Color in Image Processing and Computer Vision

Image Processing and Computer Vision

Rein van den Boomgaard   Leo Dorst

University of Amsterdam

Outline

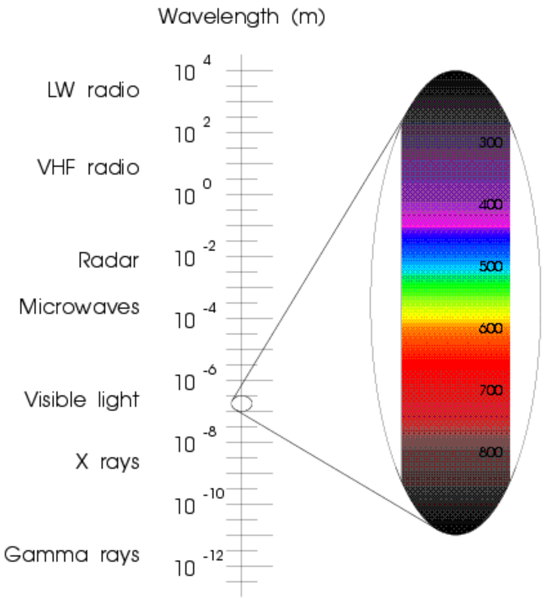
Color Images

Color Images

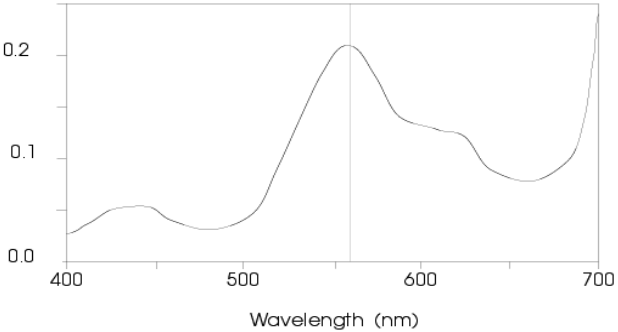
Color Images

- Color adds a lot to our human capability in recognizing objects
- Color is a local surface property that is (almost) view invariant and largely independent of resolution

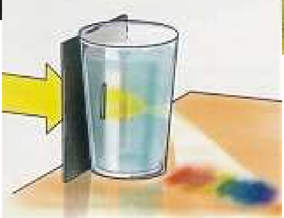
Electromagnetic Radiation



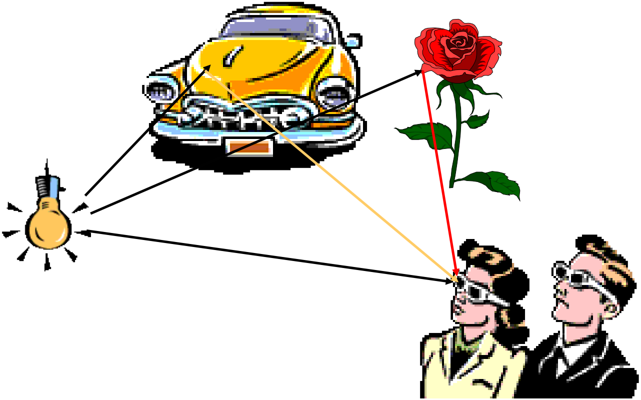
Spectra



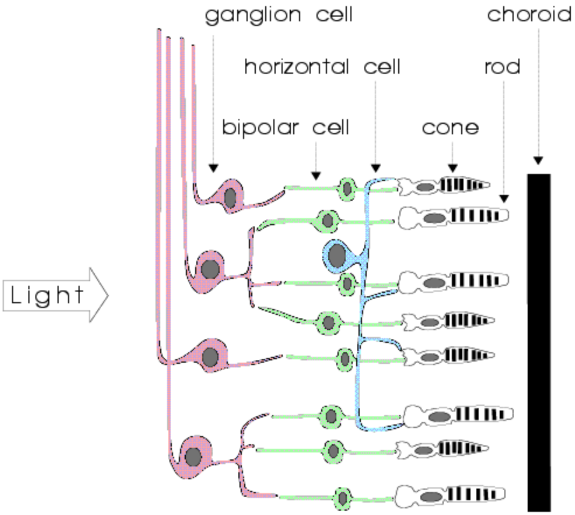
Spectrum of White Light



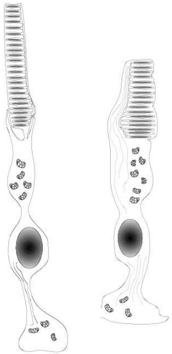
Beam Color versus Object Color



Photoreceptors (I)

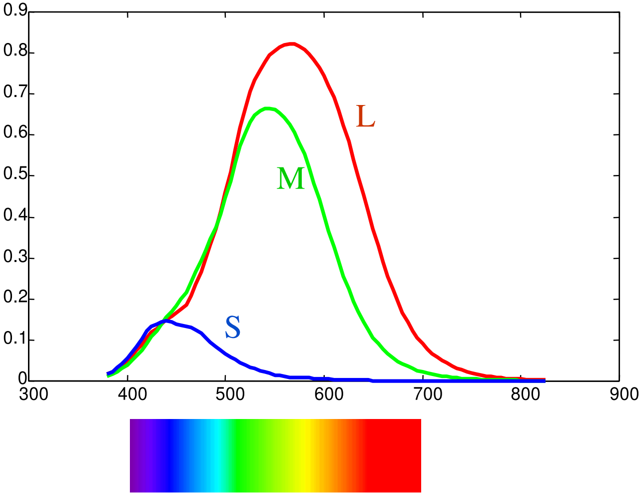


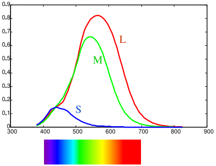
Photoreceptors (II)



- Rods
  - used in low light levels (B&W only)
- Cones
  - normal light levels
  - color vision (3 different types of cones)

Cone Sensitivity (I)





The responses are:

$$c_i = \int s_i(\lambda) t(\lambda) d\lambda$$

for $i = S, M, L$. For a sampled spectrum and sensitivity curve:

$$c_i = s_i^T t$$

Combining these :

$$c = S^T t$$

where

$$c = \begin{pmatrix} c_1 & c_2 & c_3 \end{pmatrix}^T, \qquad S = (\; s_1 \quad s_2 \quad s_3$$

Grassman Laws

1. perceived brightness of a beam is linear:

$$c \longrightarrow \alpha c$$

   changes only the brightness and not the color (tone)

2. mixture of colorbeams is additive:

$$c_1 = S^T t_1, \quad c_2 = S^T t_2$$

$$t_1 + t_2 \longrightarrow c_1 + c_2$$

These are *experimental facts*, and show that we can use linear algebra for colorimetry.

Color Equivalence

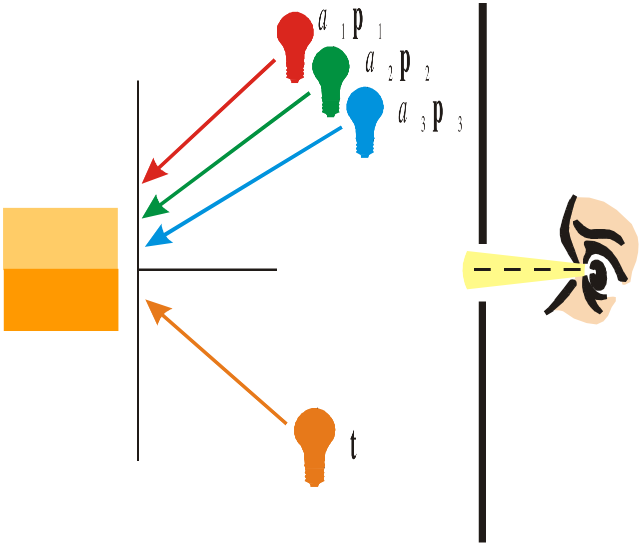
Colorimetry is the science of color equivalences. Let t_1 and t_2 be the spectra of two color beams. These beams are perceived as equivalent colors in case:

$$S^T t_1 = S^T t_2$$

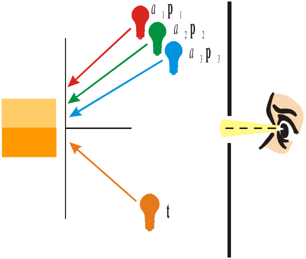
Note:

- equivalent colors does not necessarily mean that the beams have equal spectra.
- color is characterized in 3D space, spectra in $N \gg 3$ dimensional space. There will be many spectra with the same color sensation
- color equivalence is not changed in case S^T is replaced with AS^T where A is a 3×3 non-singular matrix.

Color Matching Experiment (Setup)



Color Matching Experiment (Math)



Equality of color perception requires:

$$S^T t = S^T(a_1 p_1 + a_2 p_2 + a_3 p_3)$$

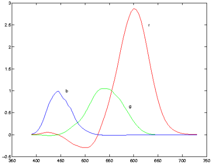
or equivalently:

$$S^T t = S^T P a$$

and thus:

$$a = (S^T P)^{-1} S^T t$$

rgb-Color Matching Functions



For the standard choice of monochromatic primaries $P = \begin{pmatrix} p_R & p_G & p_B \end{pmatrix}$ the three columns of $C = \begin{pmatrix} r & g & b \end{pmatrix}$ are depicted.

The matrix

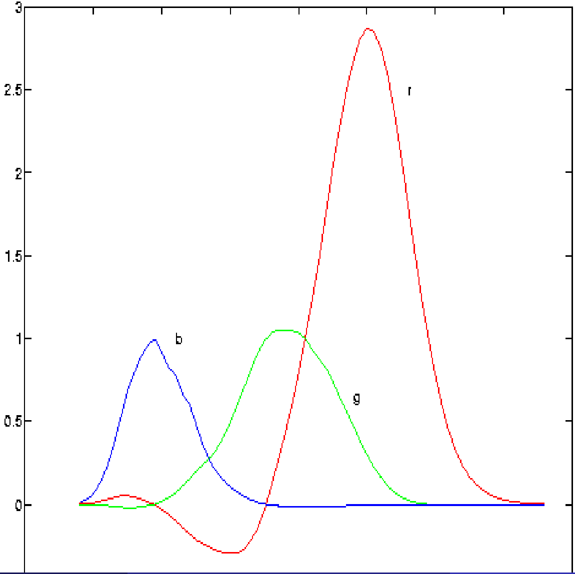
$$C = S(S^T P)^{-T}$$

is dependent on the eye sensitivity matrix S and the primaries used in the colormatching experiments. Let δ_i be a monochromatic beam, i.e. a vector of all zeros except a one at index i. Then we let an average human select the a to match the monochromatic beam:

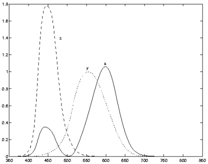
$$a = C^T \delta_i$$

Now the i-th row of C is given by a. Repeating the experiment for all monochromatic beams we find the entire C-matrix.

rgb-Color Matching Functions



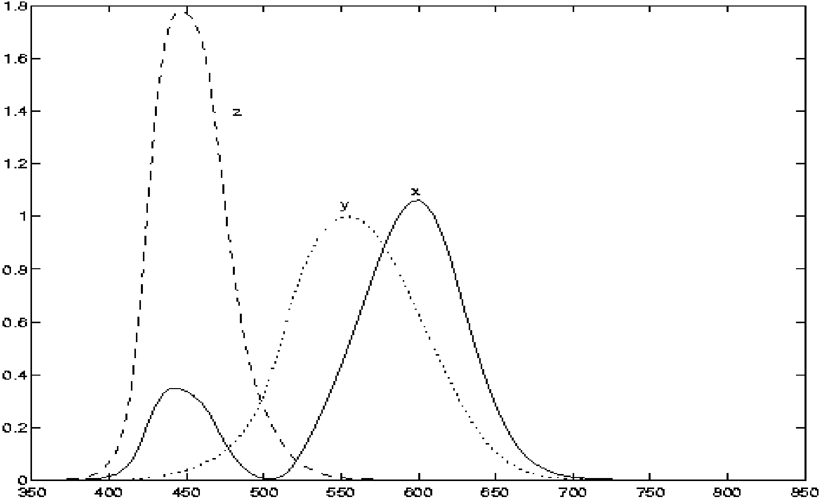
xyz-Color Matching Functions



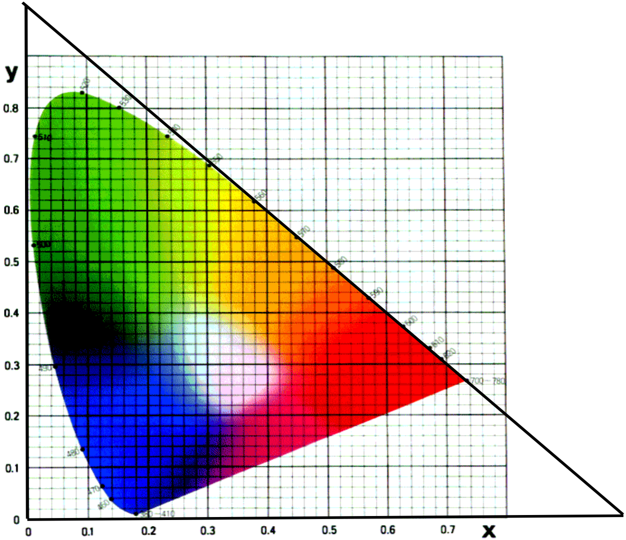
- At the time the *rgb*-color matching curves where measured, the use of negative numbers was considered to be a serious disadvantage. Therefore it was decided to transform the C_{rgb} colormatching matrix into one containing no negative numbers.
- Furthermore one of the 'new' coordinates was chosen to be proportional to the perceived brightness of the beam.
- The XYZ colormatching matrix is denoted as C_{XYZ}. Given a beam spectrum t the color coordinates are:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = C_{XYZ}^T t$$

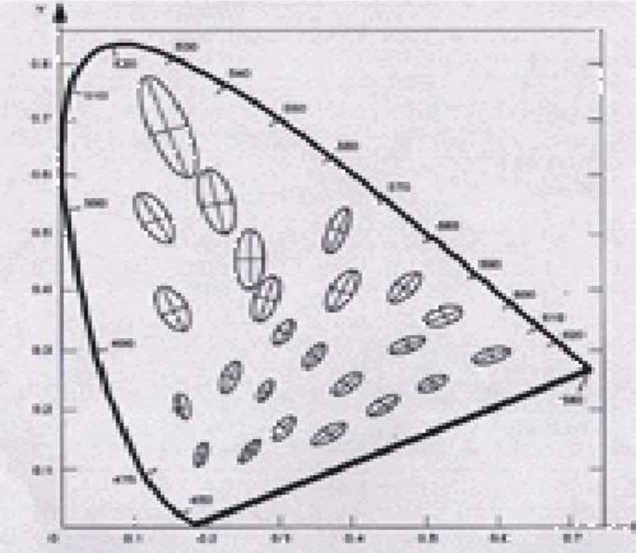
XYZ Color Matching Curves

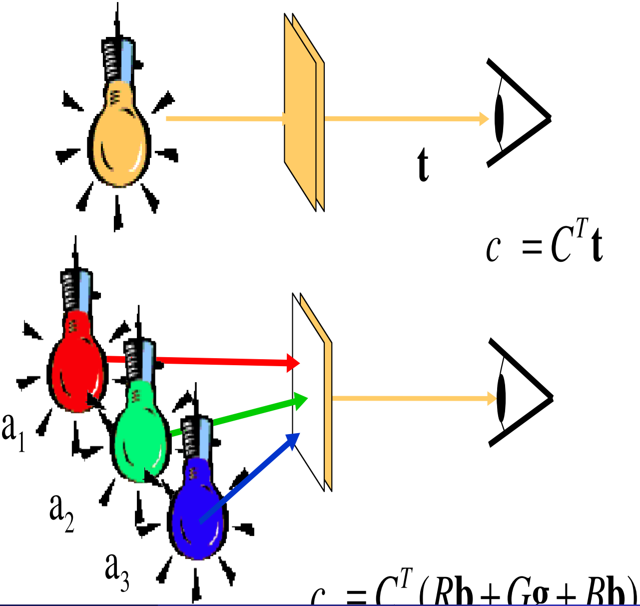


Chromaticity Plane

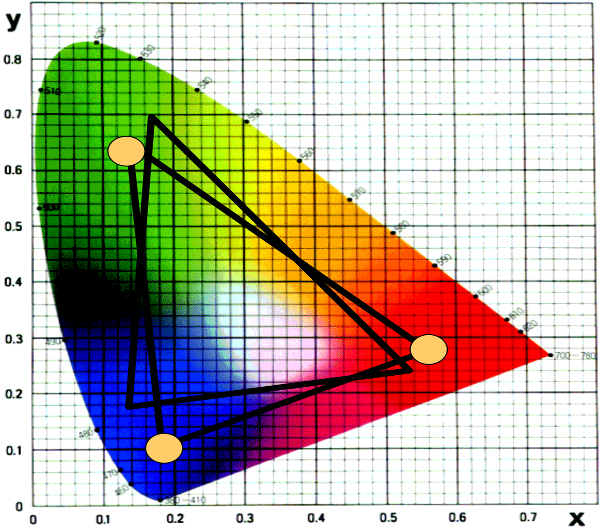


Perceptual Color Differences

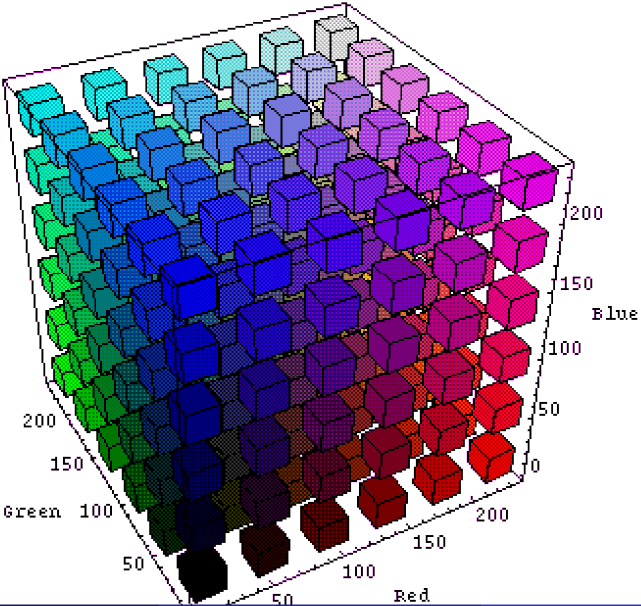



t
$c = C^T \mathbf{t}$
a_1
a_2
a_3
$c = C^T (R\mathbf{b} + G\mathbf{g} + B\mathbf{b})$

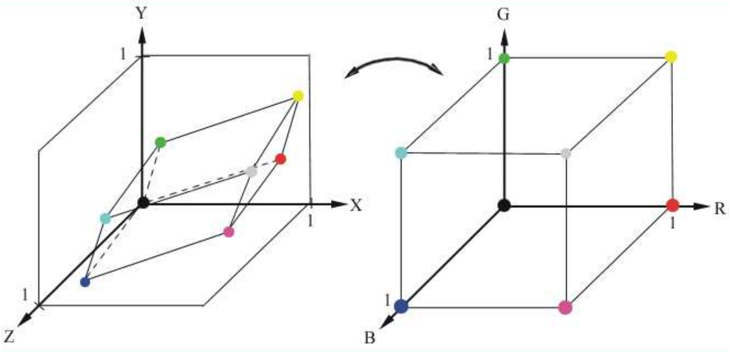
Gamut



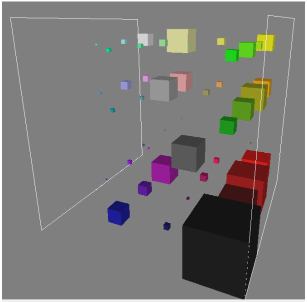
RGB Color Model



RGB Color Model in XYZ-Space

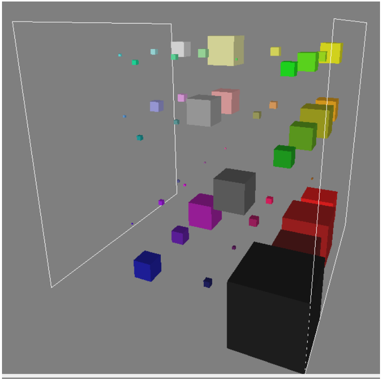


Color Histograms



- A RGB color histogram counts the frequency of occurance of a color (R,G,B) in an image.
- For 8 bit R,G and B colorplanes, we have $2^{24} = 16777216$ different colors: a histogram with that many bins is unfeasible an probably won't show the 'real' color distribution.
- Subsampling the RGB colorcube into $K_R \times K_G \times K_B$ bins is a usual way to deal with this.
- We consider here a $4 \times 4 \times 4$ color histogram.

Color Histograms



- Let b be the mapping from a color (RGB triplet) to a bin in the histogram
- For a subsampling of the RGB color cube we might have

```
def bin(c):
   return (c[0]>>4)<<8 & \
          (c[1]>>4)<<4 & \
          (c[2]>>4)
```

- Then we make a histogram with

```
for x,y in image:
   c = image(x,y)
   b = bin(c)
   histogram(b) += 1
```

Histogram Intersection (I)

Histogram Intersection

Let h_M be the histogram of a the image of an object we are looking for (in other images). This is the *model* and h_M is the *model histogram*. Let h_I be the histogram of the image we want to compare to our model. Is the object visible or not?
Swain and Ballard defined the *histogram intersection*, to characterize the equality of the histograms:

$$H(h_M, h_I) = \frac{\sum h_M \wedge h_I}{\sum h_M}$$

Histogram Intersection (II)

The rationale for this particular 'equality function' are:

- $0 \leq H \leq 1$
- H is not much influenced by the colors in the image I that are not in the model (e.g. consider a histogram of an object against an unknown but colorful background).
- Note that H is an assymmetrical function in the two histograms.

Finding Waldo (I)

- Find ‘Waldo’ in the large image
- Ballard and Swain showed that this can be done based on the color distribution alone (no spatial coherence of colors).

Histogram Back Projection

Indexing Via Color Histograms

Michael J. Swain†and Dana H. Ballard

Department of Computer Science
University of Rochester
Rochester, NY 14627, USA

Abstract

The color spectrum of multicolored objects provides a a robust, efficient cue for indexing into a large database of models. This paper shows color histograms to be stable object representations over change in view, and demonstrates they can differentiate among a large number of objects. It introduces a technique called *Histogram Intersection* for efficiently matching model and image histograms. Color can also be used to search for the location of an object. An algorithm called *Histogram Backprojection* performs this task efficiently in crowded scenes.

1 Introduction

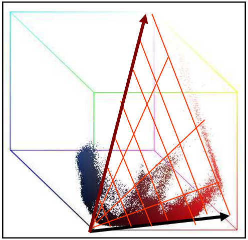
Computer vision is moving into a new era in which the aim is to develop visual skills for robots that al-

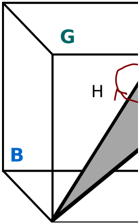
color constancy can be achieved, color has enormous value in recognition because it is a local surface property that is view invariant and largely independent of resolution. Shape cues, by contrast, are highly resolution dependent, and only a highly restricted set are view invariant (e.g. corners, zeros of curvature).

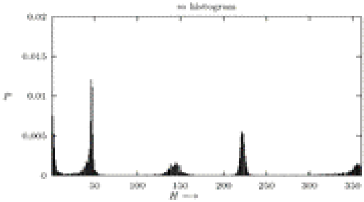
Perhaps another reason that color has not been used is that it is not intrinsically related to the object's identity in the way that other cues, e.g., form, are. This view is well represented by Biederman [1]:

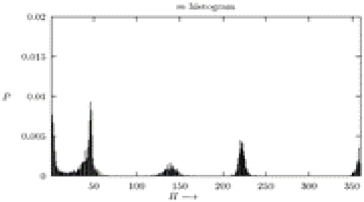
> "Surface characteristics such as color and texture will typically have only secondary roles in primal access ... we may know that a chair has a particular color and texture simultaneously with its volumetric description, but it is only the volumetric description that provides efficient access to the representation of CHAIR."

Color Models (HSV)






0.02
0.015
0.01
0.005
0
50
100
150
200
250
300
350


0.02
0.015
0.01
0.005
0
50
100
150
200
250
300
350

Skin Color Detection