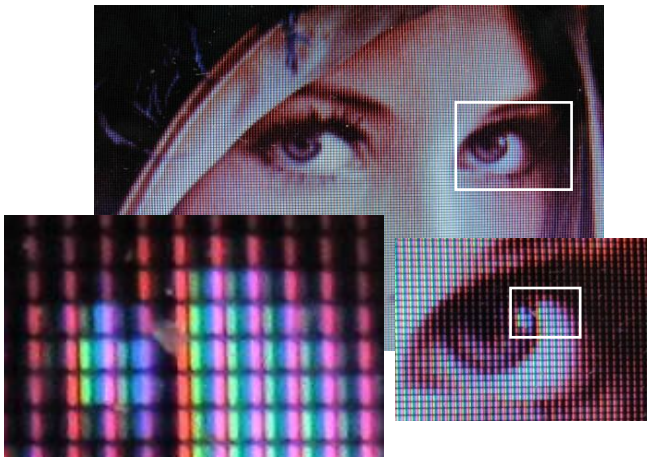
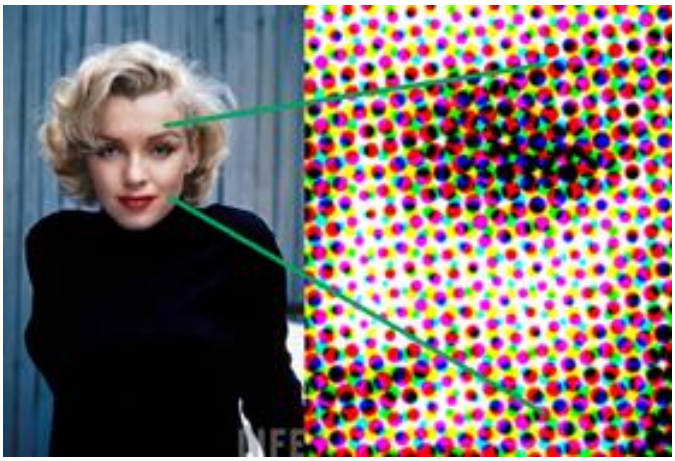
Color

En esta unidad veremos algunos elementos básicos de la teoría del color; en particular analizaremos que es y cómo se percibe el color. La teoría se analizará con una profundidad solamente adecuada para la comprensión de algunos aspectos tecnológicos de la Computación Gráfica.

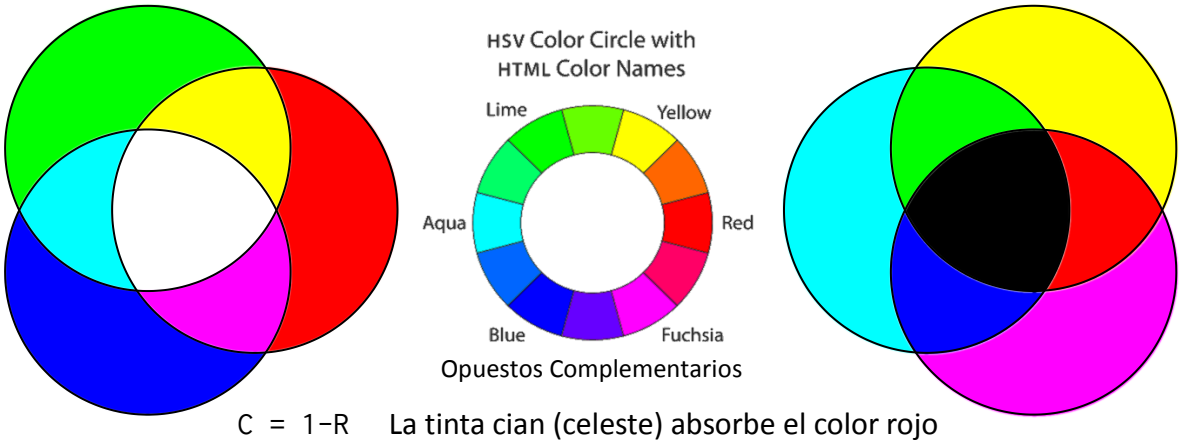




A esta altura parece innecesario decir que la imagen de cualquier monitor o televisor se forma mediante puntos luminosos rojos, verdes y azules (RGB) y que la imagen de una impresora en color se puede formar con tintas cian (turquesa), magenta (fucsia) y amarilla (CMY), con el posible agregado del negro (CMYK). Lo que no resulta nada clara es la razón de esa selección de colores en particular ¿Por qué tres? ¿Por qué esos tres? ¿Acaso mezclando tres sabores se puede reproducir cualquier sabor?

Cualquiera que haya utilizado un programa de dibujo sabe que se puede formar cualquier color con distintas proporciones de rojo, verde y azul. Eso es sencillamente falso.

Comencemos por diferenciar RGB de CMY: antes de mostrar la imagen en colores, la pantalla es negra y el papel blanco, esa es la diferencia fundamental entre el monitor y la impresora. El negro es ausencia de luz (0); el blanco es más difícil de definir, pero digamos que es el máximo (1). Podemos decir que los puntos del monitor emiten luz sumando al negro mientras que las tintas absorben luz restando al blanco, de allí la denominación de colores aditivos RGB y sustractivos CMY.

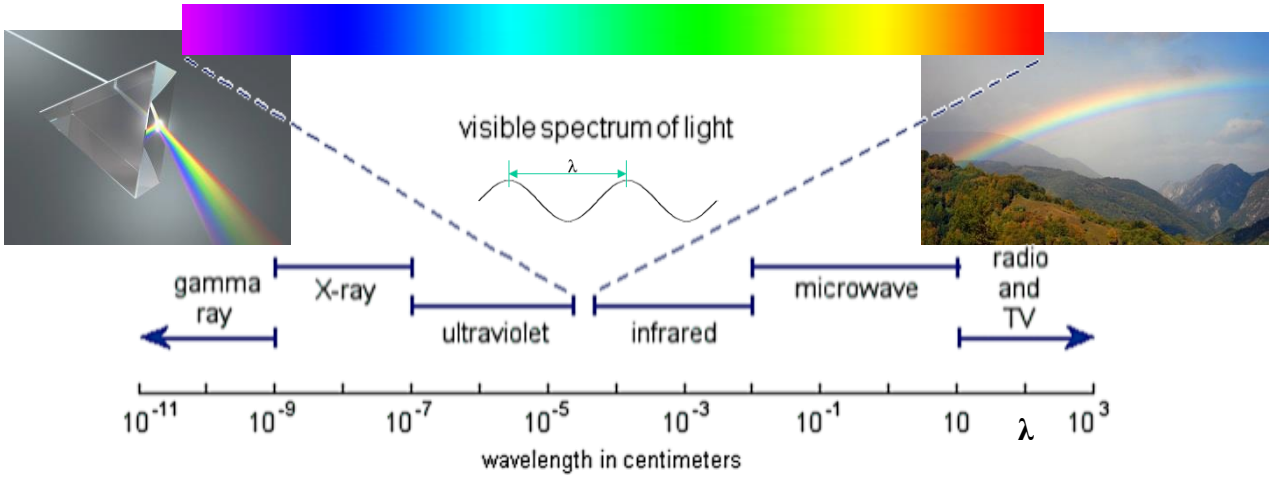


C = 1−R La tinta cian (celeste) absorbe el color rojo
M = 1−G La tinta magenta (fucsia) absorbe el color verde
Y = 1−B La tinta amarilla absorbe el color azul

Los tres colores R,G,B, al máximo emiten todo lo posible ⇒ R+G+B = blanco
Las tres tintas C,M,Y, al máximo, absorben todo el color ⇒ C+M+Y = negro
Las impresoras usan tinta negra más económica K=min(C,M,Y)

Un objeto envía luz a nuestros ojos, la luz puede ser emitida o reflejada por cada punto de la superficie. El monitor encendido emite, el papel pintado refleja. La mayoría de los objetos naturales reflejan la luz, en general del sol, absorbiendo una parte y devolviendo un resto que define su color.
Si un monitor emite azul y verde en máxima intensidad y nada de rojo, vemos el color cian, lo cual puede traducirse como que el cian es ausencia de rojo. Efectivamente, cuando la luz del sol llega a un papel pintado de cian, la tinta absorbe el color rojo y refleja el resto; el papel pintado se ve cian. En la gráfica de arriba se muestra la relación entre estos colores primarios aditivos y sustractivos.

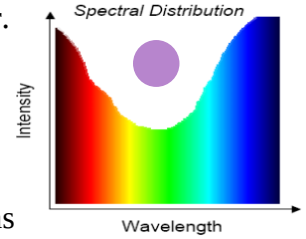
Sabemos que la luz es energía electromagnética, en cantidad proporcional a la intensidad y la frecuencia de oscilación del campo (velocidad de la luz dividida por la longitud de onda). En un determinado rango de frecuencias o longitudes de onda, el ojo las capta y las envía al cerebro, quien las interpreta.



Si el ojo recibe luz de una determinada longitud de onda pura, el cerebro la interpreta igual que alguno de los colores del arcoíris, la relación entre el color percibido y la longitud de onda se ilustra en la figura de arriba. Entonces, una longitud de onda se asocia a un determinado color; pero eso no define un color. Podemos ver que no hay ninguna longitud de onda que corresponda al marrón, por ejemplo.

El arcoíris se forma porque las micro-gotas de agua de la atmósfera desvían la luz del sol, en una medida angular que depende de la longitud de onda. La luz del sol es una mezcla de frecuencias, el ojo sólo percibe entre el rojo y el violeta; el infrarrojo nos da calor y el ultravioleta cáncer de piel.

En general, las superficies opacas reflejan en forma variable, una parte de la luz la absorben y la otra parte la reflejan; la fracción reflejada define su color. Una naranja refleja preferencialmente las ondas de mayor longitud. Llamamos gris a una superficie que refleja en igual medida todas las longitudes de onda visibles de la luz del sol, negra si esa medida es nula y blanca si refleja el 100% (en forma difusa como un papel y no como un espejo o un acero pulido). Si se iluminan con luz de color, las superficies reflejan en las mismas proporciones, pero el efecto es que parecen de otro color.



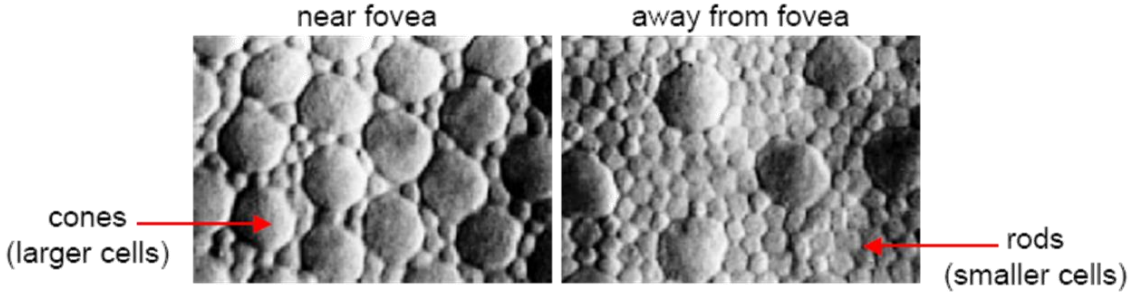
Podemos decir entonces que un color se forma mediante una mezcla de longitudes de onda, una distribución o espectro I(λ) de intensidad de energía luminosa (emitida o reflejada) en función de la longitud de onda.

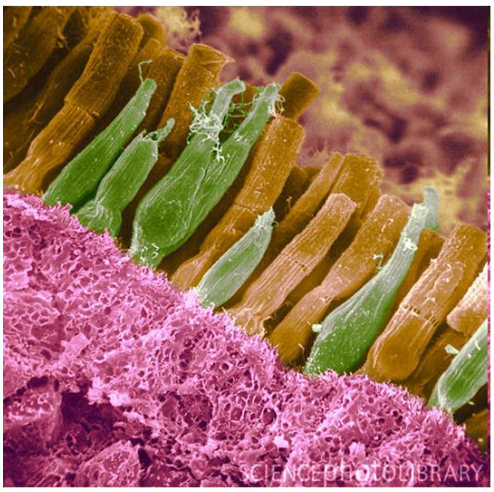
Ahora bien ¿Que es el color salmón o, incluso, el rojo? Tratándose de sentidos fisiológicos los estándares se definen analizando estadísticamente las respuestas de la gente, observadores entrenados y en situaciones de laboratorio. Si a un niño se le dice que una naranja es de color naranja esa será su referencia de comparación; pero habría que saber si el niño ve una banana del mismo color que una naranja; o si ve distinto color en dos naranjas, que para la mayoría tienen el mismo color. En el fondo esto equivale a preguntar si el sistema sensor del color funciona igual en todos los humanos; la respuesta es un obvio no; pero, salvo por el daltonismo (que es muy frecuente), existe poca desviación y se puede hablar de un "observador estándar".

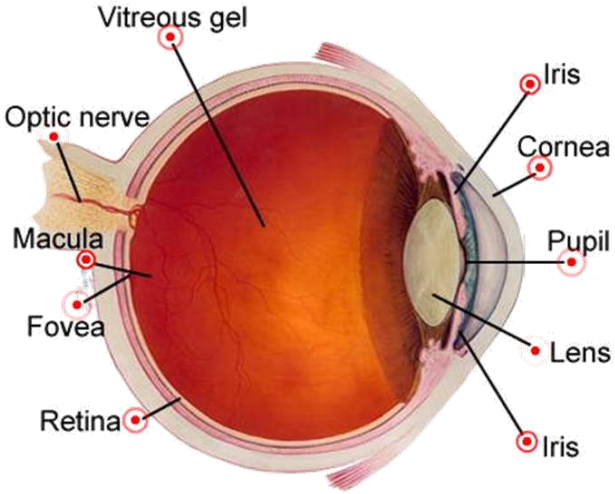
El ojo es el receptor de la imagen visual. Si bien la piel siente la luz que absorbe, sobre todo el calor de las ondas infrarrojas, el ojo está especializado para sentir la luz visible y discernir su distribución espectral y espacial; es capaz de transmitir al cerebro la información necesaria para construir una imagen del entorno. Los humanos distinguimos una fracción determinada de todas las mezclas posibles de frecuencias (hay distribuciones distintas que "se ven" igual) y distinguimos desde donde viene la información, a través de una percepción distribuida a lo ancho y alto; más un sentido de la distancia, gracias a que tenemos dos ojos separados (visión estereoscópica).

El ojo humano consta de un pequeño disco negro: la pupila, que limita la cantidad de luz que entra en el ojo, mediante músculos autónomos que cambian su tamaño; es negra justamente porque no refleja luz, todo lo que entra se aprovecha. La luz entrante se enfoca para que los rayos que provienen de la zona observada se crucen en un único punto, el foco; y así la imagen de esa zona será nítida; de ello se encarga el cristalino, una lente (tejido transparente) que se acomoda voluntariamente mediante músculos. La luz impacta en la retina, que es la capa que contiene los transductores de luz en impulsos nerviosos, que viajan al cerebro a través del nervio óptico. Finalmente, es el cerebro quien construye la imagen.

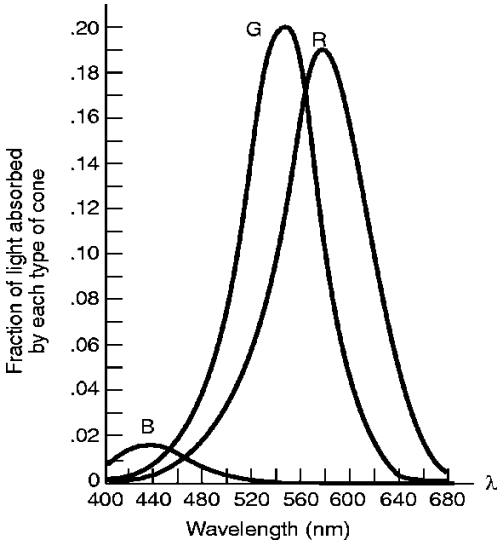
Para la teoría que estamos analizando son centrales los transductores de la retina, que son células especializadas. Hay dos grandes tipos: los bastones o cilindros (*rods*) y los conos (*cones*).







Los bastones se distribuyen densamente por toda la retina y son los transductores más sensibles a la cantidad de luz, nos permiten ver formas en la oscuridad, pero no colores; en condiciones de luz de día se saturan y no aportan casi nada al proceso de la visión. Los conos, en cambio, son menos sensibles y no aportan a la visión nocturna; se concentran más en la fóvea, la región donde chocan los rayos de luz que provienen de la zona a la cual miramos en forma directa. Son los conos los que nos permiten ver colores.



Hay **<u>tres</u>** tipos de conos, que filtran las longitudes de onda de distinto modo; es decir que no dejan pasar toda la luz en la misma medida. De los tres tipos de conos, uno es más sensible a la luz en la zona de longitudes de onda cortas (S por *short* o B por *blue*), otro en la zona media (M por *mid* o G por *green*) y el tercero privilegia las longitudes de onda más largas (L por *large* o R por *red*). Las sustancias responsables de la transducción de luz en señales nerviosas son las opsinas: rodopsinas en los bastones, fotopsinas S, M y L en los conos. Una falla en la génesis de esas proteínas puede provocar la percepción alterada de los colores; siendo la más común el Daltonismo. El hecho de que esas proteínas se codifican en el cromosoma X es la razón por la cual menos mujeres padecen Daltonismo y el hecho de que las G y R son muy parecidas explica que la mayoría de los daltónicos no distinguen entre rojo y verde.

Para comprender algunos porqués de las tecnologías del color, lo más relevante de lo antedicho es que, habiendo tres tipos de receptores de color, el espacio funcional de todas las distribuciones posibles se puede representar en tres dimensiones, para un humano con visión estándar.

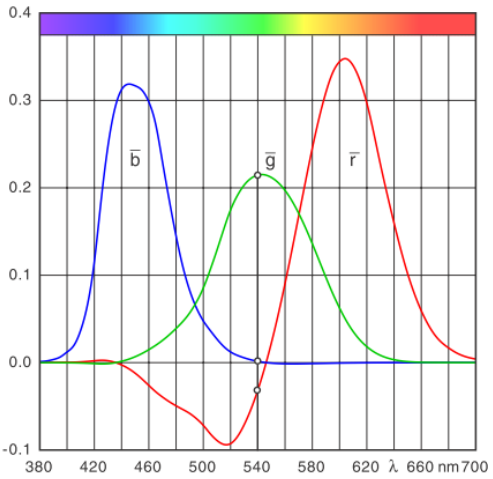
El espectro de luz que incide en una pequeña porción de la retina es una función $I_i(\lambda)$, que integrada mide el total de energía luminosa entrante. A su vez, un dado cono – digamos R – absorbe una fracción de la luz incidente según una función $A_R(\lambda)$ acotada entre cero y uno, que multiplica a la energía recibida para definir la absorbida. El total de energía absorbida por el cono es $I_R = \int I_i \, A_R \, d\lambda$ y es esa cantidad de energía la que define la magnitud de la señal que el cono envía hacia el cerebro; con muy poca precisión podríamos decir que es la cantidad de rojo que percibimos en ese punto de la retina.

Habiendo tres tipos de conos, desde cada pequeña zona del ojo se envía al cerebro una terna de valores que definen el color percibido. Cualquier otra distribución de luz incidente, que integrada dé como resultado los mismos tres números, se debería percibir del mismo color.

Ya encontramos la razón fisiológica de que sean tres colores. Los espectros de absorción de los tres tipos de conos se relacionan con el hecho de que se utilicen el rojo, el verde y el azul, aun cuando la correspondencia es irreal.

Esto no es todo ni es totalmente correcto, pero es suficiente para entender la mayor parte.

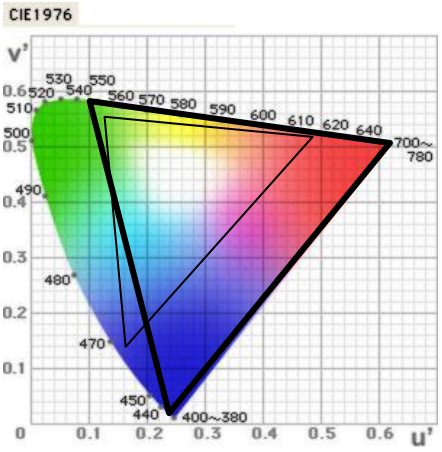
Desde el siglo IXX (Young - Helmholtz) se cree que alcanza con rojo, verde y azul para reproducir cualquier color natural. Los experimentos clave los realizaron Wright y Guild cerca de 1930. La idea es presentar a los sujetos un panel dividido en dos campos. En el campo de referencia hay un color, natural o no, puede ser una flor o un color de pintura iluminado por una luz blanca estandarizada o bien un color del arcoíris iluminando un fondo blanco. Del otro lado hay tres lámparas estandarizadas, roja (700nm), verde (546.1nm) y azul (435.8nm), iluminando un fondo blanco y con comandos para graduar cada intensidad. El sujeto debe modificar los tres controles para reproducir el color de referencia. Algunos colores no podían ser reproducidos a menos que se agregue luz roja del otro lado, a la referencia. En la gráfica se representa la intensidad de las lámparas que reproducen los colores puros del arcoíris, los valores negativos indican que se agregó luz roja al campo de referencia.



En una PC común, puede hacer el experimento de recortar un pétalo de flor y buscar la terna RGB que representa el mismo color. Puede que la encuentre o no, pero ya debería darse cuenta de que la "distribución espectral" no es la misma, el pétalo tiene un espectro continuo de frecuencias mientras que el monitor tiene solo tres picos.

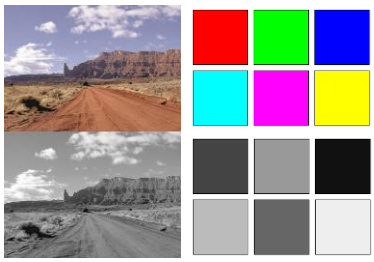
Podemos identificar las tres intensidades como coordenadas en un espacio donde los puntos representan colores. El conjunto de colores que puede distinguir un ojo humano se agrupa en una zona de ese espacio tridimensional. Para un aparato, el conjunto de colores que puede captar (cámara) o reproducir (monitor) también está en una zona limitada. Esa zona se denomina **gamut**, ya sea para el ojo o para un dispositivo.

El espacio normalizado para representar el gamut está definido por la norma CIE, a la derecha podemos ver una representación bidimensional (intensidad máxima) del espacio, con las coordenadas transformadas de modo que la distancia entre dos puntos se relacione con la diferencia percibida entre sus colores (espacio CIELUV). Hay otras formas que aparecen en los textos, pero la idea es que el gamut de un determinado dispositivo cubre una zona determinada del gamut del ojo humano estándar. En tal sentido, cualquier representación gráfica del diagrama completo es falsa, pues este monitor o el papel en el cual está impreso este texto, no puede reproducir todos los colores identificables por el ojo, la figura es solamente indicativa de donde están más o menos los colores.



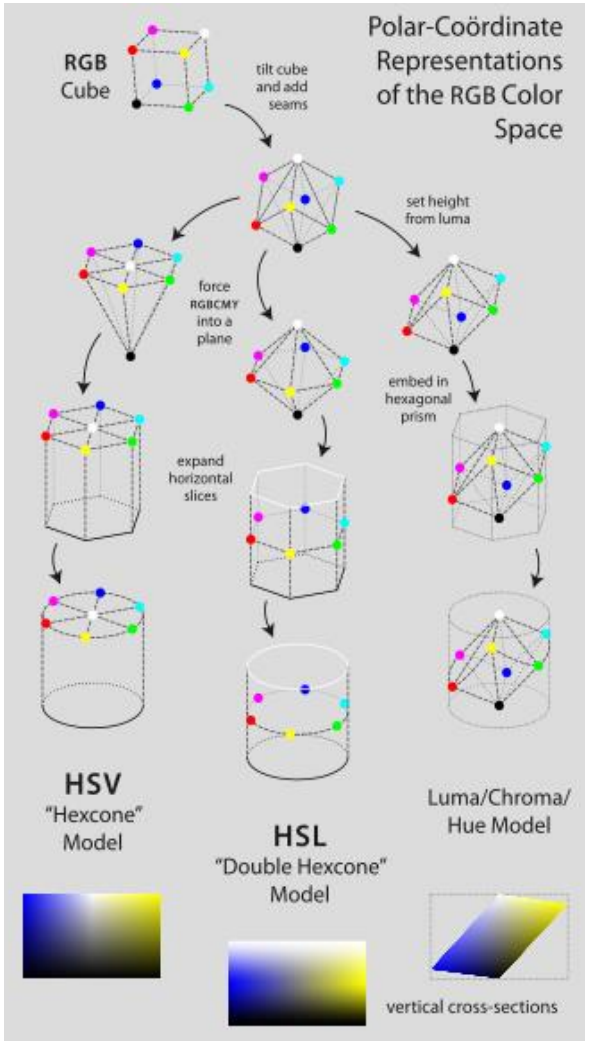
Una determinada terna de colores puede formar cualquier color que se encuentre en el interior del triángulo que definen esos tres colores, como puntos, en el diagrama. La línea gruesa es el conjunto de colores con intensidades "positivas" del experimento original. Se puede entender por qué hacía falta agregar luz roja para cubrir todo el campo por extrapolación, faltan muchos colores a la izquierda de la línea que va del verde al azul; agregándoles rojo "se mueven" hacia el rojo, hasta entrar en el triángulo. Para computación, se utiliza el sRGB estándar de 1996, que se muestra aproximado en línea delgada.

Si al ojo le entra luz gris, entre negro = 0% y blanco = 100%, la diferencia percibida entre 3% y 5%, es mucho mayor que entre 95% y 97%. La forma más usual y simple de hacer la corrección es con una curva potencial donde la intensidad de luz emitida por un monitor, normalizada entre 0 y 1, sea una potencia γ (gama o gamma) de la intensidad (~ eléctrica) del dispositivo emisor o numérica de un programa de manipulación de imágenes. Actualmente la función estándar del sRGB no es exactamente potencial, pero sigue llamándose curva gama (https://en.wikipedia.org/wiki/SRGB).



En el diagrama anterior, el blanco no se logra con iguales intensidades de R, G y B standard (no está en el centro del triángulo sRGB). Para convertir una imagen en colores a escala de grises, primero se deshace la corrección gamma y luego se calcula la luminancia (R,G,B → L,L,L) mediante la siguiente fórmula: L = 0.2126 R + 0.7152 G + 0.0722 B. Puede verse que el verde influye mucho más que el rojo en la percepción de luminosidad y el azul no aporta casi nada.

La forma más común de representar el espacio de colores en computación es definir un cubo RGB que va desde RGB={0,0,0} (negro) a RGB={1,1,1} (blanco).



Más orientado a lo perceptual, el lenguaje artístico ha dado lugar al sistema HSL, por matiz (*hue*), saturación y luminosidad. El matiz indica el color de base o puro del espectro y se suele cuantificar en forma circular o angular [0º, 360º). La saturación es cero para el gris y uno para el color puro y se indica en forma radial. La luminancia varía entre cero para el negro y uno para el blanco y se indica en dirección vertical. Ese espacio suele representarse mediante un doble cono; a veces inclinado, puesto que el amarillo puro es más luminoso que el azul puro; pero en computación debe entenderse como un cilindro de eje gris, piso negro y techo blanco; pues las coordenadas son independientes entre sí y cada una puede adquirir cualquier valor en su rango. Este sistema es mucho más intuitivo que el RGB, pues puede especificarse fácilmente un matiz (digamos: verde) y luego terminar de definir el color por medio de la saturación y la luminosidad. Hay que aclarar que en el lenguaje se utiliza mucho el amarillo como un "cuarto primario" para terminar de definir el matiz ("tirando al amarillo" o "amarillento").

El modelo perceptual más utilizado para comparar colores en computación es el Lab. También se relaciona con la fisiología del ojo: después de los conos y bastones, y antes de recolectarse en el nervio óptico, las señales se procesan en unas células ganglionares que, en el humano, están delante de la retina. Se cree que estas células transforman las señales individuales de los conos en una señal **L** de luminancia o suma (donde son mucho más importantes R y G que B) una señal **a** = R − G que identifica cuanto de rojo – verde tiene un color y una señal **b** = **L** − B que mide cuanto de azul – amarillo tiene. No hay rojos verdosos ni azules amarillentos. Este espacio de color permite discernir mejor en la computadora si una zona "se parece" a otra o si cambió con el tiempo. En la figura del diagrama CIE de la página anterior (Luv) ya podemos ver que se nota una variación horizontal del verde al rojo y una vertical del amarillo al azul.

En cuanto al análisis de las tecnologías, ya sean las disponibles como las propuestas, hay que tener además un mejor conocimiento del gusto del consumidor, un análisis que pertenece a la sicología o sociología aplicada. En las tecnologías de audio ya sea para muestreo o *sampling* (datos por segundo), almacenamiento (cantidad de bits) o para la linealidad de la reproducción, el consumidor es muy exigente, en los años 90 aún se defendía la superioridad de los equipos analógicos con válvulas por sobre los equipos digitales con componentes de estado sólido. En las imágenes y el video el usuario es muchísimo menos exigente, sobre todo con los colores. Si nos detenemos a examinar una excelente imagen tomada con una excelente cámara y mostrada en un excelente monitor, veremos grandes diferencias de color con la realidad y aun así nos impresiona favorablemente. Ello ha permitido que la compresión de imágenes con pérdidas (jpg) resulte aceptable para el consumidor. Lo que el usuario no ve con buen agrado es el "*banding*", las transiciones bruscas entre bandas de color diferente, cuando se

supone que el degradé debería ser suave; el efecto es muy visible en imágenes oscuras y sobre todo en video; la solución al *banding* aún no está desarrollada ¿Pasará por más bits o por un gamma variable?

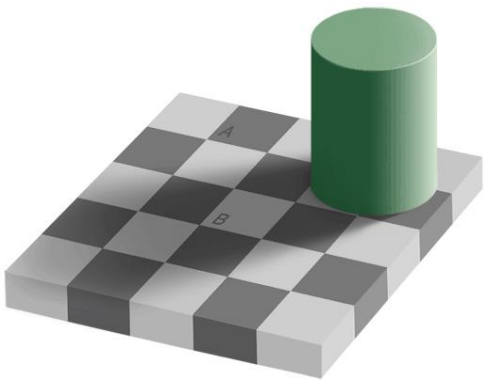
En las primeras impresoras a color, el negro se formaba acumulando las tres tintas CMY eso da un gris o verde oscuro que no llega a ser un negro aceptable, además las tintas de color son caras y además gran parte del texto impreso es negro. Parece muy sano, lógico y tecnológicamente sencillo medir el mínimo de las tres tintas y aplicar esa cantidad de tinta negra: K=min(C,M,Y), con lo cual el negro es efectivamente negro y se ahorran tintas de color: $C^*=C-K$; $M^*=M-K$, $Y^*=Y-K$. Un ejemplo de avance.

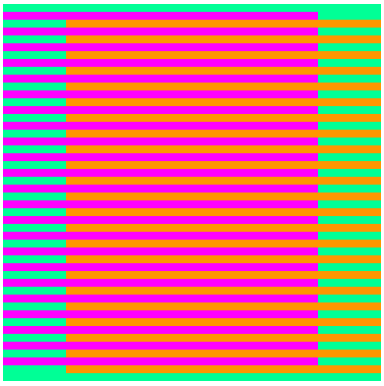
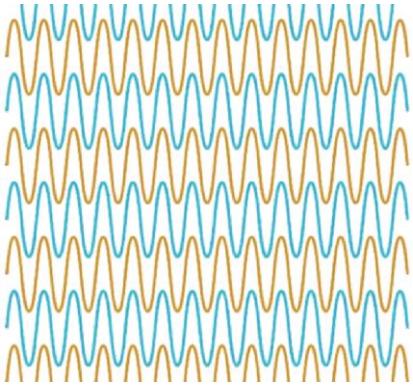
En 2010 la firma Sharp introdujo un cuarto subpíxel amarillo en sus televisores argumentando que con ello ampliaba la gama de colores (entre muchos otros argumentos de venta). En primer lugar, toda la tecnología actual de captura y trasmisión está basada en RGB, un reproductor de RGBY no tiene forma de obtener ese cuarto dato de la fuente; solo puede hacer un análisis estadístico y una reconversión funcional (deducir algorítmicamente el nivel de amarillo). Pero además, si miramos el diagrama CIE, vemos que el amarillo no agrega un área importante al gamut, más interesante sería agregar un color centrado en los 480 o 490nm (azul índigo), aunque puede ser que haya mucha mayor variedad y cantidad de cosas naturales amarillas que de cosas índigo o verde-azuladas. El agregado de amarillo podría tener sentido por otro lado: la luminosidad; el rojo puro consiste en un subpíxel a pleno y los otros dos apagados, el amarillo se hace con rojo y verde a pleno y el blanco con los tres a pleno; es imposible hacer un rojo o incluso un amarillo muy intenso y sin embargo la naturaleza nos ofrece rojos y amarillos intensos, por ese lado se puede justificar un poco mejor el agregado de amarillo.

Con todo eso en vista, a muchos se les ocurrió una mejor solución tecnológica: agregar un subpíxel blanco. No necesita ser transmitido porque se enciende al máximo de entre R, G o B con gran ahorro de energía (vital en equipos portátiles), el ahorro es mayor si se trata de mostrar texto con fondo habitualmente blanco. Además, permite aumentar la intensidad de los colores, sobre todo de los no-saturados (pasteles) y hacerlos más vívidos. La desventaja que lo hizo fallar fue que más subpíxeles agregan granularidad y serruchado visible en el texto y eso es muy molesto para los usuarios.

Esta teoría del color es la punta de un iceberg en cuanto a la representación y percepción del color, quien desee dedicarse a las ciencias, técnicas o artes gráficas debería aprender muchas cosas más.

Para ver algunas de las cuestiones fundamentales que pasamos por alto en esta teoría básica, basta con analizar algunas ilusiones ópticas. En particular, dijimos que los tres números integrales de los conos definen el color; esto es falso pues el color también depende del contexto, como puede verse comparando los grises A y B de la figura izquierda o el verde y el celeste de la central, en ambos casos, el espectro entrante es el mismo, el resultado de la integración es el mismo y los colores se perciben distintos. Lo mismo sucede con los tenues naranja y celeste, en el fondo de la figura derecha, que es blanco.



Hay otras ilusiones de color “falso” provocadas por acostumbramiento (persistencia y *afterimages*) o por movimiento (buscar “*Benham’s top*”) que denotan que el tiempo es también una variable importante.

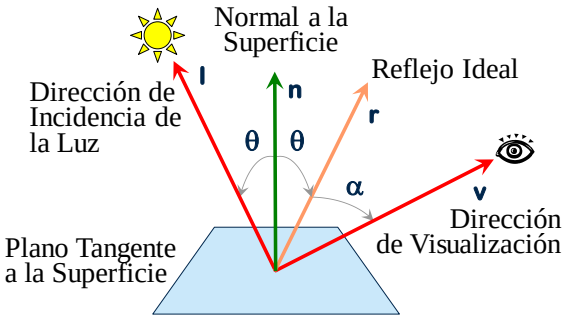
Con esto queda claro que no queda nada claro lo que “es” un color ¿Es lo que se recibe como distribución espectral (color físico: los grises A y B “son” iguales) o es lo que se percibe en el cerebro comparando con patrones aprendidos e inmerso en un contexto (color percibido: los grises A y B “son” diferentes)?

Iluminación y sombreado

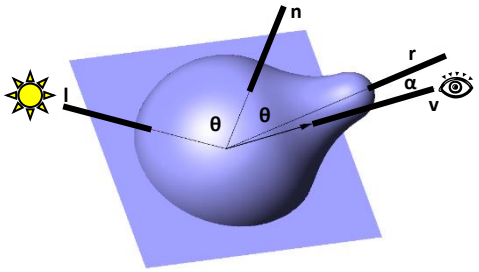
La física de la iluminación comprende la emisión de luz, su incidencia en las superficies y los procesos de refracción y reflexión complejas que sufre en los objetos. El objetivo del renderizado fotorealista es producir una imagen que haga llegar luz a nuestros ojos tal como lo haría la escena real. Esa empresa es imposible. En CG se modela la iluminación mediante simplificaciones que permiten calcular y representar una escena con mayor o menor grado de realismo. Los modelos más complejos se utilizan cuando se dispone de mayor tiempo y poder de cálculo, pero en general se utiliza un simple modelo heurístico (que "funciona", pero no tiene mucho sustento físico) que estudiaremos en detalle.

Se entiende por "sombreado" (*shading*) al degradé de color que vemos en un objeto por el hecho de no recibir luz en forma uniforme. No debe confundirse con la proyección de sombras (*shadow casting*). El sombreado es el responsable de que la imagen de una bola roja no se vea como un disco rojo uniforme.

El sombreado de objetos opacos se modela mediante una función denominada BDRF, por *bidirectional reflectance distribution function* o función de distribución bidireccional de la reflectancia, que indica, para un punto de la superficie del objeto, la proporción de luz reflejada, en función de la longitud de onda, de la dirección de incidencia y de la dirección hacia el ojo u observador. Para esa función, la superficie se supone suave en el punto, es decir que tiene un plano tangente y un vector normal, respecto al cual se definen el ángulo de incidencia y el de visión. Para cada par de ángulos, la superficie refleja una proporción variable de la luz incidente en cada longitud de onda.



La reflectancia es la fracción [0,1] reflejada. Suele representarse en coordenadas esféricas. A la derecha se muestra un diagrama que indica la forma de la función; para una dada dirección de incidencia fija, se asigna una reflectancia a cada dirección visual, que se mide como indica el vector: al cortar con la superficie al rayo que va del punto al ojo.



Una función BDRF modelada (a diferencia de la medida en un experimento) puede incorporar mayor o menor complejidad física. Las más simples son isotrópicas, es decir que la superficie no tiene propiedades distintas en distintas direcciones como tendría, por ejemplo, un acero lijado en una dirección. Hay distintos modelos; algunos de mediana complejidad consideran a la superficie como un conjunto aleatorio de micro-espejos; los más complejos tienen en cuenta efectos de óptica física y/o capas diferenciadas de superficie; pero en la práctica estándar de CG solo se utiliza el sencillo modelo de Phong, que enseguida veremos.

El BDRF simulado supone siempre a la superficie lisa y uniforme. Si la superficie es rugosa (naranja, roca) o no uniforme (mármol, madera veteada), debería definirse la geometría rugosa y cambiante; pero en la práctica se recurre a la aplicación de texturas y otros trucos complejos sobre superficies simples, algunas de esas técnicas se verán más adelante.

Respecto a la luz incidente existen los modelos locales, que solamente consideran la luz proveniente de un reducido número de fuentes (lámparas) puntuales y los modelos globales, que calculan la luz proveniente de fuentes extensas o del reflejo en otros objetos del entorno y admiten, además, objetos translúcidos que refractan la luz. Los modelos globales producen efectos muy realistas, pero requieren una gran cantidad de cálculos y aún no son utilizables en videojuegos o animaciones interactivas en "tiempo real", requieren el procesamiento "*batch*" u *"off-line"*, que consiste en preparar todo para mandar a renderizar la imagen o la animación en el tiempo que insuma, sin interacción humana.

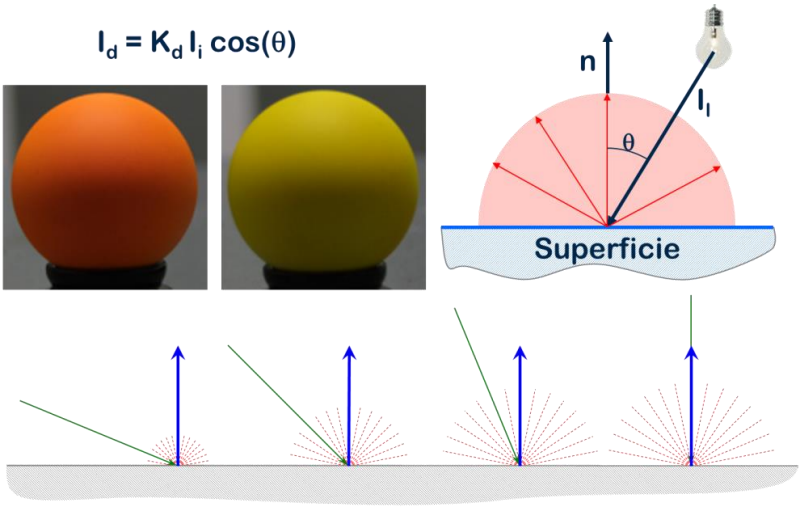
Entre los modelos globales más útiles podemos considerar el análisis integral de energía radiante recibida y reflejada, un método conocido como "*radiosity*" que analiza cuanto recibe cada superficie de cada otra superficie (costo algorítmico cuadrático) y se utiliza también en el cálculo de radiación del calor; por ejemplo, para calcular la disipación de energía en un satélite y las temperaturas de sus componentes.

En la práctica habitual, interactiva o en tiempo real (OpenGL o DirectX y sus *wrappers* o envoltorios de más alto nivel) se modela la luz puntual (posición, distribución angular y dependencia con la distancia) y el sombreado BDRF simple, también puede definirse un modelo de neblina (*fog*).

Modelo de iluminación de Phong

Se trata de un modelo BDRF local y lineal que suma tres mecanismos de reflexión de la luz de las fuentes: ambiente, difusa ideal y especular no ideal. Veremos los modelos parciales uno por uno.

Reflexión difusa ideal:



Según este modelo debido a Lambert (1760), la luz que refleja el objeto depende solamente de su color y el ángulo con el que incide la luz. La superficie "difunde" la misma cantidad de luz en todas las direcciones posibles del hemisferio exterior al plano tangente. Ese modelo sólo es realista para superficies difusas; éstas son superficies mate, con la textura de un polvo comprimido (tiza) o bien lijadas en forma fina, sin ningún brillo.

El sombreado del objeto no depende de la posición del observador. Depende sólamente del coseno del ángulo θ entre la dirección del rayo incidente y la normal; que es el producto escalar entre **l** y **n**, si ambos están normalizados (módulo 1) y apuntan hacia el exterior de la superficie.

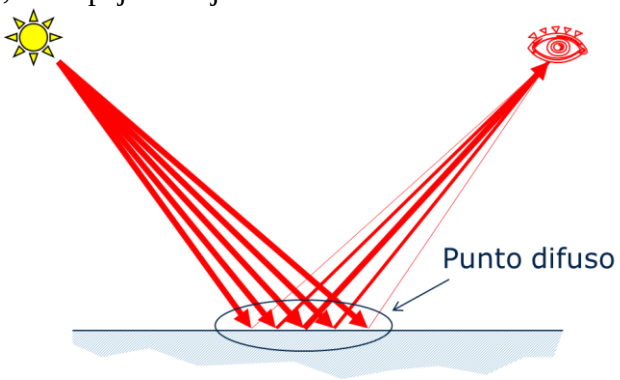
El ángulo de incidencia de la luz no puede ser > 90º desde el exterior del objeto; por lo tanto, la zona iluminada está dentro de un cono de rayos que parten desde la fuente de luz e interceptan al objeto. En los límites, el cono roza tangencialmente a la superficie, con incidencia de 90º. (Las esferas de arriba deberían verse negras en la parte inferior si la única fuente de luz estuviera arriba de ellas)

El color del objeto se define por medio de una reflectancia difusa $K_d(\lambda) \in [0,1]$, que define la fracción de luz reflejada en cada longitud de onda. Para CG se utiliza una terna de valores RGB: (K_{dR}, K_{dG}, K_{dB}). Obviamente, si la luz incidente no es blanca (1,1,1), habrá que multiplicar cada componente de color de la luz incidente por el correspondiente coeficiente de reflectancia difusa del material.

Reflexión especular:

La reflexión especular ideal es la de un espejo o un objeto muy bien pulido, de modo que el rayo de luz que incide sobre un punto desde una dirección **l** es reflejado en una única dirección **r**, con **l** y **r** formando el mismo ángulo con la normal **n** a la superficie y los tres vectores en el mismo plano.

Cuando un espejo recibe luz desde una fuente puntual, el único punto visible de la superficie es aquel para el cual se dan las condiciones de reflexión. Para pensar ese caso es necesario suponer fuentes puntuales de luz y nada de luz ambiente; por ejemplo, un espejo reflejando estrellas en la noche. Si el espejo estuviese mal pulido, la imagen de cada estrella sería borrosa. Habría un centro más luminoso, donde coinciden los ángulos de incidencia y visión, y una periferia cada vez menos iluminada a medida que aumenta el desvío angular entre **v** y **r**.

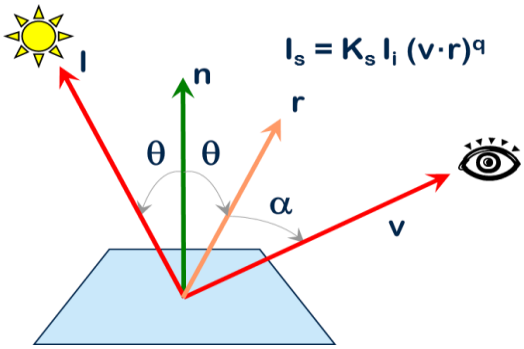


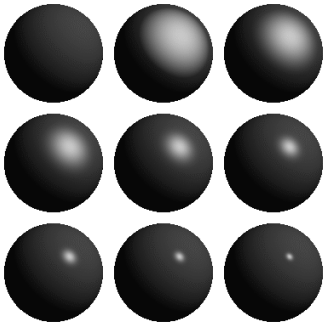
El grado de perfección del espejo se relaciona con la velocidad de decaimiento de la intensidad, en un espejo ideal decae inmediatamente; cuanto menos pulido esté el espejo, más lentamente caerá la intensidad reflejada con el ángulo, dando un mayor tamaño del punto difuso en la superficie.

En su tesis de 1973, Bui Tuong Phong desarrolló un modelo general de iluminación con un modelo heurístico de la reflexión especular no-ideal, basado en un decaimiento cosenoidal a partir del rayo reflejado y cuya velocidad se gradúa mediante un exponente de brillo o *shinniness*.

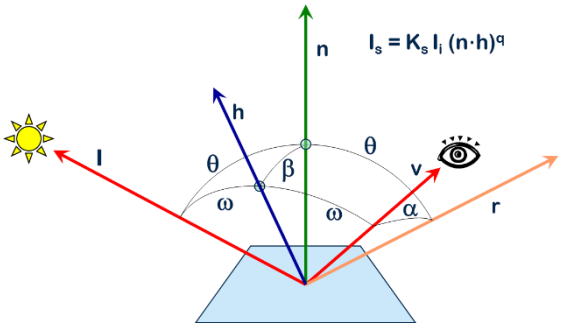
En este modelo, la intensidad de luz no decae con el ángulo de incidencia, aunque la dirección de la luz importa pues define el rayo reflejado ideal. Fijados el objeto y la luz, la posición del punto brillante en el objeto depende de la ubicación del observador.

En la imagen de la izquierda se esquematiza el modelo y a la derecha se ve el resultado de utilizar exponentes de *shinniness* crecientes, de izquierda a derecha y de arriba hacia abajo. El punto brillante es tanto más pequeño cuanto mayor es el exponente q del coseno del ángulo de desvío ($\cos(\alpha) = \mathbf{v}\cdot\mathbf{r}$).





En 1977 Blinn introduce un truco que define el modelo que se utiliza actualmente. La suma normalizada de **l** y **v** es un vector unitario **h**, a medio camino (*halfway*) entre ellos. El ángulo que se utiliza en lugar de α es β, cuyo coseno es el producto escalar **h**·**n**. El ángulo β es prácticamente la mitad de α (analizar los triángulos esféricos, la mitad no es exacta). Usando **h**·**n** (Blinn) o **r**·**v** (Phong) se puede obtener prácticamente el mismo resultado visual, pero después de ajustar adecuadamente el exponente q. A diferencia del vector **r**, ni **l** ni **v** dependen de **n**, entonces cuando el observador y la luz se consideran en el infinito (como casi siempre) el vector **h** es constante para cada luz, en toda la escena.



Del mismo modo que para la reflexión difusa, se considera que el material refleja parcialmente cada componente de la luz incidente. La terna de reflectancias, definidas como K_s, definen la fracción de luz reflejada. La inmensa mayoría de los objetos tienen brillo blanco, aún los negros brillantes (como puede verse en las esferas de arriba), de modo que K_s suele fijarse en {1,1,1} y solo se define el exponente q de *shinniness* o brillo de cada material.

Reflexión ambiente:

Sumando los términos difuso y especular, las zonas que no reciben luz directa quedarían negras. Para solucionar eso se agrega una componente ambiente uniforme, que simula la iluminación de origen incierto, provista por el ambiente. La idea es definir una proporción reducida del color del objeto para sumar a las otras componentes, de modo que, donde no haya luz incidente, esa será la única iluminación.

Una bola, reflejando sólo la componente ambiente, se vería como un disco de color uniforme que no depende de la ubicación de las luces ni la del observador.

Las luces del ambiente se consideran incidentes en cualquier punto con igual intensidad I_a y la fracción reflejada por el material está determinada por una terna K_a, del mismo modo que las otras. En CG (o al menos en OpenGL) se suele definir una contribución al ambiente por cada luz encendida; pero, además, una intensidad de luz ambiente general I_{ag}.

Emisión del material:

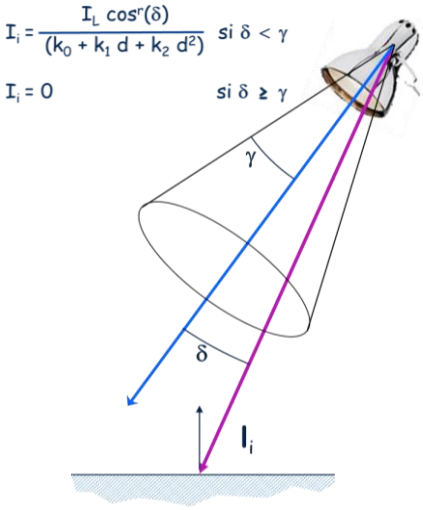
La suma de las tres componentes reflexivas resulta en el modelo compuesto de reflexión de Phong, pero en su implementación suele agregarse un término que permita simular superficies que emiten luz, aun cuando todas las luces estén apagadas. La emisión se modela mediante una terna K_e que define una intensidad de luz emitida en cada punto del objeto y por igual en cualquier dirección. Una esfera emisiva y que no refleja luz se vería como un disco uniforme. Se utiliza para modelar fuego, luciérnagas, luces de neón, halos u otros objetos que tienen luz propia. Estas superficies emisivas no iluminan otros objetos, el modelo sigue siendo local.

Modelado de la luz incidente

La luz direccional simula la luz del sol, que viene desde el infinito, con una dada dirección **l** (vector unitario) e intensidad I_i, con las que incide en cualquier punto. Es el modelo más simple de luz.

Cuando se requiera una luz cercana a la escena y que ilumine en forma diferente a los objetos alrededor de ella se define una luz posicional. La luz de una lámpara se modela como un punto en posición

definida; y es posible simular el decaimiento con la distancia y el comportamiento de un *spot*, en el que la intensidad también decae con el ángulo respecto a un eje. A la derecha se muestra el modelo. El exponente del coseno modela la velocidad de caída de la intensidad a medida que aumenta el ángulo δ con el eje (*spot direction*). También se incluye un ángulo límite de corte γ (*cutoff*) a partir del cual la intensidad se anula abruptamente.



El decaimiento con la distancia se define mediante un polinomio cuadrático, pero la realidad física es que una fuente puntual ilumina con una intensidad inversamente proporcional al cuadrado de la distancia. Cuando la fuente es una línea (como un tubo fluorescente) el decaimiento resulta inversamente proporcional a la distancia (integrando en la línea una sucesión de fuentes puntuales se demuestra que el decaimiento resulta lineal). Normalmente se utiliza el coeficiente lineal (k_1) o el cuadrático (k_2) pero no ambos, uno con valor y el otro nulo. El término independiente k_0 tiene doble funcionalidad: si hay otro (k_1 o k_2) no nulo, un pequeño k_0 impide la división por cero, cuando la distancia d se anula; por otro lado, cuando ambos son nulos, k_0 se define unitario (default) y la intensidad de la luz no decae con la distancia.

La cantidad de luz I_i que recibe un punto del objeto es espectralmente variable, de acuerdo al color I_L de la luz, pero normalmente se modela como una terna de componentes RGB y con intensidad unitaria, es decir una luz blanca. De todos modos, cada luz modelada es responsable de definir cuanto aporta a cada componente de reflexión, es decir que por cada luz encendida habrá tres ternas RGB, una para ambiente I_a, una para difusa I_d y otra I_e para especular.

Modelo global de Phong + emisión:

En OpenGL el modelo completo queda definido mediante la ecuación que suma todas las componentes:

$$I = K_a I_{ag} + \Sigma_j [K_a I_{aj} + K_d I_{dj} \, n \cdot l_j + K_s I_{sj} (n \cdot h_j)^q] + K_e$$

El primer término es la componente ambiente general, luego se suman, para cada luz (j) encendida, una componente ambiente, una difusa y una especular y finalmente se agrega un término de emisión. Un material se define mediante un juego de constantes K y el exponente de brillo q.

La ecuación representa en realidad tres ecuaciones idénticas en R, G y B. Son las componentes RGB en las K del material y en las I de las luces, las que varían entre ecuaciones.

Dado que I es una suma de factores reales entre 0 y 1, un resultado mayor que 1, en una o más componentes RGB se clampea a 1, es decir que cualquier valor mayor a 1 pasa a ser 1 (entero de 8 bits > 255 → I = 255).

Por la forma en que se envían los datos desde la CPU a la GPU, el cambio de materiales resulta costoso. Pero, en OpenGL, se puede recurrir a un truco: La luz ambiente se define de baja intensidad (gris) y la difusa y especular blancas; se define un material base cualquiera, pero con la componente especular blanca. Todos los objetos de ese material serán igualmente brillantes, pero podrán tener distinto color. A medida que se dibuja, se define un color C que será utilizado como componente ambiente y difusa del material ($K_a = K_d = C$). La técnica se conoce como *color-material* y consiste en definir uno o pocos materiales (plástico de brillo medio, tiza mate, metal brillante) y solo se cambia el color a medida que se dibujan los objetosm agrupados por material; minimizando así los cambios.

Otra aceleración importante se logra definiendo luz y observador en el infinito. Con ese truco $\mathbf{l}_j$, $\mathbf{v}$ y $\mathbf{h}_j$ son constantes y se ahorran muchos cálculos, sin que el efecto visual sea importante.

Una luz posicional puede servir solo cuando es una luz de mano o linterna, en la exploración de una caverna, por ejemplo; o, como ya se mencionó cuando ilumina objetos a su alrededor.

El punto de ubicación del observador es importante para la vista en perspectiva; pero para la iluminación solo importa la dirección visual. Aún para realidad virtual, no es imprescindible que el brillo esté corrido en la imagen generada para cada ojo.

Interpolación del sombreado (*shading*)

Un objeto complejo se representa normalmente mediante una malla: una aproximación poliédrica y facetada de la superficie, mediante primitivas poligonales simples, como triángulos o cuadriláteros. Cuando se rasteriza una primitiva y se definen los fragmentos para cada píxel, a cada fragmento debe asignarse un color antes de presentarlo en la imagen de salida. Debido al costo computacional del cálculo de iluminación en cada fragmento, se puede utilizar algún procedimiento simplificador.

Para calcular la iluminación se utilizan la normal a la superficie y las direcciones hacia el ojo y hacia la luz. Normalmente la primitiva no tiene la forma deseada; un triángulo, por ejemplo, es plano y con una normal constante. Si se calcula la iluminación en algún punto y se asigna el color resultante a todo el polígono, el objeto se verá facetado, pero con variaciones de color entre primitivas, dando una sensación aproximada del sombrado esperado. Este resultado solo es aceptable para poliedros simples o cuando se quiere resaltar la subdivisión en primitivas. También es valioso como alternativa rápida, cuando la imagen es intermedia y no final; por ejemplo, cuando se está moviendo interactivamente un objeto, a una velocidad que impide un renderizado más costoso, una vez que el objeto se detiene se puede mejorar la imagen final contando con mayor tiempo.

Si bien son las caras (primitivas) las que tienen normal, en CG las normales se calculan y definen solamente en los vértices, las variaciones en el interior, supuesto suave, se simulan por interpolación. Los vértices de la malla son puntos angulosos, sin normal; es el programador el que debe calcular y asignar una normal en cada vértice de la superficie suave. Alcanza con un promedio de las normales de las caras en cada vértice, normalmente sin ponderar, pero es mejor (y mucho más caro) ponderarlas con el ángulo interno de la cara en el vértice.

El método de Gouraud es un poco más costoso que el facetado: Se calcula la iluminación en cada vértice y los colores resultantes se interpolan en los fragmentos interiores. Es el suavizado estándar en OpenGL y el resultado visual es mucho mejor que un facetado, sin un costo excesivo, pues la GPU interpola varias cosas y tres números más no influyen demasiado. De todos modos, tiene defectos muy visibles, sobre todo en la componente especular, cuando la discretización es gruesa, es decir cuando las primitivas son grandes en relación a la curvatura del objeto.

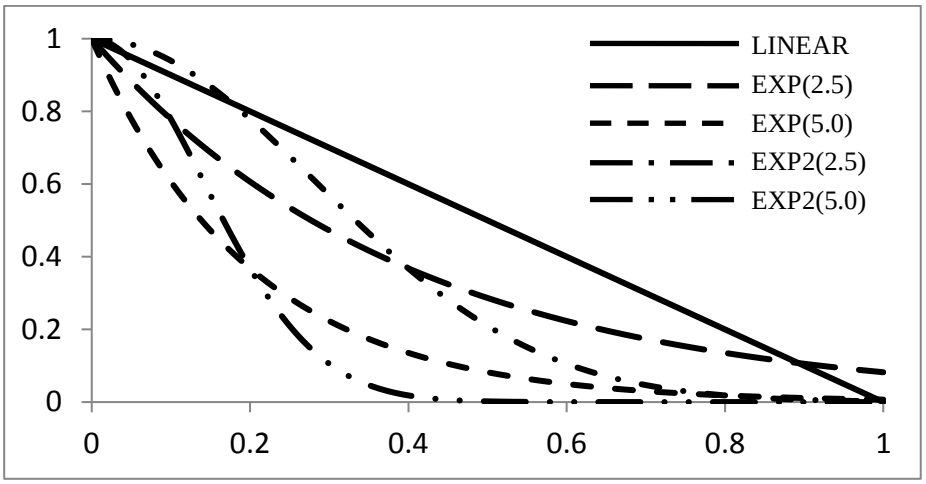
El método con mejores resultados fue desarrollado por Phong, en la misma tesis sobre iluminación. Consiste en interpolar las normales y calcular la iluminación en cada fragmento. Este método no está implementado en OpenGL, pero es muy sencillo de realizar definiendo un programa para el *fragment shader*, una de las etapas programables del pipeline, admitidas en las versiones modernas de OpenGL y las GPU (por eso, las etapas programables se llaman *"shaders"*). Aunque es sencillo de implementar, puede resultar muy costoso en tiempo y entorpecer el trabajo interactivo cuando la malla es muy densa.

En las figuras de arriba se pueden apreciar los efectos de aplicar los tres métodos mencionados sobre una misma malla. Los perímetros visuales siempre se ven facetados, esto es porque el sombreado otorga una sensación de suavidad en el interior, pero la geometría sigue siendo facetada y los límites visuales están gobernados por la geometría. Para arreglar esos defectos, cualquier GPU moderna permite otras instancias programables en el pipeline gráfico (*geometric and tessellation shaders*).

En la práctica, un sombreado más efectivo que el de Gouraud, pero sin tantos cálculos como el de Phong, se puede simular utilizando técnicas basadas en texturas que veremos más adelante.

Niebla

La aplicación de niebla (*fog*) se realiza mezclando el color f de la neblina (normalmente blanco) con el color s (*source*) de cada fragmento, mediante una interpolación lineal (promedio ponderado). El peso del color del fragmento es inversamente proporcional a la distancia del fragmento al ojo y se define mediante una función y sus parámetros.



$f = e^{-(density \cdot z)}$ (GL_EXP)

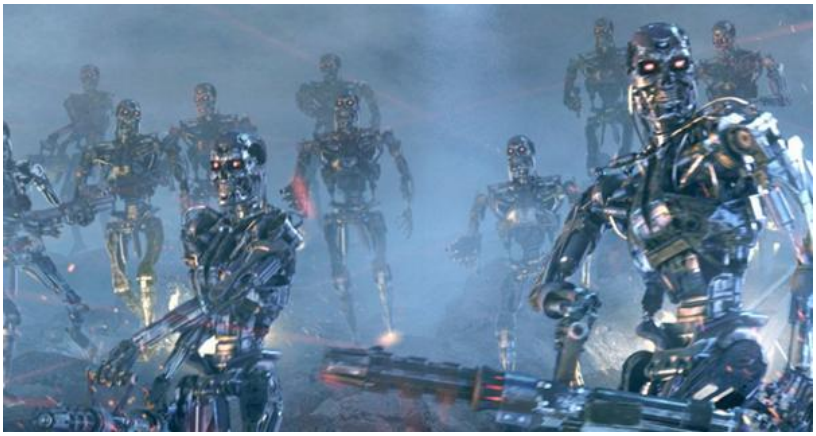
$f = e^{-(density \cdot z)^2}$ (GL_EXP2)

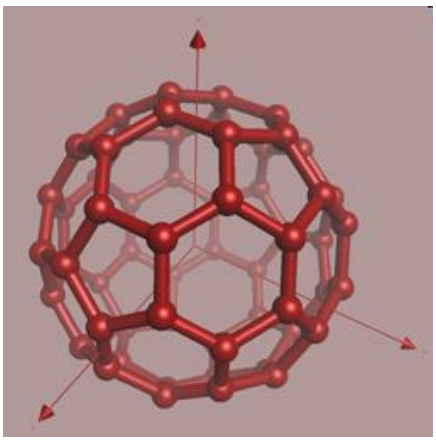
$f = \frac{end - z}{end - start}$ (GL_LINEAR)

$$C = f(z_s)\, C_s + (1 - f(z_s))\, C_f$$

En OpenGL se puede elegir una de tres funciones posibles: exponencial, exponencial cuadrática y lineal. Las funciones, sus parámetros y la representación gráfica de las mismas se muestran en la figura de arriba. En el caso lineal, se considera que la niebla tiene una distancia inicial, hasta la cual el color del fragmento no se ve afectado y una final a partir de la cual la neblina no deja ver los objetos.

OpenGL no usa la distancia $\sqrt{x^2+y^2+z^2}$ sino solo la coordenada z del fragmento como una medida aproximada (normalmente $z \gg x$ o y). Eso hace que los fragmentos periféricos de la imagen tengan menos niebla que la que les correspondería. Puede solucionarse calculando la verdadera distancia e implementando la niebla en un *shader program*, aunque con mayor costo computacional.



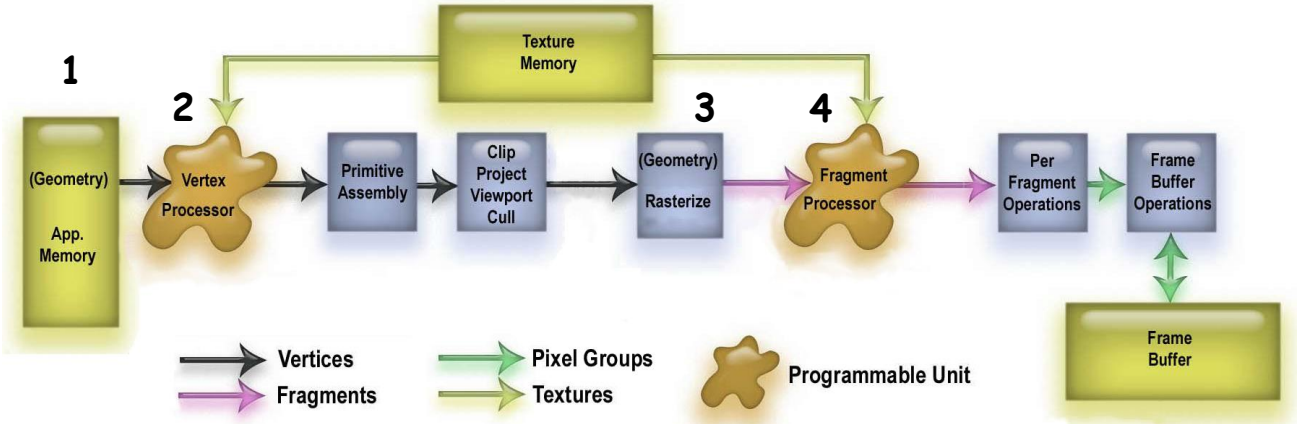


Además del uso obvio para simular niebla, también sirve para simular una escena bajo el agua; el efecto y el tratamiento es exactamente el mismo: en la escena bajo el agua, cuanto mayor es la distancia, más prevalece el color del agua sobre el del objeto.

También sirve para evitar la necesidad de renderizar fondos muy complejos, simulando una visibilidad limitada y para aumentar la sensación de profundidad (*depth-cuing*) usando el color de fondo como color de niebla y la función lineal.

Color e iluminación en el pipeline

Los materiales se definen en el programa (**1**), en la CPU: se define un material y se envían a renderizar los objetos de ese material. Lo mismo sucede con el color, pero el color y la normal se pueden variar entre vértice y vértice de cada primitiva, mientras que (típicamente) un material es para un objeto o un conjunto de numerosas primitivas.



En el procesador de vértices (**2**) hay dos opciones: automático o programado. En ambos casos, el color o material activo es asignado a los vértices enviados a la GPU. En el caso automático, si hay un material, se realizan los cálculos del modelo de Phong en los vértices, en el caso programado se puede calcular la iluminación con el mismo u otro modelo y/o modificar a voluntad el color (y alpha) asignado a cada vértice. En esta etapa también se asignan normales, neblina (fog) y otros datos de vértices que veremos.

En la etapa de rasterización (**3**) cada primitiva se subdivide en fragmentos que corresponden a los píxeles de la imagen final. En el procesador de fragmentos (**4**), si actúa en forma autónoma (no programado) los valores asignados a los vértices se interpolan en los fragmentos interiores, si el modelo de sombreado elegido es Gouraud: glShadeModel(GL_SMOOTH). En cambio, si el modelo es facetado: glShadeModel(GL_FLAT), todos los fragmentos reciben el color del último vértice de la primitiva. Por otra parte, si el programado introduce un programa de fragmentos, el color de iluminación y/o fog de cada fragmente individual lo puede calcular mediante cualquier otro modelo (ej: Phong *shading* o algún método no-fotorealista).

Nestor Calvo
Computación Grafica – FICH
septiembre de 2018