UNIT - IV

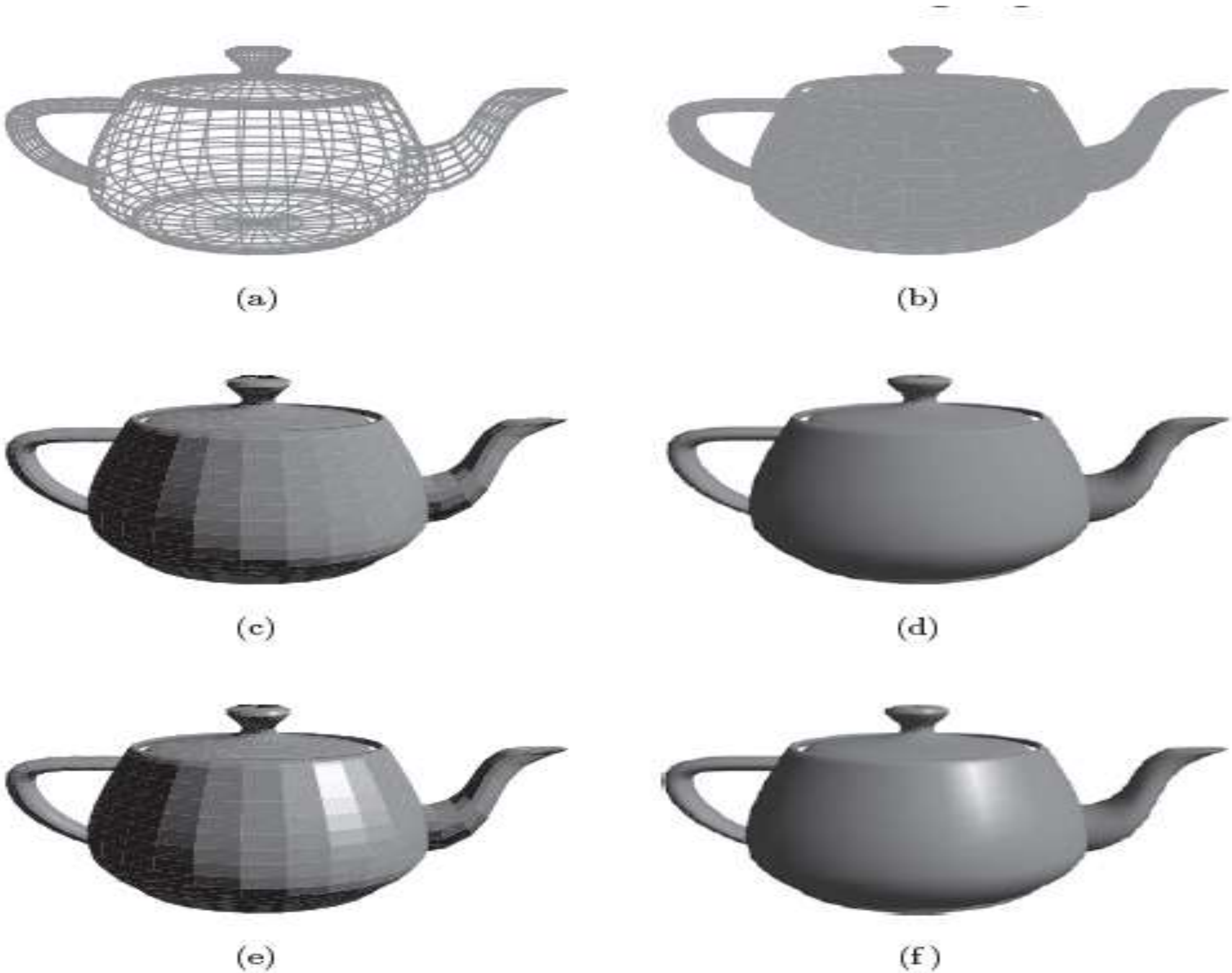
SHADING AND COLORS ISSUSES

INTRODUCTION

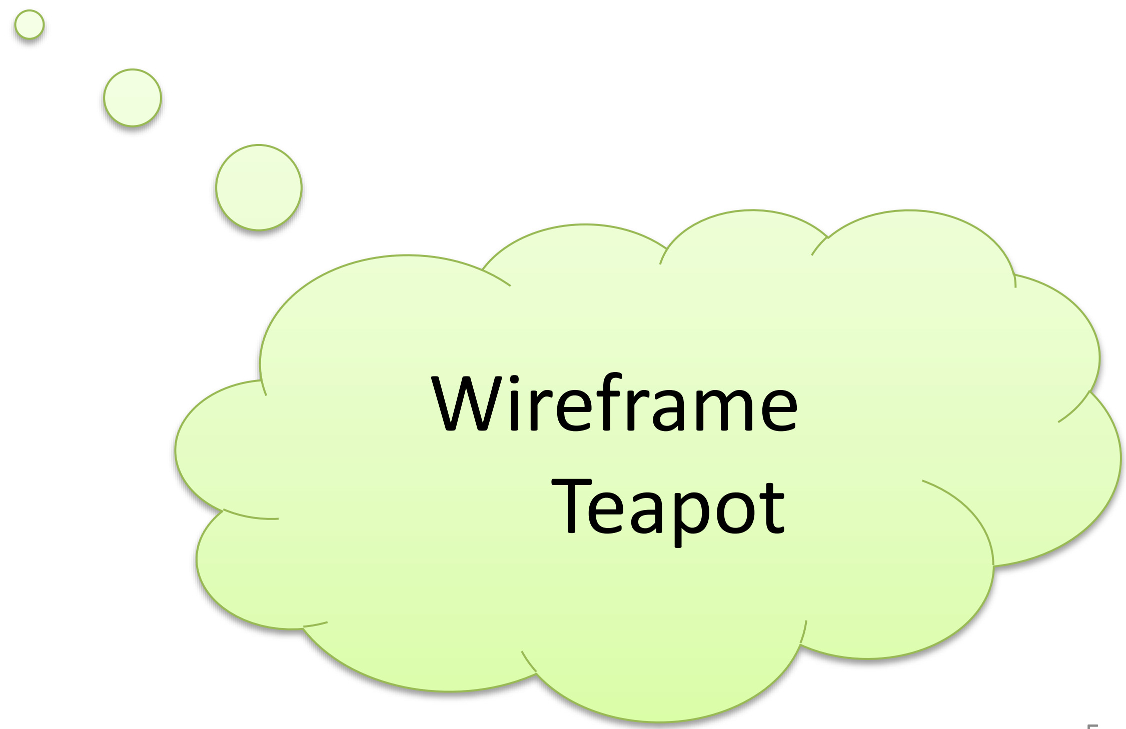
Lighting and shading are important tools for making graphics images appear more realistic and more understandable. Lighting and shading can provide crucial visual cues about the curvature and orientation of surfaces and are important in making three-dimensionality apparent in a graphics image. Indeed, good lighting and shading are probably more important than correct perspective in making a scene understandable. Lighting and illumination models in computer graphics are based on a modular approach where in the artist or programmer specifies the positions and properties of light sources, and, independently, specifies the surface properties of materials. The properties of the lights and the materials interact to create the illumination, color, and shading seen from a given viewpoint.

For an example of the importance of lighting and shading for rendering three-dimensional images, refer to Figure III.1. Figure III.1(b) shows a teapot rendered with a solid color with no shading. This flat, featureless teapot is just a silhouette with no three-dimensionality. Figure III.1(c) shows the same teapot but now rendered with the Phong lighting model. This teapot now looks three-dimensional, but the individual polygons are clearly visible. Figure III.1(d) further improves the teapot by using Gouraud interpolation to create a smooth, rounded appearance. Finally, Figures III.1(e) and (f) show the teapot with specular lighting added; the brightly reflecting spot shown in (e) and (f) is called a *specular highlight.*

"Shading" refers to the practice of letting colors and brightness vary smoothly across a surface. The two most popular kinds of shading are Gouraud interpolation (Gouraud, 1971) and Phong interpolation (Phong, 1975). Either of these shading methods can be used to give a smooth appearance to surfaces; even surfaces modeled as flat facets can appear smooth, as shown in Figure III.1(d) and (f).


(a)
(b)
(c)
(d)
(e)
(f)

(a)



(b)

shows a teapot rendered with a solid color with no shading

Teapot
drawn with solid
color but no
lighting or shading

(c)

shows the same teapot but now rendered with the Phong lighting model. This teapot now looks three-dimensional, but the individual polygons are clearly visible.

Teapot with flat shading with only ambient and diffuse lighting.

(d)

further improves the teapot by using Gouraud interpolation to create a smooth, rounded appearance.

Teapot drawn with Gouraud interpolation with only ambient and diffuse reflection.

(e)

show the teapot with specular lighting added; the brightly reflecting spot shown in figure e is called a *specular highlight.*

Teapot drawn with flat shading with ambient, diffuse, and specular lighting.

(f)

show the teapot with specular lighting added; the brightly reflecting spot shown in figure f is called a *specular highlight.*

Teapot with Gouraud shading with ambient, diffuse, and specular lighting.

Figure III.1. Six teapots with various shading and lighting options.

(a) Wireframe teapot

(b) Teapot Drawn with solid color but no lighting or shading

(c) Teapot with flat shading with only ambient and diffuse lighting.

(d) Teapot drawn with Gouraud interpolation with only ambient and diffuse reflection.

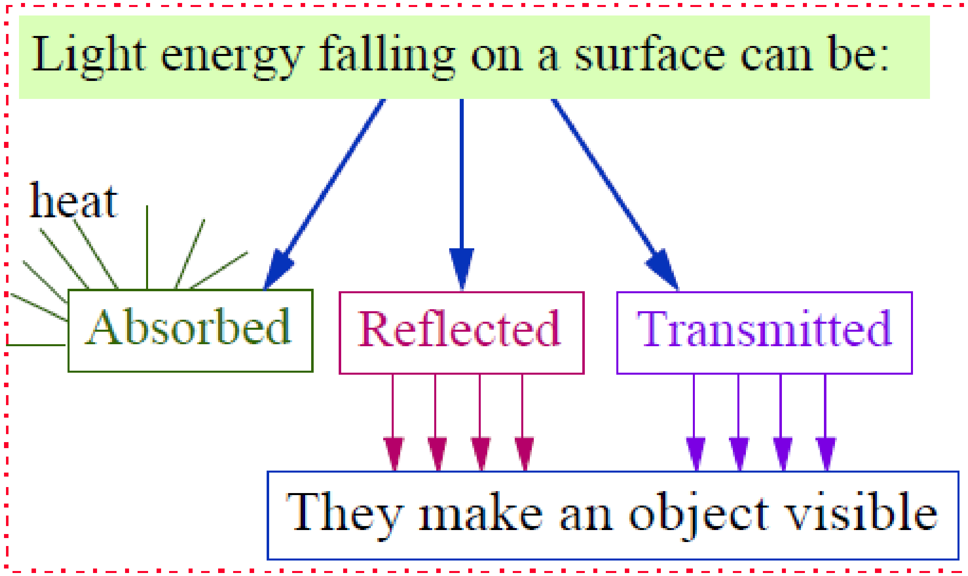
(e) Teapot drawn with flat shading with ambient, diffuse, and specular lighting.

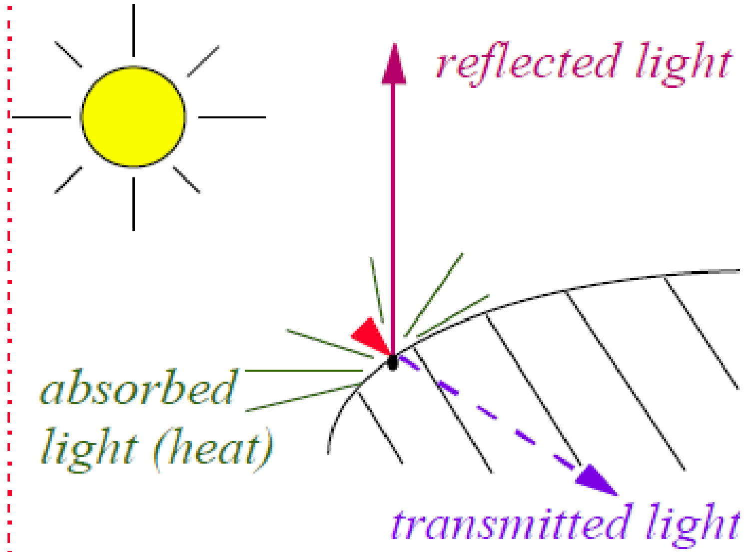
(f) Teapot with Gouraud shading with ambient, diffuse, and specular lighting.

CONTENTS :

Illumination model for diffused & specular reflection, Computing reflection vector, Gouraud and Phog tracing, Band Illusion, Lateral inhibition, Texture mapping & their characteristics, Parametric Texture mapping, 2D Texture mapping and Bump mapping, Handling shadows, Radiosity: Lambert's Law, Basic element, Recapitulation, Modeling transparency, Visualization of data sets, volume rendering, Color issues : Additive, Subtractive primaries, Wavelength spectrum, JCM color.

Illumination Models





The amount of energy absorbed, transmitted or deflected depends on the *wavelength* (w.l.) of the light.

If ... **then, the object**

all the incident light energy is absorbed...is invisible

nearly all the incident light energy is absorbed.................................. appears black

only a small fraction is absorbed ....appears white

the incident light energy is nearly equally reduced for all w.l. ....appears gray

the incident light energy is selectively reduced for all w.l. ....appears colored

Illumination Models

The character of the light reflected or transmitted...

depends on:

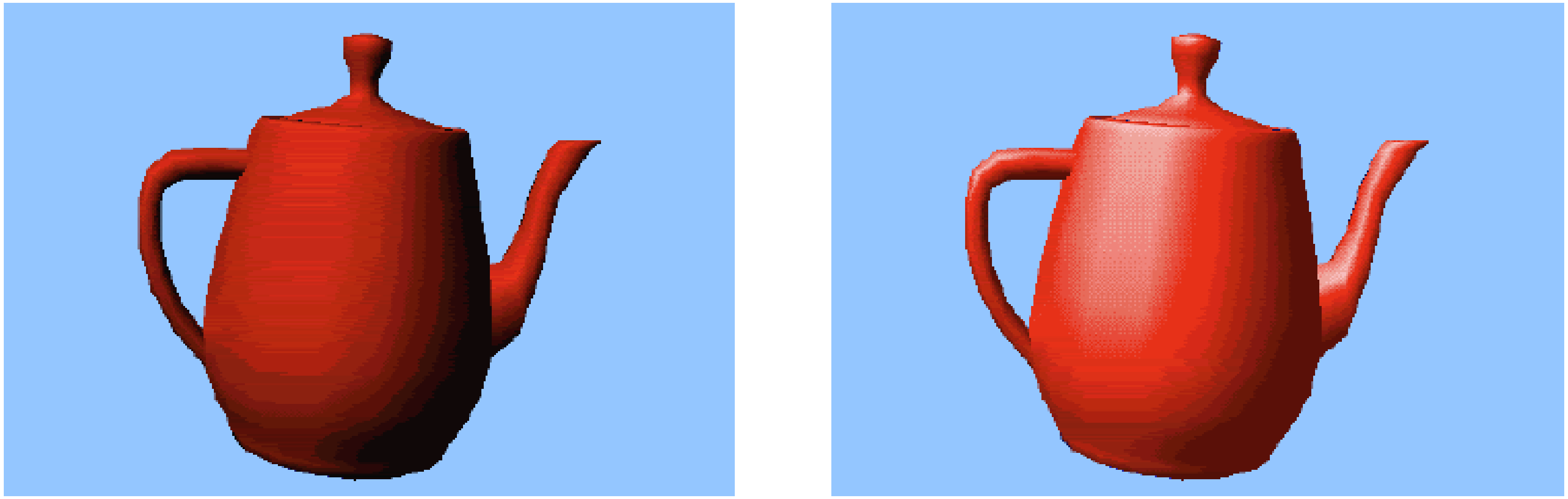
- Composition of the light source
- Direction of the light source
- Geometry of the light source
- The surface orientation
- The surface properties of the object

Illumination Models

Each point may have several sources of illumination:

direct illumination
light arrives straight from the light sources

indirect illumination
light arrives after interacting with the rest of the scene

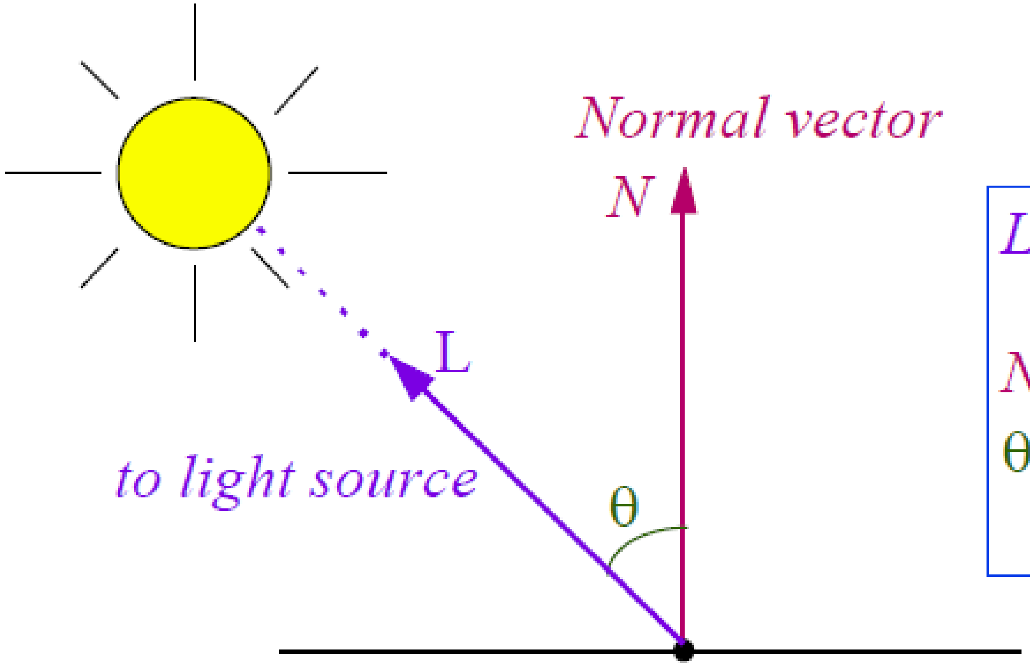


According to how they handle these sources, algorithms can be grouped into:

global illumination algorithms
Both kinds of sources are considered

local illumination algorithms
Only direct lights are taken into account

Some useful definitions:



L - *light source direction*
N - *Normal vector*
θ – *angle of incidence*

- **N** is the **surface normal**
- **L** is the **direction to light source**
- Vectors **N** and **L** are *unit* vectors
- θ is the **angle of incidence**

Model I – Ambient Light

- A object may be visible even it is not directly exposed to a light source.
- That is because, some light is always scattered from the nearby illuminated objects and surrounding is known as ambient model.
- This light is diffused and non-directional in nature and is assumed to be incident with uniform intensity on all object in a scene.
- $I_{amb} = I_a$ (Incident light Intensity) K_a (Ambient reflection coefficient)

Model II – Diffused Light

- Light from the source is incident at different angle at different point on the surface.
- The model which represents such diffused reflection in based on the Lambert's Cosine Law and is given by
- $I_{diff} = I_l \, k_d \cos \Theta$
- Where
- I_{diff} is the reflected intensity
- I_l is the incident light intensity
- k_d is the diffuse reflection coefficient
- Θ is the angle between the incident light direction.

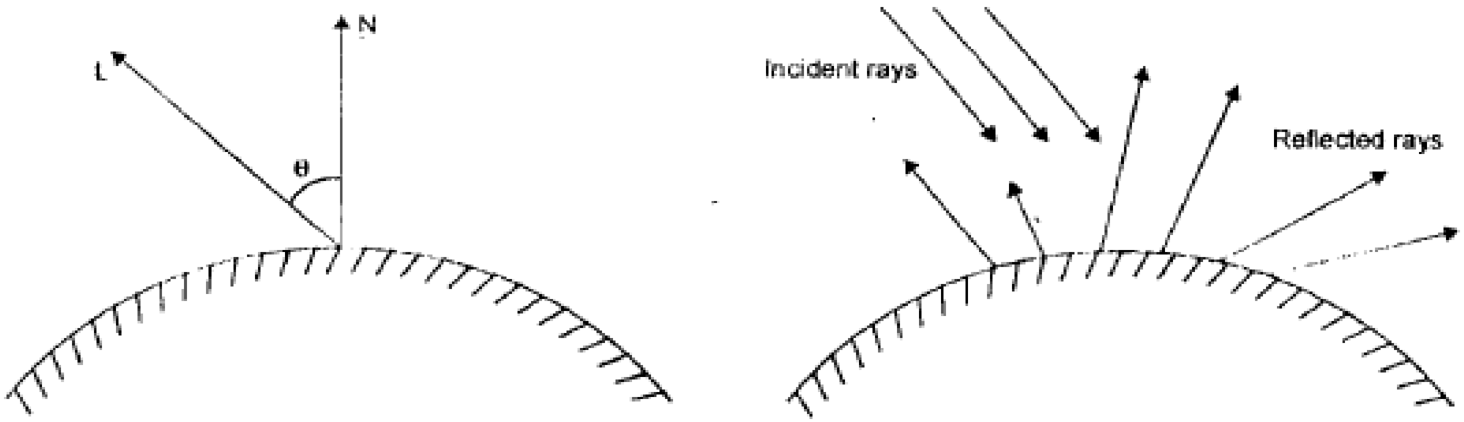


Fig. 8.1

- If the unit vectors representing the surface normal and incident light directions are N and L respectively …. rewritten as

$$»I_{diff} = I_l \, k_d \, (N.L)$$

- Dull mate surface like chalk or cardboard that ideally exhibit diffuse reflection scatter light with equal intensity in all direction.

Illumination model 2: Ambient + diffuse light

Lambert's Cosine Law

incident intensity from a point light source

wavelenght

diffuse reflection function $(0 \le K_d(\lambda) \le 1)$

$$I_d(\lambda) = I_l(\lambda)\, K_d(\lambda) \cos(\theta) \qquad 0 \le \theta \le \frac{\pi}{2}$$

intensity of reflected diffuse light

Therefore, the Lambertian illumination model becomes:

$$I(\lambda) = \underbrace{I_l(\lambda)\, K_d(\lambda) \cos(\theta)}_{\text{diffuse light}} + \underbrace{K_\alpha(\lambda)\, I_\alpha(\lambda)}_{\text{ambient light}}$$

diffuse light ambient light

Illumination model 2: Ambient + diffuse light

In practice, dependence on the wavelength λ is usually omitted:

$$I = \underbrace{I_l\, K_d \cos(\theta)}_{\text{diffuse light}} + \underbrace{K_\alpha\, I_\alpha}_{\text{ambient light}}$$

$$0 \le \theta \le \frac{\pi}{2}$$

$$K_\alpha + K_d < 1$$

Since N and L are unit vectors, it holds that: $\cos(\theta) = N.L$ (↑ *dot* product)

$$I = \underbrace{I_l\, K_d\, (N.L)}_{\text{diffuse light}} + \underbrace{K_\alpha\, I_\alpha}_{\text{ambient light}}$$

$$0 \le \theta \le \frac{\pi}{2}$$

$$K_\alpha + K_d < 1$$

Illumination model 1: Ambient light

Ambient light

- Uniform from all directions
- K_α measures reflectivity of surface for diffuse light (values in the range: 0-1)

$$I = K_\alpha \; I_\alpha$$

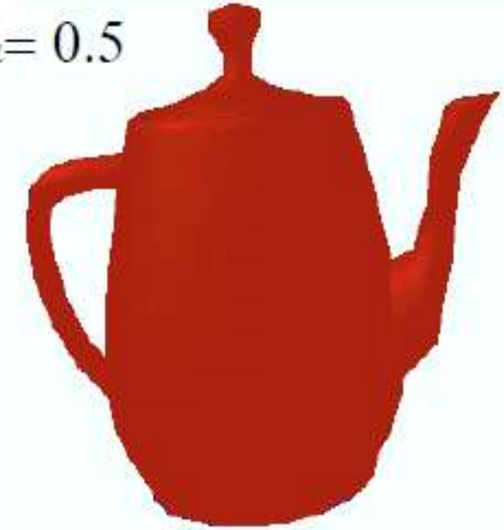
I_α → Intensity of ambient light

K_α → Ambient reflection coefficient

Problem: an object is illuminated uniformly

$K_\alpha = 0.7$

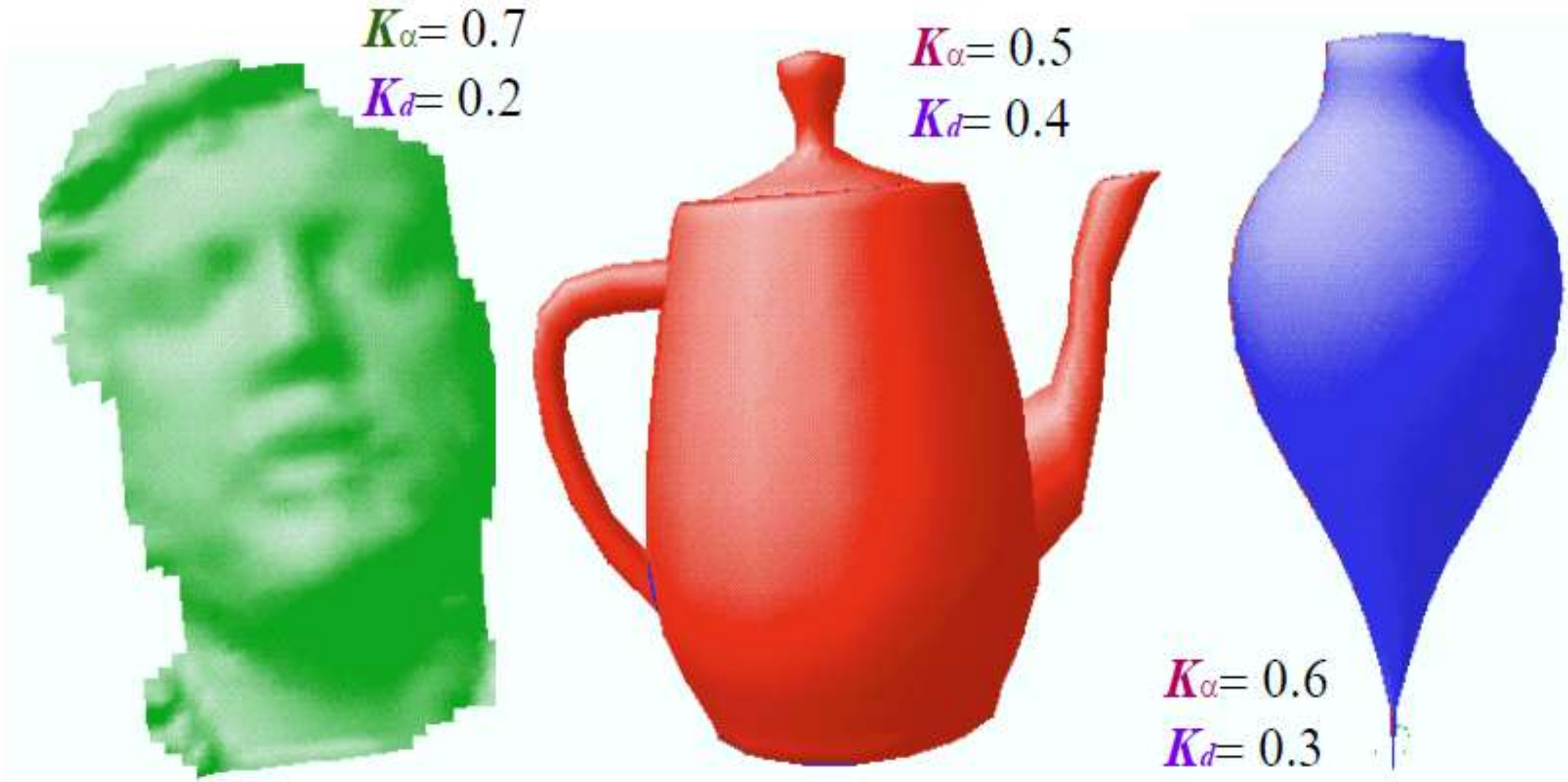
$K_\alpha = 0.5$



$K_\alpha = 0.6$

Illumination model 2: Ambient + diffuse light

Surfaces with a simple Lambertian diffuse reflection appear to have a *dull matte* surface:



Model III – SPECULARLY REFLECTED LIGHT

- Shiny surface : Polished Metal , Oily Skin … nor ideal reflector ….
- Ideal Reflector

Illumination model 3: Ambient + diffuse + specular light

Many surfaces are *shiny*

At certain viewing angles, *shiny* surfaces produces *highlights*

Highlights are due to *specular reflection*

Specular reflection is *directional*

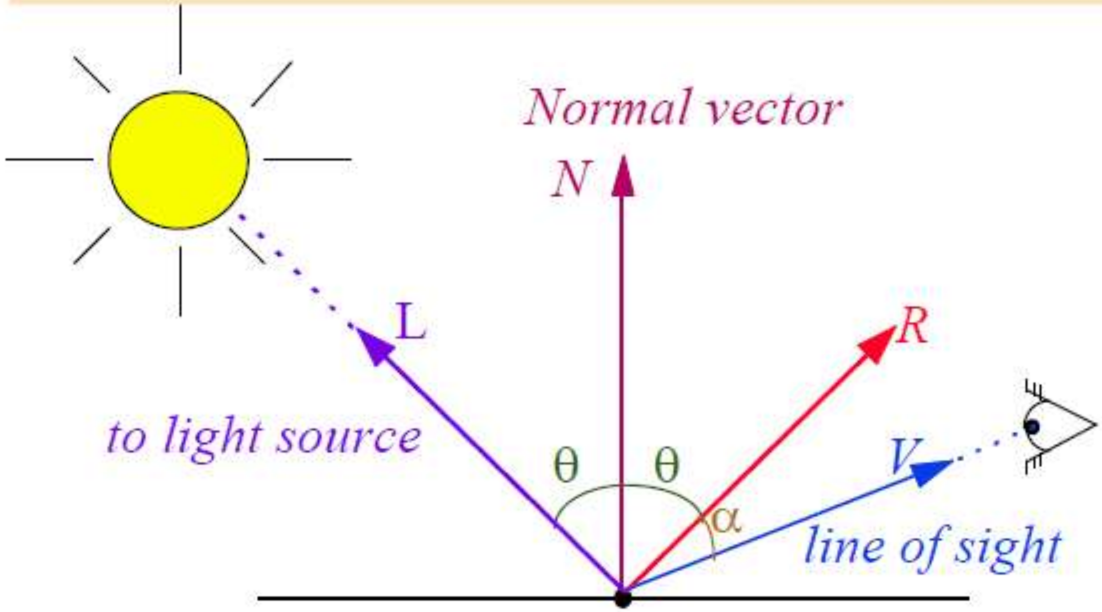
Highlights occur over a narrow range of angles

Highlights color same as the illuminating light rather than the color of the surface

Highlights move with the observer

- For a perfect reflecting surface (a mirror) the angle of reflection is equal to the angle of incidence
- For smooth surfaces, the spatial distribution of specular light is narrow.
- For rough surfaces, it is spread out.

Illumination model 3: Ambient + diffuse + specular light



L - light source direction
N - Normal vector
θ – angle of incidence
V - line of sight
R - direction of ideal specular reflection
α – angle between R and V

Illumination model 3: Ambient + diffuse + specular light

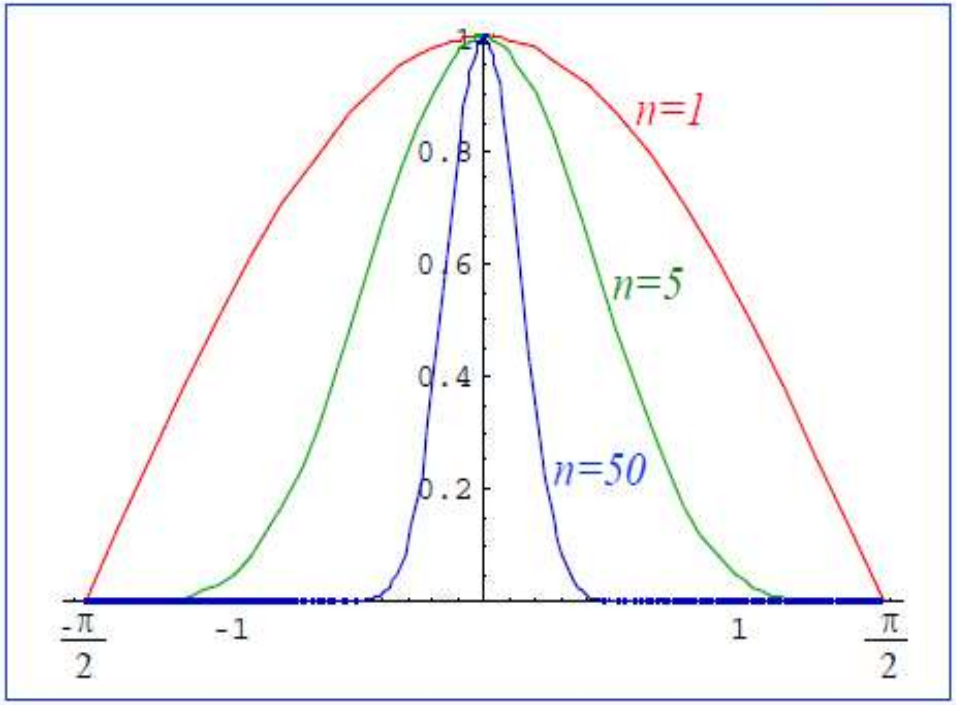
Phong Model

Because of the complex physical characteristics of the specular light, an *empirical model* based on taking the function:

$$f(\alpha) = \cos^n(\alpha)$$

where n depends on surface properties. For:

- a perfect reflector, $n = \infty$
- very poor reflector $n=1$
- in practice use $1 \leq n \leq 200$



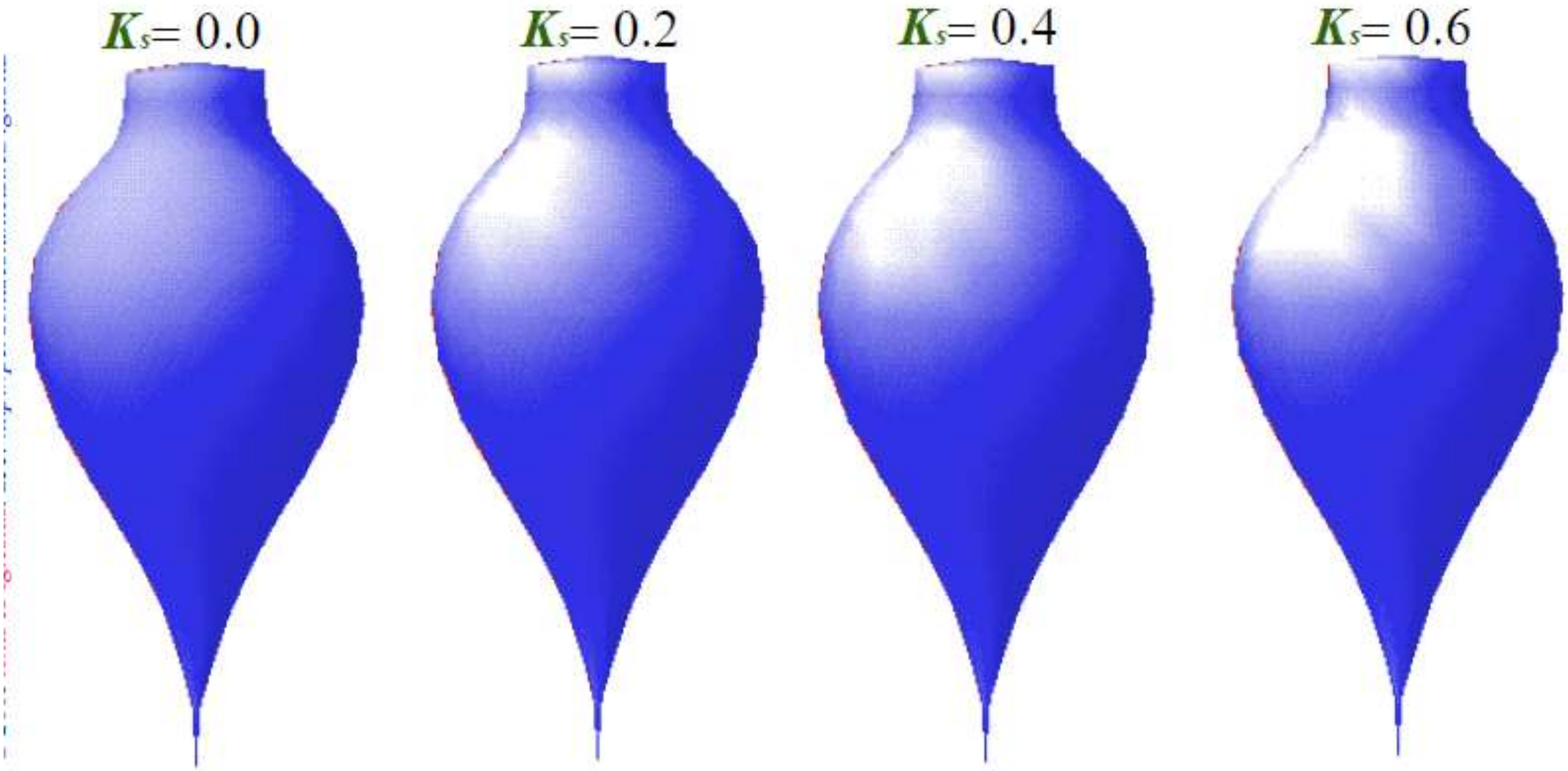
In general, we use:

Larger values of n for metals and other shiny surfaces

Small values of n for nonmetallic surfaces (e.g., paper)

Illumination model 3: Ambient + diffuse + specular light

In practice, dependence on the wavelength λ is usually omitted. In addition, $w(i,\lambda)$ is a very complex function, so it is replaced by an aesthetically or experimentally determined constant k_s



Illumination model 3: Ambient + diffuse + specular light

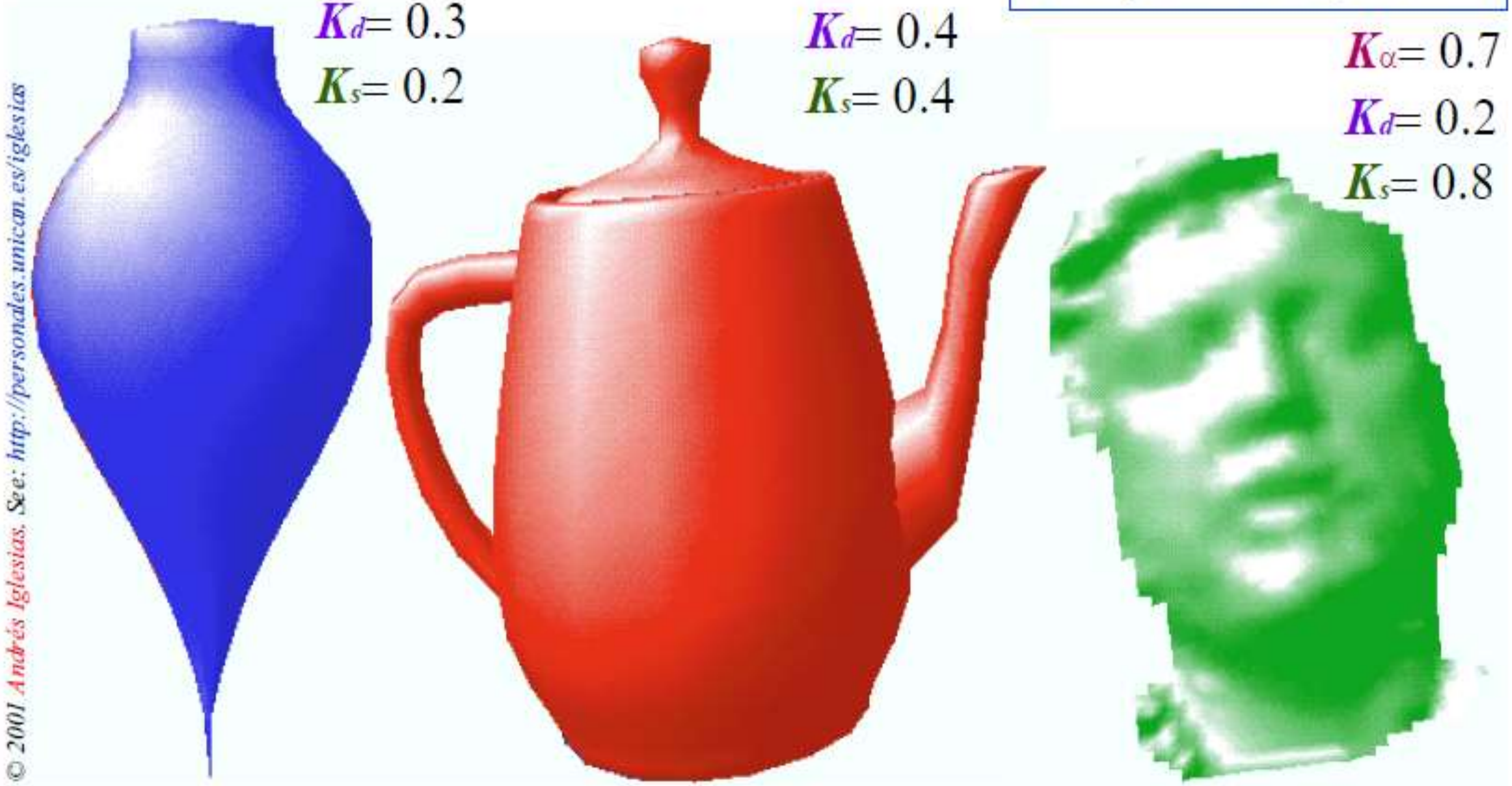
$$I = K_\alpha I_\alpha + I_l K_d \cos(\theta) + I_l K_s \cos^n(\alpha)$$

- K_α *Ambient reflection*
- K_d *Diffuse reflection*
- K_s *Specular reflection*

$K_\alpha = 0.6$, $K_d = 0.3$, $K_s = 0.2$

$K_\alpha = 0.5$, $K_d = 0.4$, $K_s = 0.4$

$K_\alpha = 0.7$, $K_d = 0.2$, $K_s = 0.8$



Illumination model 3: Ambient + diffuse + specular light

Noting that: the model becomes:

$\cos(\theta) = N.L$

$\cos(\alpha) = R.V$

$$I = K_a I_a + I_l K_d (N.L) + I_l K_s (R.V)^n$$

However, two objects *at different distances* but with *the same orientation* to the light source exhibit *the same intensity*.

Computing Reflection Vector

To simulate lighting condition in a real environment we have to assume presence of both distributed and point light source. The intensity of light reflected from a surface will be the resultant of ambient, diffused and specular reflection. Mathematically this can be expressed as,

$$\begin{aligned} \mathbf{I} &= \mathbf{I}_{amb} + \mathbf{I}_{diff} + \mathbf{I}_{spec} \\ &= \mathbf{I}_a\,\mathbf{k}_a + \mathbf{I}_l\,\mathbf{k}_d\,(\mathbf{N}\,.\,\mathbf{L}) + \mathbf{I}_l\,\mathbf{k}_s\,(\mathbf{R}\,.\,\mathbf{V})^{n_s} \\ &= \mathbf{I}_a\,\mathbf{k}_a + \mathbf{I}_l\,[\mathbf{k}_d\,(\mathbf{N}\,.\,\mathbf{L}) + \mathbf{k}_s\,(\mathbf{R}\,.\,\mathbf{V})^{n_s}] \qquad (7) \end{aligned}$$

In the above expression a single point light source is assumed. For multiple (say m numbers) light source the effect of illumination is simply additive and resultant I can be expressed as,

$$I = I_a\,k_a + \sum^{m} I_l[k_d(\mathbf{N.L}) + k_s(\mathbf{R.V})^{n_s}] \qquad (8)$$

Distance Factor

$$I_{diff} = f_{att}\, I_l\, k_d\, (N \cdot L)$$

$$I_{spec} = f_{att}\, I_l\, k_s\, (R \cdot V)^{n_s}$$

$$I = I_a k_a + \sum^{m} f_{att} I_l [k_d (N \cdot L) + k_s (R \cdot V)^{n_s}]$$

Color Factor

For a surface exposed to white light from a point light source the reflections at any point can be modeled by following 3 equations.

$$I_{RED} = I_{a, RED}\, k_{a, RED} + f_{att}\, I_{l, RED}\, [k_{d, RED}\, (N . L) + k_{s, RED}\, (R . V)^{n_s}]$$

$$I_{BLUE} = I_{a, BLUE}\, k_{a, BLUE} + f_{att}\, I_{l, BLUE}\, [k_{d, BLUE}\, (N . L) + k_{s, RED}\, (R . V)^{n_s}]$$

$$I_{GREEN} = I_{a, GREEN}\, k_{a, GREEN} + f_{att}\, I_{l, GREEN}\, [k_{d, GREEN}\, (N . L) + k_{s, GREEN}\, (R . V)^{n_s}]$$

The RGB intensities (I_{RED}, I_{BLUE}, I_{GREEN}) thus calculated can be used to adjust the electron beam intensity of the 3 electron guns in any RGB monitor to generate a pixel in true color.

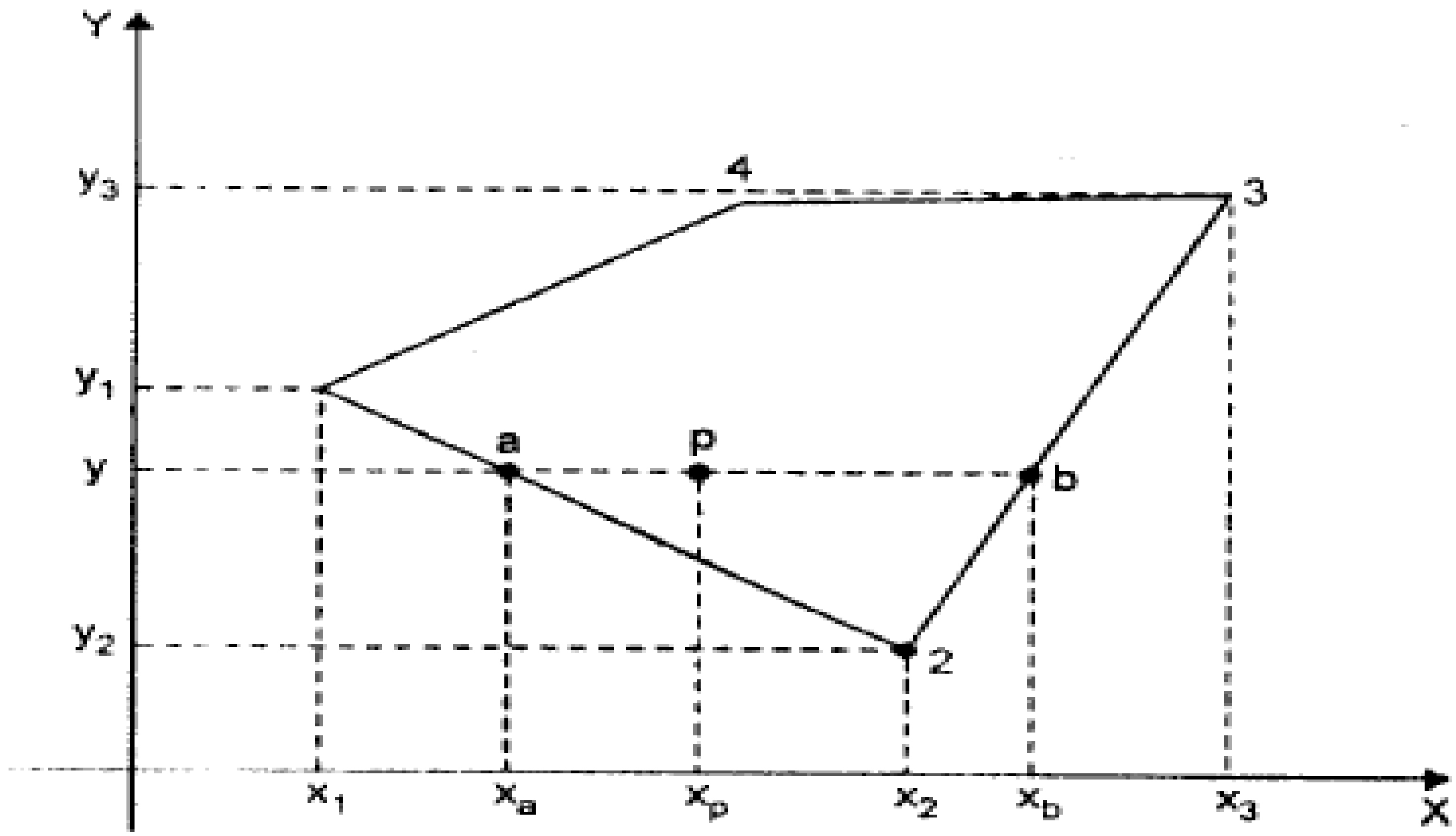
Shading

We have learnt how to calculate intensity of light reflected from a point in a given lighting and surface condition. Now we have to use shading techniques for finding intensity profile of a surface as a whole to simulate it's appearance under given lighting condition. Such techniques calls for implementation of illumination model at each visible point of surfaccs defined by polygon meshes.

Interpolating Shading

Gouraud Shading

Phong Shading



Let $\mathbf{I}_1$, $\mathbf{I}_2$, $\mathbf{I}_3$ be the intensities calculated at vertices 1, 2 and 3 respectively using illumination model at each of these vertices. To determine $\mathbf{I}_p$ at a point p along any arbitrary scan line (y), $\mathbf{I}_a$ is first found at the intersection a of the scan line with the edge 1-2 by linearly interpolating $\mathbf{I}_1$ and $\mathbf{I}_2$.

$$\frac{\mathbf{I}_1 - \mathbf{I}_a}{\mathbf{I}_1 - \mathbf{I}_2} = \frac{y_1 - y}{y_1 - y_2}$$

$$\Rightarrow \qquad \mathbf{I}_a = \mathbf{I}_1 - (\mathbf{I}_1 - \mathbf{I}_2)\frac{y_1 - y}{y_1 - y_2}$$

Similarly $\mathbf{I}_b$ is found at intersection b with edge 3-2 by linearly interpolating $\mathbf{I}_2$ & $\mathbf{I}_3$.

$$\frac{\mathbf{I}_3 - \mathbf{I}_b}{\mathbf{I}_3 - \mathbf{I}_2} = \frac{y_3 - y}{y_3 - y_2}$$

$$\Rightarrow \qquad \mathbf{I}_b = \mathbf{I}_3 - (\mathbf{I}_3 - \mathbf{I}_2)\frac{y_3 - y}{y_3 - y_2}$$

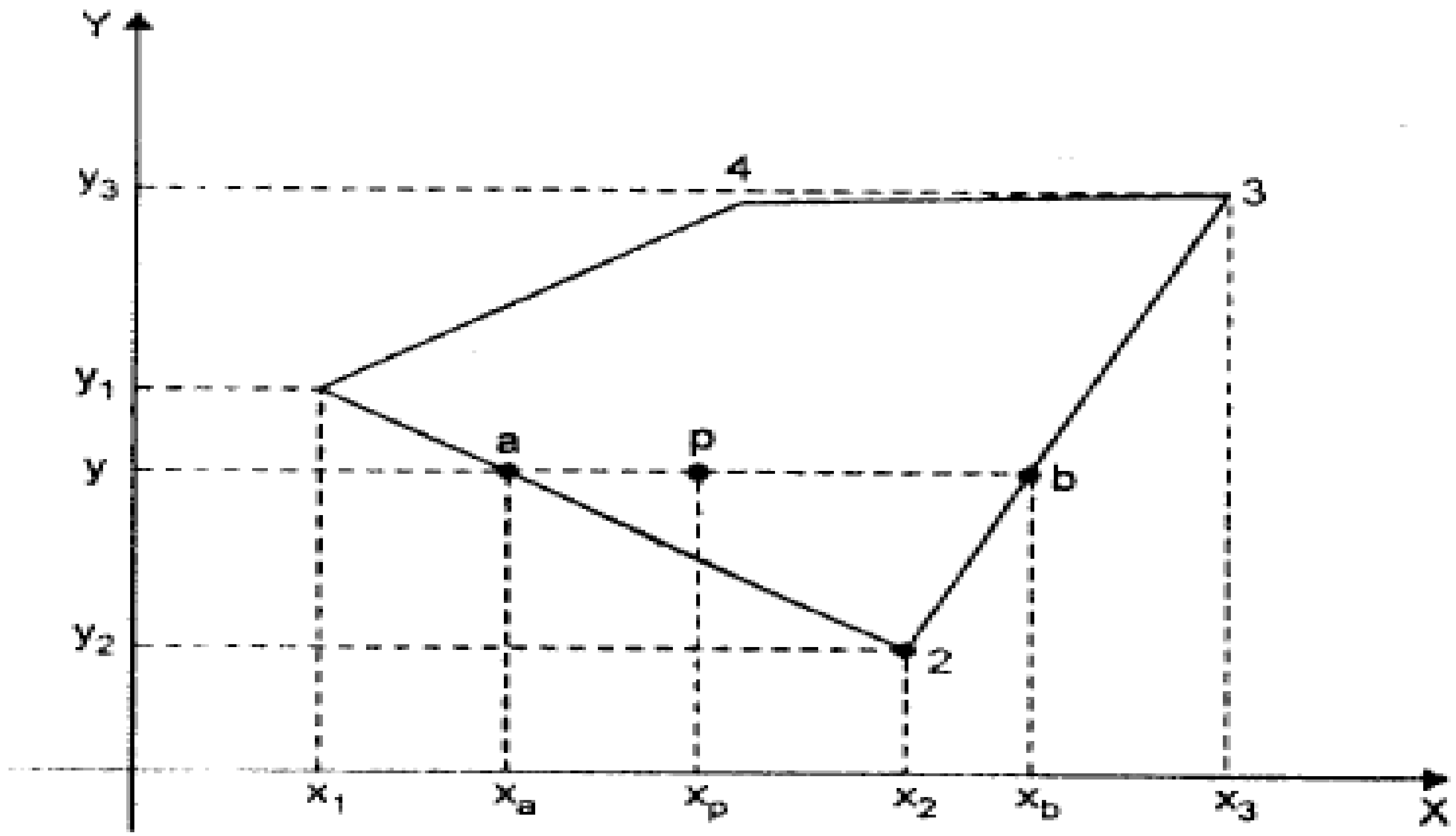
- Finally $\mathbf{I}_p$ is found by linearly interpolating $\mathbf{I}_a$ & $\mathbf{I}_b$.

$$\frac{\mathbf{I}_p - \mathbf{I}_a}{\mathbf{I}_b - \mathbf{I}_a} = \frac{x_p - x_a}{x_b - x_a}$$

$$\Rightarrow \qquad \mathbf{I}_p = \mathbf{I}_a - (\mathbf{I}_a - \mathbf{I}_b)\frac{x_p - x_a}{x_b - x_a}$$

Thus intensity of all points along the scan line y and also along other scan lines within the domain of the area can be determined.

Phong Shading



Step 1: Find $\mathbf{N}_1$, $\mathbf{N}_2$, $\mathbf{N}_3$ the unit normal vectors at vertices 1, 2 and 3 respectively. (Refer Fig 8.5)

Step 2: Find $\mathbf{N}_a$ by interpolating $\mathbf{N}_1$ & $\mathbf{N}_2$ using the following formula

$$\frac{\mathbf{N}_1 - \mathbf{N}_a}{\mathbf{N}_1 - \mathbf{N}_2} = \frac{y_1 - y}{y_1 - y_2}$$

Step 3: Find $\mathbf{N}_b$ by interpolating $\mathbf{N}_3$ & $\mathbf{N}_2$ using the following formula

$$\frac{\mathbf{N}_3 - \mathbf{N}_b}{\mathbf{N}_3 - \mathbf{N}_2} = \frac{y_3 - y}{y_3 - y_2}$$

Step 4: Find $\mathbf{N}_p$ by interpolating $\mathbf{N}_a$ & $\mathbf{N}_b$ using the following formula

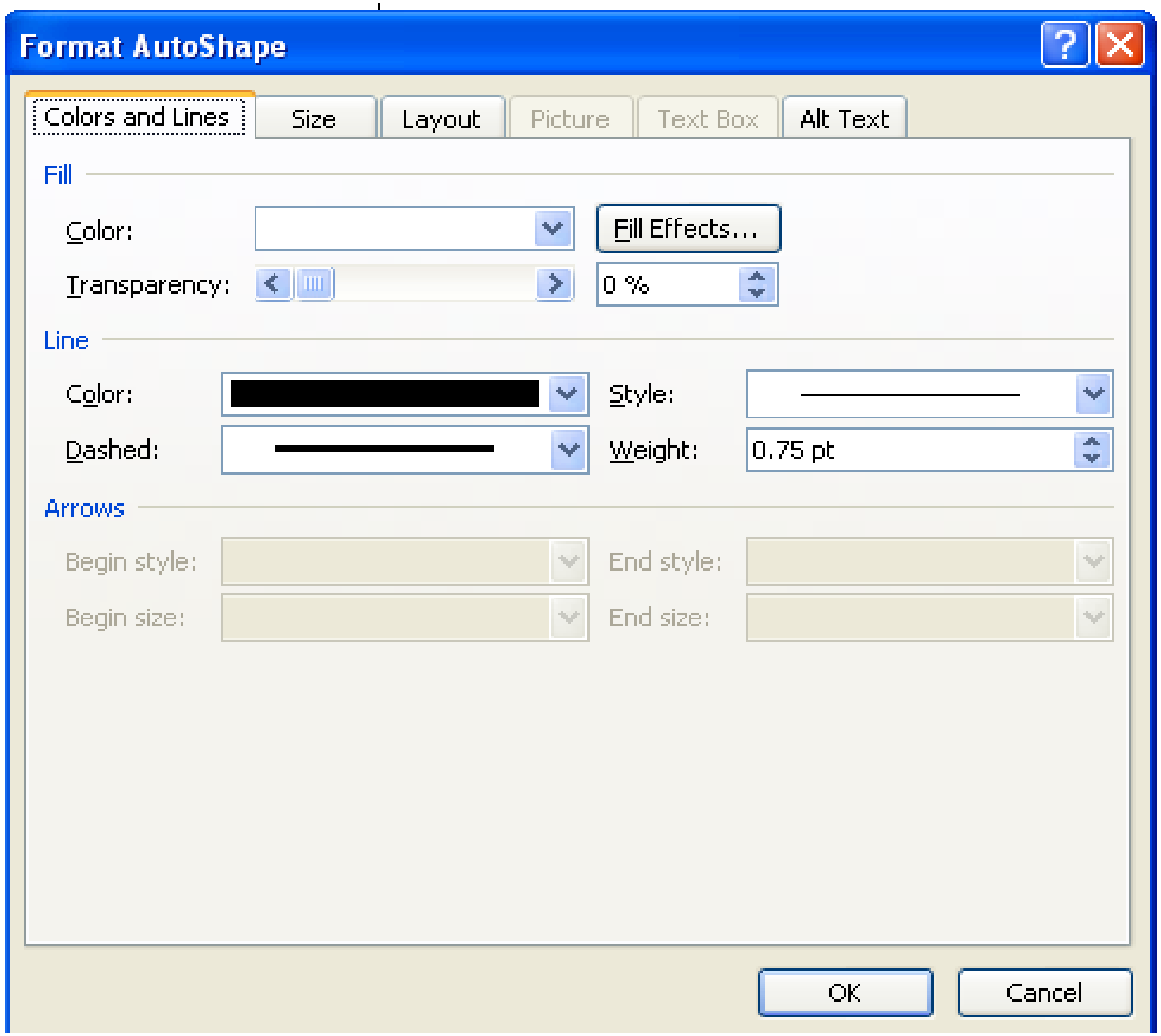
$$\frac{\mathbf{N}_p' - \mathbf{N}_a}{\mathbf{N}_b - \mathbf{N}_a} = \frac{x_p - x_a}{x_b - x_a}$$

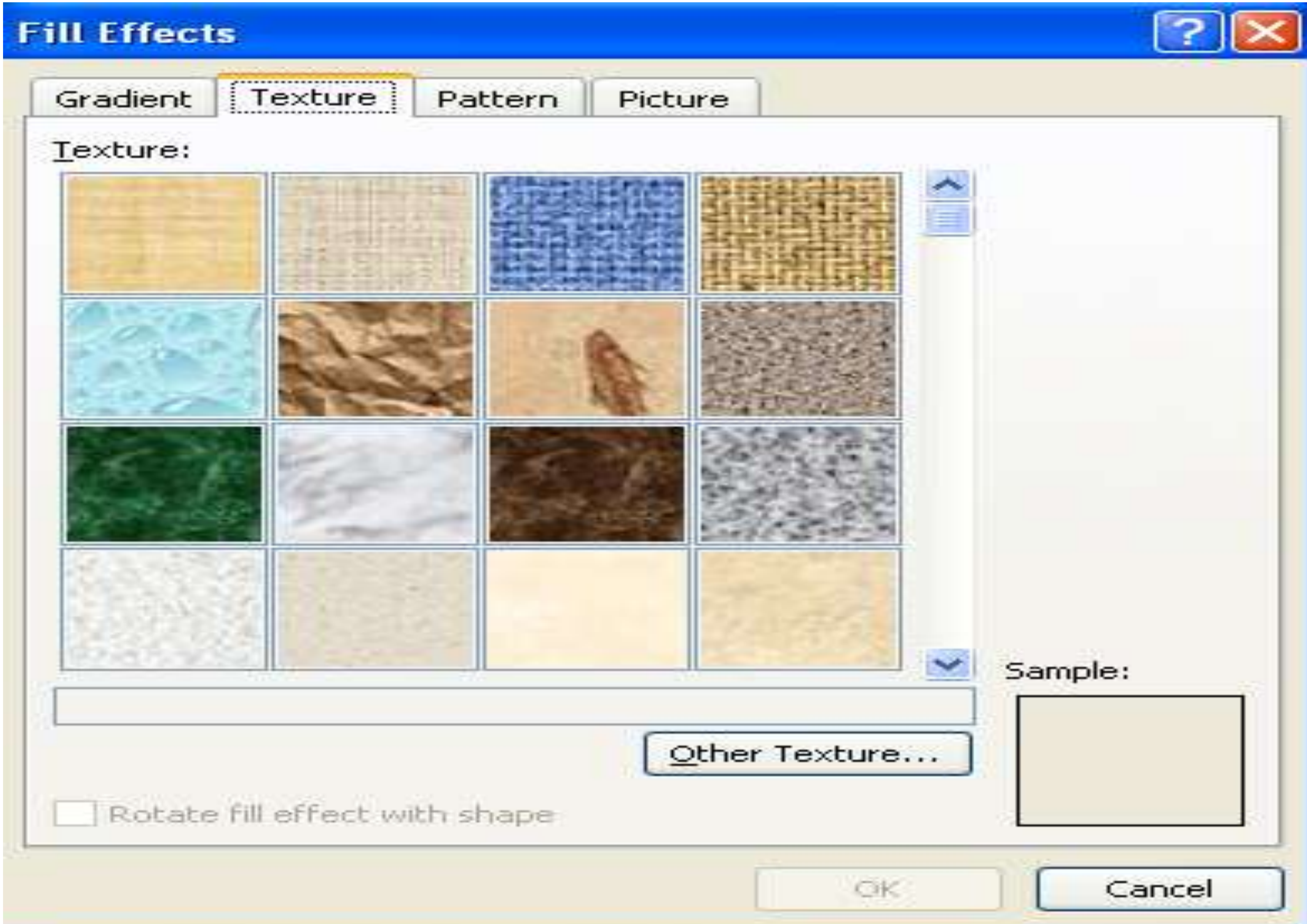
Step 5: Find $\mathbf{I}_p$ at p using $\mathbf{N}_p$ in a standard illumination model

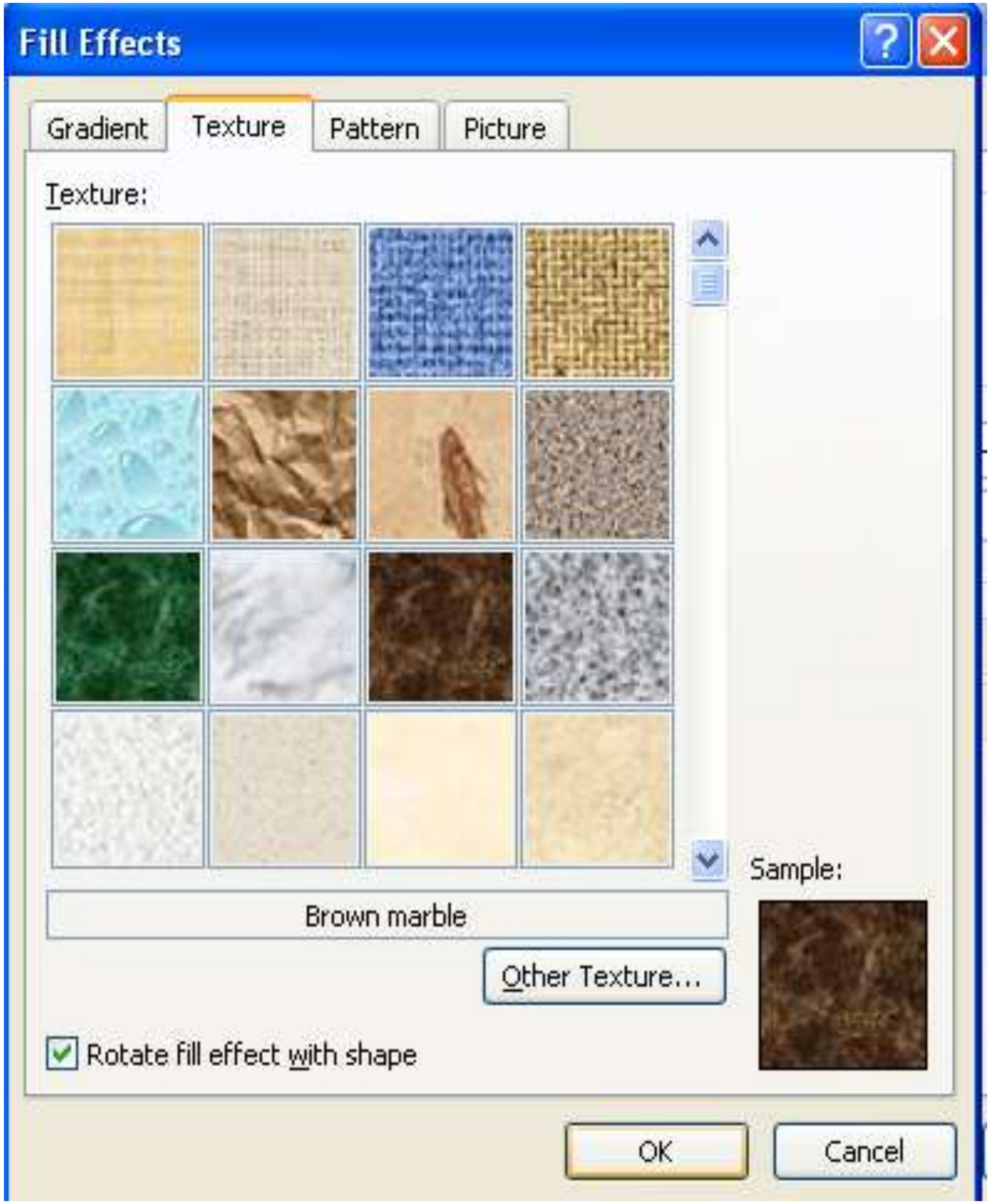
Texture Mapping

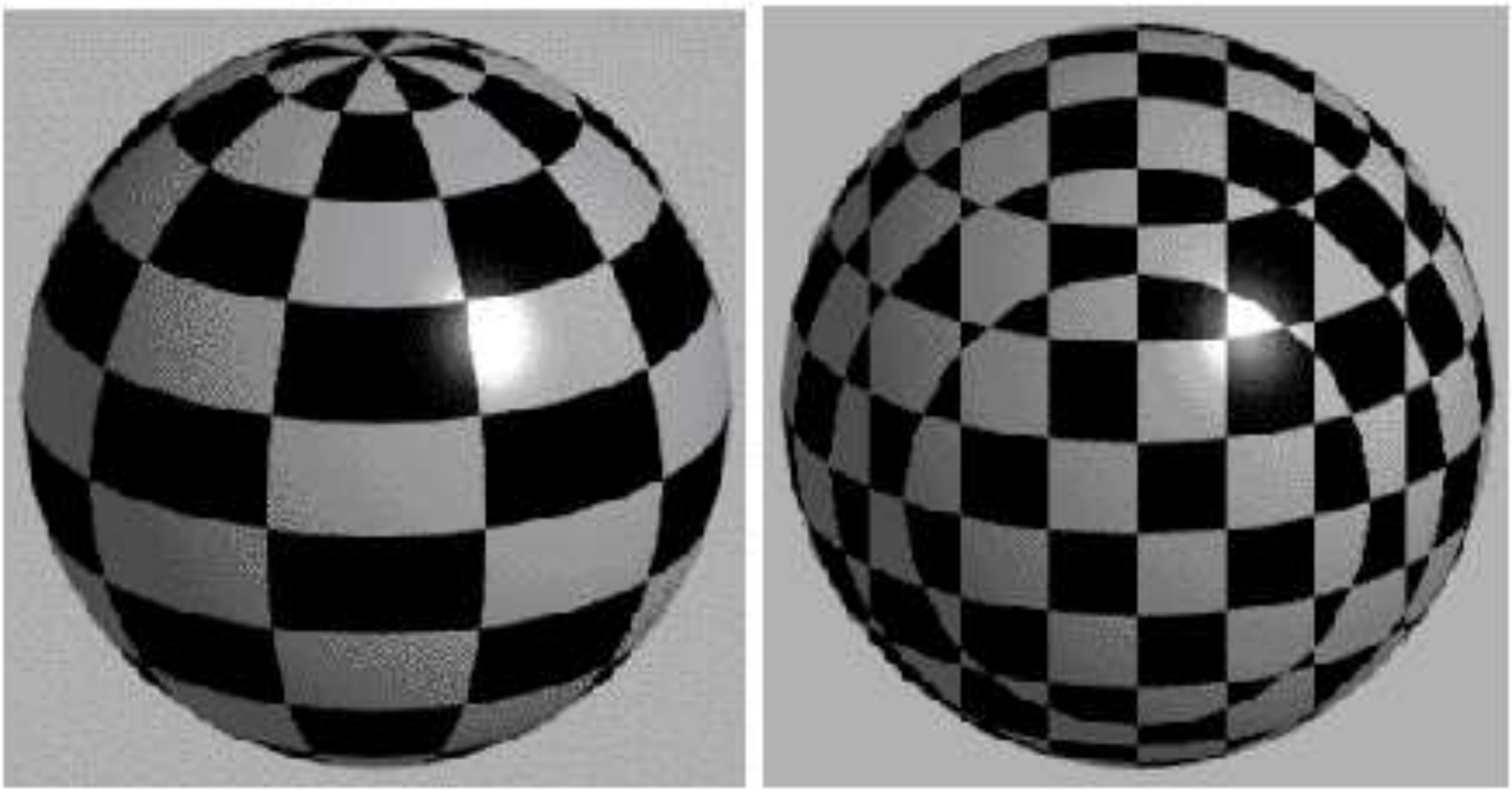
- What is Texture ?
- What is Mapping?
- What is Texture Mapping?

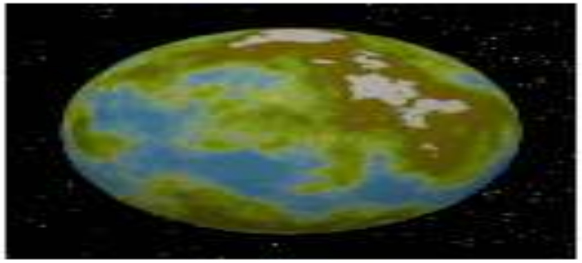
Example of Texture

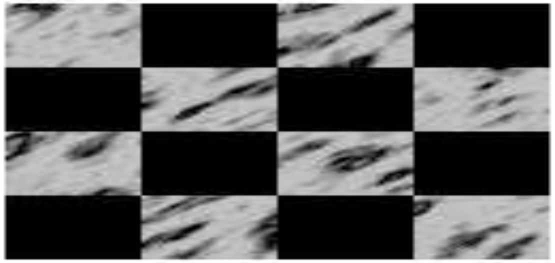
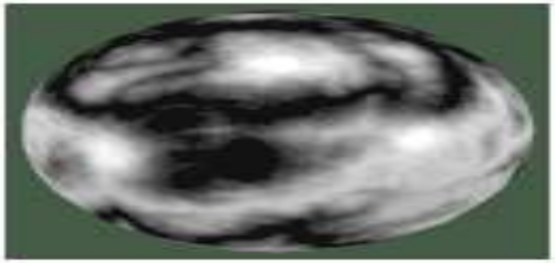



Fill Effects
Gradient
Texture
Pattern
Picture
Texture:
Other Texture...
Rotate fill effect with shape
Sample:
OK
Cancel


Fill Effects
Gradient
Texture
Pattern
Picture
Texture:
Brown marble
Other Texture...
Rotate fill effect with shape
Sample:
OK
Cancel





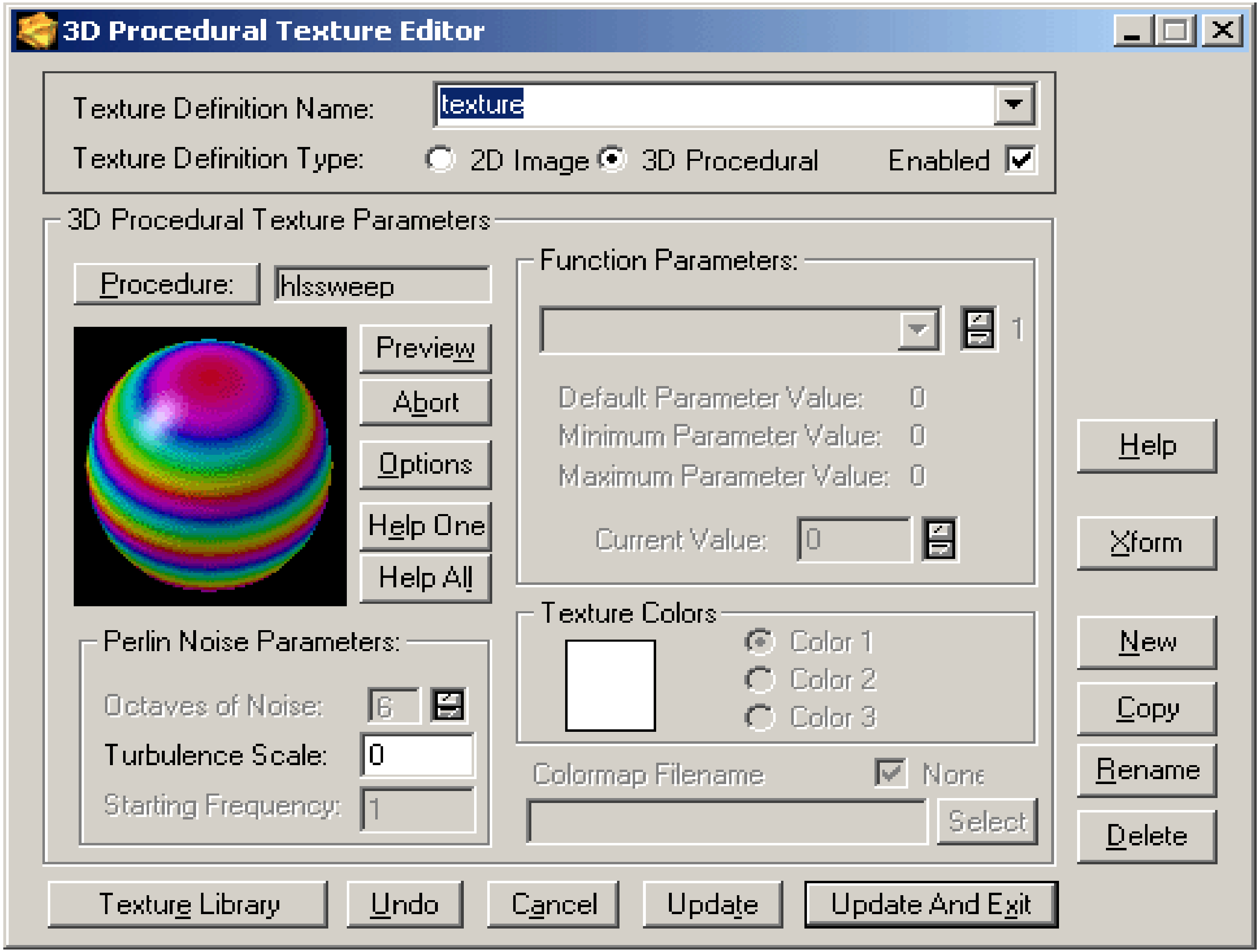
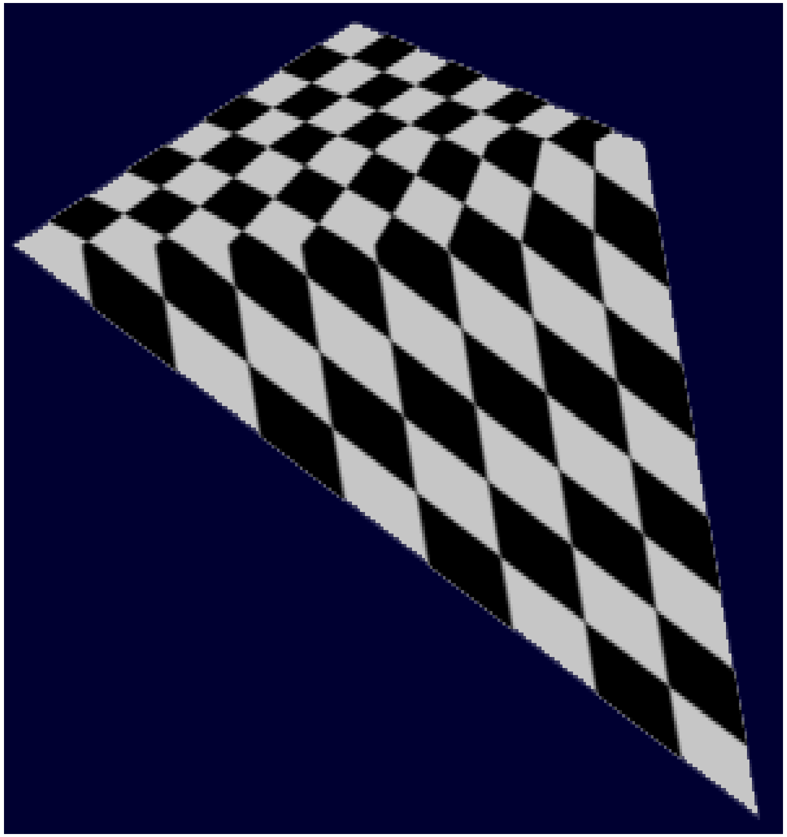
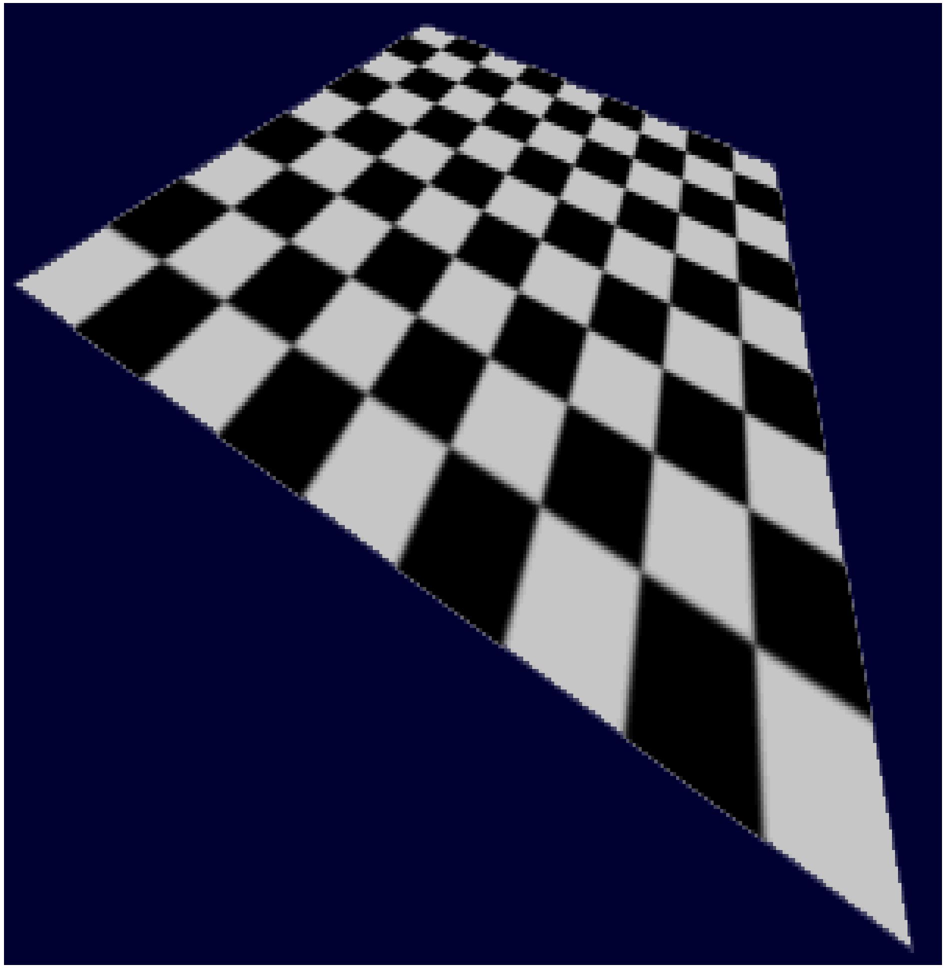



Figure 3: NuGraf's "3D Procedural Texture Editor"





Texture may also refer to:

- Texture (painting), feel of the canvas based on the paint used and its method of application
- Texture (visual arts), refers to the element of design and its application in art.
- Texture (music), a way to describe the overall sound created by the interaction of aspects of a piece of music
- Texture (crystalline), the property of a material's individual crystallites sharing some degree of orientation
- Texture (geology), the physical appearance or character of a rock
- Texture mapping, a bitmap image applied to a surface in computer graphics
- Texture (food), physical and chemical interaction of food in the mouth (Mouthfeel)
- Textures (band), a metal band from the Netherlands
- Soil texture, describes the relative proportion of grain sizes of a soil or any unconsolidated material
- Texture (cosmology), a type of theoretical topological defect in the structure of spacetime.
- Character structure, various life experiences resulting in the "texture" of one's character
- Texture (roads), road surface characteristics with waves shorter than road roughness
- Textures (software), a well known software program to typeset TeX and LaTeX on Macintosh computers
- Texture (image processing), a set of metrics calculated in image processing designed to quantify the perceived texture of an image.
- Microtexture of road surfaces
- Macrotexture of road surfaces
- Megatexture (roads) of road surfaces

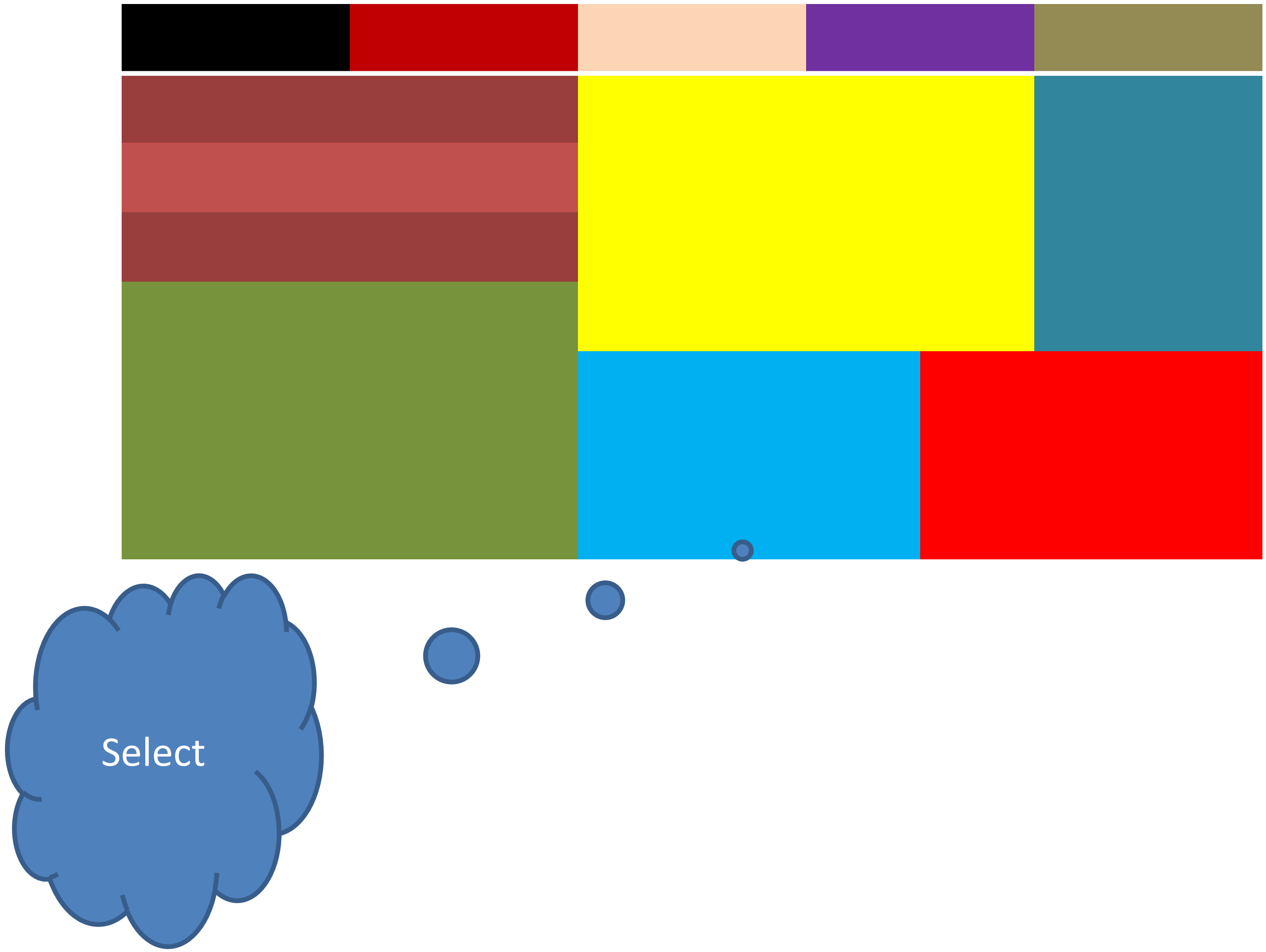
Example : Textures

- Brick Wall
- Wooden Furniture
- Mosaic Floor
- Skins
- Pineapple
- Tyre

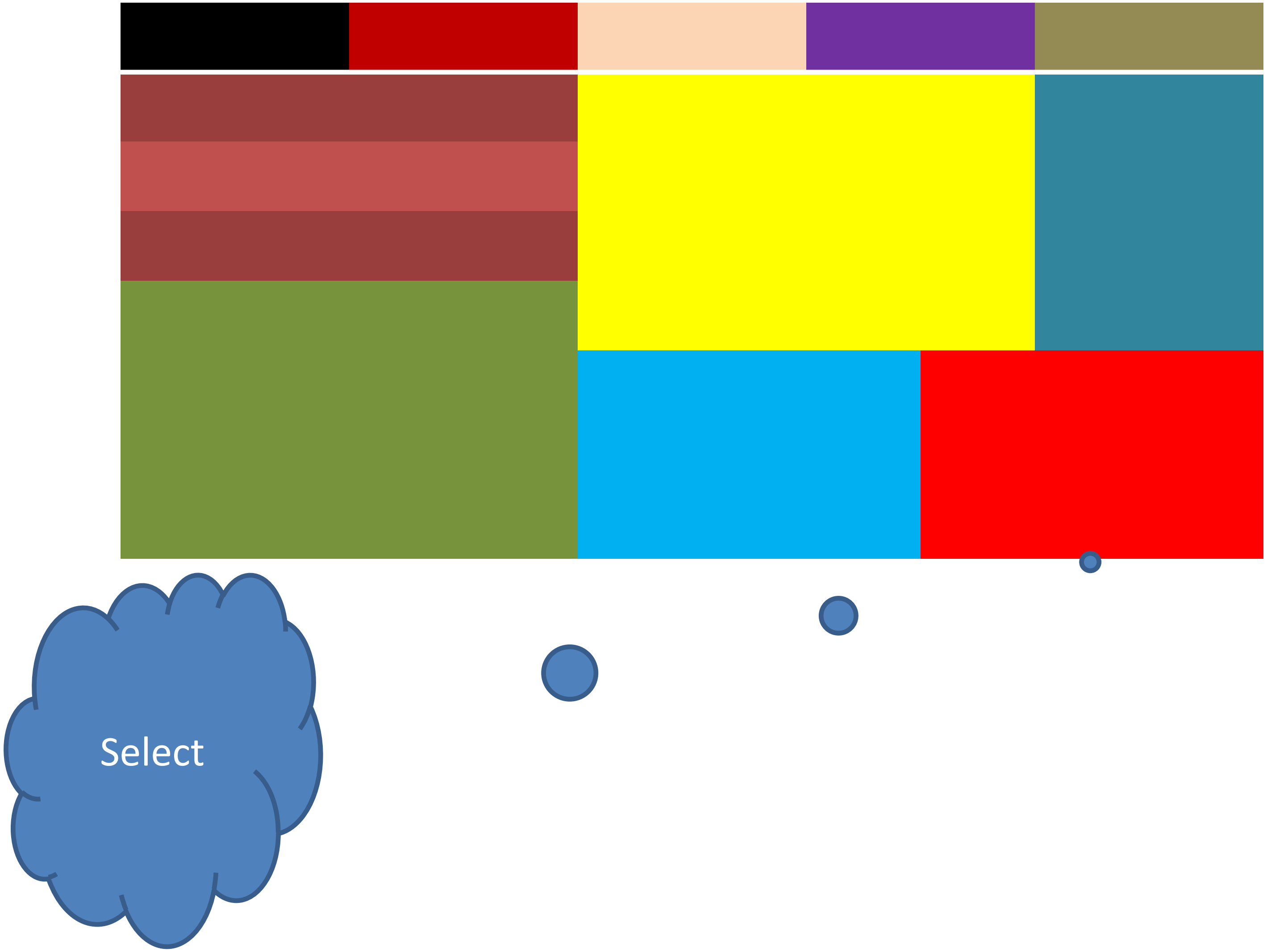
Mapping may refer to:

- The making of maps, as in **cartography, surveying, and photogrammetric**
- **In biology and neuroscience:**
- **Gene mapping**, the assignment of DNA fragments to chromosomes
- **Brain mapping**, set of techniques to study the brain
- In mathematics:
- Map (mathematics), often a synonym for function
- Functional predicate, a logical symbol that may be applied to an object term to produce another object term
- **In computing:**
- **Data mapping**, data element mappings between two distinct data models
- Level design, the creation of levels, locales, stages, or missions for a video game
- Memory-mapped I/O, hardware pretending to be memory
- Page mapping, or paging, in virtual memory systems
- Cache mapping, the mapping of main memory locations into entries of a cache (computing)
- Texture mapping, in computer graphics
- Device mapping, the assignment of I/O devices to file descriptors, file names, file numbers, etc.
- In logic, linguistics, and psychology:
- Conceptual metaphor, an understanding one conceptual domain in terms of another conceptual domain
- Metaphor, cross mapping across two or more seemingly unrelated subjects
- Analogy, inference from a particular to another particular

Example of Texture Mapping



Example of TextureMapping



Other Example

@@@@@@®®®®®®¥¥¥¥¥¥££££££
€€€€€€€€

Result

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
¥¥¥
¥¥¥
¥¥
€€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€€€€€€€€€€€
€€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€€€€€€€€€€€
€€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€€€€€€€€€€€
€€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€ €€€€€€€€€€€€€€€€€€

Parametric form of Texture Mapping

Now assume a co-ordinate (x,y) of texture mapping which to be represented in parametric form of (p,q) using the mapping technique.

So in this case the mapping relationship can be established as :

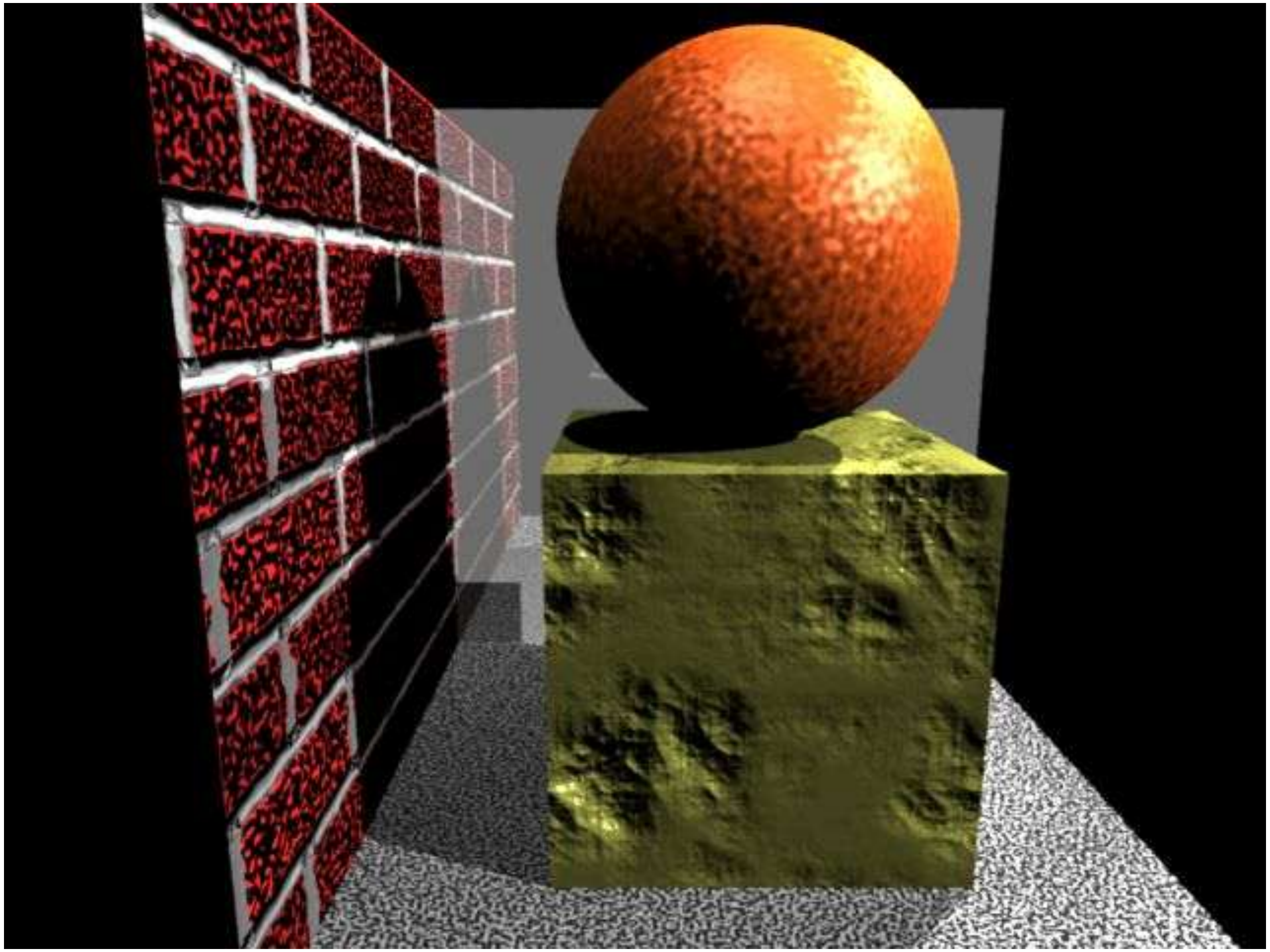
$$p = Ax + B$$
$$q = Cy + D$$

Where the constants A,B,C and D are obtained by using these relationship equations to some known value of co-ordinates. These co-ordinates are normally obtained from the known points of the corners of the texture map and the corresponding surface points.

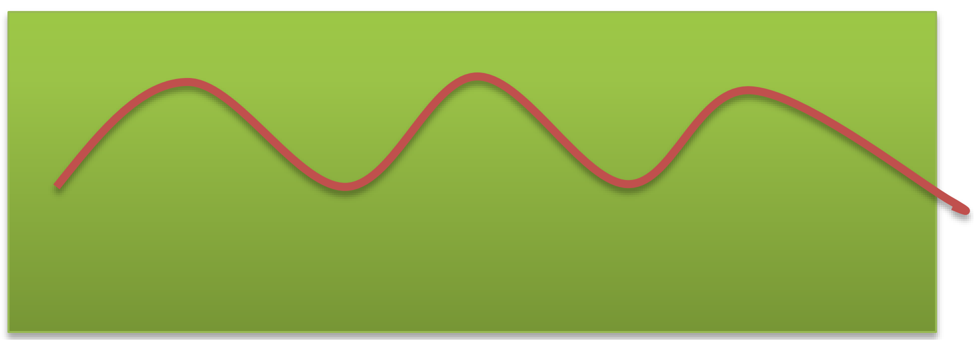
There are many methods present and are used to generate this effect. One of the popular such method is bump mapping method.

Bump Mapping(Roughness) : To add Color , Magnitude, direction with original color attributes and used in Curve.

Bump Mapping



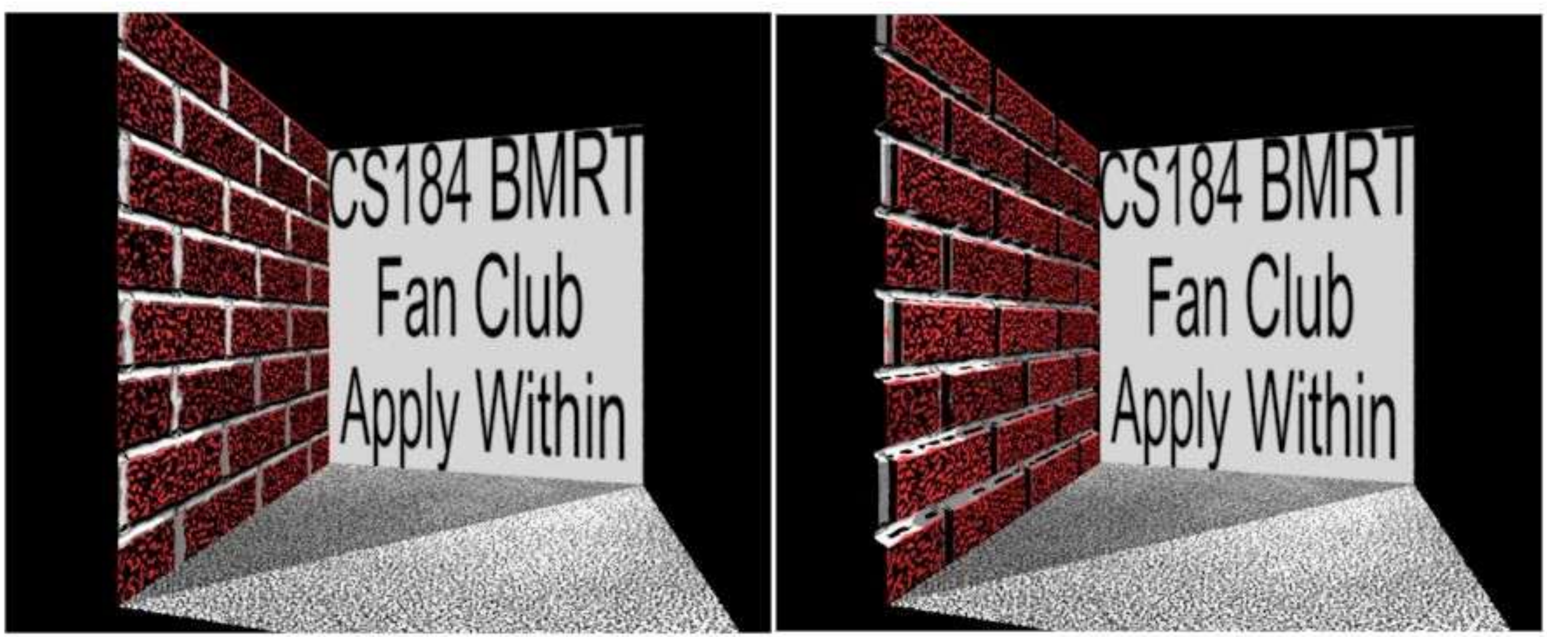
Before Bump Mapping



After Bump Mapping

Displacement Mapping

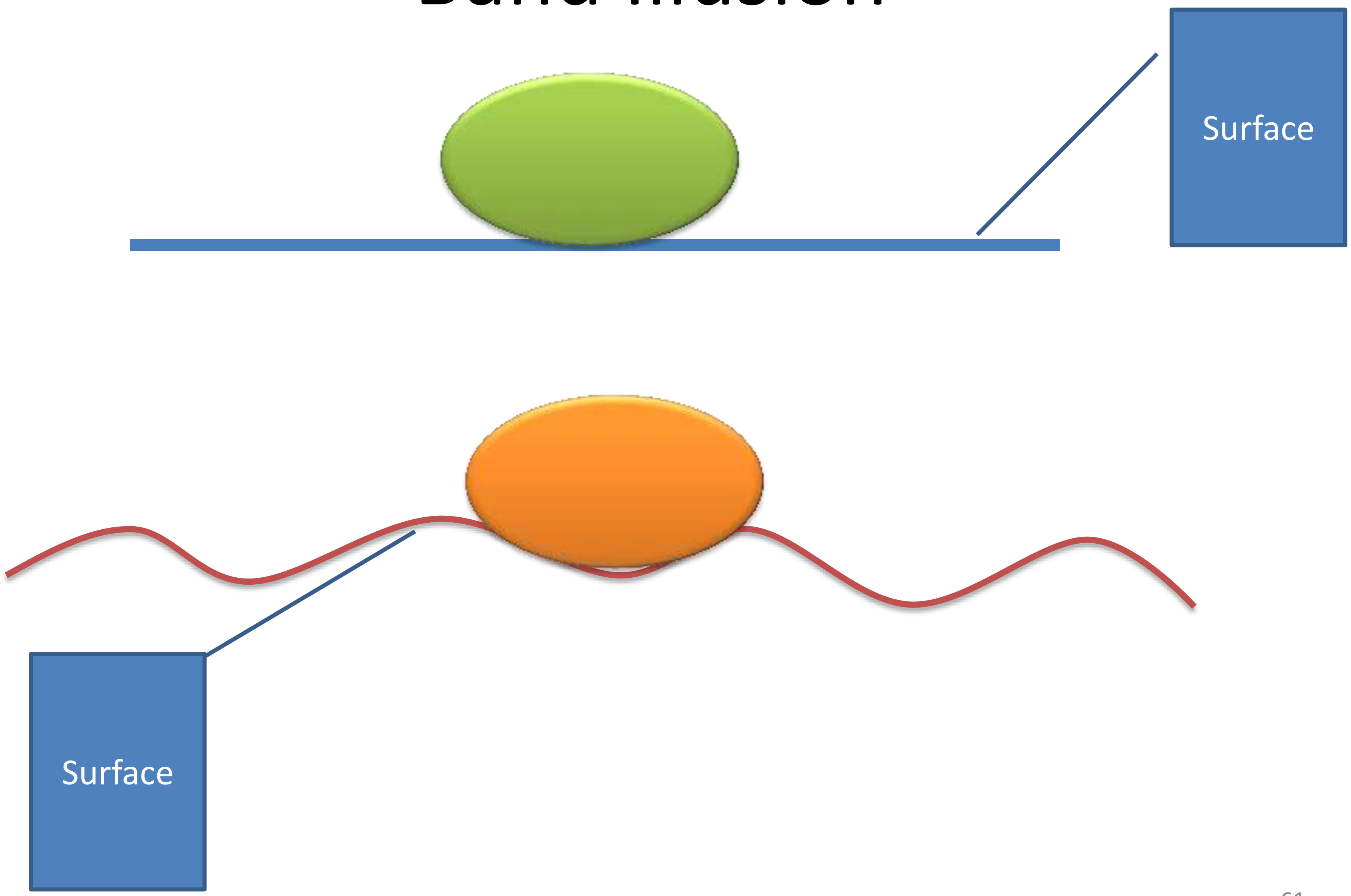
- Bump mapped normals are inconsistent with actual geometry. Problems arise (shadows).
- Displacement mapping actually affects the surface geometry



Texture Mapping Characteristics

- It is rendering (picture) technique.
- Most real-life objects.
- Roughness (Bump Mapping).
- Add specified pattern but the surface still appear smooth.
- Rectangular Shape.
- Different Colors.
- Intensity
- Shading

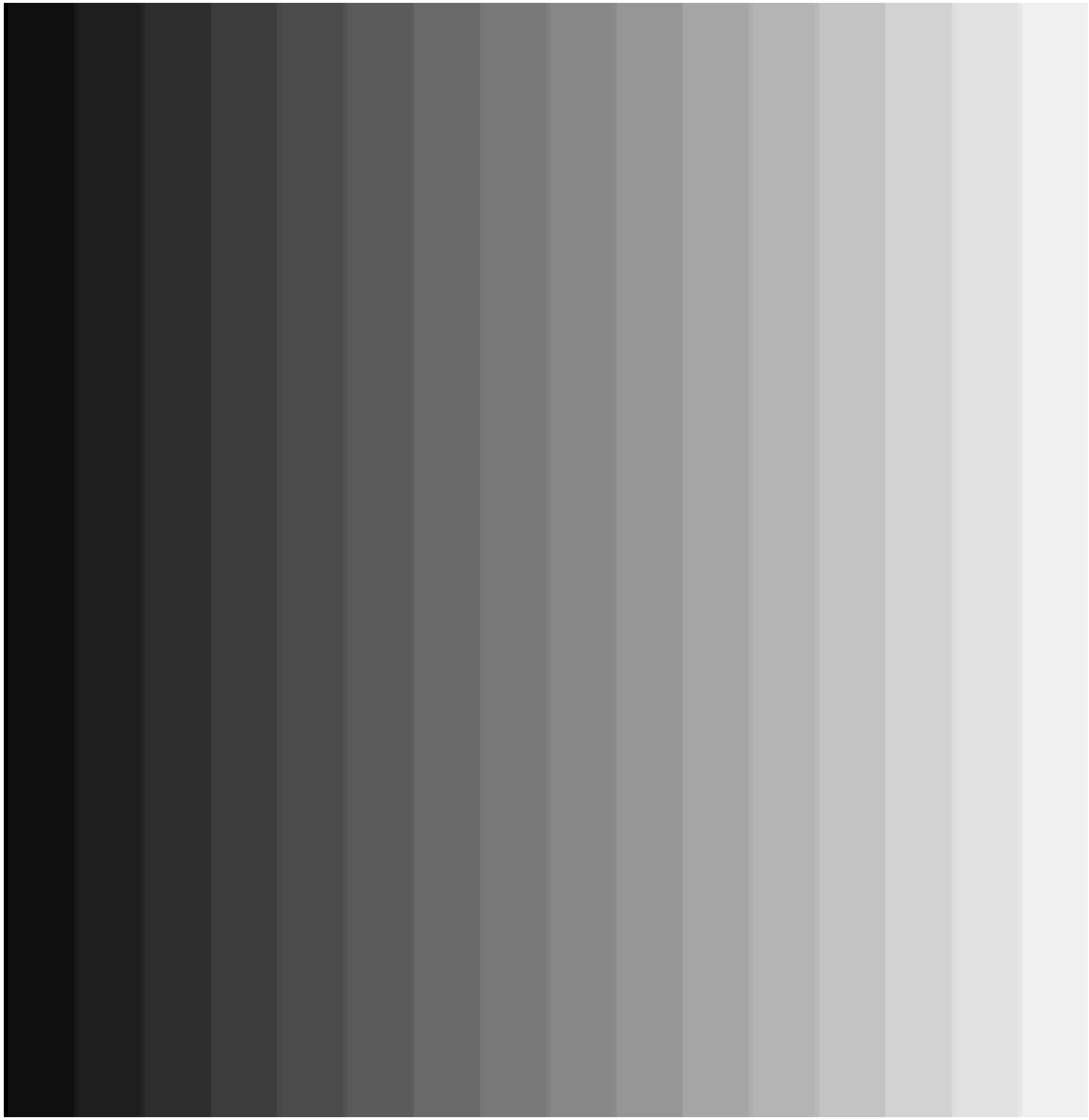
Band Illusion



Lateral Inhibition

- Biological word.
- Human Visual System.
- Lateral inhibition increases the contrast and sharpness in visual response.


This is a text in red
This is a text in green
This is a text in blue
This is a text in red
This is a text in green
This is a text in blue
This is a text in green



RECAPITULATION

1. The act or process of recapitulating.
2. A summary or concise review.

Transparency

Transparency

- Transparency is possible in a number of graphics file formats.
- The term transparency is used in various ways by different people, but at its simplest there is "full transparency" i.e. something that is completely invisible.
- There are many different ways to mix colors.
- Raster file formats that support transparency include GIF, PNG, and TIFF, through either a *transparent color* or an alpha channel.
- A suitable bitmap graphics editor shows transparency by a special pattern, e.g. a chessboard pattern.

- A bitmap graphics editor is a computer program that allows users to paint and edit pictures interactively on the computer screen and save them in one of many popular "bitmap" or "raster" formats such as JPEG, PNG, GIF and TIFF.
- Usually an image viewer is preferred over a bitmap graphics editor for viewing images.
- Some of the features common to many bitmap graphics editors are:
- Select a region for editing.
- Draw lines with brushes of different color, size, shape and pressure
- Fill in a region with a single color, gradient of colors, or a texture.
- Select a color using different color models (e.g. RGB, HSV), or by using a color dropper.
- Add typed letters in different font styles.
- Remove scratches, dirt, wrinkles, and imperfections on photo portraits.
- Composite editing by using layers.
- Edit and convert between various color models.
- Apply various filters for effects like sharpening and blurring.
- Convert between various image formats .

Visualization of Data Sets

Visualization

- **Visualization** is any technique for creating images, diagrams, or animations to communicate a message. Visualization through visual imagery has been an effective way to communicate both abstract and concrete ideas since the dawn of man. Visualization today has ever-expanding applications in science, education, engineering (e.g. product visualization), interactive multimedia, medicine, etc. Typical of a visualization application is the field of computer graphics. The invention of computer graphics may be the most important development in visualization since the invention of central perspective in the regeneration period.

Data Sets

- A **data set** (or **dataset**) is a collection of data, usually presented in tabular form. Each column represents a particular variable. Each row corresponds to a given member of the data set in question. Its values for each of the variables, such as height and weight of an object or values of random numbers. Each value is known as a datum. The data set may comprise data for one or more members, corresponding to the number of rows.

The following are examples of some common visualization techniques:

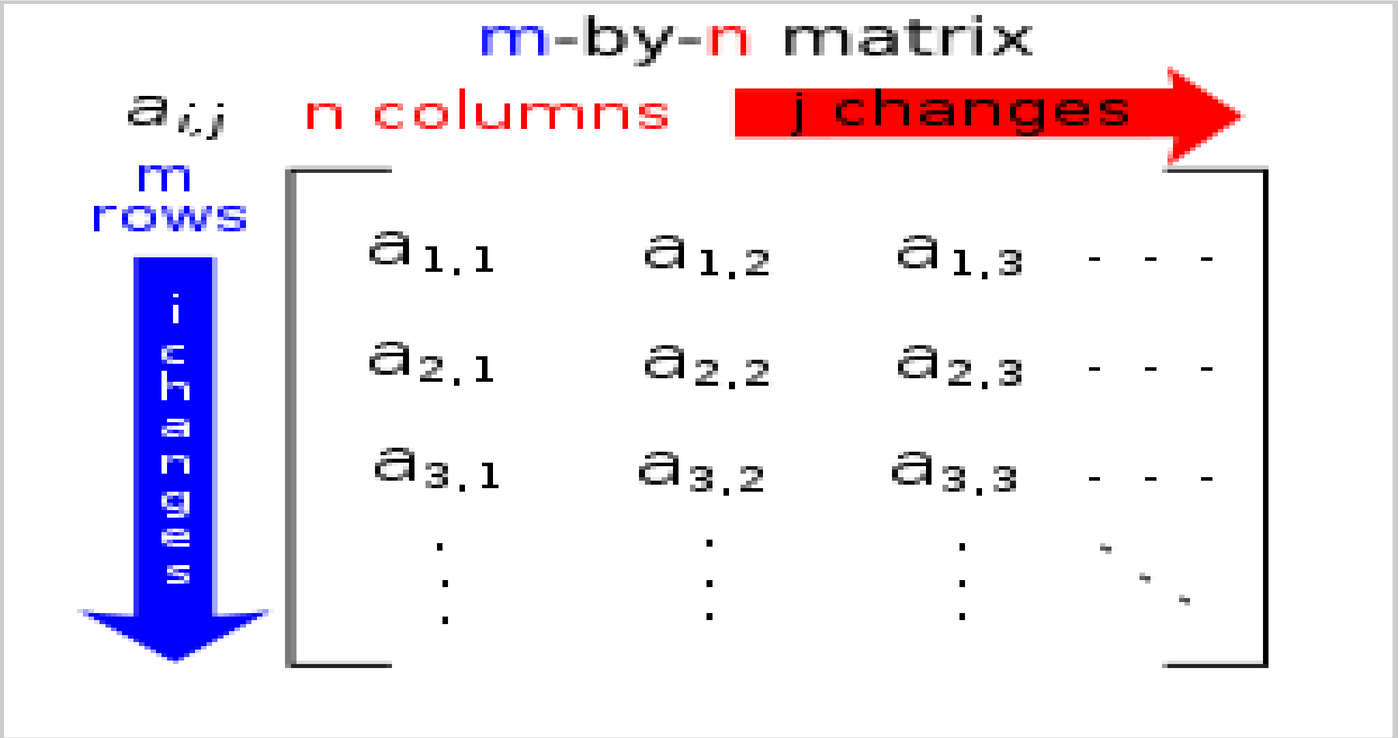
- direct volume rendering
- Streamlines, streaklines, and pathlines
- table, matrix
- charts (pie chart, bar chart, histogram, function graph, scatter plot, etc.)
- graphs (tree diagram, network diagram, flowchart, existential graph, etc.)
- Maps
- parallel coordinates - a visualization technique aimed at multidimensional data
- Tree map - a visualization technique aimed at hierarchical data
- Venn diagram
- Timeline
- Euler diagram
- Chern off face
- Hyperbolic trees
- brushing and linking
- Cluster diagram or dendrogram
- Ordino gram

Example1: TABLE

Purchased Equipments (June, 2006)			
Item Num#	**Item Picture**	**Item Description**	**Price**
		Shipping Handling, Installation, etc	**Expense**
1.		IBM Clone Computer.	$ 400.00
		Shipping Handling, Installation, etc	$ 20.00
2.		1GB RAM Module for Computer.	$ 50.00
		Shipping Handling, Installation, etc	$ 14.00
Purchased Equipments (June, 2006)			

An example table rendered in a web browser using HTML.

Example2: MATRIX

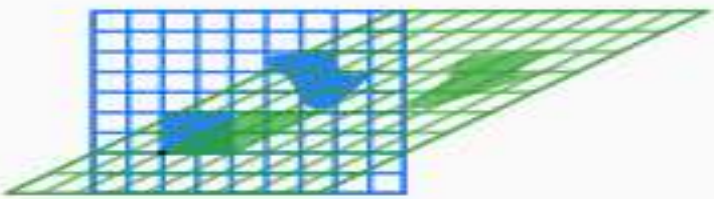
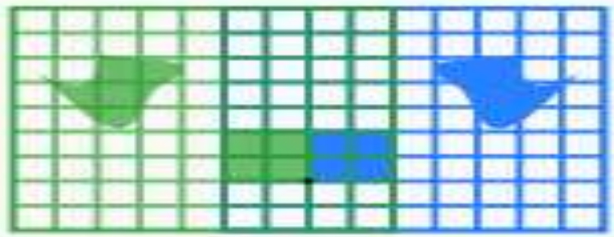


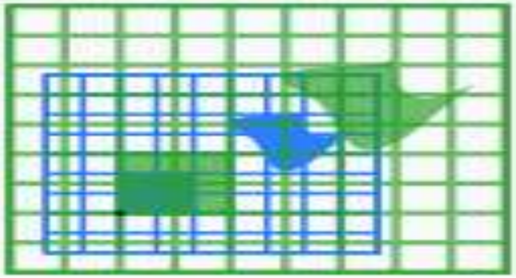
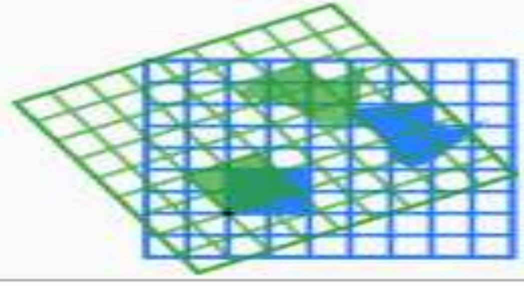
Specific entries of a matrix are often referenced by using pairs of subscripts.

BASIC OPERATIONS

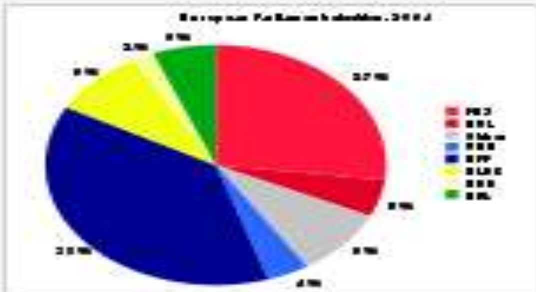
Operation	Example
Addition	$\begin{bmatrix}1 & 3 & 1\\1 & 0 & 0\end{bmatrix} + \begin{bmatrix}0 & 0 & 5\\7 & 5 & 0\end{bmatrix} = \begin{bmatrix}1+0 & 3+0 & 1+5\\1+7 & 0+5 & 0+0\end{bmatrix} = \begin{bmatrix}1 & 3 & 6\\8 & 5 & 0\end{bmatrix}$
Scalar multiplication	$2 \cdot \begin{bmatrix}1 & 8 & -3\\4 & -2 & 5\end{bmatrix} = \begin{bmatrix}2\cdot 1 & 2\cdot 8 & 2\cdot -3\\2\cdot 4 & 2\cdot -2 & 2\cdot 5\end{bmatrix} = \begin{bmatrix}2 & 16 & -6\\8 & -4 & 10\end{bmatrix}$
Transpose	$\begin{bmatrix}1 & 2 & 3\\0 & -6 & 0\end{bmatrix}^T = \begin{bmatrix}1 & 0\\2 & -6\\3 & 0\end{bmatrix}$

TRANSFORMATION

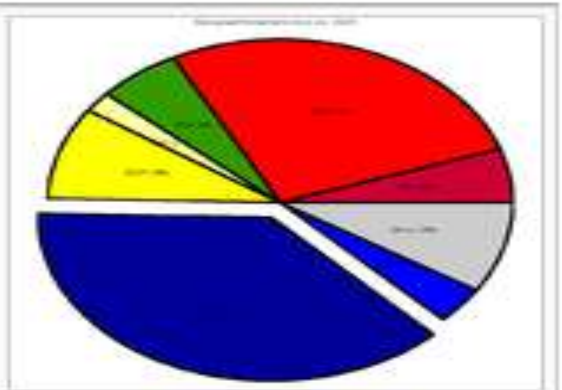
Vertical shear with m=1.25.	Horizontal flip
$\begin{bmatrix} 1 & 1.25 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$
	

Scaling by a factor of 3/2	Rotation by π/6 = 30°
$\begin{bmatrix} 3/2 & 0 \\ 0 & 3/2 \end{bmatrix}$	$\begin{bmatrix} \cos(\pi/6) & -\sin(\pi/6) \\ \sin(\pi/6) & \cos(\pi/6) \end{bmatrix}$
	

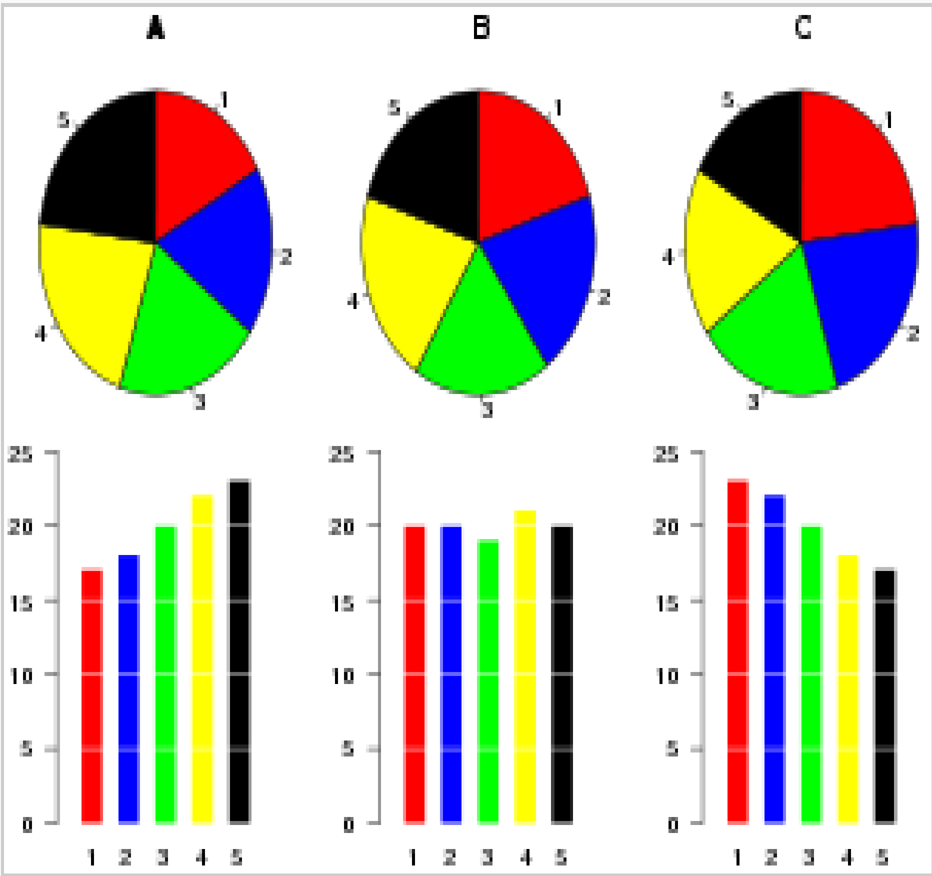
Example3: PIE & BAR CHART



A pie chart for the example data.

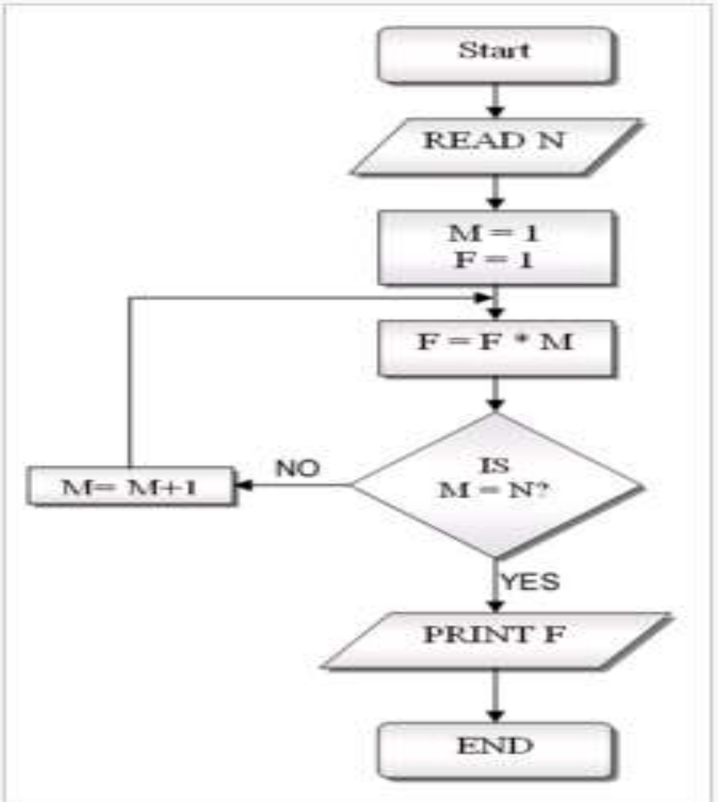


An exploded pie chart for the example data, with the largest party group exploded.

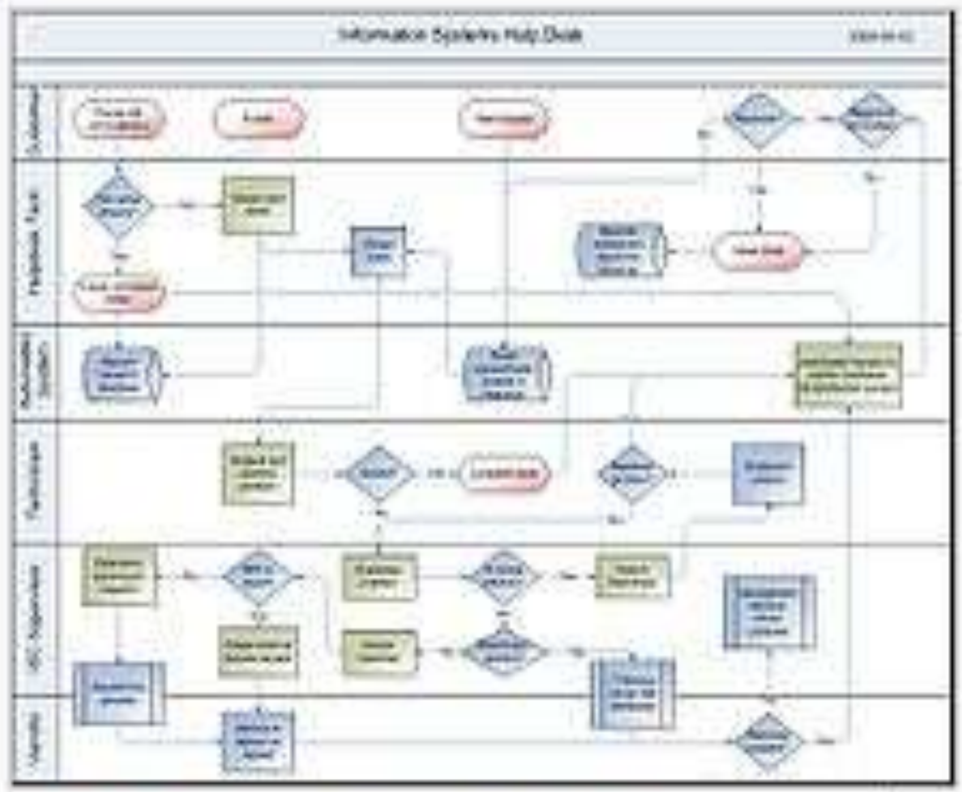


Three sets of data plotted using pie charts and bar charts.

EXAMPLE 4 : FLOWCHART

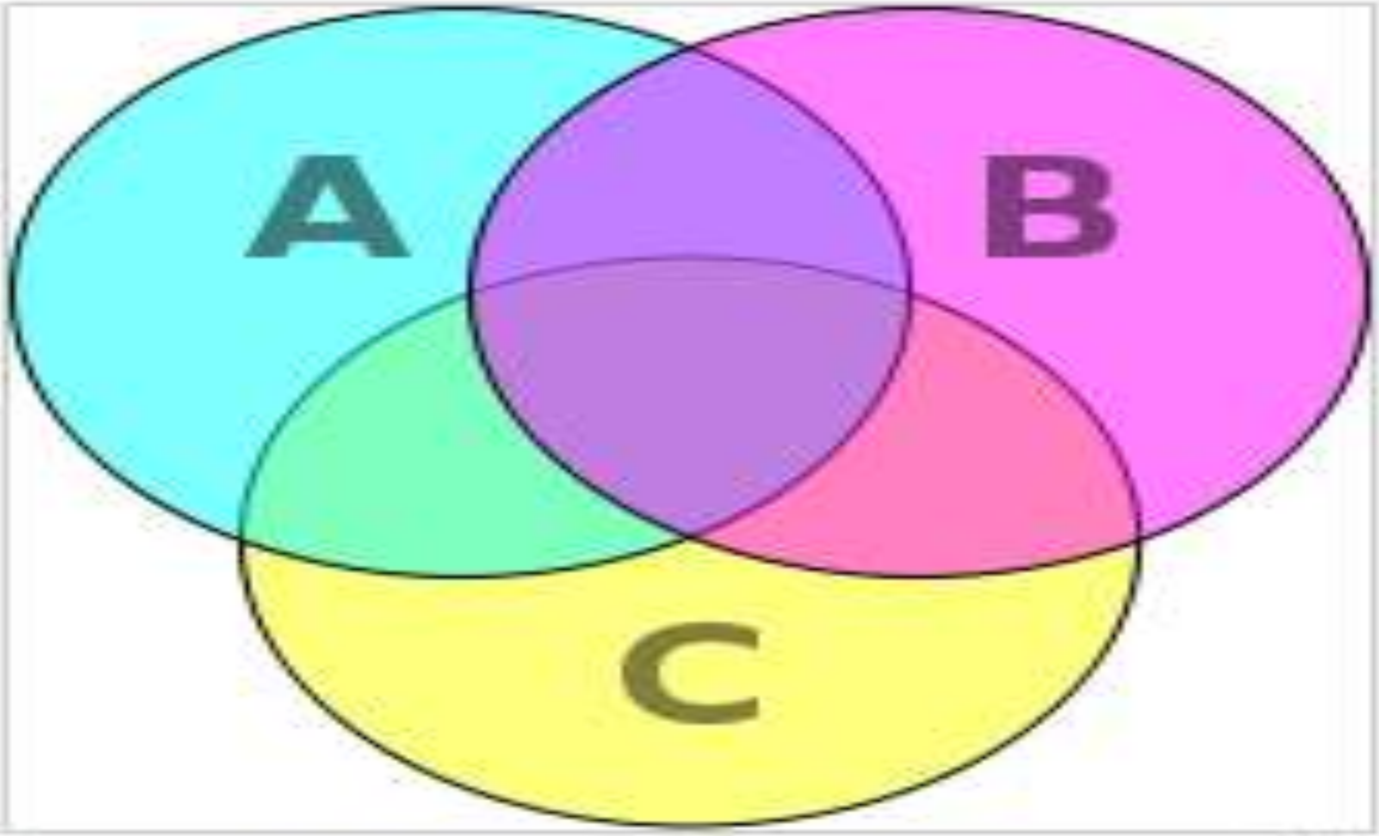


A simple flowchart for computing factorial N (N!)



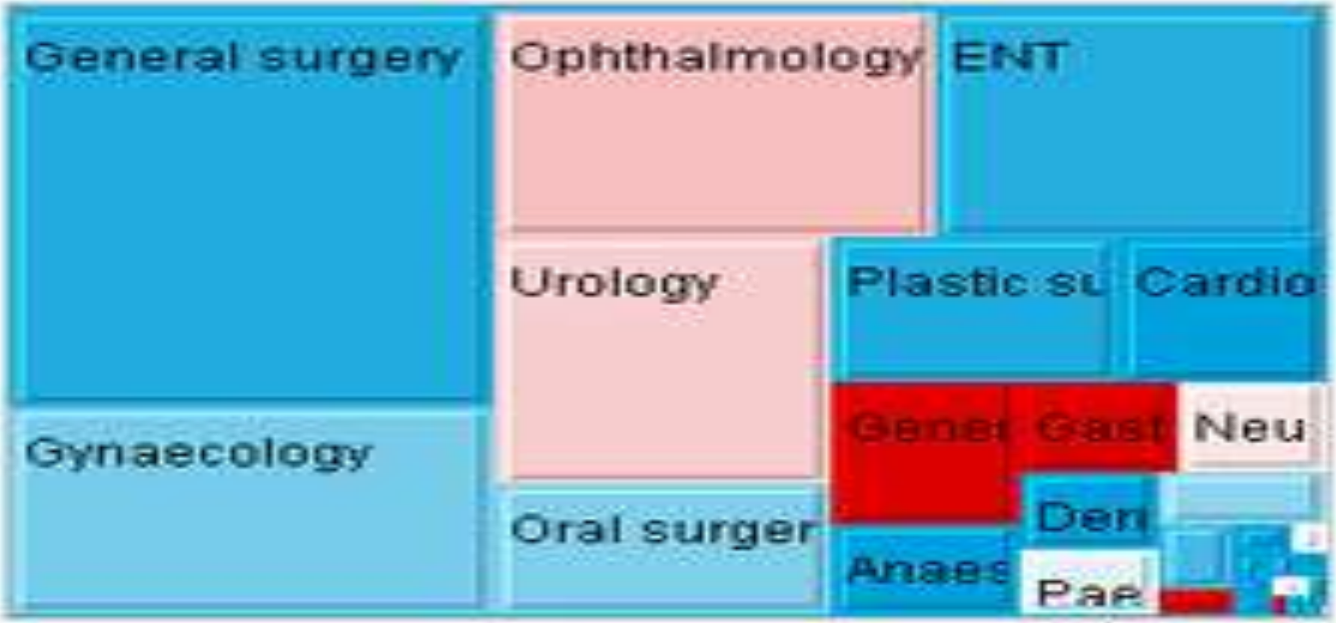
Example of a system flowchart.

EXAMPLE 5 : VENN DIAGRAM



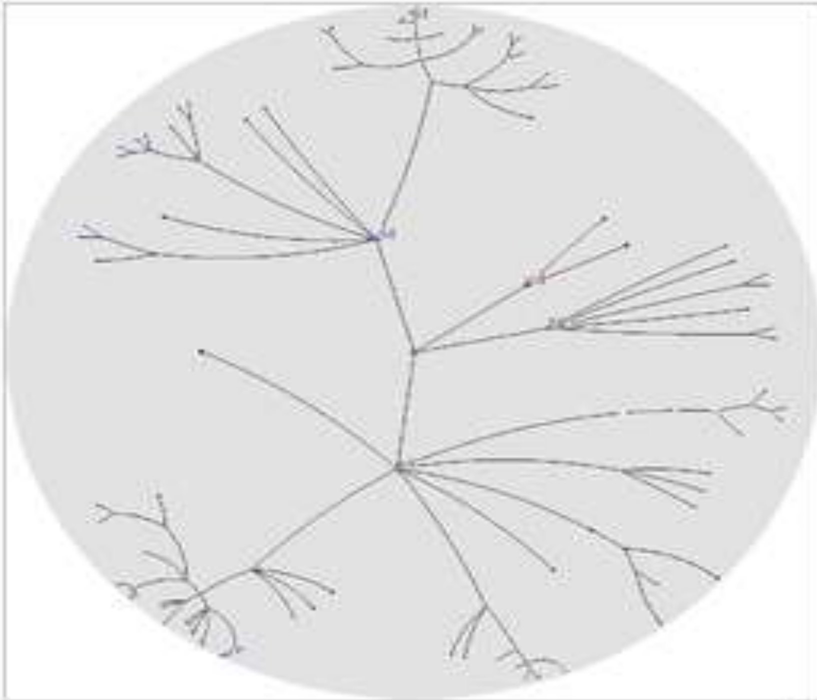
A Venn diagram of sets A, B, and C

EXAMPLE 6 : TREEMAPPING

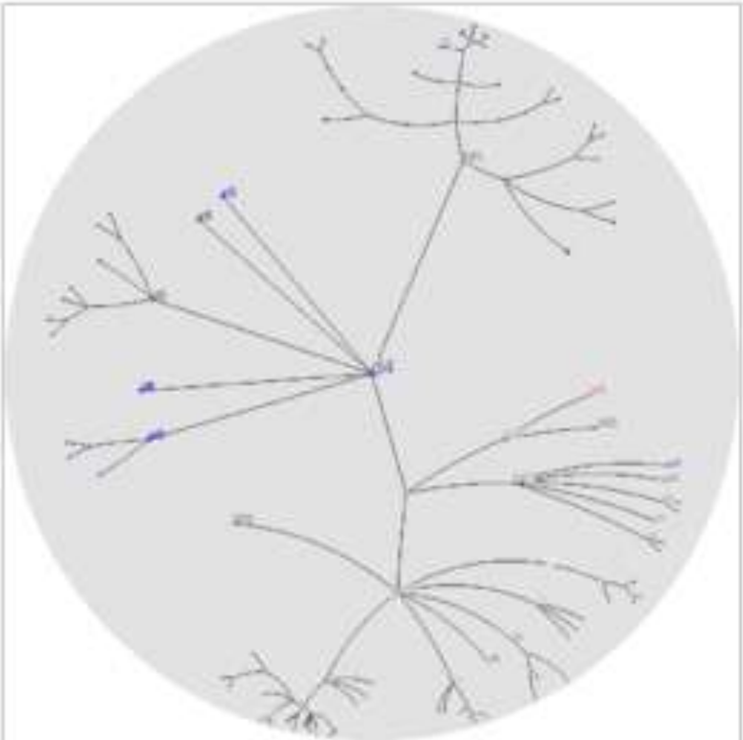


Treemap showing changes in waiting times for patients of English PCTs. Image released under Creative Commons by Incito Ltd.

EXAMPLE 7- HYPERBOLIC TREE

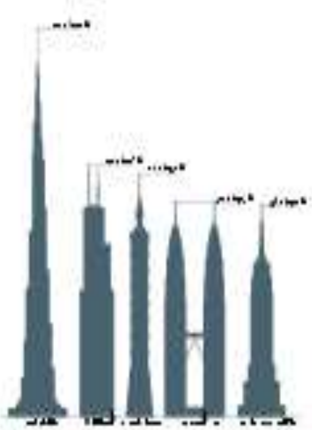


A basic hyperbolic tree. Nodes in focus are placed in the center and given more room, while out -of-focus nodes are compressed near the boundaries.

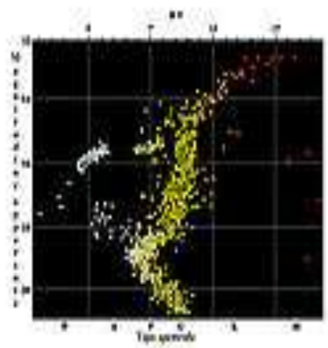


Focusing on a different node brings it and its children to the center of the disk, while uninteresting portions of the tree are compressed.

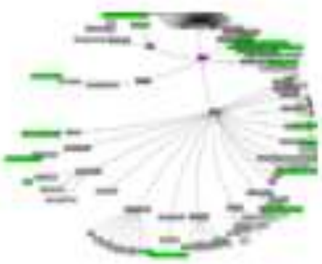
EXAMPLE 8: CLUSTER DIAGRAM



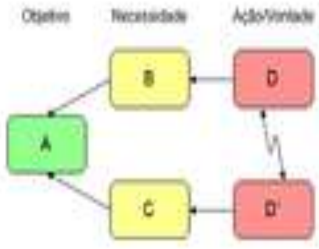
Comparison diagram of sky scraper



Astronomic cluster diagram of the Messier 3 globular cluster

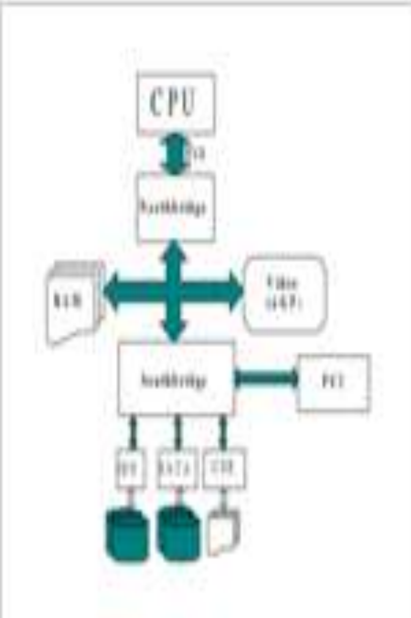


Biositemap diagram

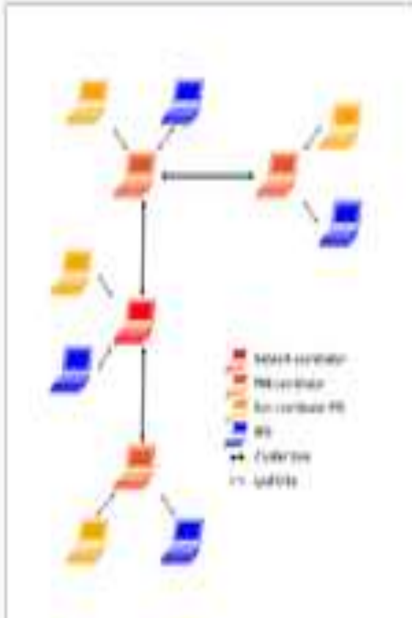


Cluster chart in brainstorming

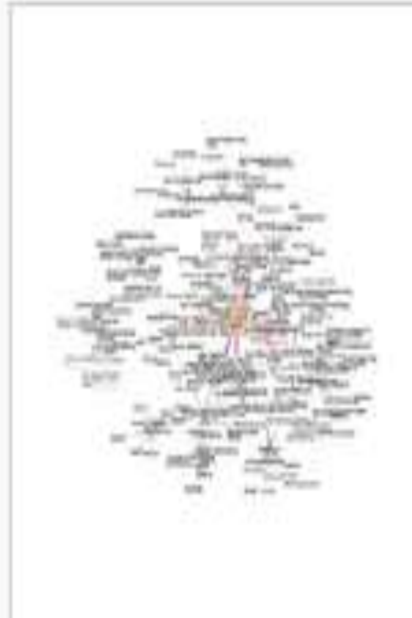
EXAMPLE 8: CLUSTER DIAGRAM



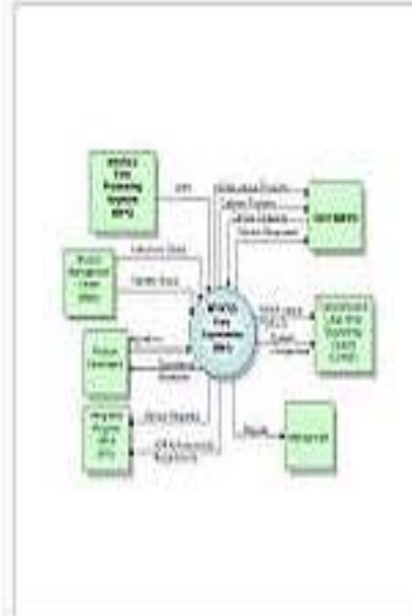
Computer architecture diagram of a PC



Computer network diagram

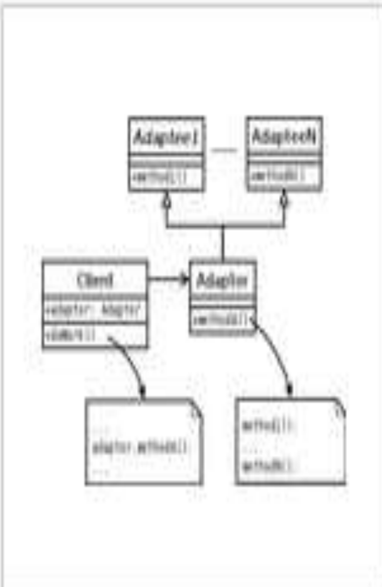


Internet diagram

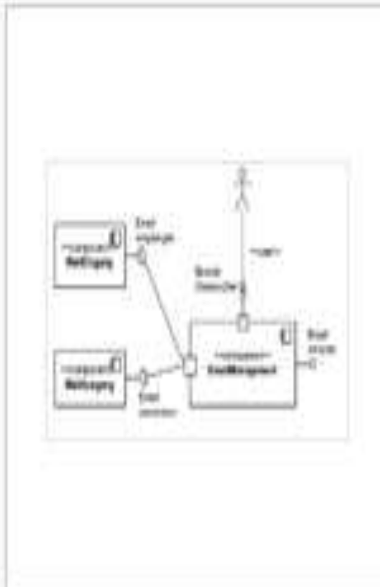


System context diagram

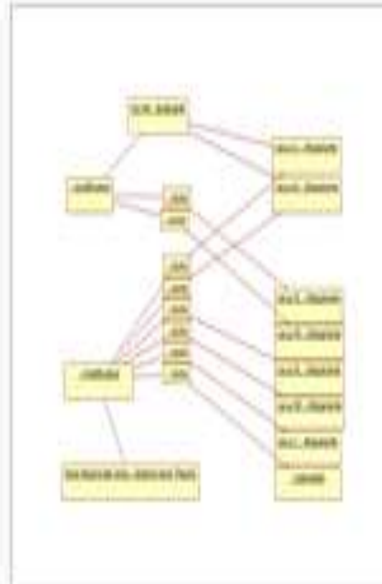
EXAMPLE 8: CLUSTER DIAGRAM



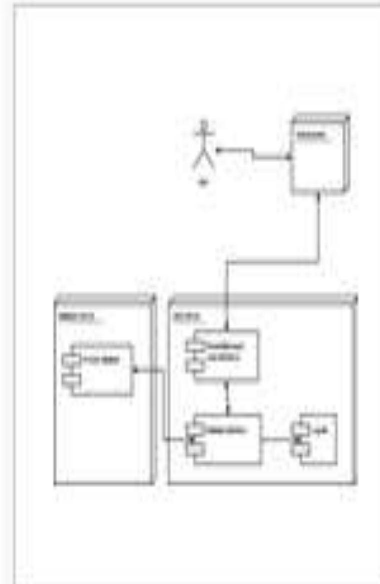
UML Class diagram



UML Component diagram

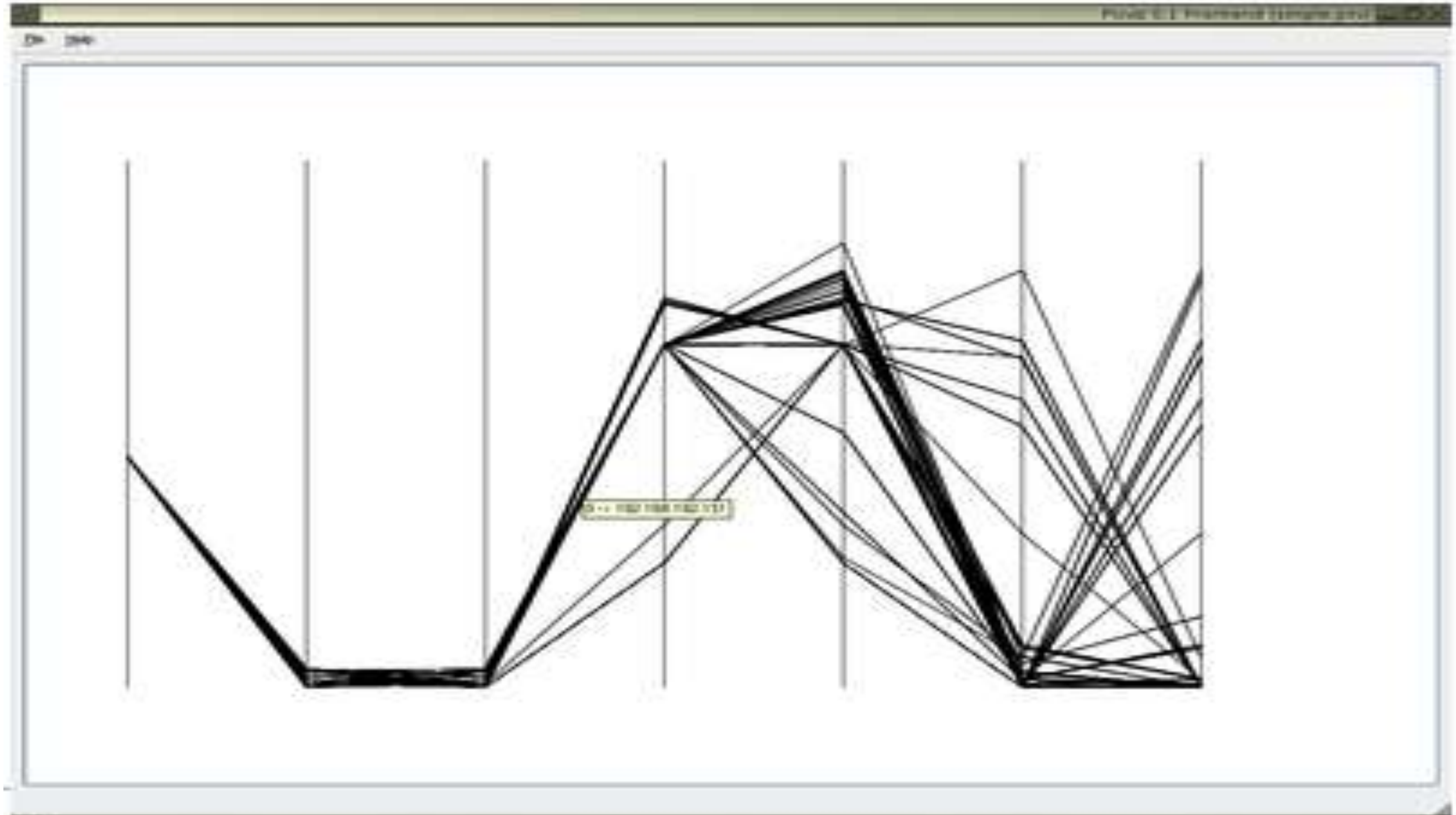


UML Composite structure diagram



UML Deployment diagram

EXAMPLE 9 : PARALLEL COORDINATES



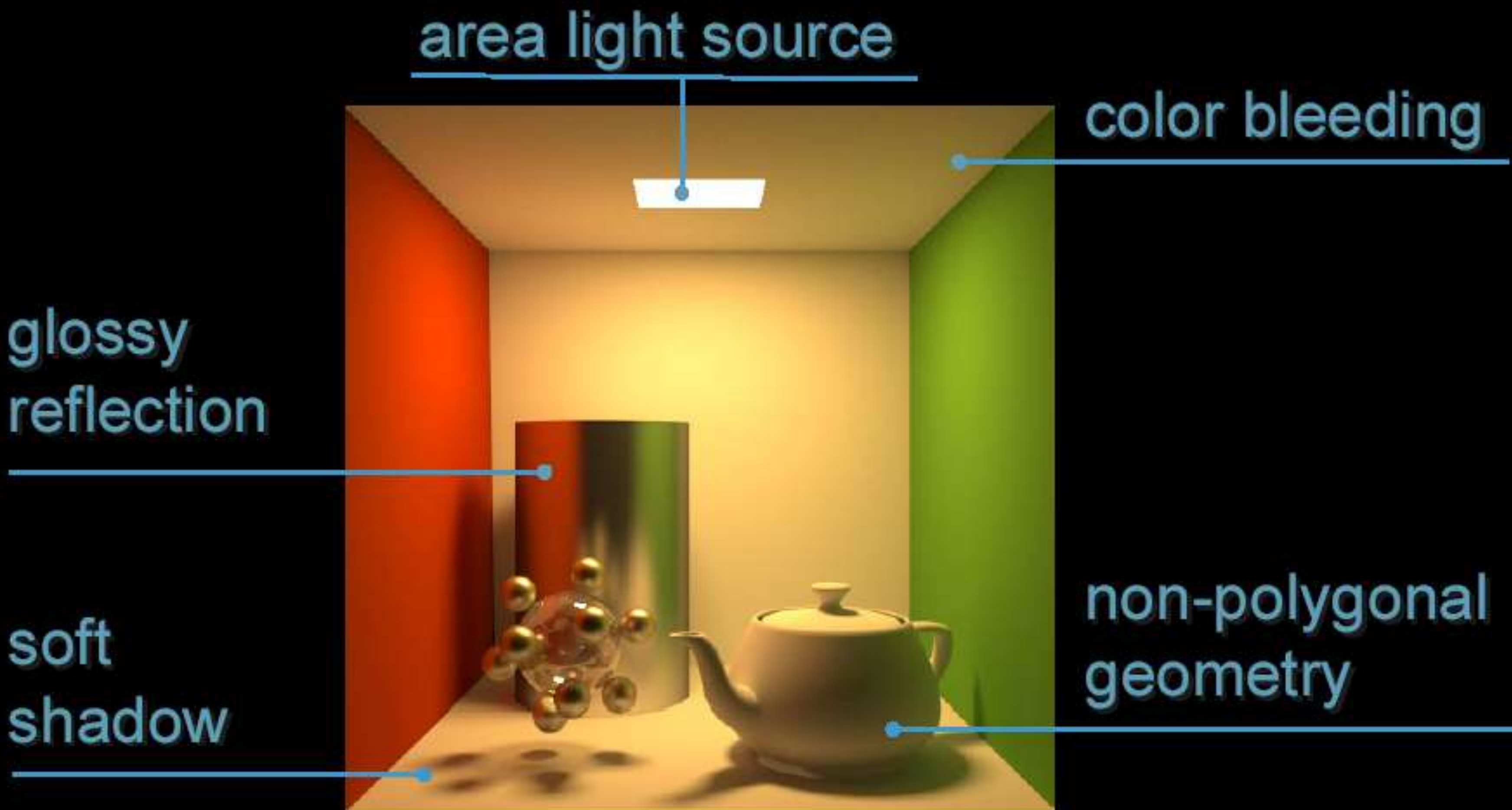
Volume Rendering

Volume rendering is a technique used to display a 2D projection of a 3D discretely sampled data set.

Radiosity

- Calculating the overall light propagation within a scene, for short **global illumination** is a very difficult problem.
- With a standard ray tracing algorithm, this is a very time consuming task, since a huge number of rays have to be shot.

Global Illumination?



Radiosity

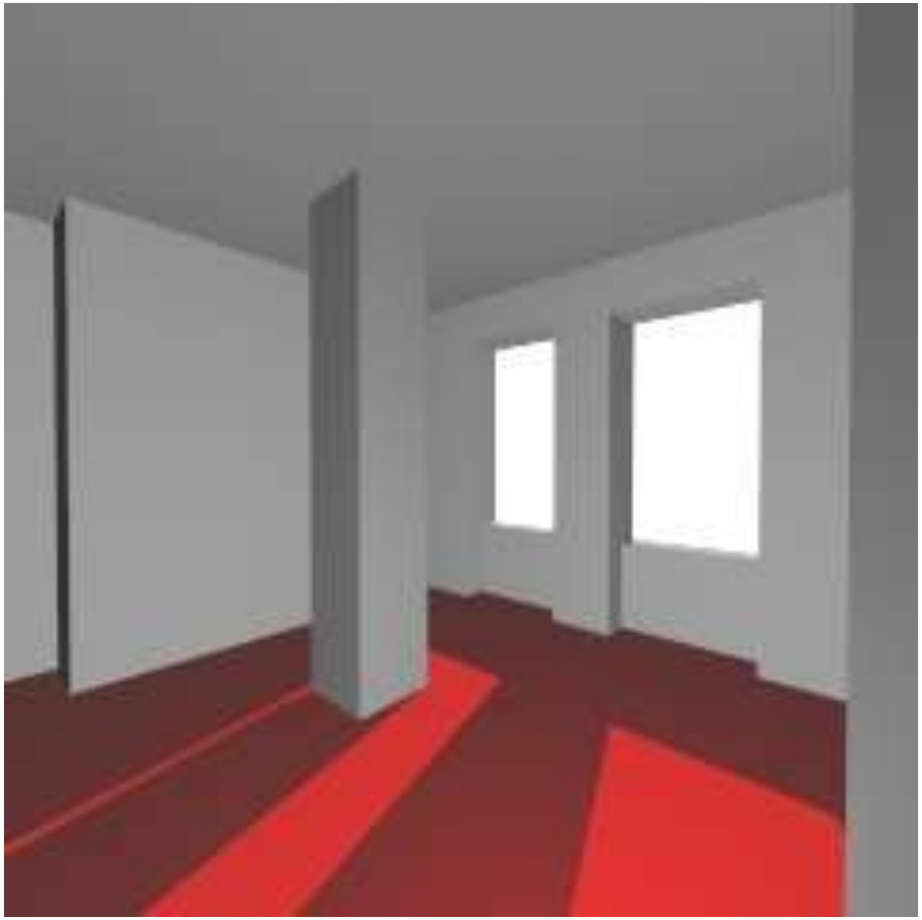
- For this reason, the radiosity method was invented.

- The main idea of the method is

  to store illumination values on the surfaces of the objects, as the light is propagated starting at the light sources.


Ray Tracing

Radiosity



Diffuse Interreflection

Shadow

- A **shadow** is an area where direct light from a light source cannot reach due to obstruction by an object. It occupies all of the space behind an opaque object with light in front of it. The cross section of a shadow is a two-dimensional shadow, or reverse projection of the object blocking the light.
- Follow the hidden surface removal concepts.
- Calculate the shadow volume.

Shadows - Close up of an abacus that is casting a shadow on...

Web Gold Icons Set Shadows Relections on White 1 - Gold...

Web Buttons Blue Set 1 in Metal with Reflection Shadow - A...

Black PDA isolated - front view of modern pda on white with...

Pixel arrow hand 3D cursors point on shadows - Pixel arrow...

Web Blue Icons Set Shadows Relections Angled 1 - Angled...

Long shadow of man - Shadow of man on dirt at sundown

Palm Tree Shadow on Beach

yellow autumn tree leave with shadow isolated over white...

eye shadows - multicolored eye shadows against the white...

multicolored eye shadows and cosmetics brush - multicolored...

Shadow People

0705ARMWRESTLING

0705CELEBRATING

0705CLEANING

0705DIVING

0705DRAWING

0705FOOTBALL

0705NEWSPAPER

0705PLAYINGCARDS

0705PLAYINGCHESS

0705PUSHUPS

0705READING

0705RUNNING

0705SHOPPING

0705SKATEBOARDING

0705STANDING

0705TEACHING

0705WATCHINGTV

0705WORKBOUND

0705WORKING

people-001

Man walking around thinking about life

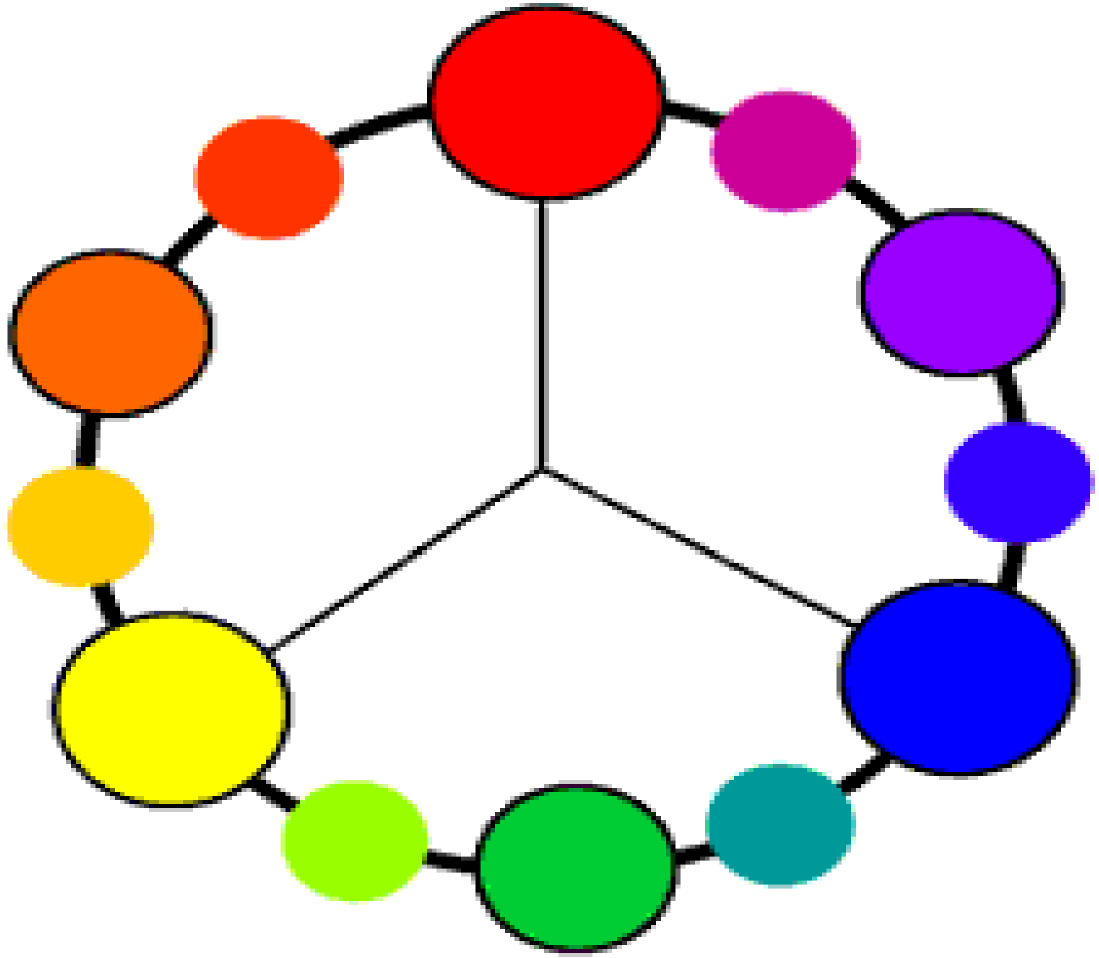
people-003

people-004

people-005

COLOR ISSUSES

The painter's color wheel

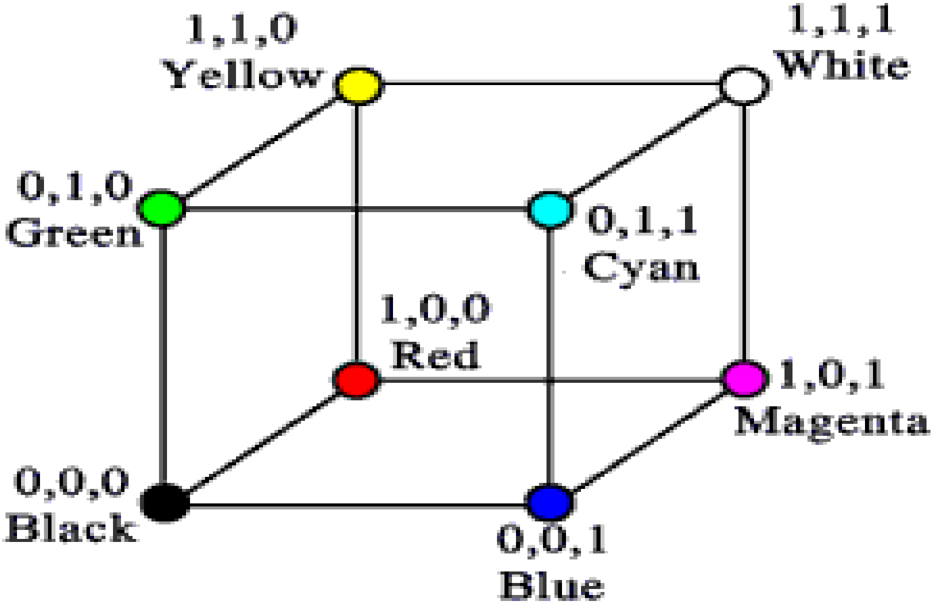


- Primary colors - the ``pure'' component colors from which are mixed other colors in a given color system
- Secondary colors - equal mixtures of two primary colors
- Tertiary colors - unequal mixtures of two or more primary colors
- A painters primaries can be somewhat aribitrarily chosen, but redish, yellowish, and blueish color primaries plus black and white often form the core of a painter's pallette

CIE (Commission Internationale de L'Eclairage) XYZ tristimulus colors

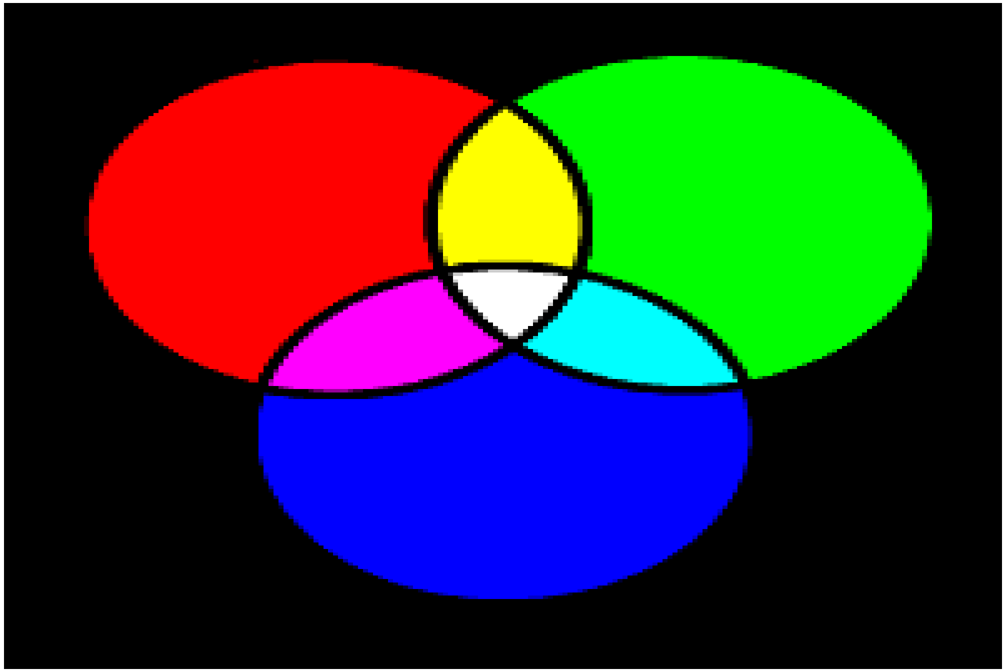
RGB Color Space

- RGB - Red, Green, Blue



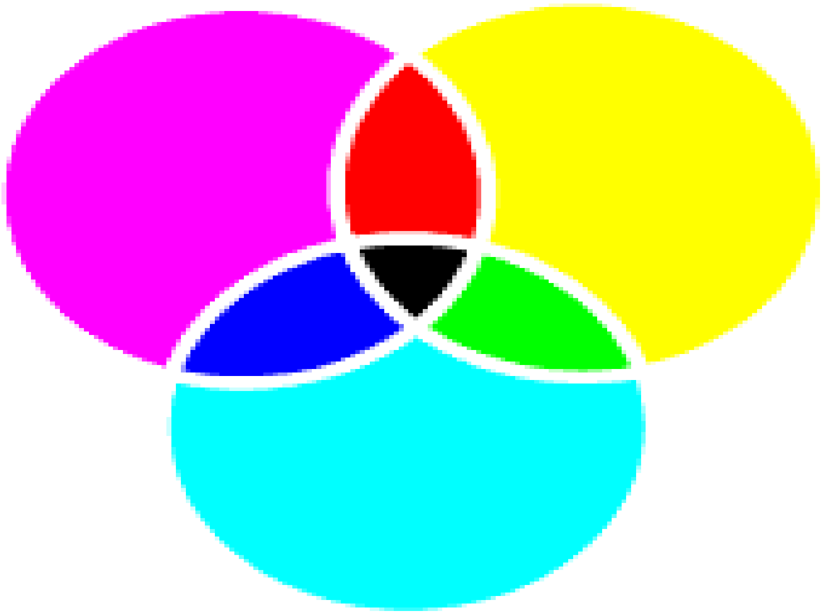
- R,G,B color cube
- R, G, B sliders
- Hardware display oriented
- Not terribly intuitive for mixing tertiary colors

R,G,B additive primaries

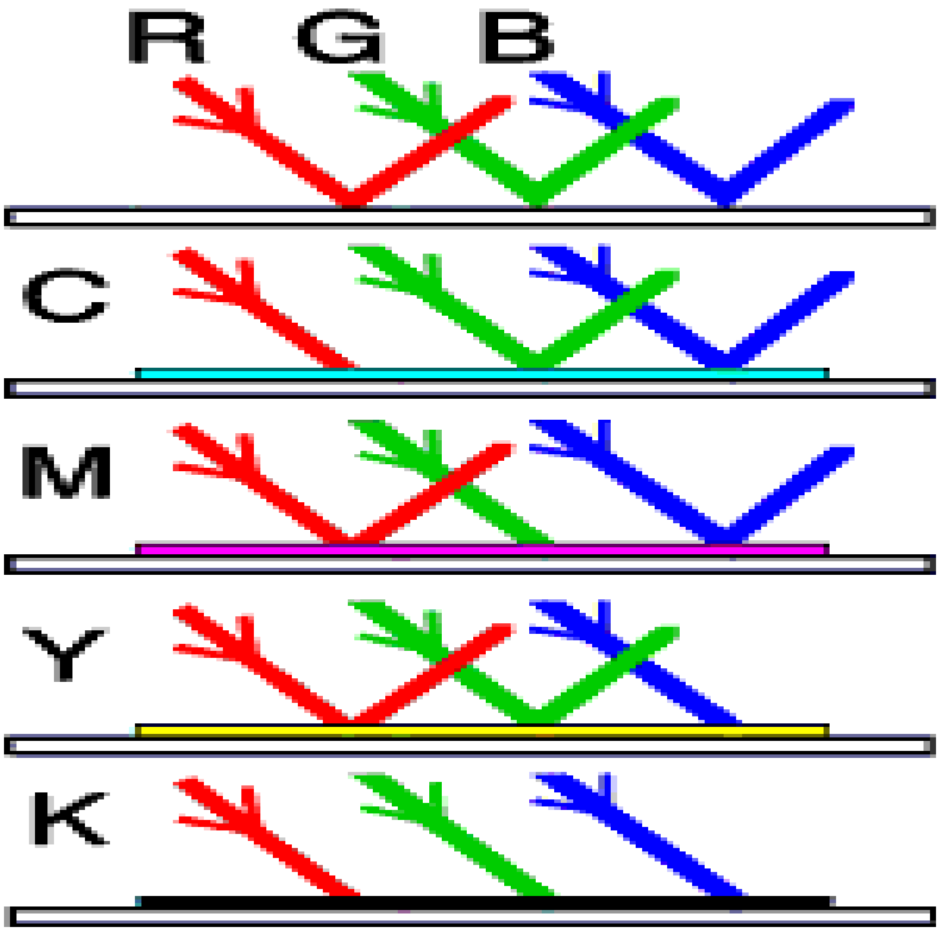


- R,G,B flashlight demo
- R,G,B color wheel
- R,G,B swatch demo

Subtractive primaries



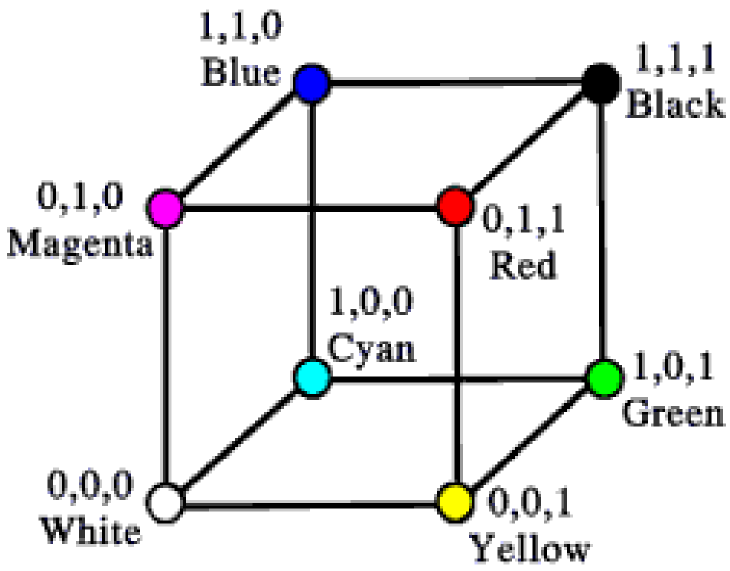
- C,M,Y color wheel
- C,M,Y sliders



- C,M,Y reflective surface

CMYK Color Space

- CMY(K) - Cyan, Magenta, Yellow, (Black)



  - C,M,Y color cube
  - Process printing oriented
  - Also not terribly intuitive beyond secondary colors

- C,M,Y/R,G,B slider demo, ideally:
  - C = RGB White - R
  - M = RGB White - G
  - Y = RGB White - B

But this is not true in practice.

Cyan = White-Red = **Blue + Green**

Magenta = White-Green = **Red + Blue**

Yellow = White – Blue = **Red+Green**

We can represent this as

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

The Conversion from RGB to CMY is

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}$$

Bits per pixel and color depth

The number of unique values a pixel can display is based upon the number of bits of memory reserved for each pixel.

- 1 bit/pixel = 2 values per pixel = B/W
- 2 bits/pixel = 4 values per pixel
- 3 bits/pixel = 8 values per pixel
- 8 bits/pixel = 256
- n bits/pixel = 2 to the nth power

- **Color can be added by mapping bits to color primaries**

A three bit (8 color) RGB scheme

0,0,0 = black

1,0,0 = red

1,1,0 = yellow

0,1,0 = green

0,1,1 = cyan

0,0,1 = blue

1,0,1 =magenta

1,1,1 = white

Perceived Attributes of Color

- Hue - the primary wavelength(s) of a color
- Lightness - the perceived luminance of a color
- Saturation - the *purity* or *vividness* of a color

HSV Color Space

- Computer scientists frequently use an intuitive color space that corresponds to tint, shade, and tone:
  - Hue - The color we see (red, green, purple)
  - Saturation - How far is the color from gray (pink is less saturated than red, sky blue is less saturated than royal blue)
  - Brightness (Luminance) - How bright is the color (how bright are the lights illuminating the object?)

HSV Color Space

- A more intuitive color space
  - H = Hue (Pure Cure)
  - S = Saturation (Purity of Colo
  - V = Value (or brightness)