SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | Approved by AICTE

COMPUTER GRAPHICS & MULTIMEDIA SYSTEMS

SCS1302

UNIT IV – PART I

Syllabus

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY FACULTY OF COMPUTING

SCS1302	COMPUTER GRAPHICS AND MULTIMEDIA SYSTEMS	L	T	P	Credits	Total Marks
		3	0	0	3	100

COURSE OBJECTIVES

- To gain knowledge to develop, design and implement two and three dimensional graphical structures.
- To enable students to acquire knowledge of Multimedia compression and animations.
- To learn creation, Management and Transmission of Multimedia objects.

UNIT 1 BASICS OF COMPUTER GRAPHICS 9 Hrs.

Output Primitives: Survey of computer graphics - Overview of graphics systems - Line drawing algorithm - Circle drawing algorithm - Curve drawing algorithm - Attributes of output primitives - Anti-aliasing.

UNIT 2 2D TRANSFORMATIONS AND VIEWING 8 Hrs.

Basic two dimensional transformations - Other transformations - 2D and 3D viewing - Line clipping - Polygon clipping - Logical classification - Input functions - Interactive picture construction techniques.

UNIT 3 3D CONCEPTS AND CURVES 10 Hrs.

3D object representation methods - B-REP , sweep representations, Three dimensional transformations. Curve generation - cubic splines, Beziers, blending of curves- other interpolation techniques, Displaying Curves and Surfaces, Shape description requirement, parametric function. Three dimensional concepts – Introduction - Fractals and self similarity- Successive refinement of curves, Koch curve and peano curves.

Syllabus

UNIT 4 METHODS AND MODELS **8 Hrs.**

Visible surface detection methods - Illumination models - Halftone patterns - Dithering techniques - Polygon rendering methods - Ray tracing methods - Color models and color applications.

UNIT 5 MULTIMEDIA BASICS AND TOOLS **10 Hrs.**

Introduction to multimedia - Compression & Decompression - Data & File Format standards - Digital voice and audio - Video image and animation. Introduction to Photoshop - Workplace - Tools - Navigating window - Importing and exporting images - Operations on Images - resize, crop, and rotate - Introduction to Flash - Elements of flash document - Drawing tools - Flash animations - Importing and exporting - Adding sounds - Publishing flash movies - Basic action scripts - GoTo, Play, Stop, Tell Target

Max. 45 Hours

TEXT / REFERENCE BOOKS

1. Donald Hearn, Pauline Baker M., "Computer Graphics", 2nd Edition, Prentice Hall, 1994.
2. Tay Vaughan ,"Multimedia", 5th Edition, Tata McGraw Hill, 2001.
3 Ze-Nian Li, Mark S. Drew ,"Fundamentals of Multimedia", Prentice Hall of India, 2004.
4 D. McClelland, L.U.Fuller ,"Photoshop CS2 Bible", Wiley Publishing, 2005.
5 James D. Foley, Andries van Dam, Steven K Feiner, John F. Hughes, "Computer Graphics Principles and Practice, 2nd
6 Edition in C, Audison Wesley, ISBN - 981 -235-974-5
7 William M. Newman, Roberet F. Sproull, " Principles of Interactive Computer Graphics", Second Edition, Tata McGraw-Hill Edition.

Course Objective(CO)

CO1: Construct lines and circles for the given input.

CO2: Apply 2D transformation techniques to transform the shapes to fit them as per the picture definition.

CO3: Construct splines, curves and perform 3D transformations

CO4: Apply colour and transformation techniques for various applications.

CO5: Analyse the fundamentals of animation, virtual reality, and underlying technologies.

CO6: Develop photo shop applications

UNIT - IV

Polygon Rendering Methods

Rendering of standard graphics objects-
those formed with polygon surfaces

Flat surface rendering

Method works when

- Polygon is one face of a polyhedron and is not a section of a curved-surface approximation mesh.
- In this method, a single intensity is calculated for each polygon.
- All points over the surface of the polygon are displayed with same intensity values.
- All light sources illuminating the polygon are far enough away from surface that N.L and attenuation function are both constant over area of poly (diffuse)
- Viewing position is far enough away from poly so that V . R is constant over area of poly (specular)

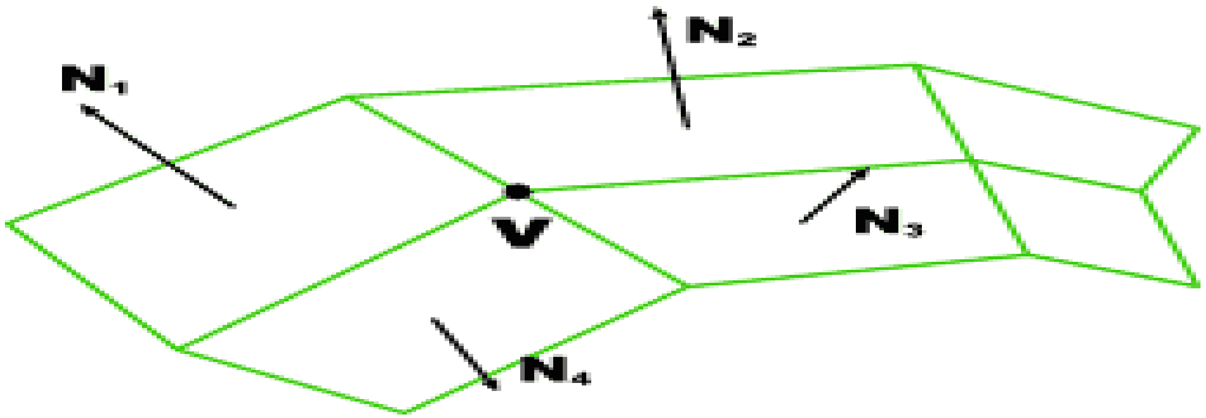
Flat surface rendering

Flat Shading Algorithm

```
For each visible polygon
    evaluate illumination model with polygon normal

    For each scanline
        For each pixel on scanline
            Fill with calculated intensity
```

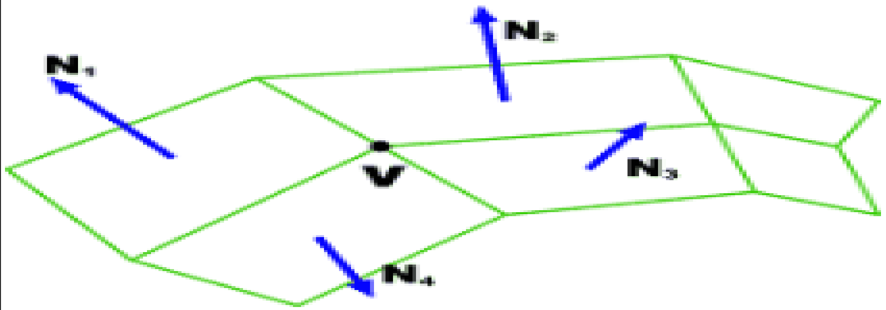
The normal vector at vertex V is calculated as the average of the surface normals for each polygon sharing that vertex



Flat surface rendering

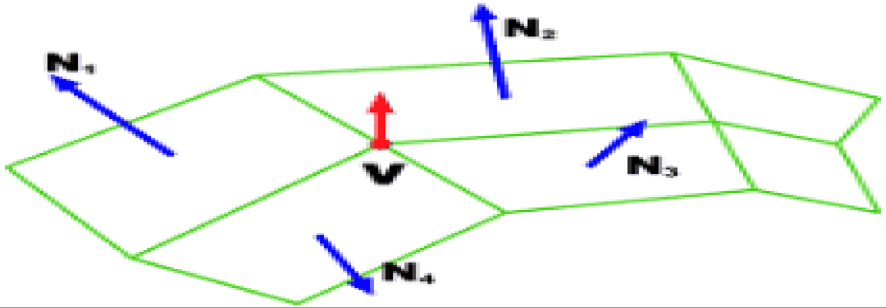
Surface Normals

The normal vector at vertex V is calculated as the average of the surface normals for each polygon sharing that vertex



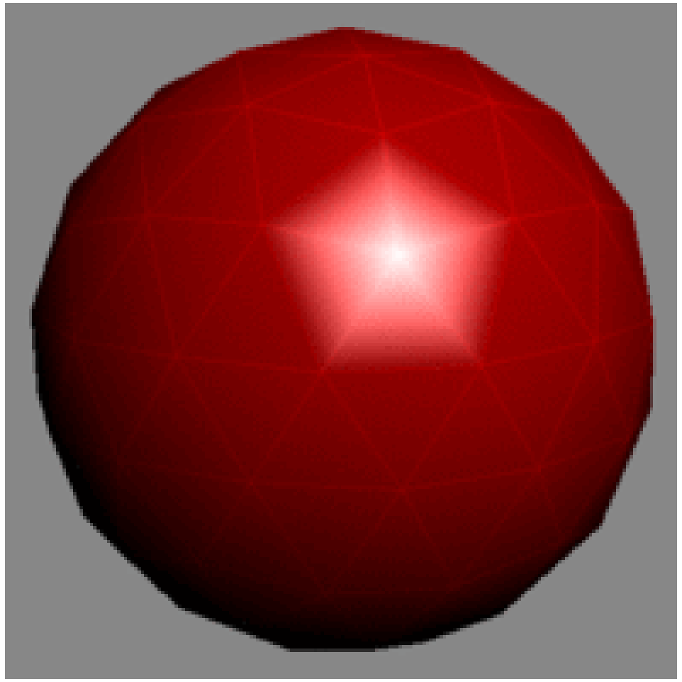
Vertex Normals

The normal vector at vertex V is calculated as the average of the surface normals for each polygon sharing that vertex



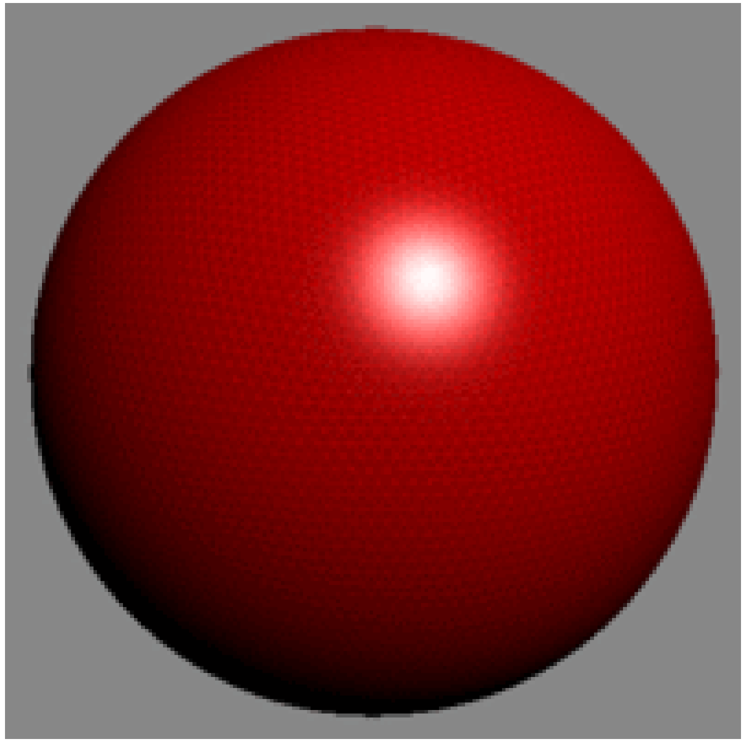
Flat surface rendering

- One problem with flat surface rendering is that discontinuities arise on boundaries of polygons.



Gouraud Shading

- A better algorithm, though still not perfect, is due to Henri Gouraud



Gouraud Shading

Gouraud Algorithm

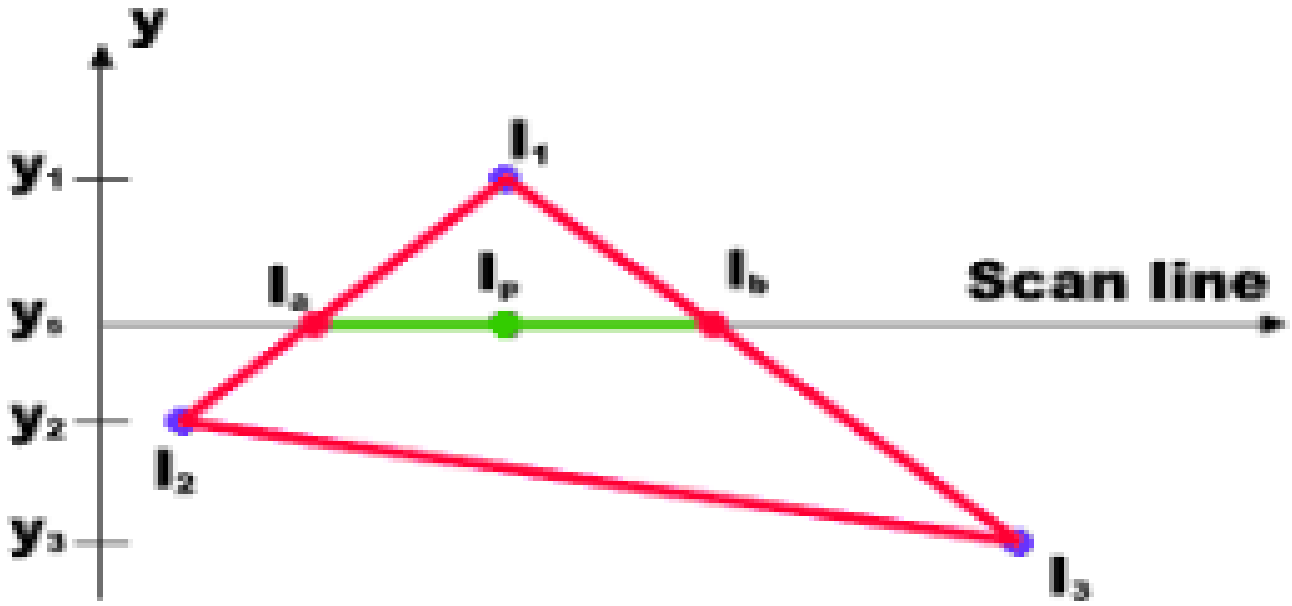
```
For each visible polygon
    evaluate illumination model at vertices
       using vertex normals

    For each scanline
        calculate intensity at edge intersections
            (span extrema) by linear interpolation

        For each pixel on scanline
            calculate intensity by interpolation of
                intensity at span extrema

(like scan conversion with vertex colors)
```

Gouraud Calculations



$I_a = I_1 - (I_1 - I_2)\,(y_1 - y_s)/(y_1 - y_2)$

$I_b = I_1 - (I_1 - I_3)\,(y_1 - y_s)/(y_1 - y_3)$

$I_p = I_b - (I_b - I_a)\,(x_b - x_p)/(x_b - x_a)$

---

(1) calculate intensity at vertices (I_1, I_2, I_3) from illumination model.

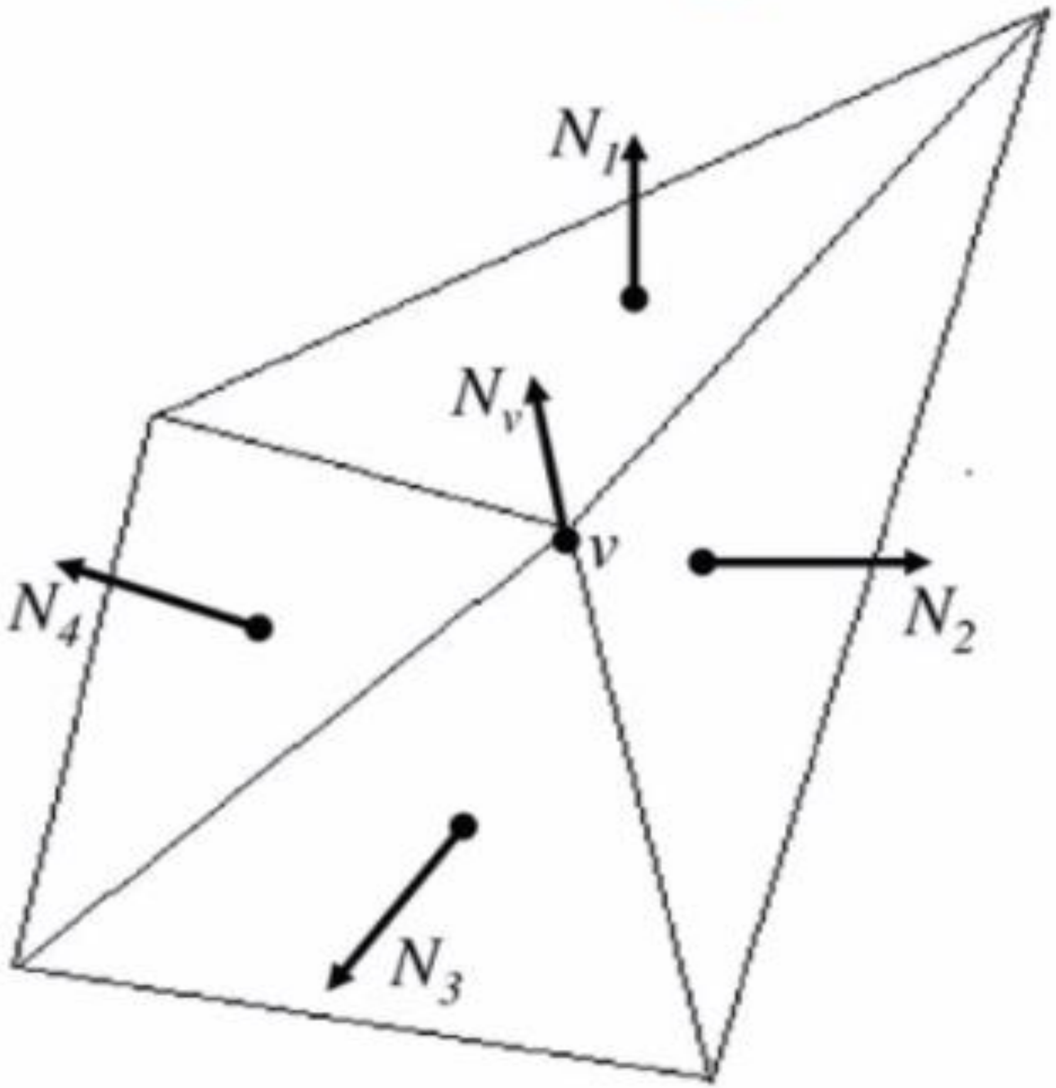
(2) interpolate vertex intensities along edges (I_a, I_b)

(3) interpolate intensities at span extrema to pixels (I_p)

Gouraud (Interploated) Shading

- Idea is to calculate intensity values at polygon vertices then, linearly interpolate these intensities across polygon faces of a lighted object.
- Eliminates discontinuities that occur with flat shading.
- Determine average unit normal vector at each vertex of poly. For a vertex v that is the meeting point for n polygons, each with normal N_k.

Determine the average unit normal vector at each vertex of the polygon



- The average unit normal vector at v is given as:

$$N_v = \frac{N_1 + N_2 + N_3 + N_4}{\left|N_1 + N_2 + N_3 + N_4\right|}$$

- or more generally:

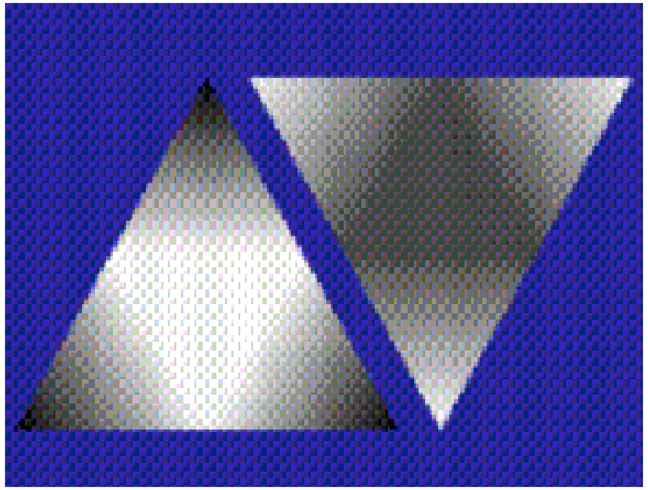
$$N_v = \frac{\sum_{i=1}^{n} N_i}{\left|\sum_{i=1}^{n} N_i\right|}$$

Gouraud Shading

- **Procedure**
- Determine average unit normal vector at each vertex of poly.
- Apply illumination model at each vertex to obtain the light intensity at that position
- Linearly interpolate the vertex intensities using scan lines over projected area of polygon.

Gouraud Shading

- Similar calculations are used to obtain intensities at successive horizontal pixel positions along each scan line.
- **Drawbacks:**
- Highlights displayed with anonymous shapes
- Linear intensity interpolation can cause bright or dark intensity streaks called **Mach bands.**

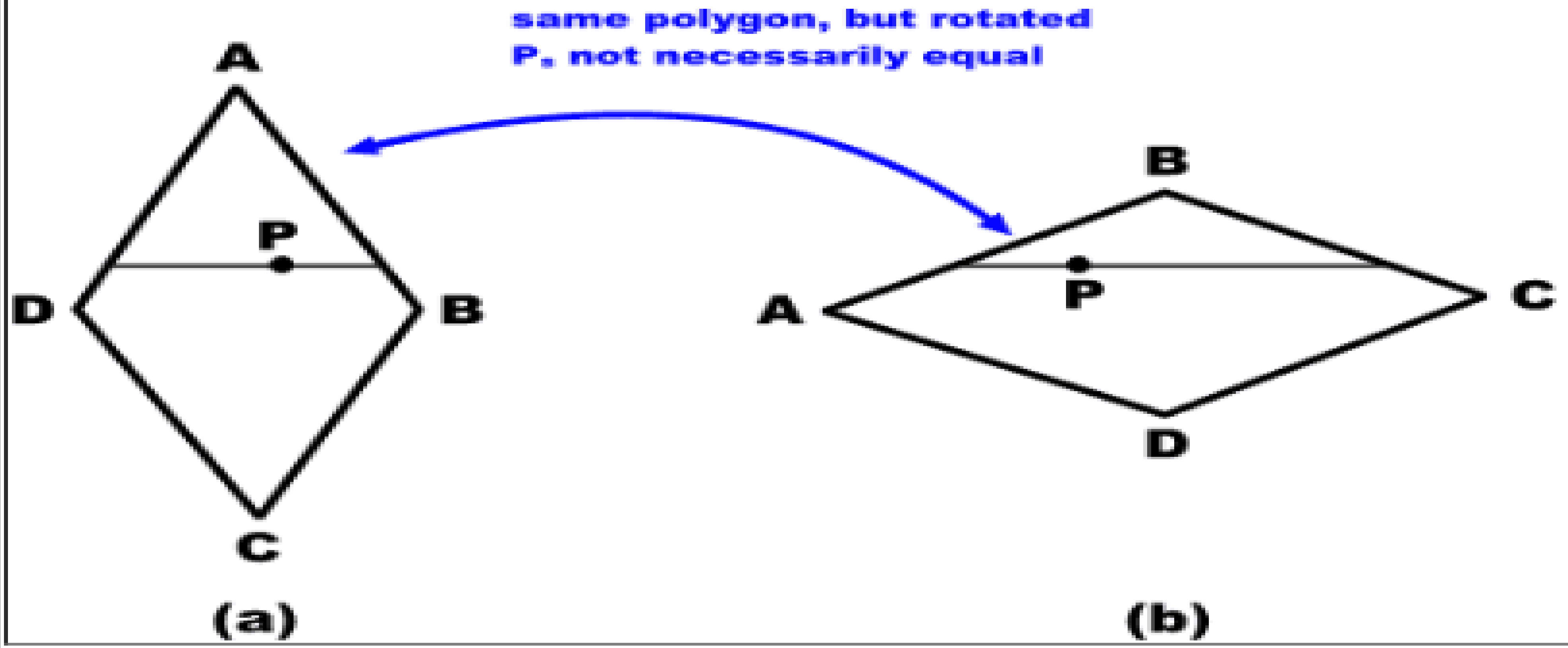


Problems with Interpolated Shading

(1) Polygon silhouette - edge still polygonal

(2) Perspective distortion - interpolation in screen space, rather than object space

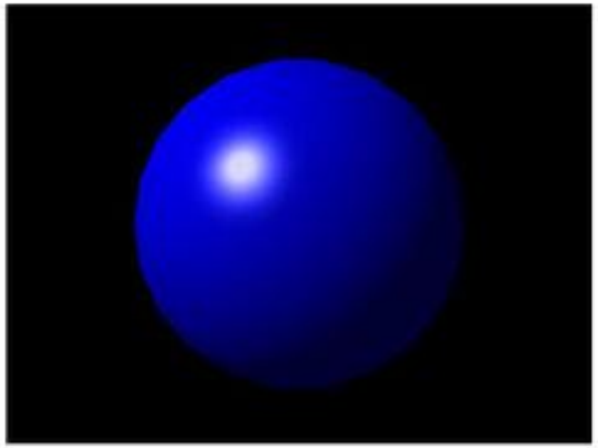
(3) Orientation dependence



Phong Shading

- Idea here is to interpolate the normal vectors instead of the light intensity.
- Provides more accurate calculation of light intensity values and more realistic surface highlights.

FLAT SHADING



PHONG SHADING

By Phong Bui Tong

- **Phong shading:**
- Handles specular highlights much better.
- Does a better job in handling Mach bands.
- But much more expensive than Gouraud shading.
- Compute color per pixel.
- Good for curved and shiny surfaces

Phong Shading

- **Procedure:**
- Find the average unit normal vector at each polygon vertex.
- Linearly interpolate the vertex normals over the surface of the polygon.
- Apply an illumination model along each scan line to calculate projected pixel intensities for the surface points.

Fast Phong shading

- It is an enhanced Phong shading technique.
- It approximates the intensity calculations using Taylor-series expansion and Triangular surface patches.

Comparison: Flat, Gouraud and Phong

Polygon Rendering (Shading)

Shading

Flat shading

- Entire surface (polygon) has one colour
- Cheapest to compute, and least accurate (so you need a dense triangulation for decent-looking results)
- OpenGL – glShadeModel(GL_FLAT)

Phong shading

- Compute illumination for every pixel during scan conversion
- Interpolate normals at each pixel too
- Expensive, but more accurate
- Not supported in OpenGL (directly)

Gouraud shading

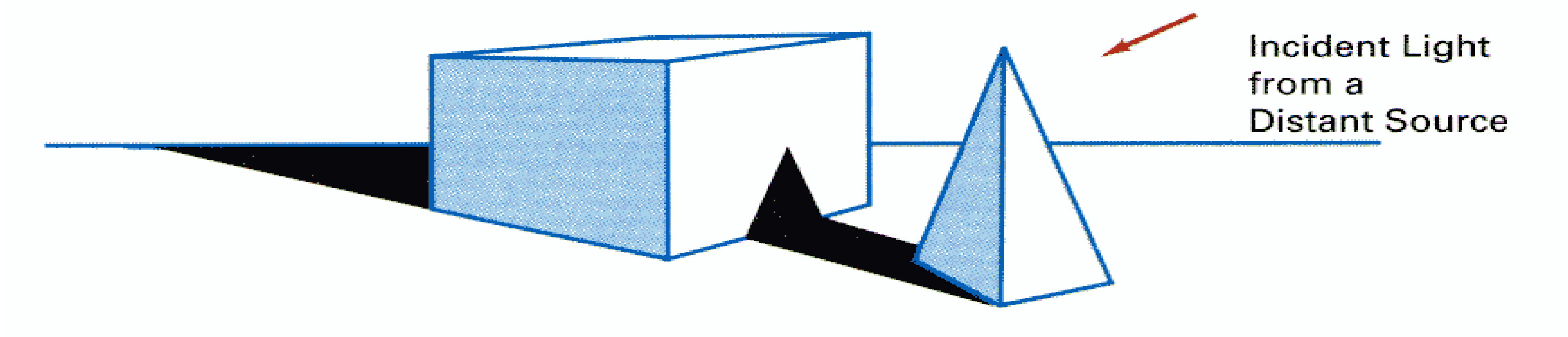
- Just compute illumination at vertices
- Interpolate vertex colours across polygon pixels
- Cheaper, but less accurate (spreads highlights)
- OpenGL - glShadeModel(GL_SMOOTH)

Phong illumination

- Don't confuse shading and illumination!
- Shading describes how to apply an illumination model to a polygonal surface patch
- All these shading methods could use Phong illumination (ambi diffuse, and specular) or any other local illumination model

Other Effects: *Shadow*

- Shadow can help to create realism. Without it, a cup, eg., on a table may look as if the cup is floating in the air above the table.
- By applying hidden-surface methods with pretending that the position of a light source is the viewing position, we can find which surface sections cannot be "seen" from the light source => shadow areas.
- We usually display shadow areas with ambient-light intensity only.

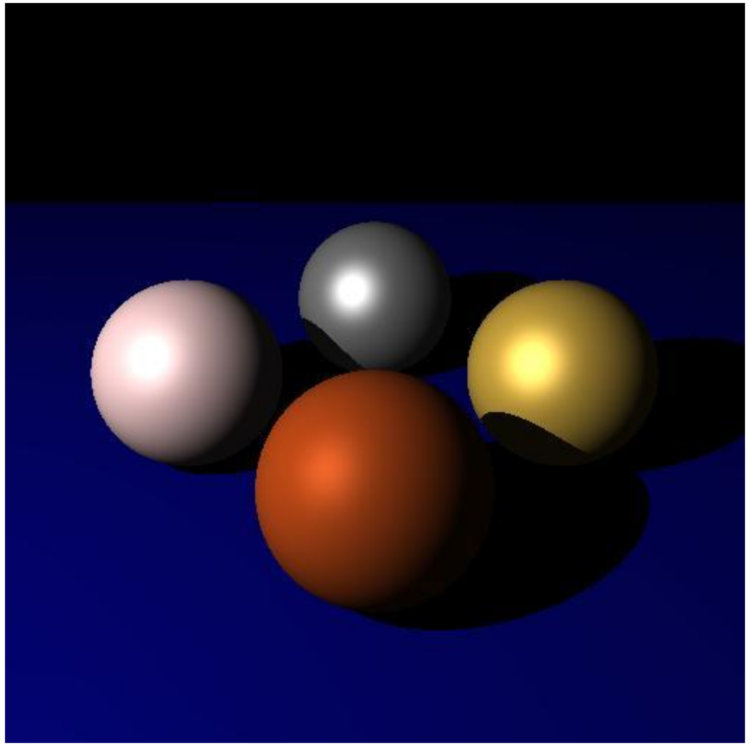


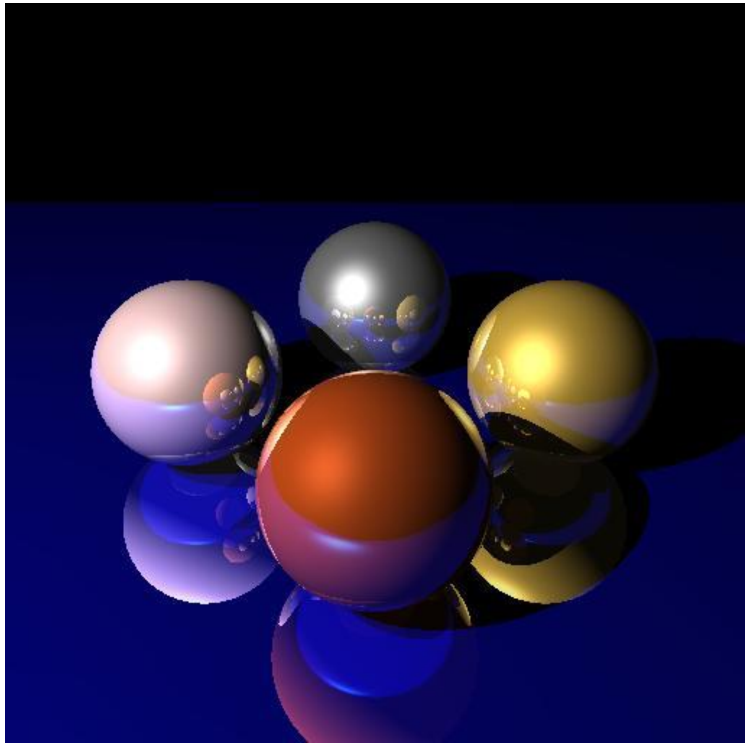
Ray-Tracing Methods

- By using the **Ray-casting,** we can have a simple and powerful rendering technique for obtaining **global reflection** and **transmission effects.**
- **Refer the link for Ray Tracing notes:**

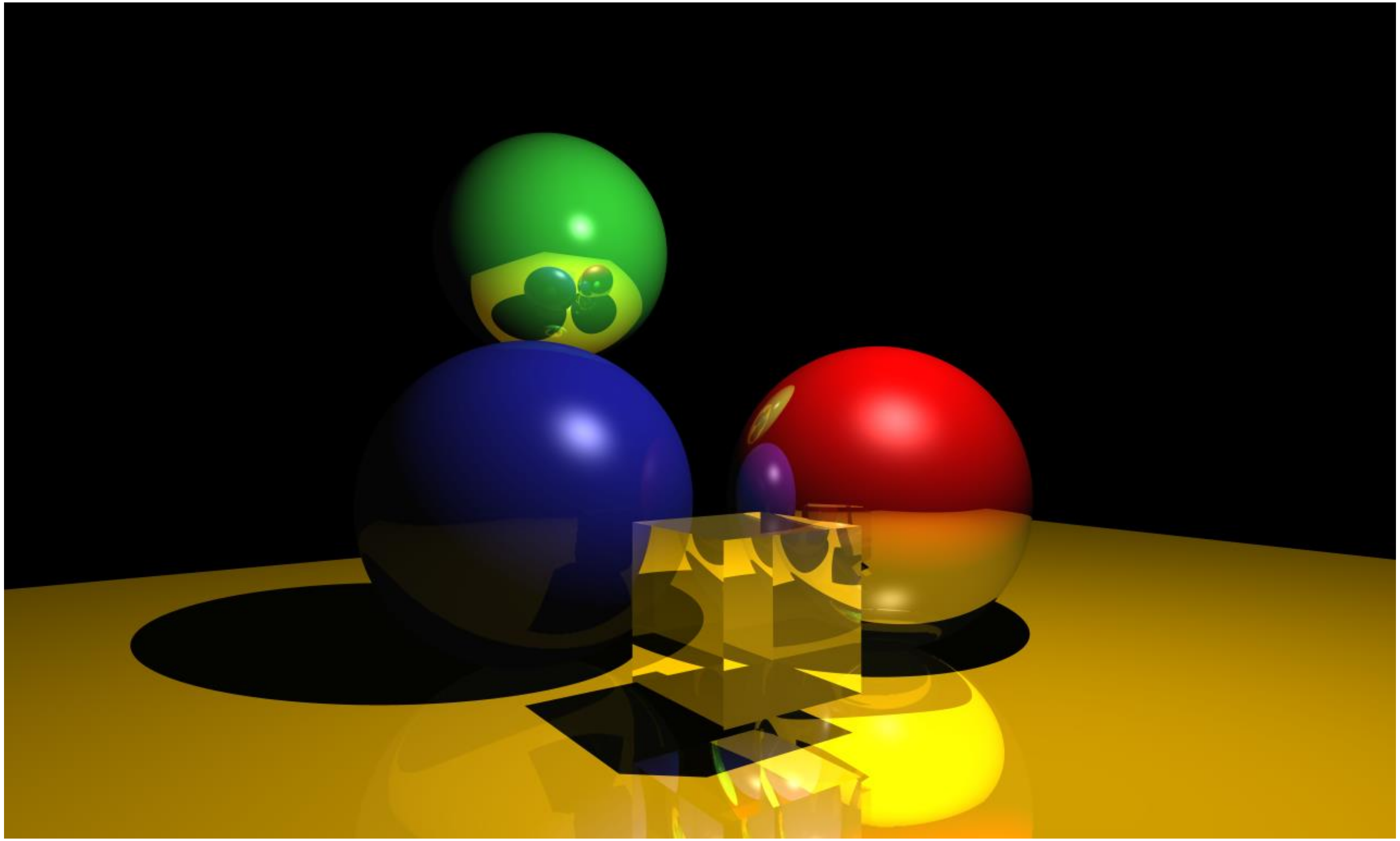
  http://math.hws.edu/graphicsbook/c8/s1.html

Photo-Realism

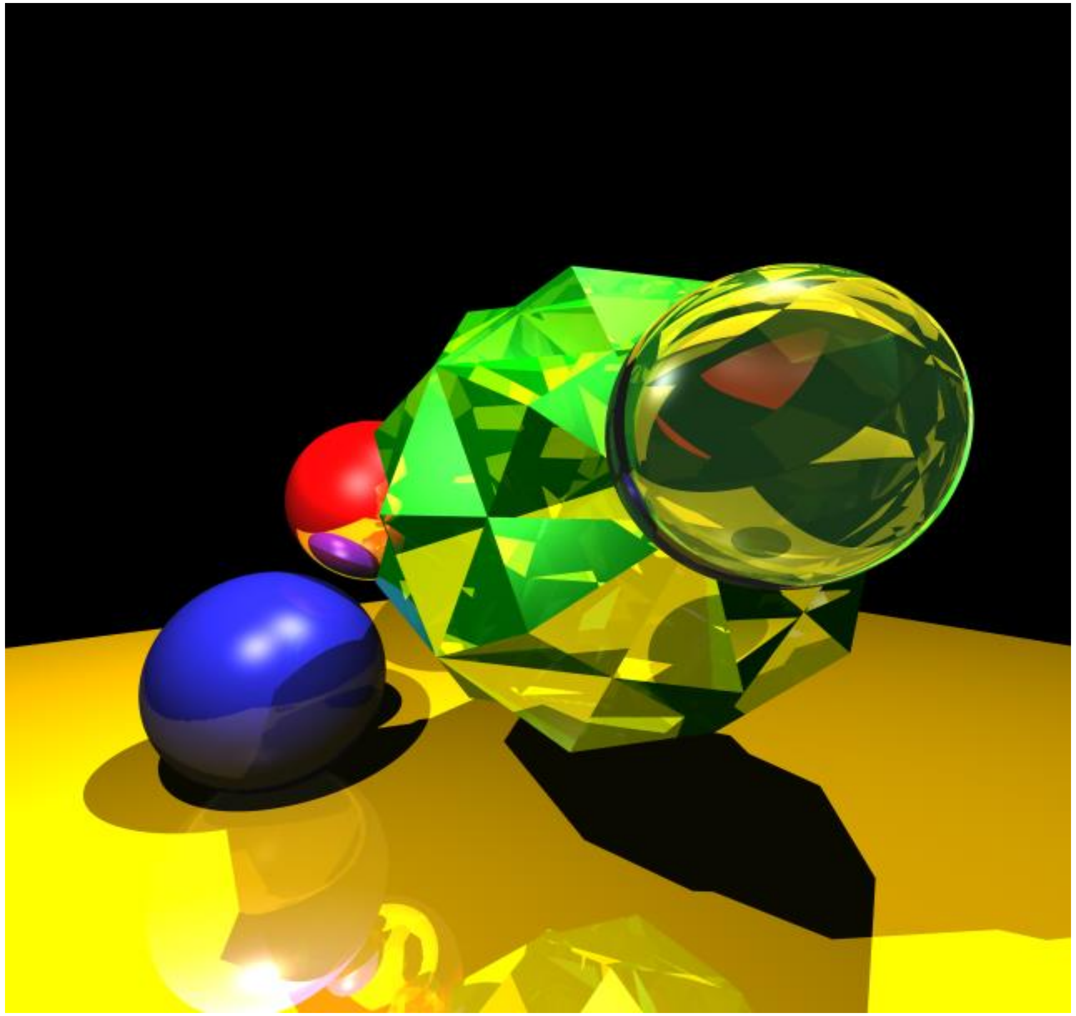


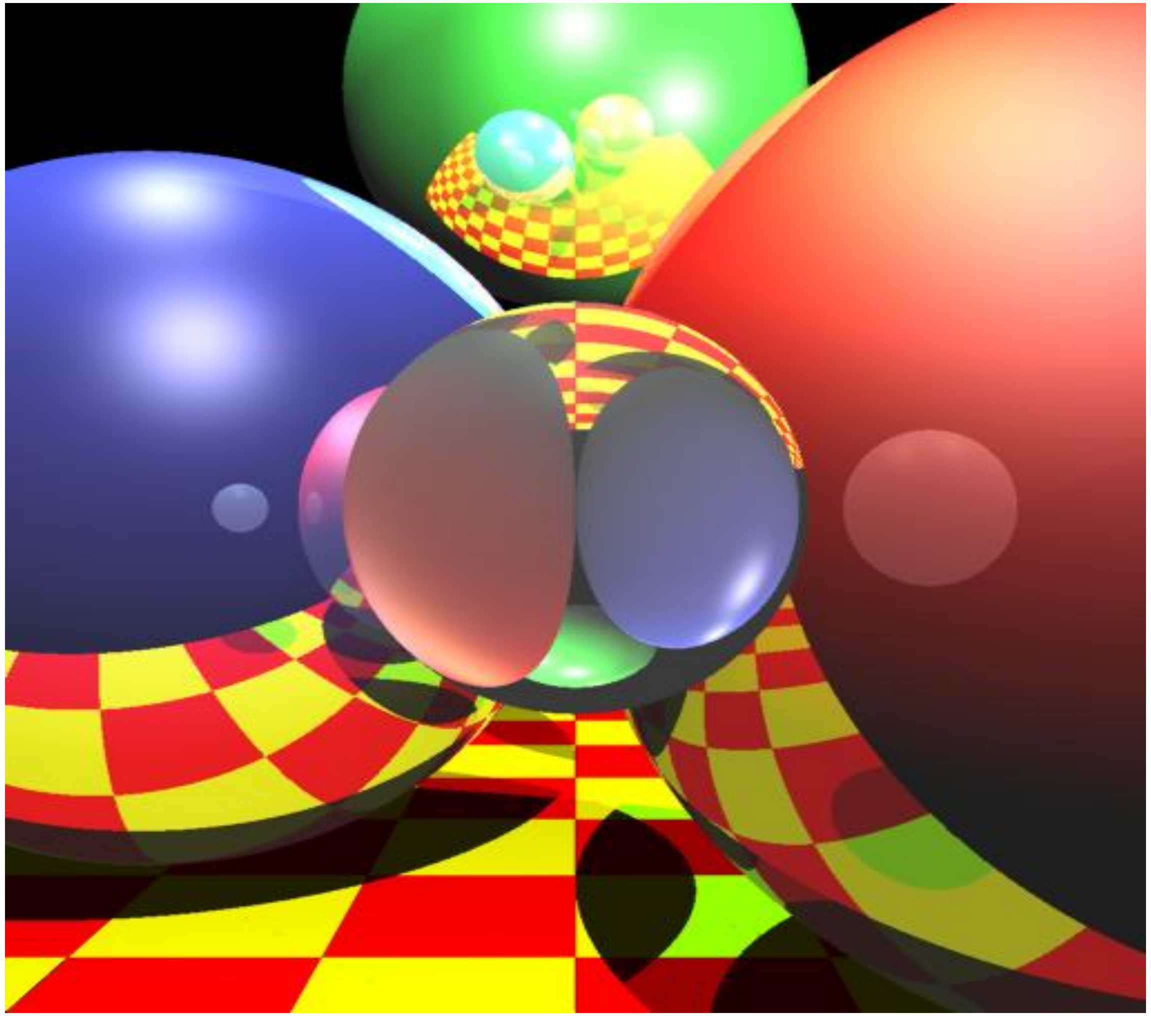


Created by David Derman – CISC 440

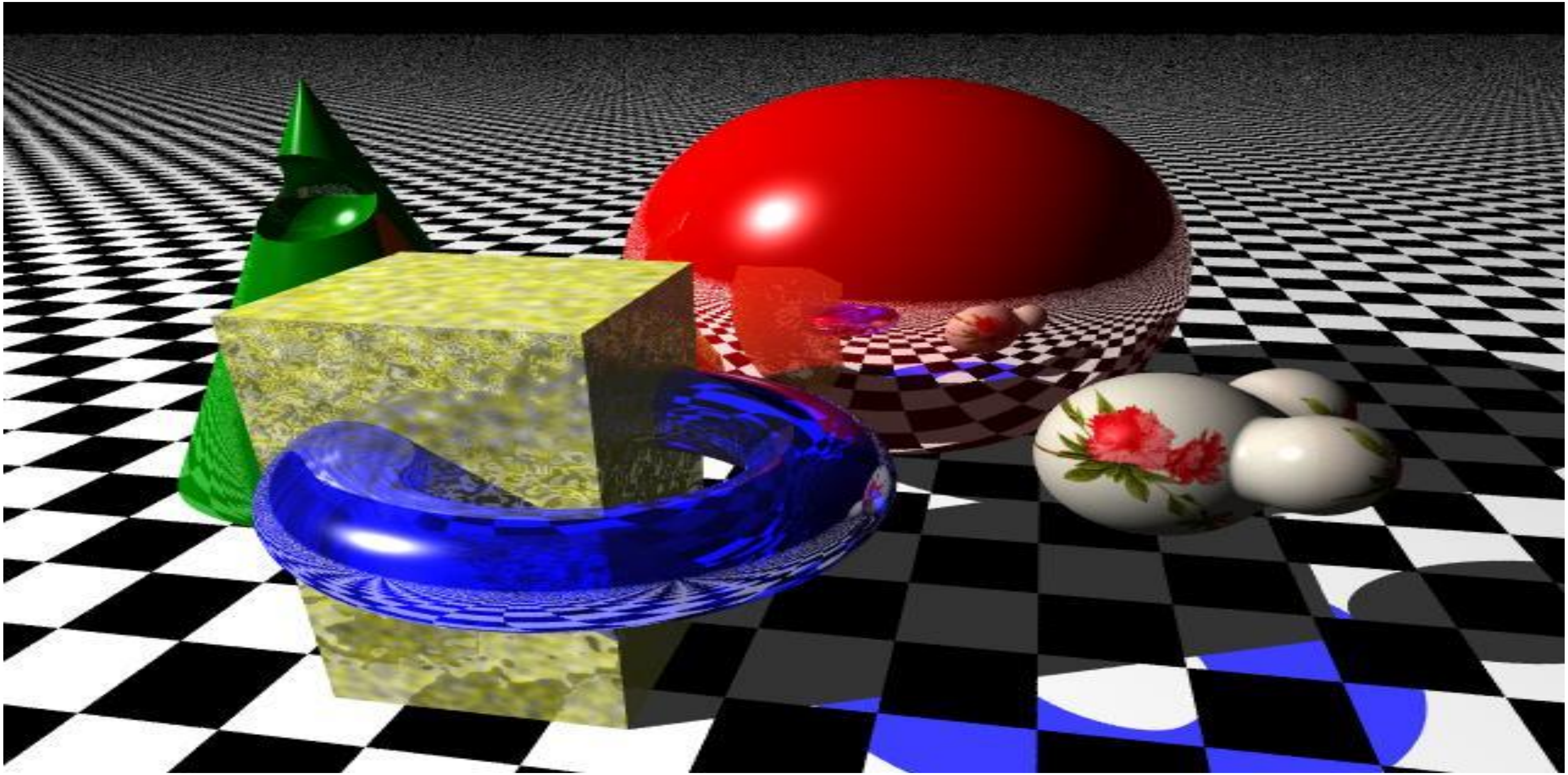


Created by Jan Oberlaender – CISC 640





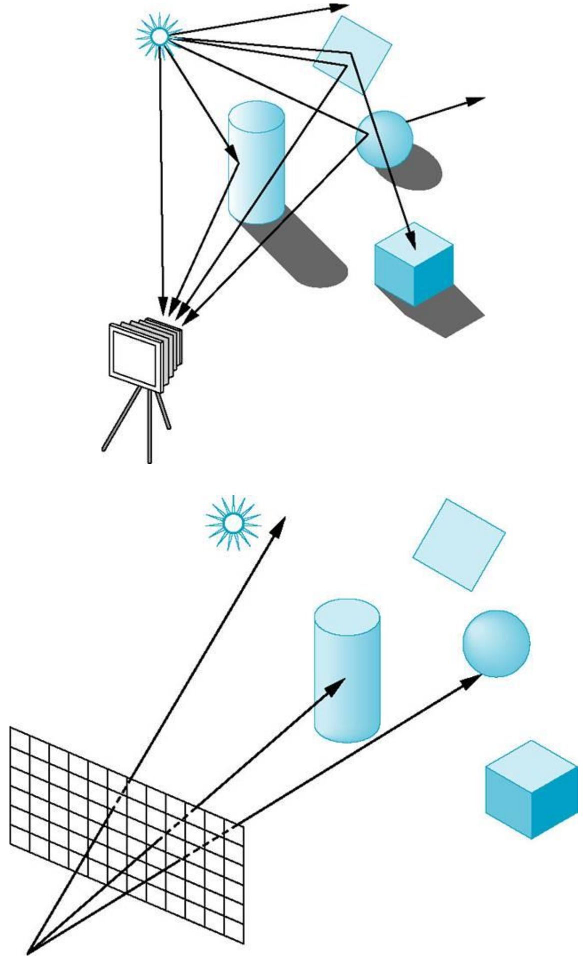
Created by Jan Oberlaender – CISC 640



Created by Donald Hyatt
http://www.tjhsst.edu/~dhyatt/superap/povray.html

- In ray tracing, a ray of light is traced in a backwards direction.
  - We start from the eye or camera and trace the ray through a pixel in the image plane into the scene and determine what it intersects
  - The pixel is then set to the color values returned by the ray.
  - If the ray misses all objects, then that pixel is shaded the background color

Overview



Courtesy: Angel

- Forward Ray tracing
  - Rays from light source bounce of objects before reaching the camera
  - Computational wastage
- Backward Ray tracing
  - Track only those rays that finally made it to the camera

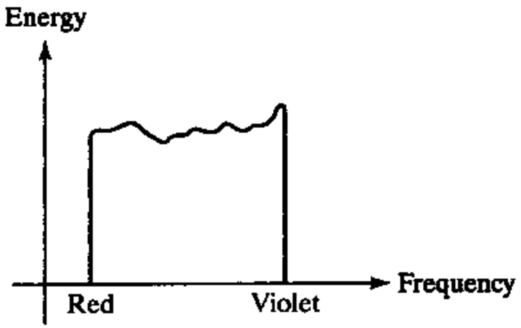
Color Models

COLOR MODEL

- Color Model- Method for explaining the properties or behavior of color within some particular context.
- No single model can explain all aspects of color.

Physical properties of light

- All kinds of light can be described by the energy of each wavelength
- The distribution showing the relation between energy and wavelength (or frequency) is called *energy spectrum*



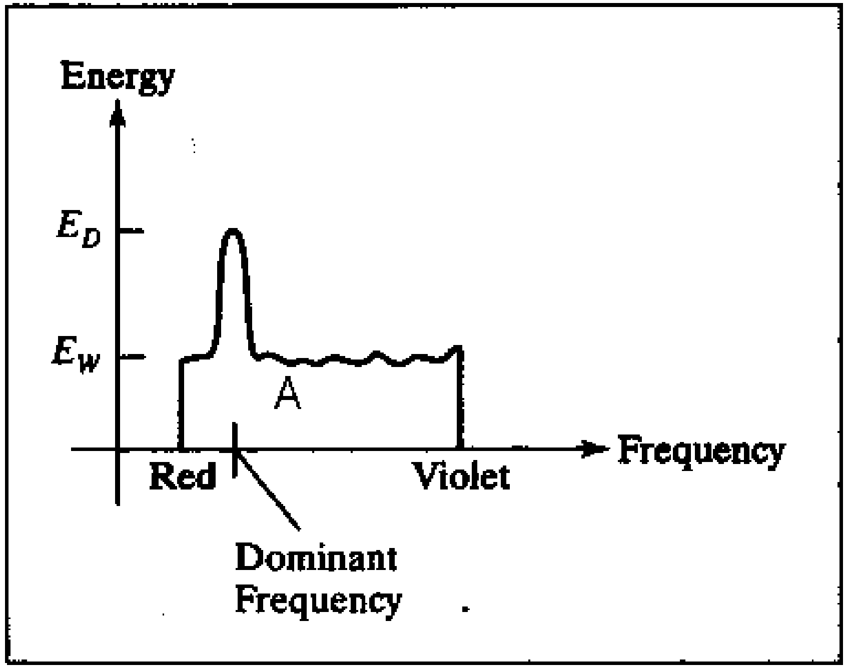
Physical properties of light

This distribution may indicate:

1) a <u>dominant wavelength</u> (or frequency) which is the color of the light (*hue*), cp. E_D
2) <u>brightness</u> (luminance), intensity of the light (*value*), cp. the area A
3) <u>purity</u> How pure the color of the light appears.

Physical properties of light

Energy spectrum for a light source with a dominant frequency near the red color

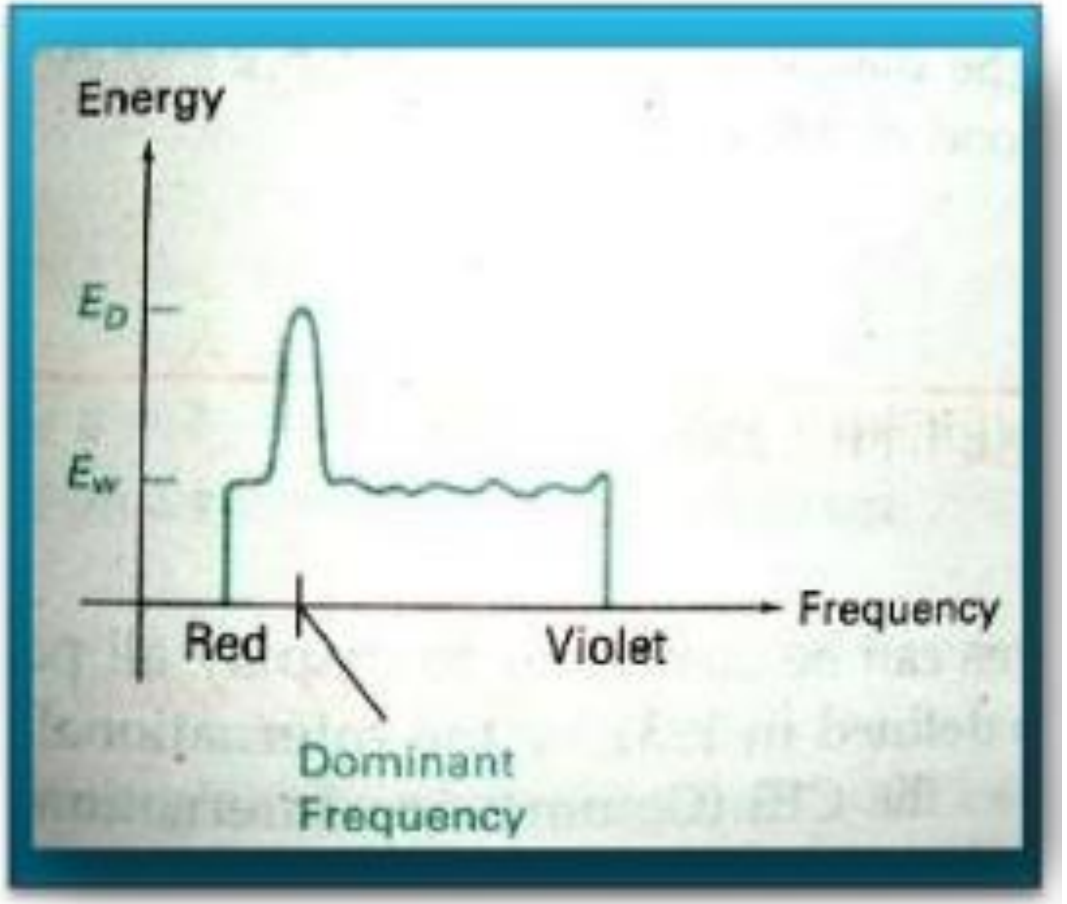


HUE COLOUR, DOMINENT FREQUENCY

BRIGHTNESS PERCIEVED INTENSITY OF LIGHT

SATURATION THE WASHING OUT EFFECT

E_D = DOMINENT LIGHT. ENERGY DENSITY
E_W = ENERGY DENSITY OF WHITE LIGHT



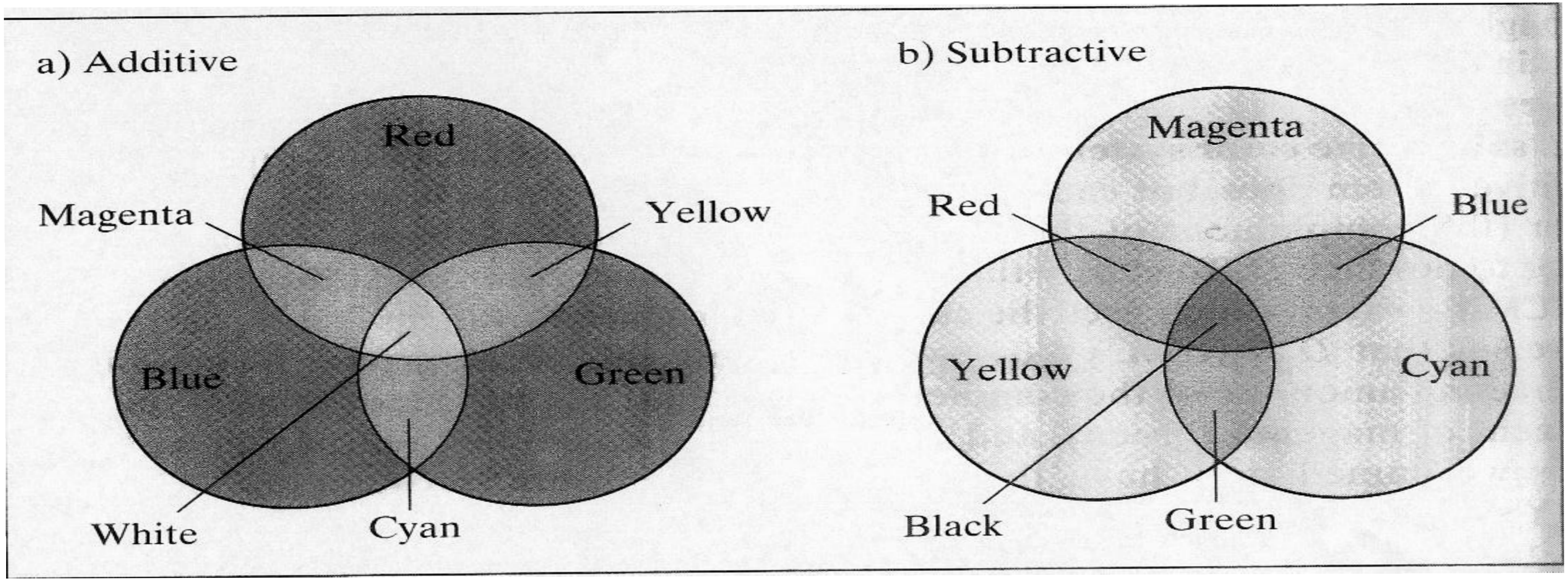
AREA UNDER THE CURVE -> BRIGHTNESS
MORE THE VALUE OF (E_D-E_W) PURER IS THE COLOUR

Color definitions

- Complementary colors - two colors combine to produce white light
- Primary colors - (two or) three colors used for describing other colors
- A color model is an orderly system for creating a whole range of colors from a small set of primary colors.
- There are two types of color models, those that are subtractive and those that are additive.
- Additive color models use light to display color while subtractive models use printing inks.
- Colors perceived in additive models are the result of transmitted light.
- Colors perceived in subtractive models are the result of reflected light.

Additive mixing

- pure colors are put close to each other => a mix on the retina of the human eye (cp. RGB)
- overlapping gives yellow, cyan, magenta and white
- the typical technique on color displays



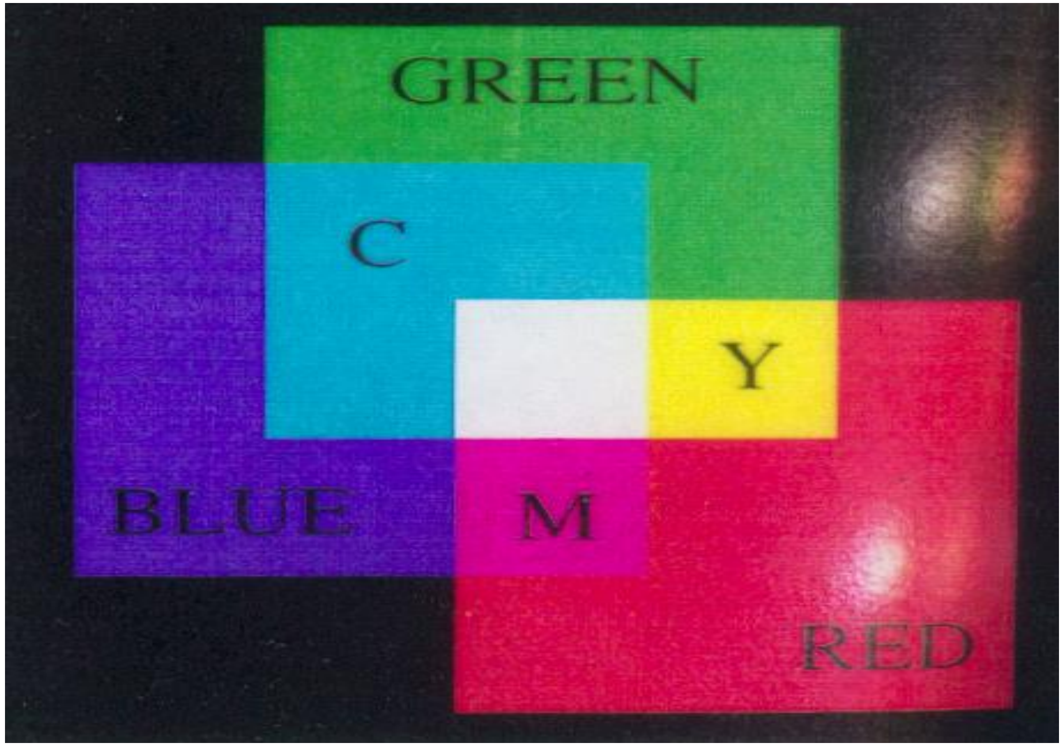
Subtractive mixing

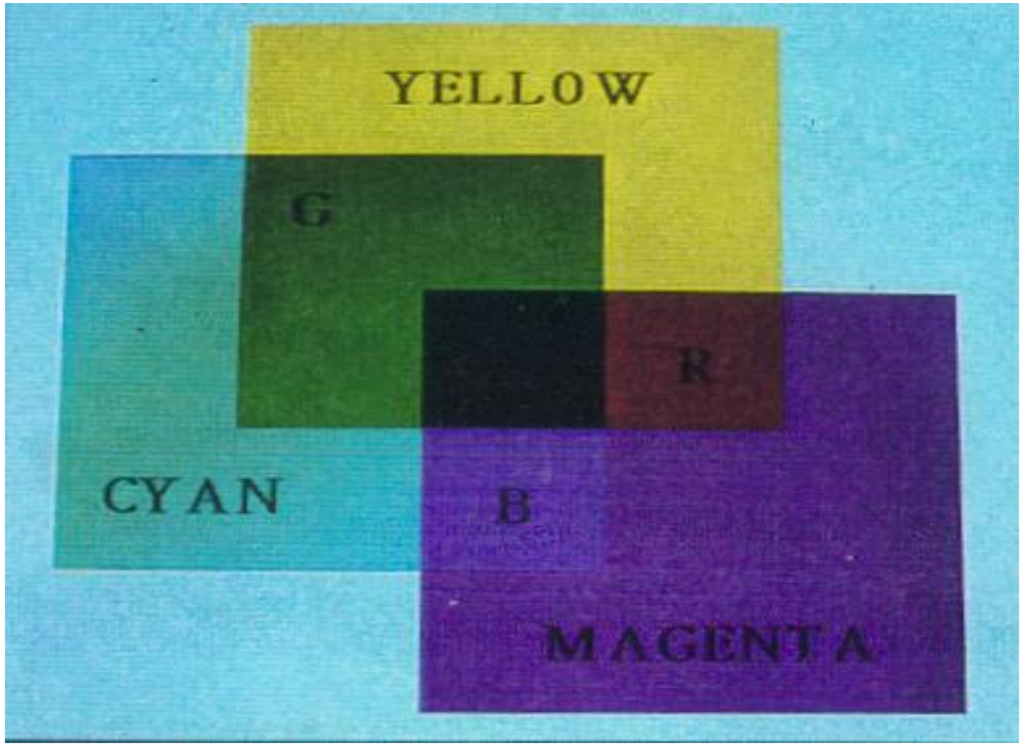
- color pigments are mixed directly in some liquid, e.g. ink
- each color in the mixture absorbs its specific part of the incident light
- the color of the mixture is determined by subtraction of colored light, e.g. yellow absorbs blue => only red and green, i.e. yellow, will reach the eye (yellow because of addition)

Subtractive mixing,cont’d

- primary colors: cyan, magenta and yellow, i.e. CMY
- the typical technique in printers/plotters
- connection between additive and subtractive primary colors (cp. the color models RGB and CMY)

Additive/subtractive mixing





C (cyan) = G + B = W - R
M (magenta) = R + B = W - G
Y (yellow) = R + G = W – B

Overview of color models

The human eye can perceive about 382000(!) different colors

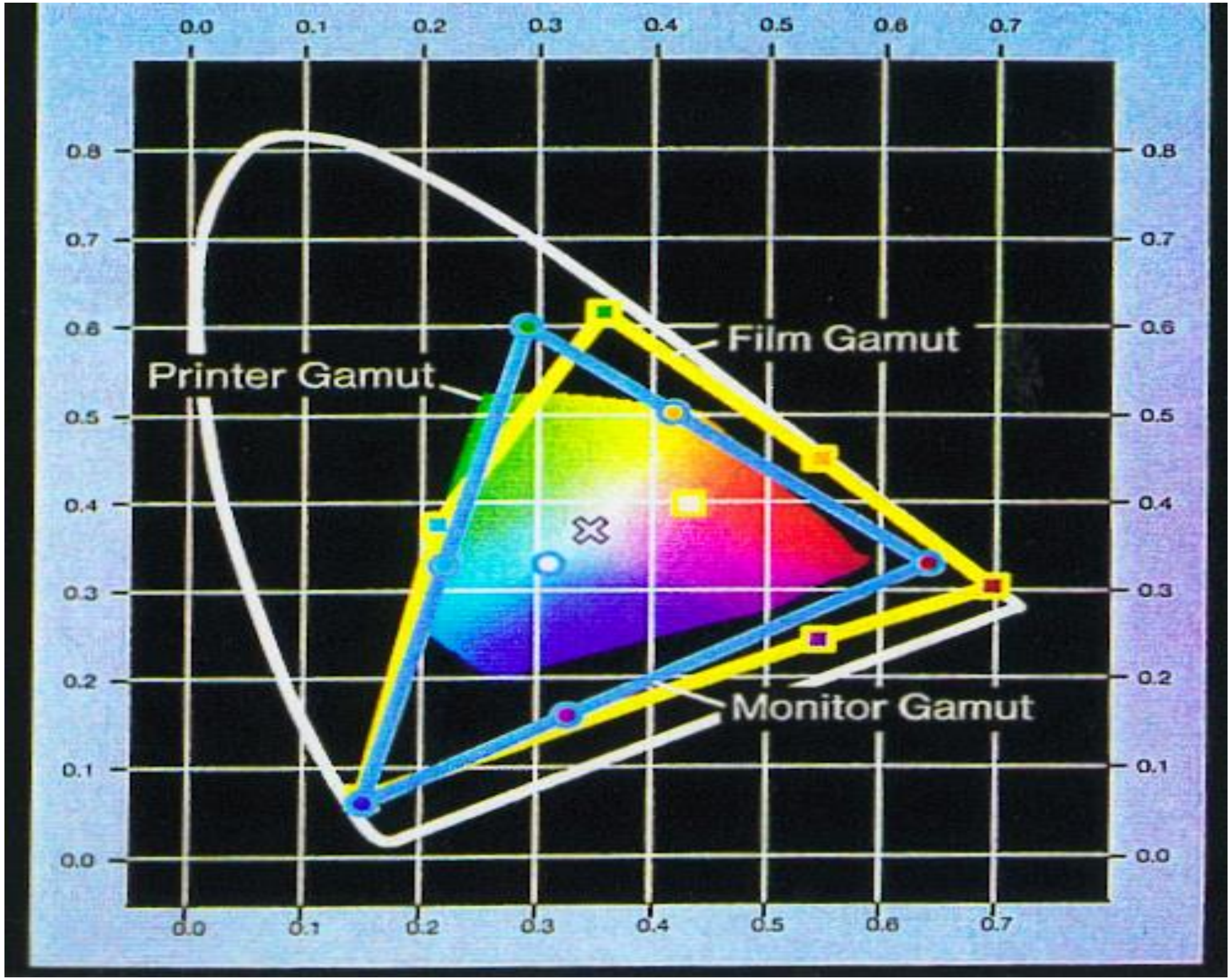
Necessary with some kind of classification sys-tem; all using three coordinates as a basis:

1) CIE standard
2) RGB color model
3) CMY color model (also, CMYK)
4) HSV color model
5) HLS color model

CIE standard

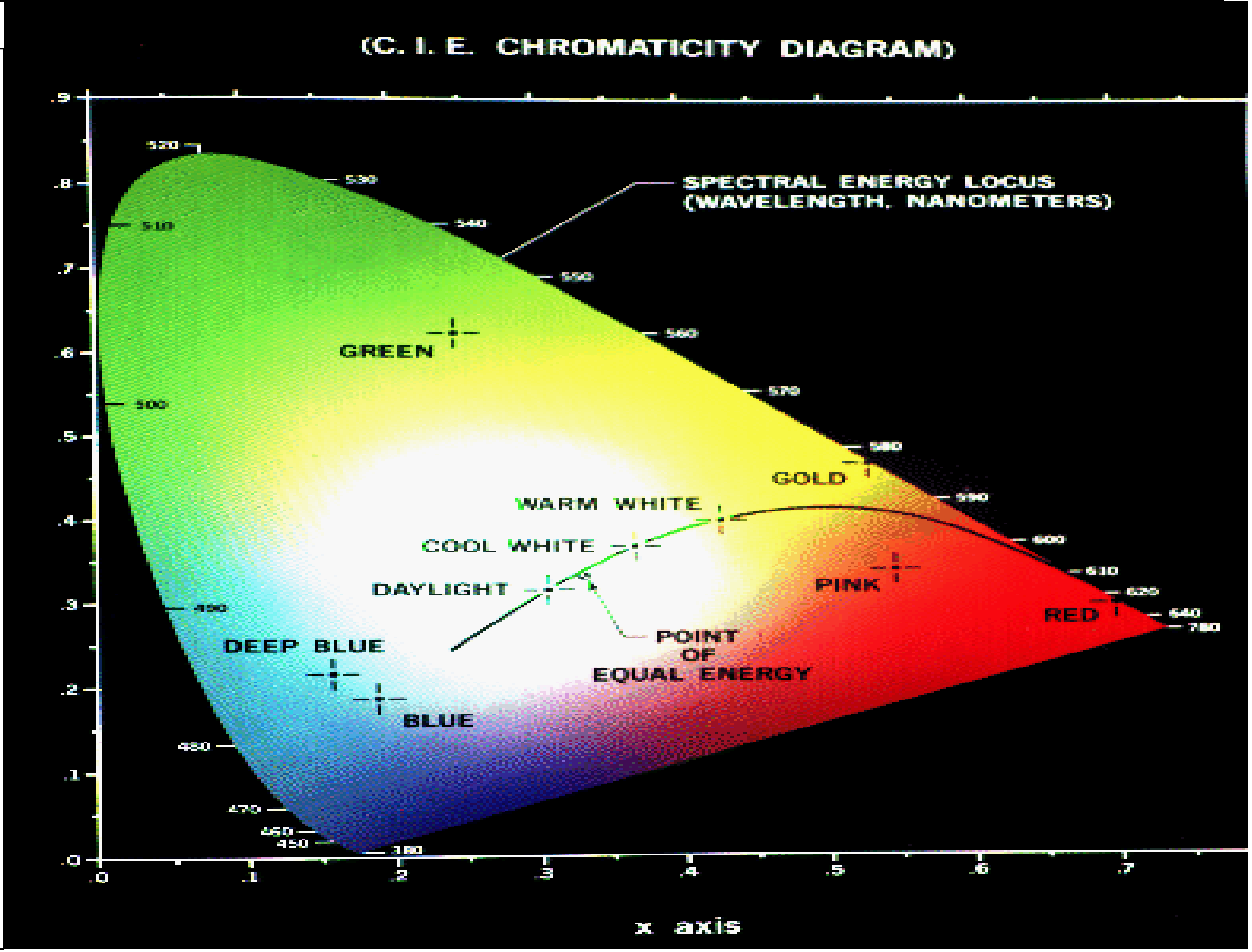
Commission Internationale de L'Eclairage (1931)

- not a computer model
- each color = a weighted sum of three imaginary primary colors



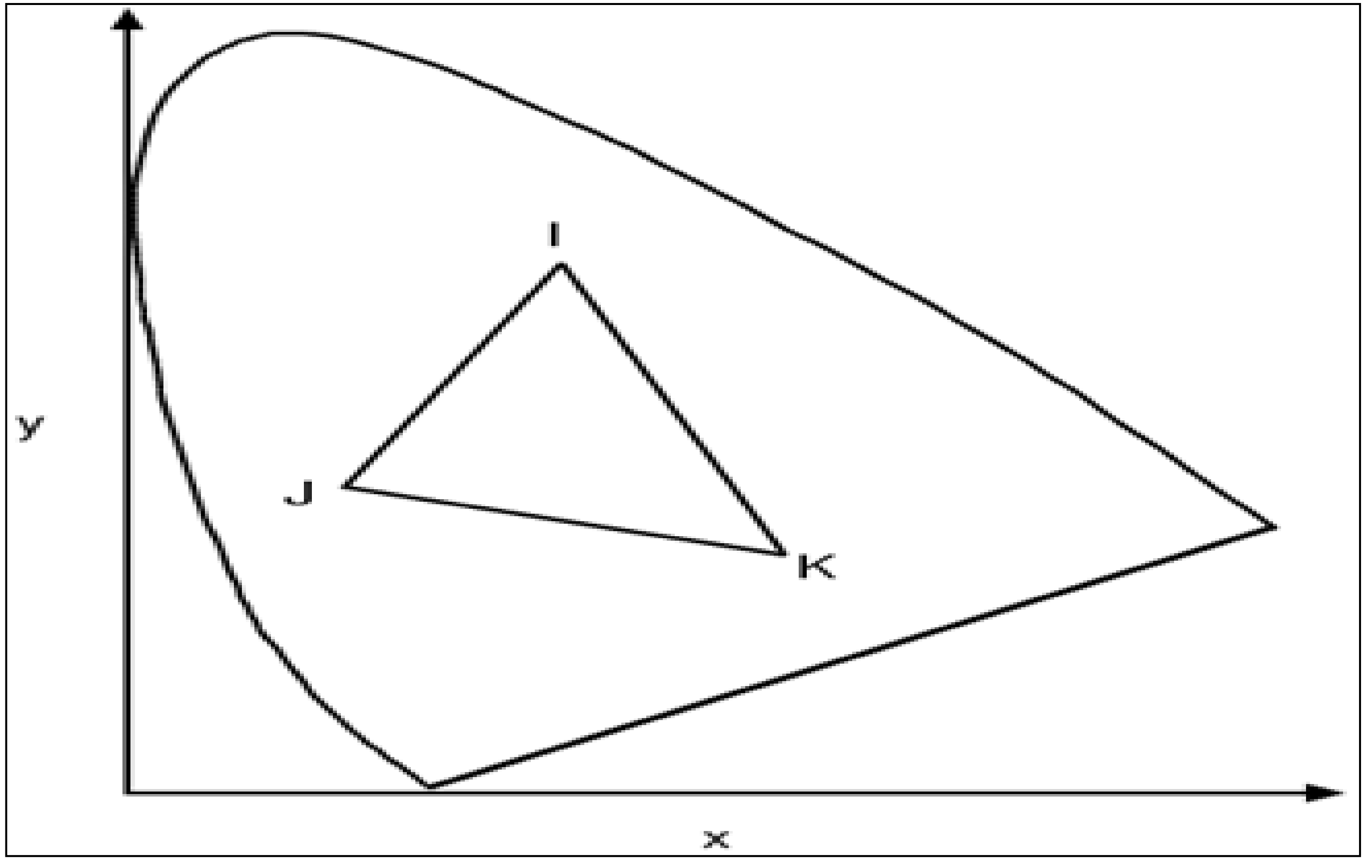
CIE primaries : Chromaticity Diagram

- The tristimulus theory of colour perception seems to imply that any colour can be obtained from a mix of the three primaries, red, green and blue, but although nearly all visible colours can be matched in this way, some cannot. However, if one of the primaries is added to one of these unmatchable colours, it can be matched by a mixture of the other two, and so the colour may be considered to have a negative weighting of that particular primary.
- In 1931, the Commission Internationale de l'Éclairage (CIE) defined three standard primaries, called X, Y and Z, that can be added to form all visible colours. The primary Y was chosen so that its colour matching function exactly matches the luminous-efficiency function for the human eye, given by the sum of the three curves in the below figure.



CIE primaries : Chromaticity Diagram

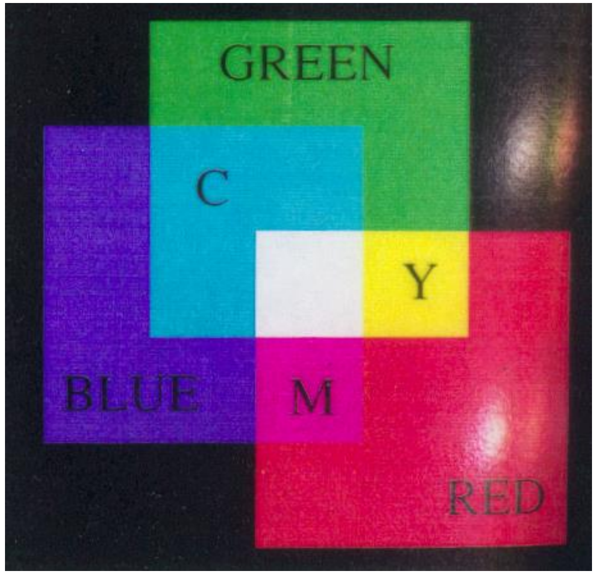
- The CIE Chromaticity Diagram showing all visible colours. x and y are the normalised amounts of the X and Y primaries present, and hence z = 1 - x - y gives the amount of the Z primary required.
- The CIE Chromaticity Diagram shows all visible colours. The x and y axis give the normalised amounts of the X and Y primaries for a particular colour, and hence z = 1 - x - y gives the amount of the Z primary required. Chromaticity depends on dominant wavelength and saturation, and is independent of luminous energy. Colours with the same chromaticity, but different luminance all map to the same point within this region.
- The pure colours of the spectrum lie on the curved part of the boundary, and a standard white light has colour defined to be near (but not at) the point of equal energy x = y = z = 1/3. Complementary colours, i.e. colours that add to give white, lie on the endpoints of a line through this point.
- As illustrated in below figure, all the colours along any line in the chromaticity diagram may be obtained by mixing the colours on the end points of the line. Furthermore, all colours within a triangle may be formed by mixing the colours at the vertices. This property illustrates graphically the fact that all visible colours cannot be obtained by a mix of R, G and B (or any other three visible) primaries alone, since the diagram is not triangular!



- Mixing colours on the chromaticity diagram. All colours on the line IJ can be obtained by mixing colours I and J, and all colours in the triangle IJK can be obtained by mixing colours I, J and K.

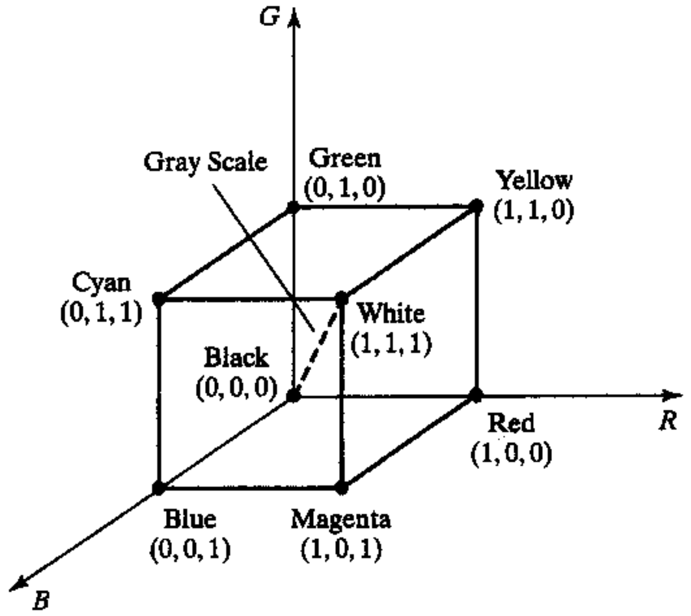
RGB model

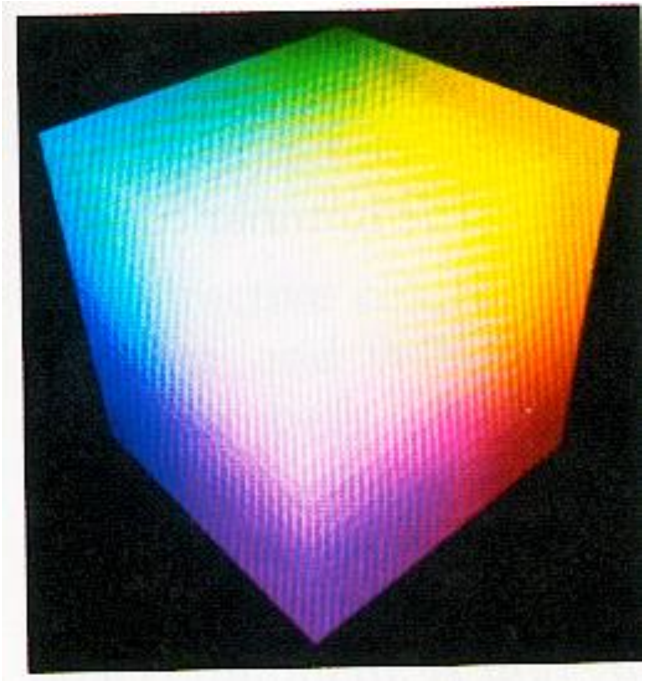
- all colors are generated from the three primaries
- various colors are obtained by changing the amount of each primary
- additive mixing (r,g,b)



RGB model,cont’d

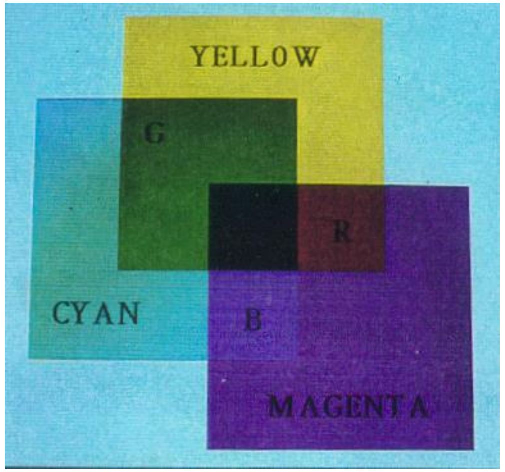
- the *RGB cube*





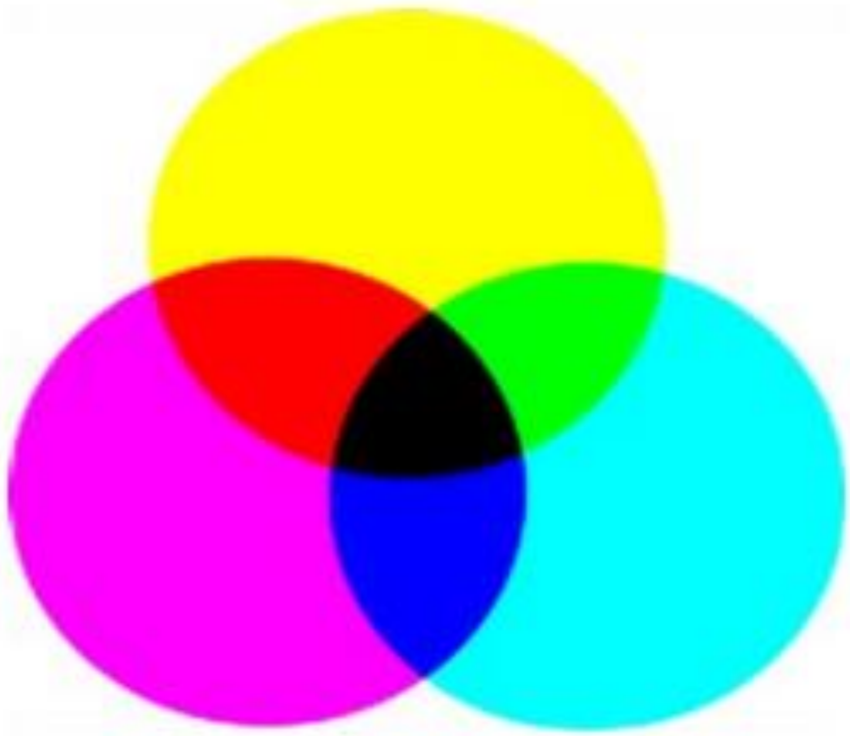
CMY model

- cyan, magenta and yellow are comple-mentary colors of red,green and blue, respectively
- subtractive mixing
- the typical printer technique



CMY model,cont’d

- almost the same cube as with RGB; only black<-> white
- the various colors are obtained by reducing light, e.g. if red is absorbed => green and blue are added, i.e cyan



RGB vs CMY

If the intensities are represented as $0 \le r,g,b \le 1$ and $0 \le c,m,y \le 1$ (also coordinates 0-255 can be used), then the relation between RGB and CMY can be described as:

$$\begin{pmatrix} c \\ m \\ y \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} r \\ g \\ b \end{pmatrix}$$

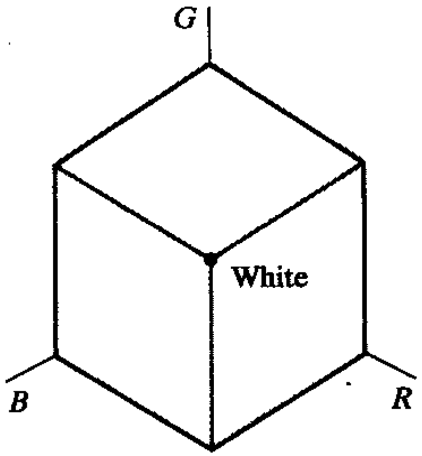
CMYK model

For printing and graphics art industry, CMY is not enough; a fourth primary, K which stands for black, is added.

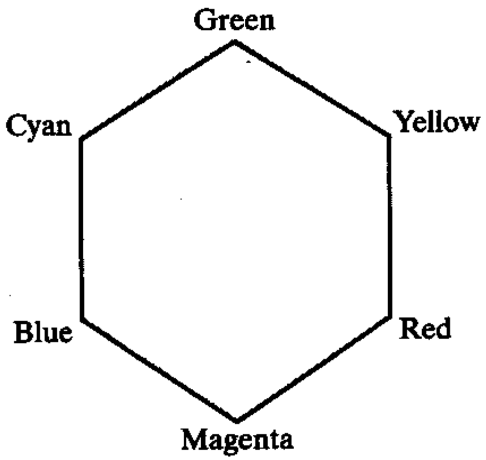
Conversions between RGB and CMYK are possible, although they require some extra processing.

HSV model

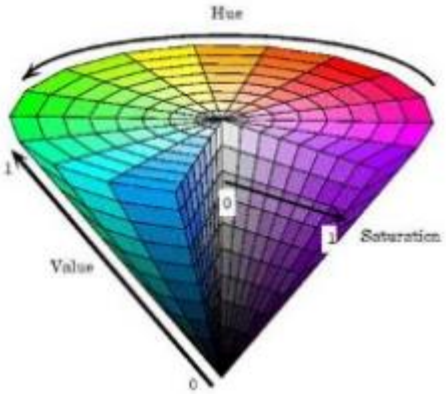
- HSV stands for Hue-Saturation-Value
- described by a hexcone derived from the RGB cube



RGB Color Cube
(a)

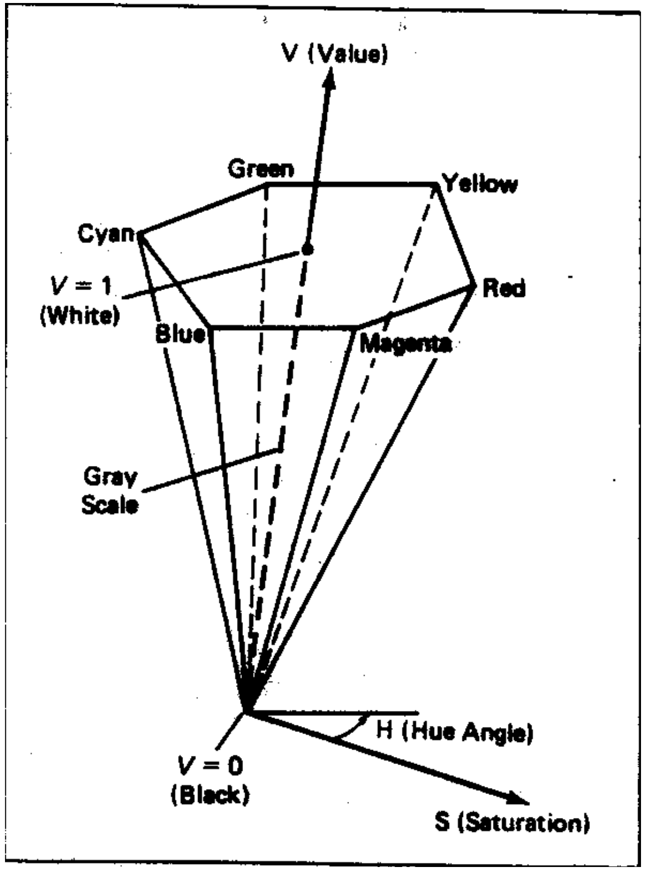


Color Hexagon
(b)



HSV model,cont’d

- Hue (0-360°); ”the color”, the dominant wave-length
- Saturation (0-1); ”the amount of white”
- Value (0-1); ”the amount of black”



HSV

- HSV Color Model Hue, Saturation, Value or HSV is a color model that describes **colors (hue or tint)** in terms of their **shade (saturation)** and **their brightness (value)**. HSV color model is based on polar coordinates; Developed in the 1970s for computer graphics applications, HSV is used today in color pickers, in image editing software, and less commonly in image analysis and computer vision.
- HSV Color Model • Hue (H), the color type (such as red, green). It ranges from 0 to 360 degree, with red at 0 degree, green at 120 degree, blue at 240 degree and so on. The two representations rearrange the geometry of RGB in an attempt to be more intuitive and perceptually relevant ,based on the color wheel.
- HSV Color Model • Saturation (S) of the color ranges from 0 to 100%. Also sometimes, it called the "purity". The lower the saturation of a color, the more "grayness" is present and the more faded the color will appear. • Value (V), the Brightness (V) of the color ranges from 0 to 100%. It is a nonlinear transformation of the RGB color space. Note that HSV and HSB are the same.

HLS model

Another model similar to HSV

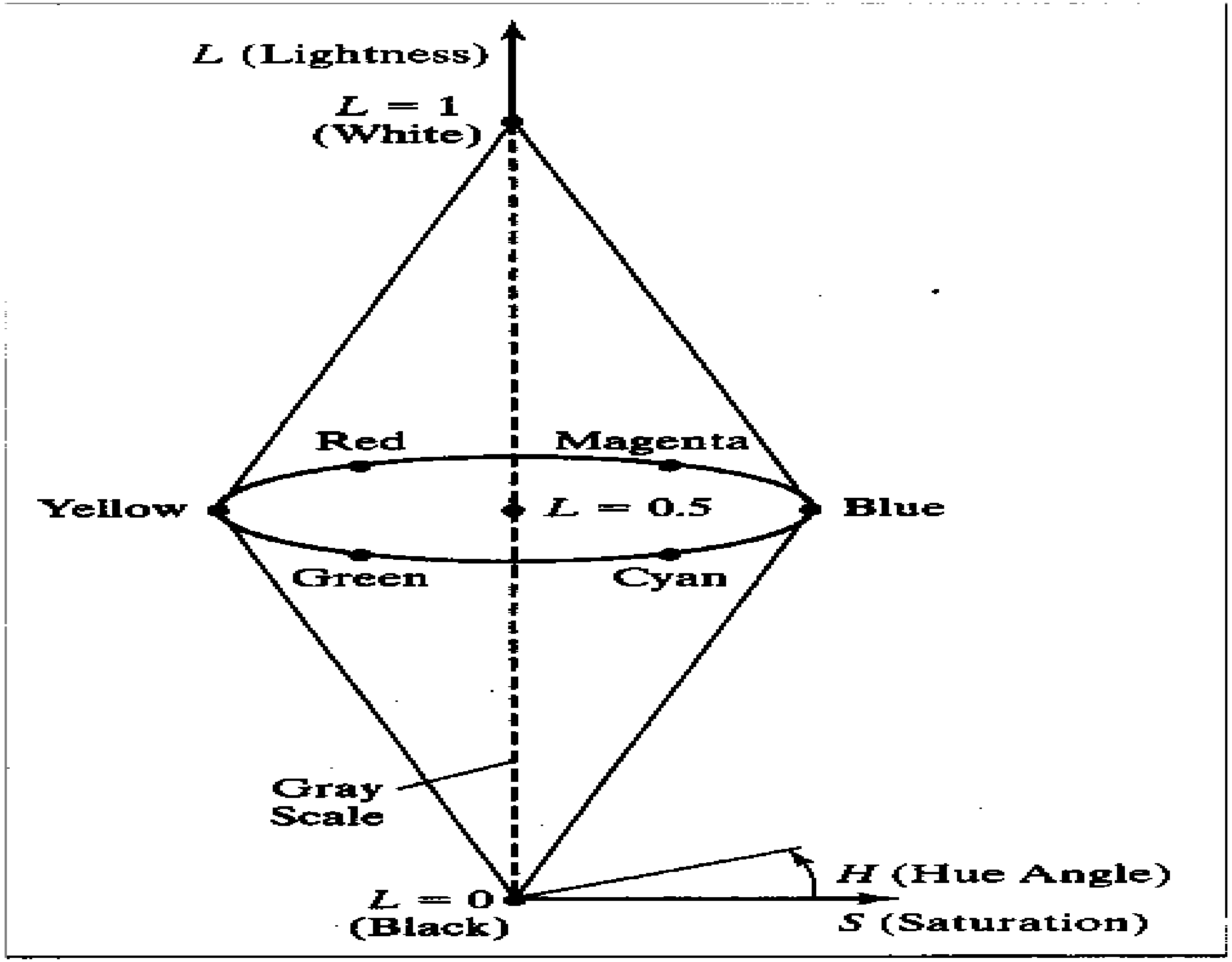
L stands for *Lightness*

Hue specifies the angle about the vertical axis that locates the specific hue.

The colours are specified around the perimeter of the cone.

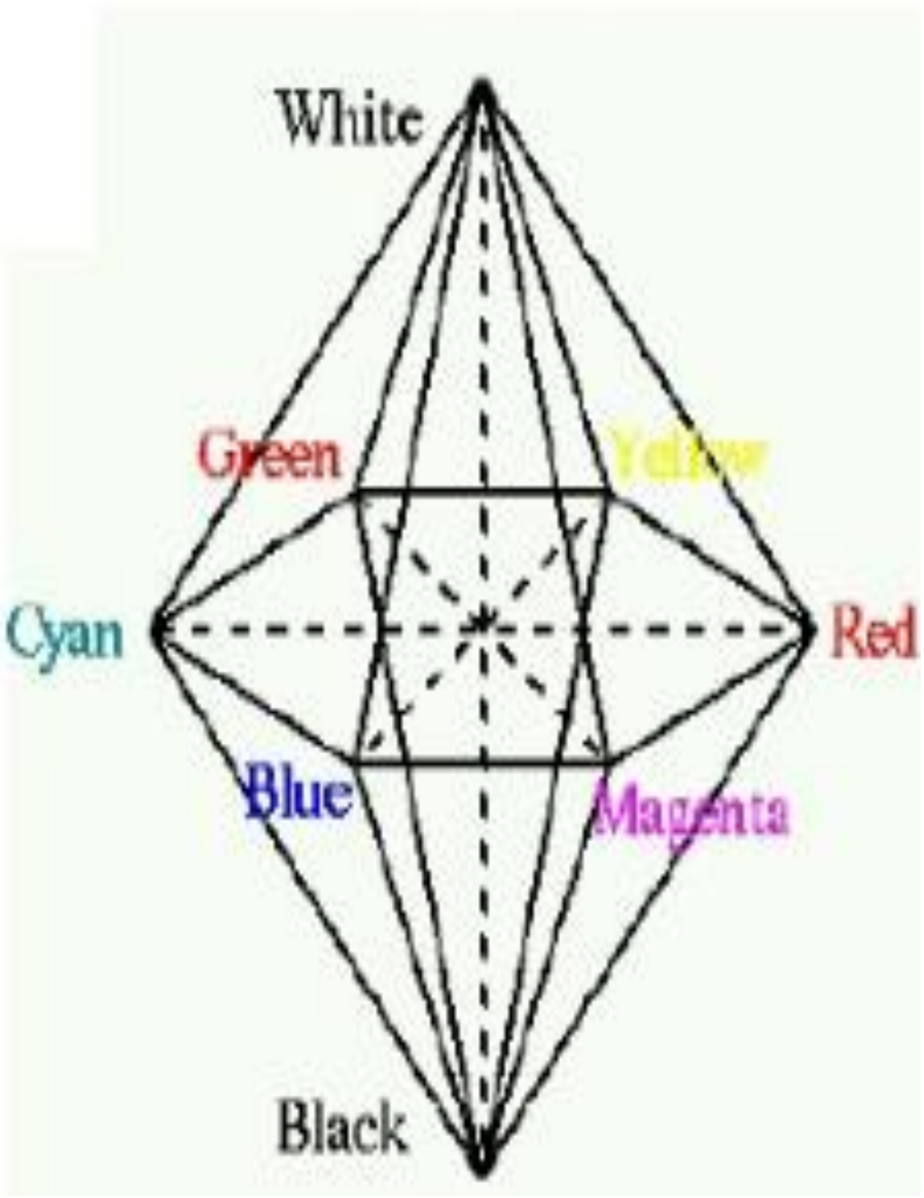
Complementary colours are 180 degrees apart

The vertical axis is called the "Lightness"



Represented by a double hexagonal Cone, with WHITE at the top and BLACK at the bottom.

The **HUE** lies on the circle,
LIGHTNESS gradually increases from Black to White
SATURATION increases from centre to the periphery



Color models

Some more facts about colors:

The distance between two colors in the color cube is not a measure of how far apart the colors are perceptionally!

Humans are more sensitive to shifts in blue (and green?) than, for instance, in yellow