGL

---

Suporte a efeitos atmosféricos como

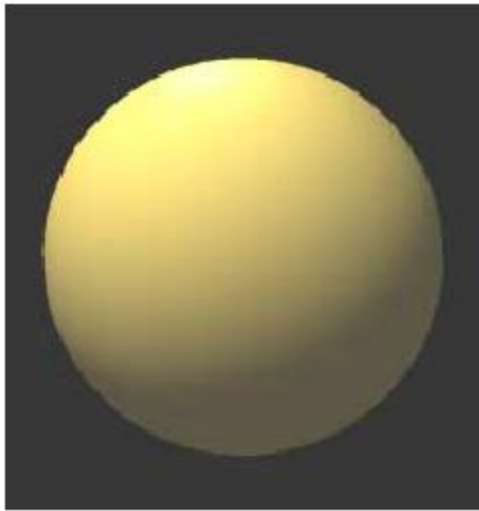
- *Fog*
- Atenuação

Modelo de iluminação é computado apenas nos vértices das superfícies

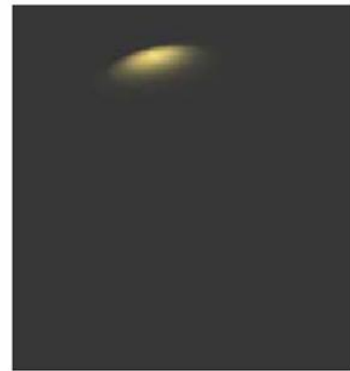
- Cor dos demais pixels é interpolada linearmente (sombreamento *Gouraud*)

# Componentes do Modelo de Phong

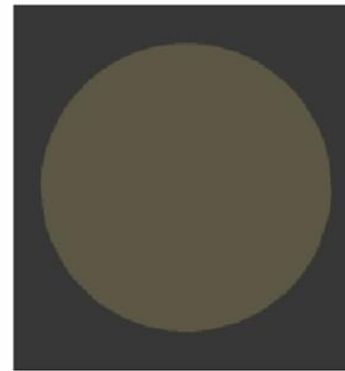
---



Difusa



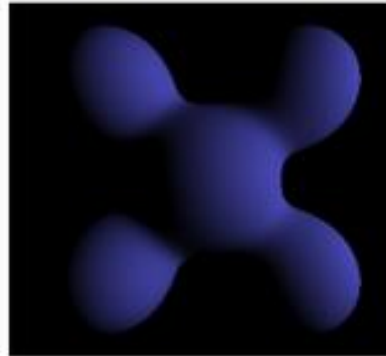
Especular



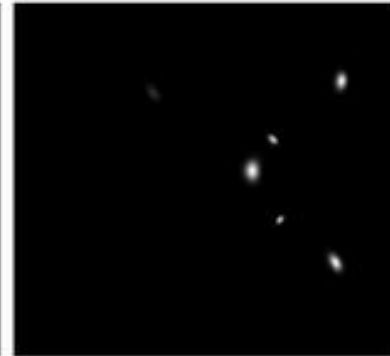
Ambiente



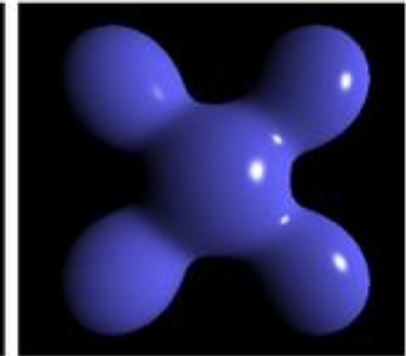
Ambient



+ Diffuse



+ Specular



= Phong Reflection

# Fontes de Luz

---

Para ligar uma fonte: **glEnable (source) ;**

- **source** é uma constante cujo nome é **GL\_LIGHT<sub>i</sub>**, começando com **GL\_LIGHT0**
- Quantas? Pelo menos 8, mas para ter certeza:
  - **glGetIntegerv( GL\_MAX\_LIGHTS, &n );**

Não esquecer de ligar o cálculo de cores pelo modelo de iluminação

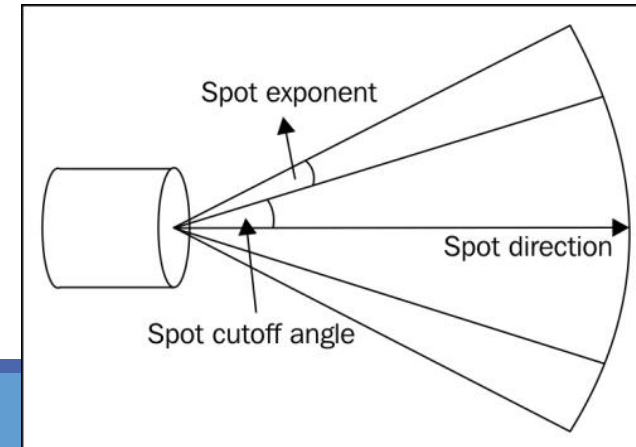
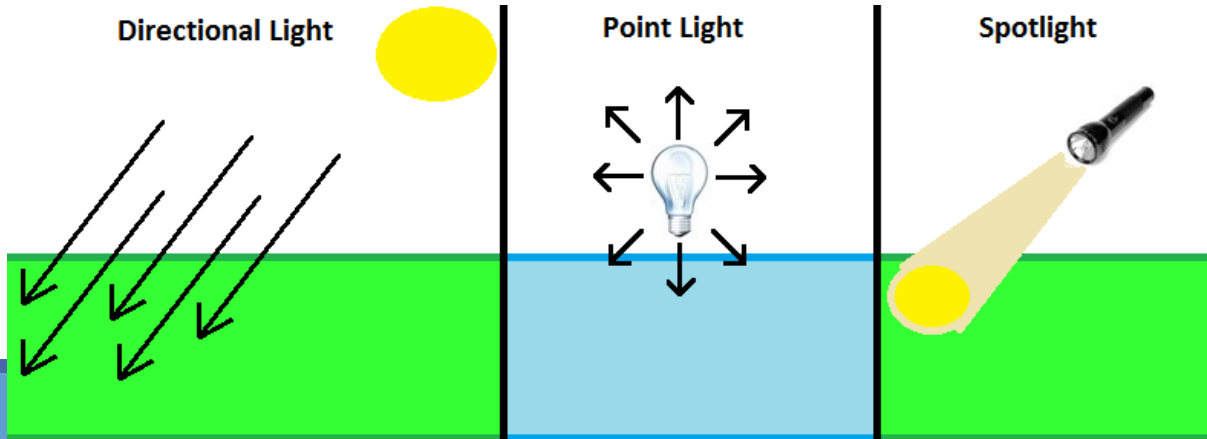
- **glEnable (GL\_LIGHTING) ;**

# Fontes de Luz

Para configurar as propriedades de cada fonte:

**glLightfv(*source*, *property*, *value*) ;**

- **Property** é uma constante designando:
  - Coeficientes de cor usados no modelo de iluminação:
    - *GL\_AMBIENT*, *GL\_DIFFUSE*, *GL\_SPECULAR*
  - Geometria da fonte
    - *GL\_POSITION*, *GL\_SPOT\_DIRECTION*, *GL\_SPOT\_CUTOFF*, *GL\_SPOT\_EXPONENT*
  - Coeficientes de atenuação
    - *GL\_CONSTANT\_ATTENUATION*, *GL\_LINEAR\_ATTENUATION*, *GL\_QUADRATIC\_ATTENUATION*



# Propriedades de Material

---

Especificados por

**glMaterialfv** (*face*, *property*, *value*)

- **Face** designa quais lados da superfície se quer configurar:
  - *GL\_FRONT*, *GL\_BACK*, *GL\_FRONT\_AND\_BACK*
- **Property** designa a propriedade do modelo de iluminação:
  - *GL\_AMBIENT*, *GL\_DIFFUSE*, *GL\_SPECULAR*, *GL\_EMISSION*, *GL\_SHININESS*

# Geometria

---

Além das propriedades da luz e do material, a geometria do objeto é também importante

- A posição dos vértices com relação ao olho e à fonte luminosa contribui no cálculo dos efeitos atmosféricos
- A *normal* é fundamental
  - Não é calculada automaticamente
  - Precisa ser especificada com **glNormal()**

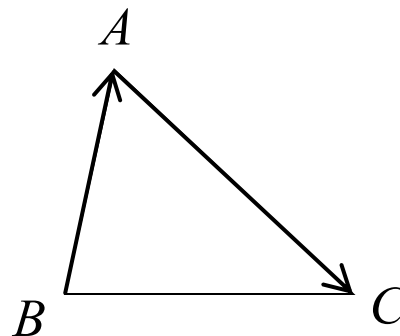
# Computando o Vetor Normal

---

## Triângulo

- Dados três vértices,

$$\vec{n} = \text{normalizar}((A - B) \times (C - A))$$



## Polígono planar

- Uma opção é usar a fórmula do triângulo para quaisquer 3 vértices
  - Sujeito a erros (vetores pequenos ou quase colineares)
- Outra opção é determinar a equação do plano
  - $ax + by + cz + d = 0$
  - Normal tem coordenadas  $(a, b, c)$



# Componentes do Modelo de Phong

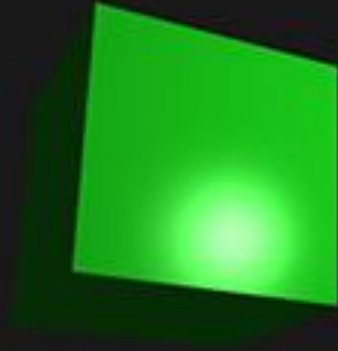
---

Emissão: contribuição que não depende de fontes de luz (fluorescência)

Ambiente: contribuição que não depende da geometria

Difusa: contribuição correspondente ao espalhamento da reflexão *lambertiano* (independe da posição do observador)

Especular: contribuição referente ao comportamento de superfícies polidas



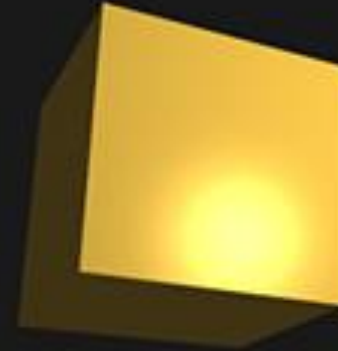
**Emerald**



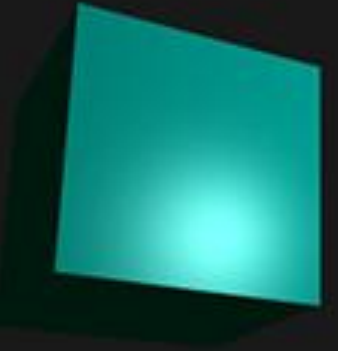
**Pearl**



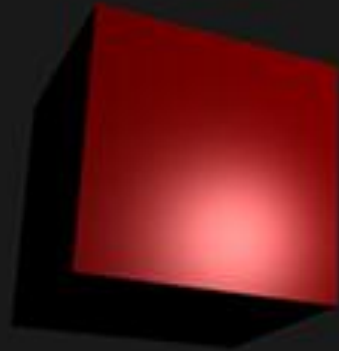
**Bronze**



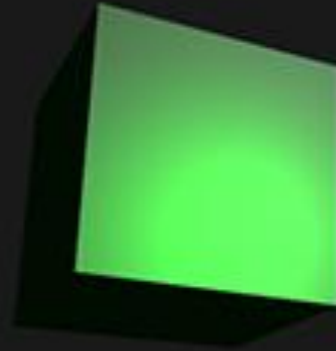
**Gold**



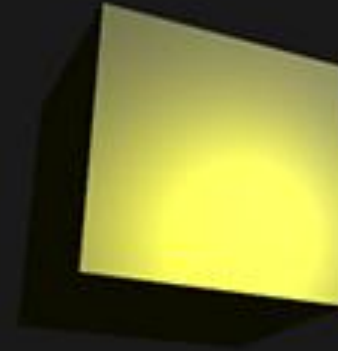
**Cyan Plastic**



**Red Plastic**



**Green Rubber**



**Yellow Rubber**

# Iluminação Ambiente

---

Componente que modela como uma constante o efeito da reflexão de outros objetos do ambiente

Depende dos coeficientes GL\_AMBIENT tanto das fontes luminosas quanto dos materiais

É ainda possível usar luminosidade ambiente não relacionada com fontes luminosas

- `glLightMaterialfv (GL_LIGHT_MODEL_AMBIENT, params)`

Contribuição é dada por  $A = I_A M_A$

# Iluminação Difusa

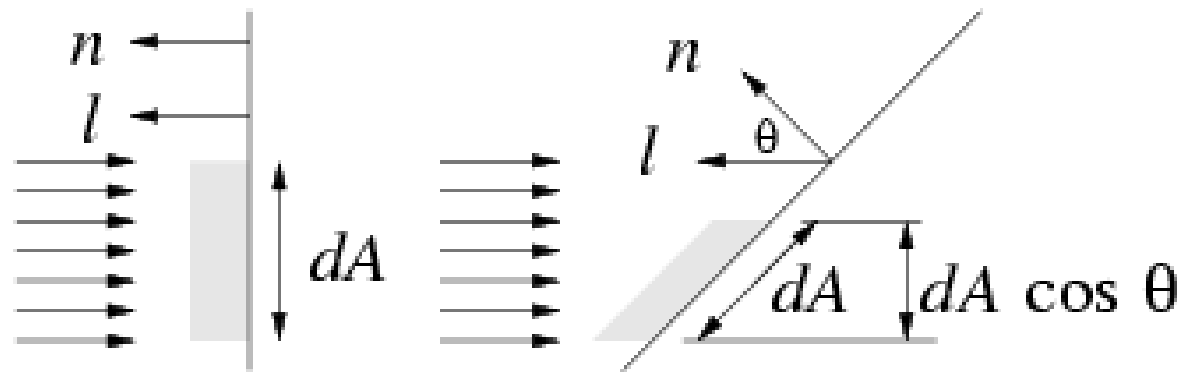
---

Iluminação recebida por uma superfície e que é refletida uniformemente em todas as direções

Característica de materiais foscos

Esse tipo de reflexão é também chamada de reflexão *lambertiana*

A luminosidade aparente da superfície não depende do observador mas apenas do cosseno do ângulo de incidência da luz

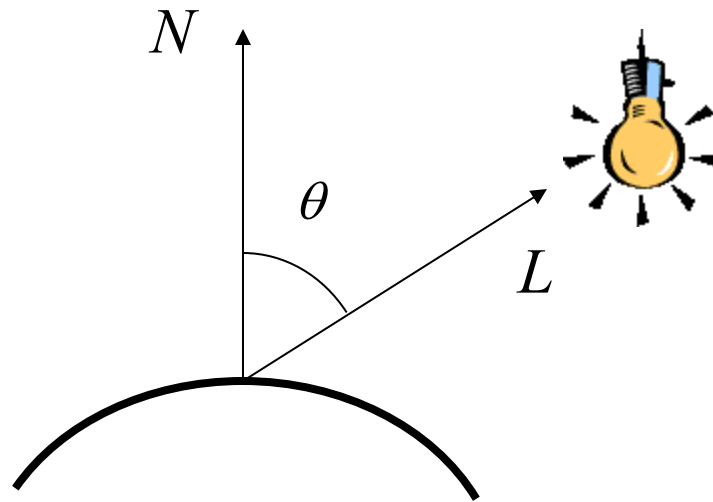


# Iluminação Difusa

---

Contribuição é dada por

$$D = I_D M_D \cos \theta = I_D M_D (N \cdot L)$$



# Iluminação Especular

---

Simula a reflexão à maneira de um espelho (objetos altamente polidos)

Depende da disposição entre observador, objeto e fonte de luz

Em um espelho perfeito, a reflexão se dá em ângulos iguais

- Observador só enxergaria reflexão de uma fonte pontual se estivesse na direção certa

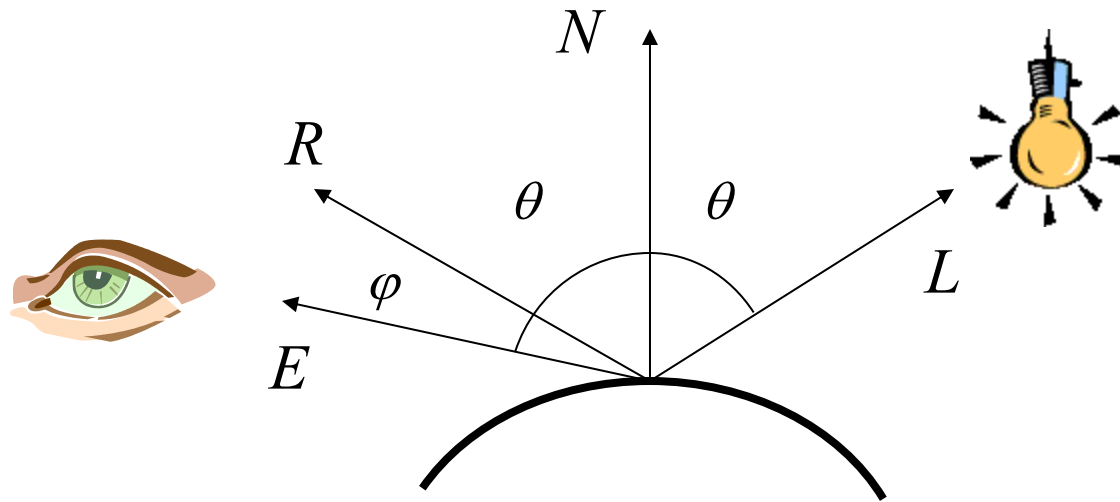
No modelo de Phong simula-se refletores imperfeitos assumindo que luz é refletida segundo um *cone* cujo eixo passa pelo observador

# Iluminação Especular

---

Contribuição é dada por

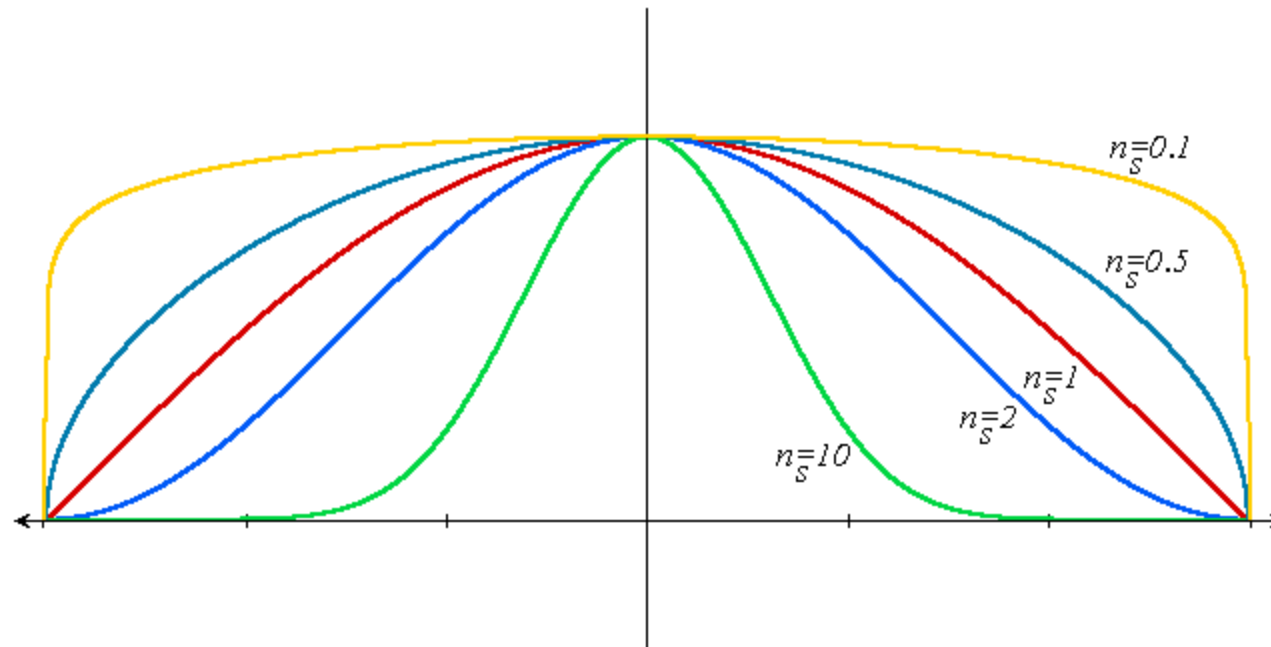
$$S = I_S M_S \cos^n \varphi = I_S M_S (R \cdot E)^n$$



# Coeficiente de Especularidade

Indica quão polida é a superfície

- Espelho ideal tem coef. especularidade infinito
- Na prática, usa-se valores entre 5 e 100





# Atenuação

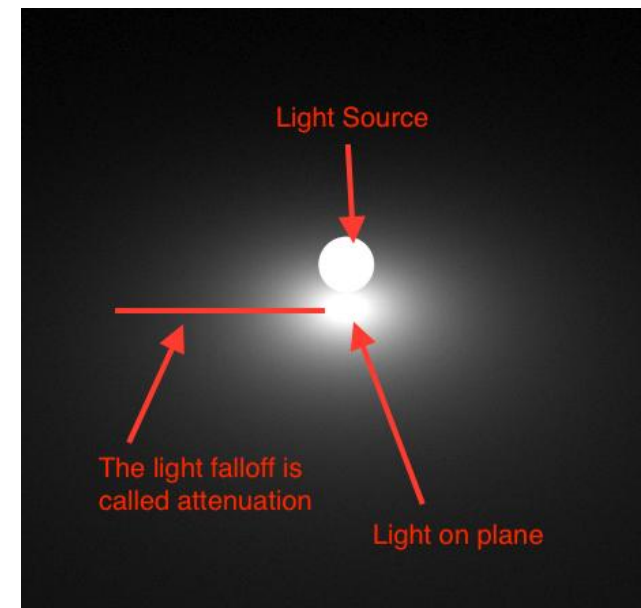
---

Para fontes de luz posicionais ( $w = 1$ ), é possível definir um fator de atenuação que leva em conta a distância  $d$  entre a fonte de luz e o objeto sendo iluminado

Coeficientes são definidos pela função `glLight()`

Por default, não há atenuação ( $c_0=1, c_1=c_2=0$ )

$$aten = \frac{1}{c_0 + c_1d + c_2d^2}$$



# Juntando tudo

---

A atenuação só é aplicada sobre às componentes difusa e especular

A fórmula que calcula a cor de um vértice devida a uma fonte luminosa  $i$  é dada por

$$C_i = \textit{aten} (D_i + S_i)$$

Portanto, no total, a cor é dada pela contribuição da iluminação ambiente (parcela não associada com fontes de luz) somada à luz emitida e às contribuições  $C_i$

$$C = A + E + \sum \textit{aten} (D_i + S_i)$$