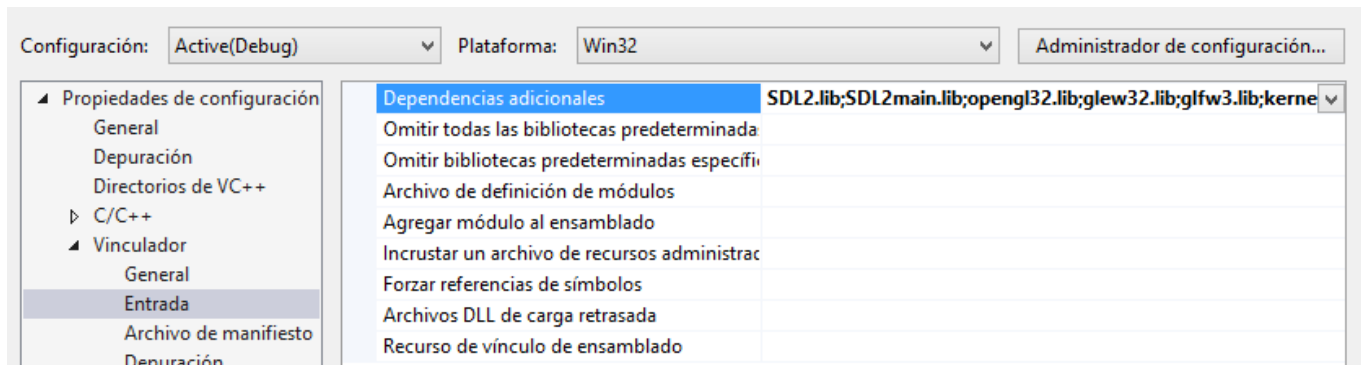


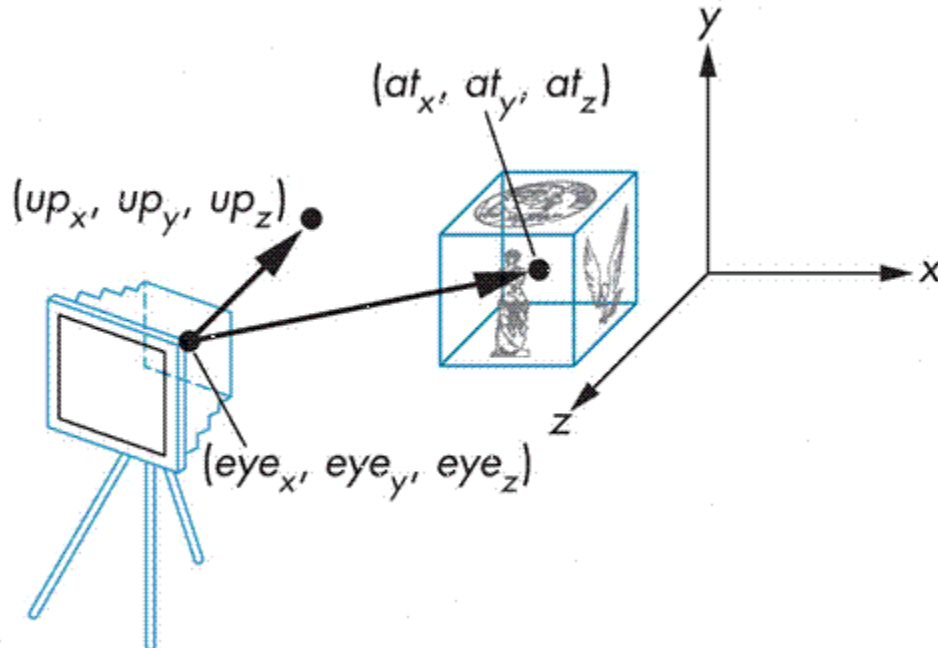
Configuración de Visual Studio

Se debe modificar la configuración de Visual Studio. En Vinculador > Entrada > Dependencias adicionales se debe agregar: **SDL2.lib;SDL2main.lib;**



Cámara Sintética

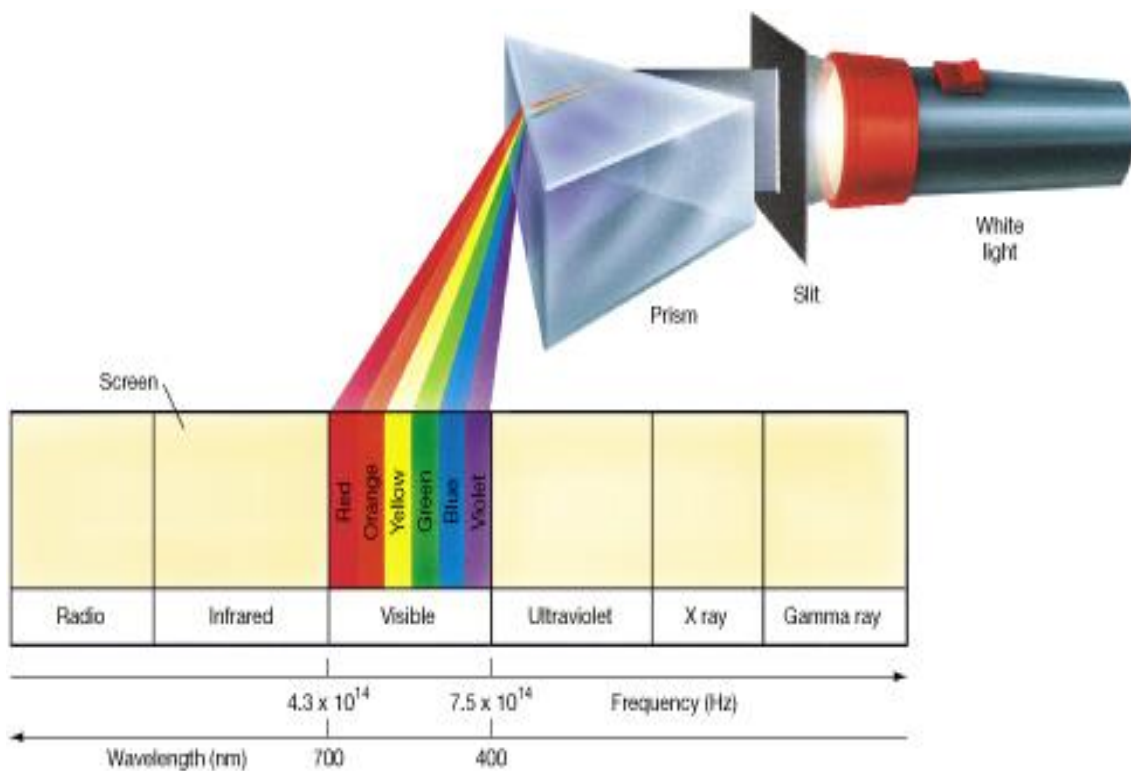
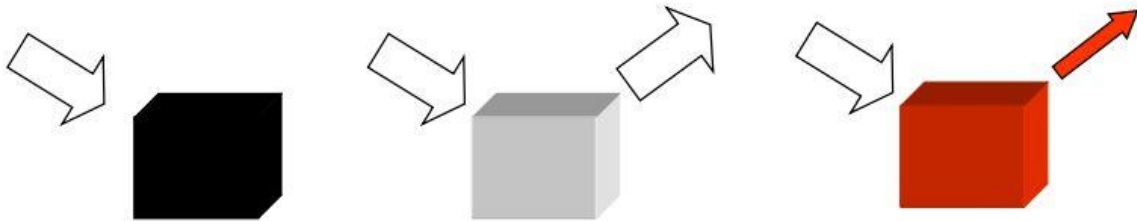
Es tratar de aproximar el comportamiento de la cámara en un ambiente virtual al que se puede realizar con una cámara en la vida real.



Modelos de Iluminación

Se utilizan para calcular la intensidad de luz que percibimos de un punto de la superficie de un objeto, es decir, determina el **color** de dicho punto.

Los colores de los objetos se deben al “material” del cual están constituidos y de las propiedades de la luz que está recibiendo.



Modelos de Iluminación Locales

Son una aproximación empírica que simulan el comportamiento de la luz y su interacción con los objetos.

Intentan ser baratos computacionalmente aunque sacrifican el realismo y calidad de la escena.

Modelos de Iluminación Globales

Son algoritmos que se basan en las leyes físicas de la luz, por lo tanto son mucho más complejos. Dan los mejores resultados visuales y permiten obtener efectos ópticos (reflejos, sombras suaves, causticas, etc.)

Los más utilizados en la industria es Raytracing, Raycasting y Radiosidad.

<https://www.electronicdesign.com/technologies/displays/article/21801219/whats-the-difference-between-ray-tracing-ray-casting-and-ray-charles>

Todos los modelos de iluminación requieren la existencia de, al menos, una fuente de luz.

En Gráficos por Computadora se manejan, al menos, tres tipos de fuentes de luz, las cuales pueden ser direccionales, puntuales o de reflector (se verán en práctica posterior).



Modelo de Iluminación de Lambert y de Phong

Analiza el comportamiento de la luz en un escenario virtual mediante tres componentes:

- a) Ambiental o Intensidad ambiental
- b) Difusa o Intensidad ambiental
- c) Especular o Intensidad especular

Al final las tres componentes determinan el color de una superficie

$$I(x) = I_a + I_d + I_s$$

Ambiental:

$$I_a = I_a \cdot K_a, \text{ donde}$$

I_a es la intensidad de la luz ambiental

K_a coeficiente de reflexión ambiental de la superficie

Difusa:

$$I_d = I_d \cdot K_d \cdot \cos \theta, \text{ donde}$$

I_d , intensidad de la luz difusa

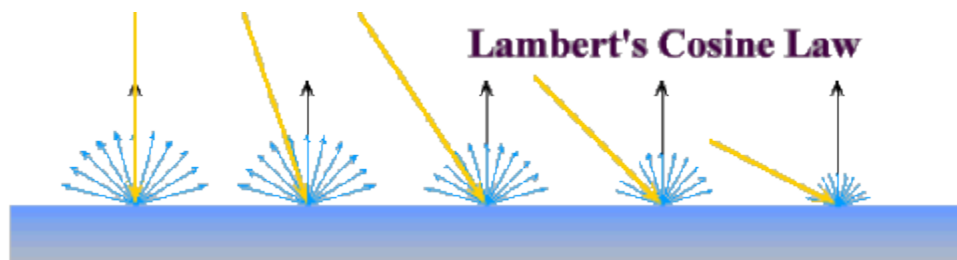
K_d , coeficiente de reflexión difusa de la superficie

θ , ángulo entre la dirección de incidencia del rayo de luz y la normal de la superficie $[0, \pi/2]$

Se recomienda trabajar con vectores normalizados, para que la ecuación se pueda representar como

$$I_d = I_p \cdot K_d \cdot (N \cdot L)$$

Donde N es el vector normal al punto de la superficie y L es el vector de dirección de los rayos luminosos.



Especular:

$I_s = I_s * K_d * (R \cdot V)^n$, donde

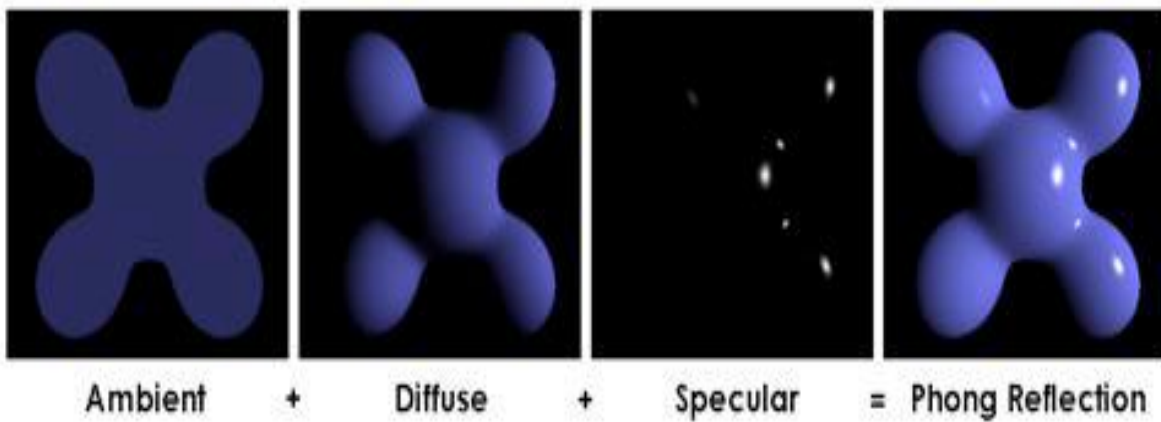
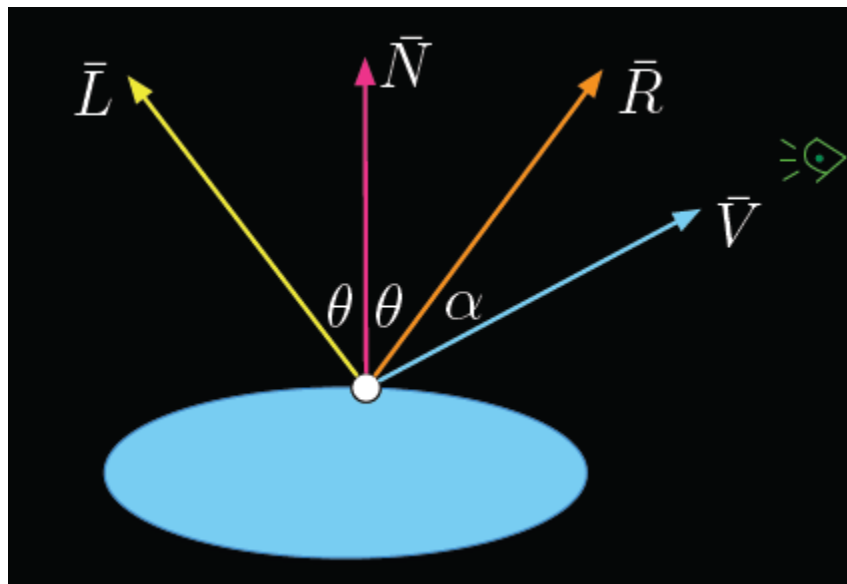
I_s , intensidad de la luz especular

K_d , coeficiente de reflexión especular de la superficie

R , vector de "reflexión", es decir la dirección en cual rebota la luz en la superficie

V , vector de vista, es decir, de la superficie hacia la posición del observador o cámara.

n , coeficiente o factor de concentración de la componente especular



Sombreado

La ecuación de Lambert, idealmente, se debería de aplicar por cada diferencial de área sin embargo esto lo haría muy cara computacionalmente. Por lo cual surgen otras aproximaciones:

- Sombreado Plano. Se aplica la ecuación de Lambert una única vez en algún punto de la superficie, se obtiene el color y éste se aplica a toda la superficie. Es barato pero da un pobre resultado visual en superficies “curvas”. Era un algoritmo implementado hasta la especificación de OpenGL 2 y DirectX 9.
- Sombreado de Gouraud. Requiere el cálculo de las normales en los vértices de la superficie, se aplica la ecuación de Lambert en cada vértice, se obtiene el color y se realiza interpolaciones lineales para determinar el color en el interior de la superficie.
- Sombreado de Phong. Requiere el cálculo de las normales en los vértices de la superficie, se interpolan linealmente los vectores normales a cada pixel (fragmento en el openGL), se aplica la ecuación de Lambert a cada fragmento para obtener el color que corresponde.