Advances in color image processing and analysis; selected applications

Lecturer:

Assoc. prof. Mihaela GORDAN

Technical University of Cluj-Napoca,
Faculty of Electronics, Telecommunications and Information Technology,
Communications Department

e-mail: Mihaela.Gordan@com.utcluj.ro

Office tel.: *+40-264-401309*

Office address: *CTMED Lab., C. Daicoviciu 15, Cluj-Napoca, ROMANIA*

Presentation contents

1. Human perception of color images (1)

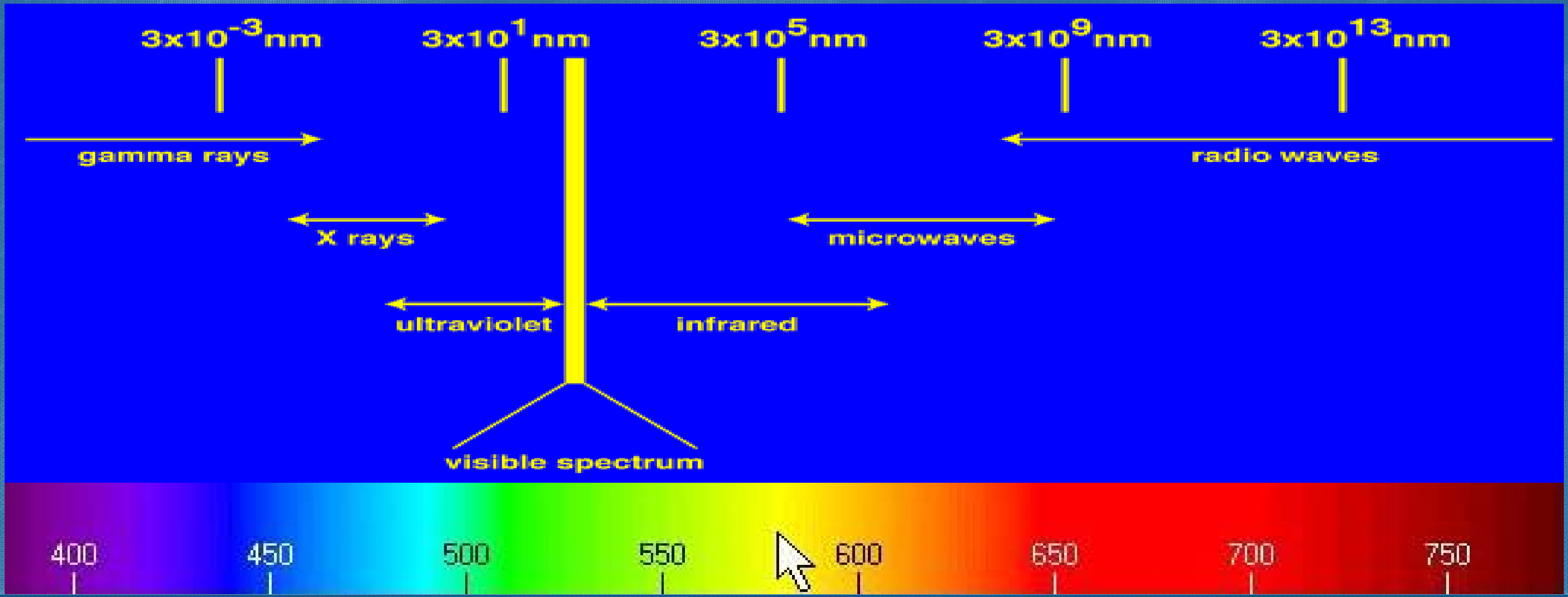
- **Perception of color** – crucial for many machine vision applications
- **General observation:**
  - most color image processing algorithms consider one pixel at a time,
  - **but in the HVS – the color perceived at a spatial location is influenced by the color of all the spatial locations in the field of view!**
- **Future issues for color image processing:** use the human visual models to describe ***the color appearance of spatial information,*** to replace the common low level (pixel-level) approaches => future trends: ***develop color image processing and analysis algorithms based on high level concepts***

1. Human perception of color images (2)

- A still image = visual representation of a 2-D scene; formed by the emission, reflection or transmission of radiation, by/through surfaces
  - **Primary sources of radiation** = sources able to emit radiation
  - **Secondary sources of radiation** = surfaces that reflect or transmit radiation (are reflective or semi-transparent); the radiation comes from a primary source
- In respect to the wavelength:
  - Visible spectrum radiation = "the light" => perceived by the HVS => the physical image = the visual reality perceived in the visible spectrum (350nm…780 nm)
  - Radiation in other wavelength ranges (e.g. IR, US, X rays)
- In respect to the energy: the radiant intensity = the energy emitted at each wavelength from the source spectrum

1. Human perception of color images (3)

- **The electromagnetic spectrum:**

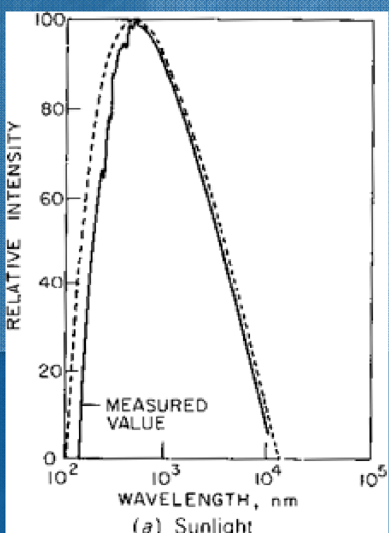


Wavelength (in nm)

White light (achromatic) = radiation composed by equal amounts of all wavelengths in the visible spectrum

1. Human perception of color images (4)

- Light = electromagnetic wave => can describe the light source by its **radiance** (= radiated flux per area, at each wavelength): $c(x, y, t, \lambda)$;
- At x, y, t given => describe the source by $c(\lambda)$ = spectral distribution of the energy of the source



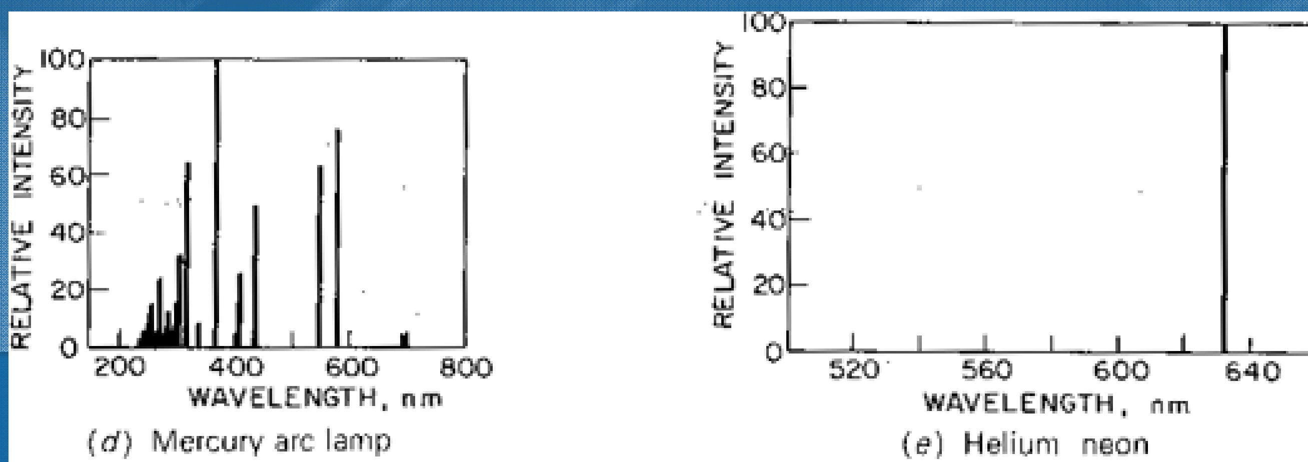
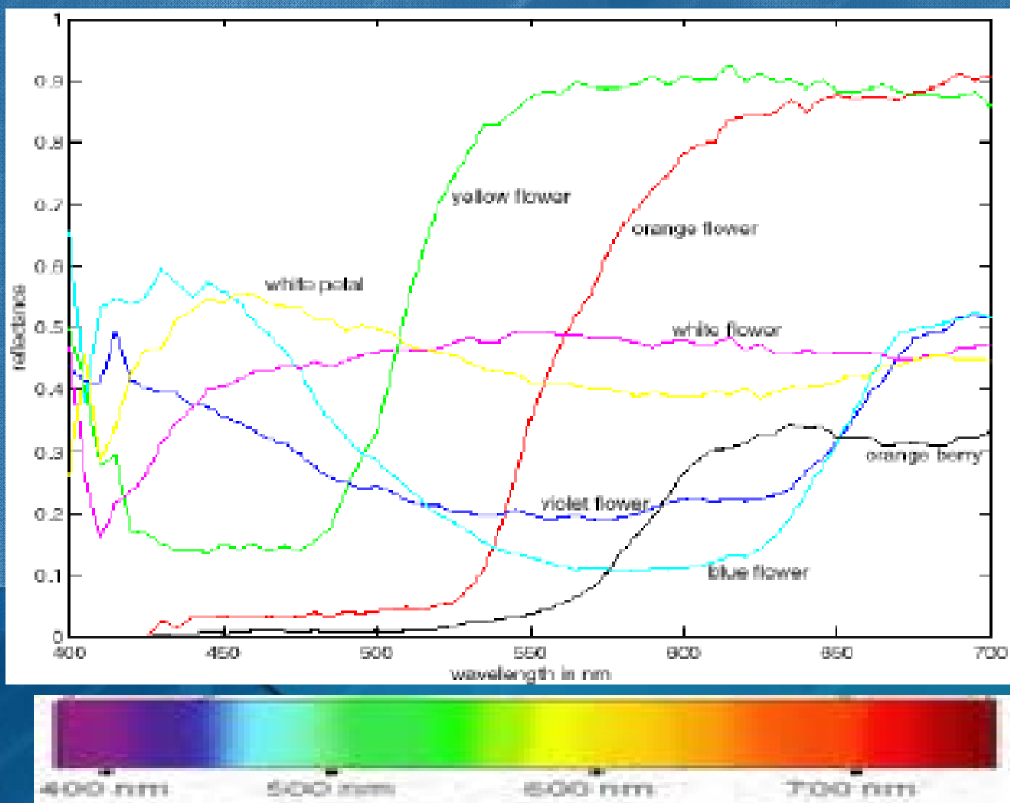
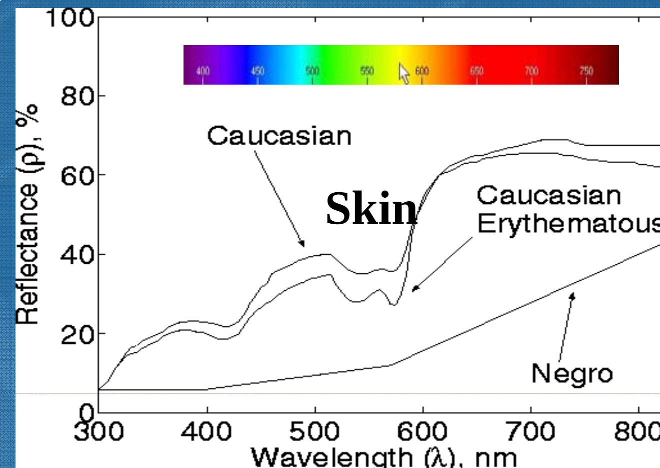


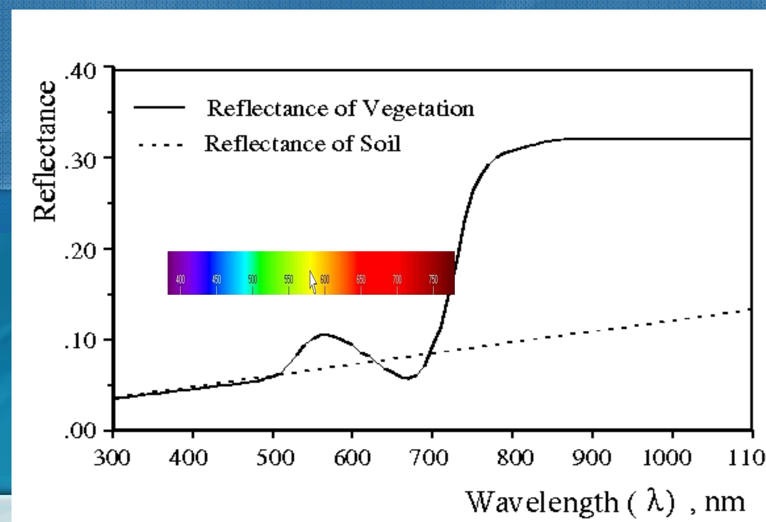
FIGURE 2.1-1. Spectral energy distributions of common physical light sources (2).

1. Human perception of color images (5)

- Some spectral distributions of secondary sources of radiation:

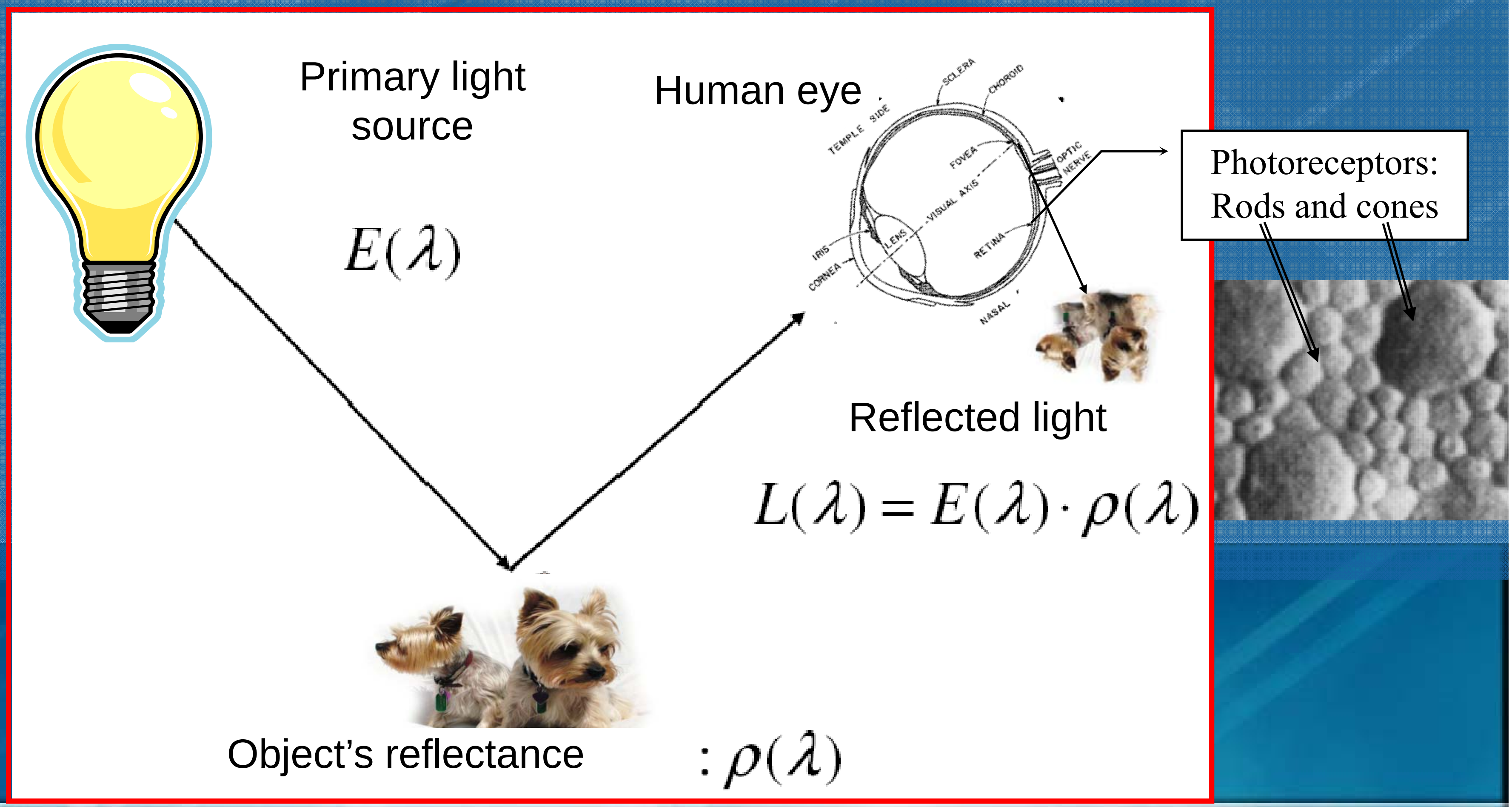






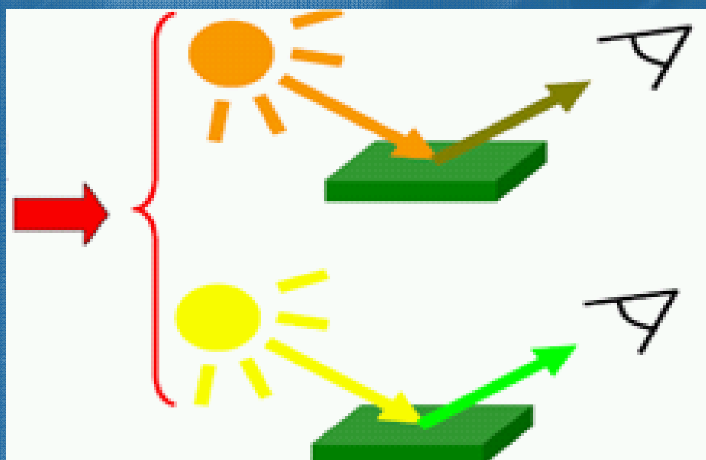
1. Human perception of color images (6)

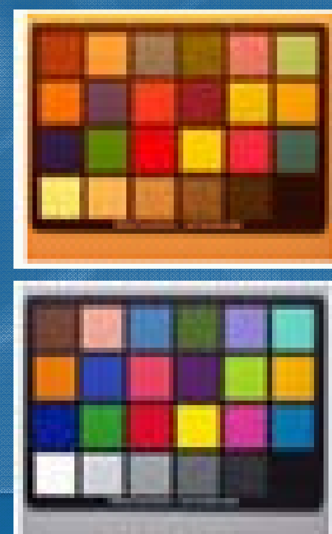
- **The human color vision system:**



1. Human perception of color images (7)

- Image formation is affected by the color of the illuminant and by the properties of the material, i.e. by the spectral distribution of the primary source of radiation and of the secondary source of radiation:





Tungsten lamp

Sunlight

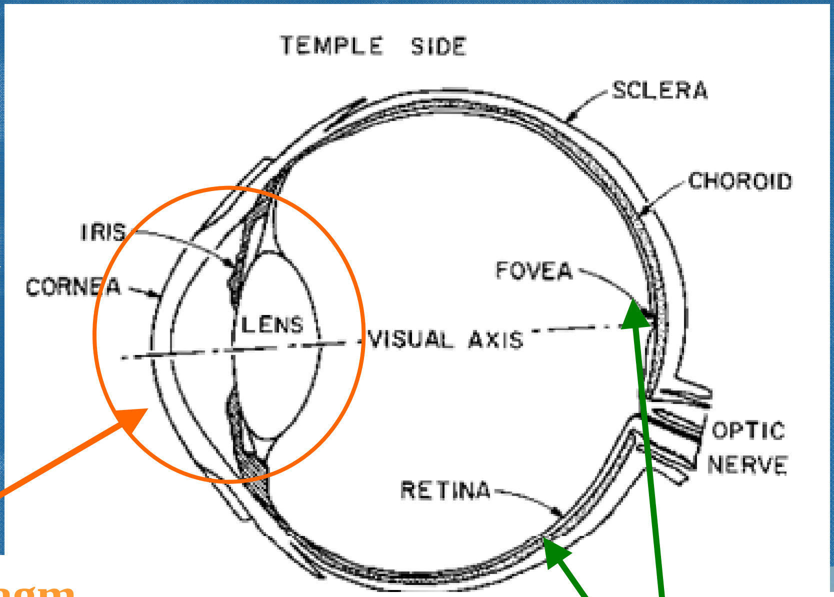
Same object, different primary radiation source => different perceived color

- Challenge in color image processing/analysis: correct the color appearance/find color descriptors invariant to illuminant

1. Human perception of color images (8)

- **The eye as an image capture system:**

Perception

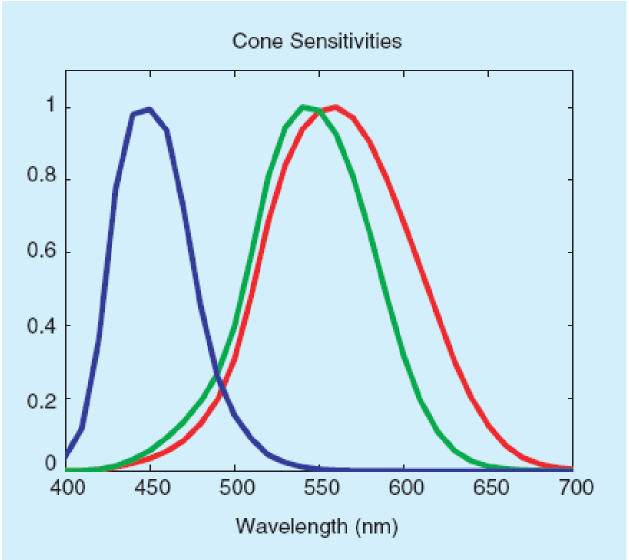


Optical system: Iris = diaphragm
Cornea, lens = focus

The optical receivers:
Matrix of cones and rods; maximum density in the fovea

1. Human perception of color images (9)

- **Photoreceptors in retina:**
  - **Rods** = sensitive to low levels of light; can't perceive color
    = absent in the fovea; maximum density in 18^0 eccentricity annulus
    => "peripheral vision field"
  - **Cones** = sensitive to normal light level (daylight); perceive color
    = 3 types of cones: long (L), medium (M), short (S) wavelength
    = maximum density in **fovea** ("central visual field", 2^0 eccentricity)



- **Types of vision (visual response):**
  - **Scotopic vision**
    = achromatic vision
    = rods only active below 0.01 cd/m^2
  - **Photopic vision**
    = color vision
    = cones only active above 10 cd/m^2
  - **Mesopic vision** => rods and cones active

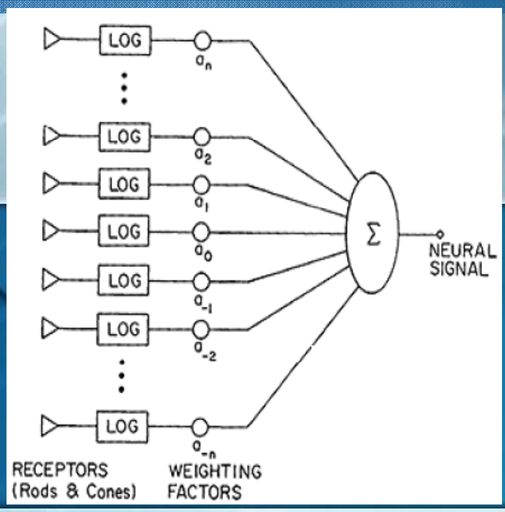
1. Human perception of color images (10)

- ***Some notes:***

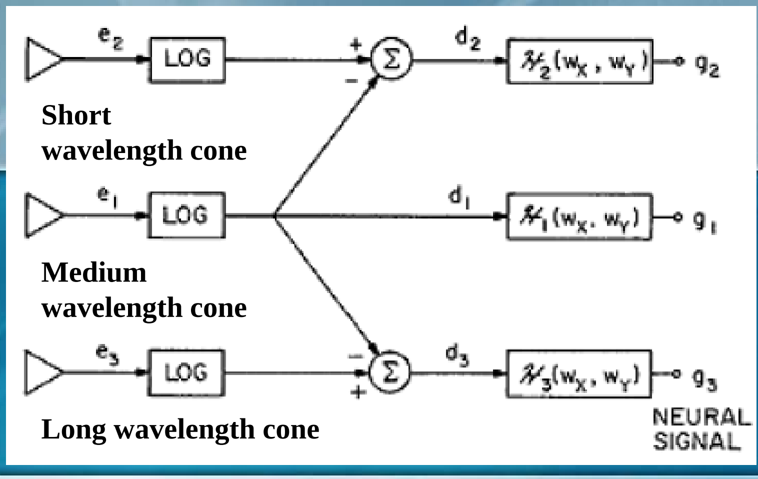
(1) finite number of photoreceptors => the spatial resolution of the eye is finite; the impression of continuous scene ⇔ the spatial interpolation in the brain;
(2) the amplitude of the spectral responses of the 3 types of cones is not equal; lowest for short wavelength, higher for medium and long wavelengths
(3) # cones and rods: > 100,000,000; # optical nerve fibers: 800,000 => the connection of optical photoreceptors to the visual cortex = many-to-one
(4) most likely – the eye response to variation of the intensity level = ***logarithmic***

- ***=> Simplified visual perception models:***

Achromatic model

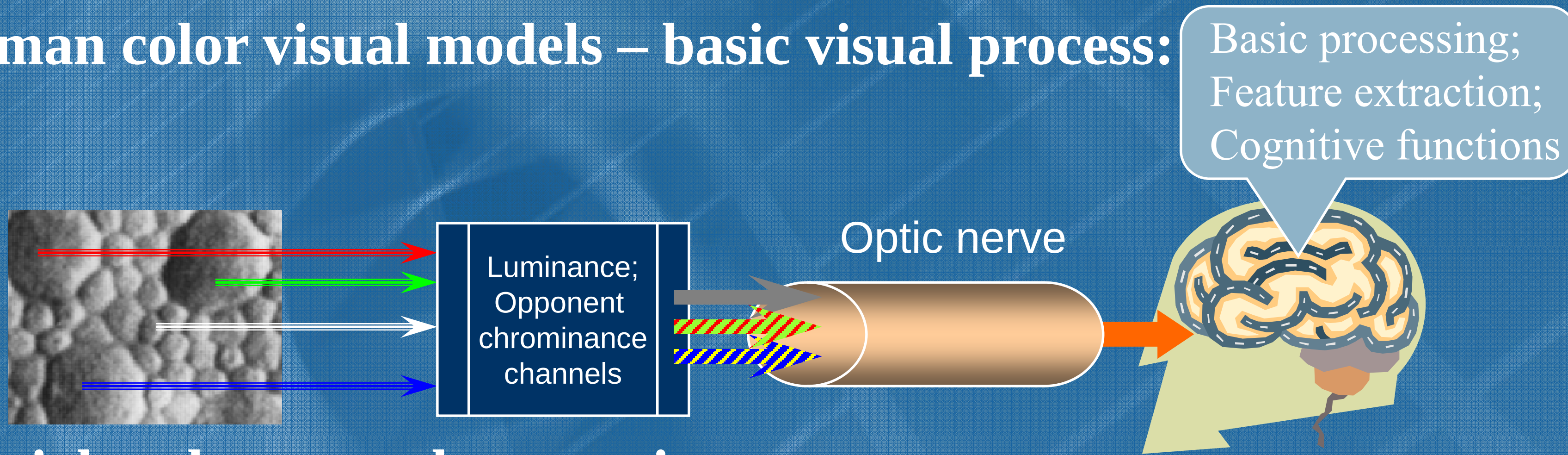


Color model



1. Human perception of color images (11)

- **Human color visual models – basic visual process:**



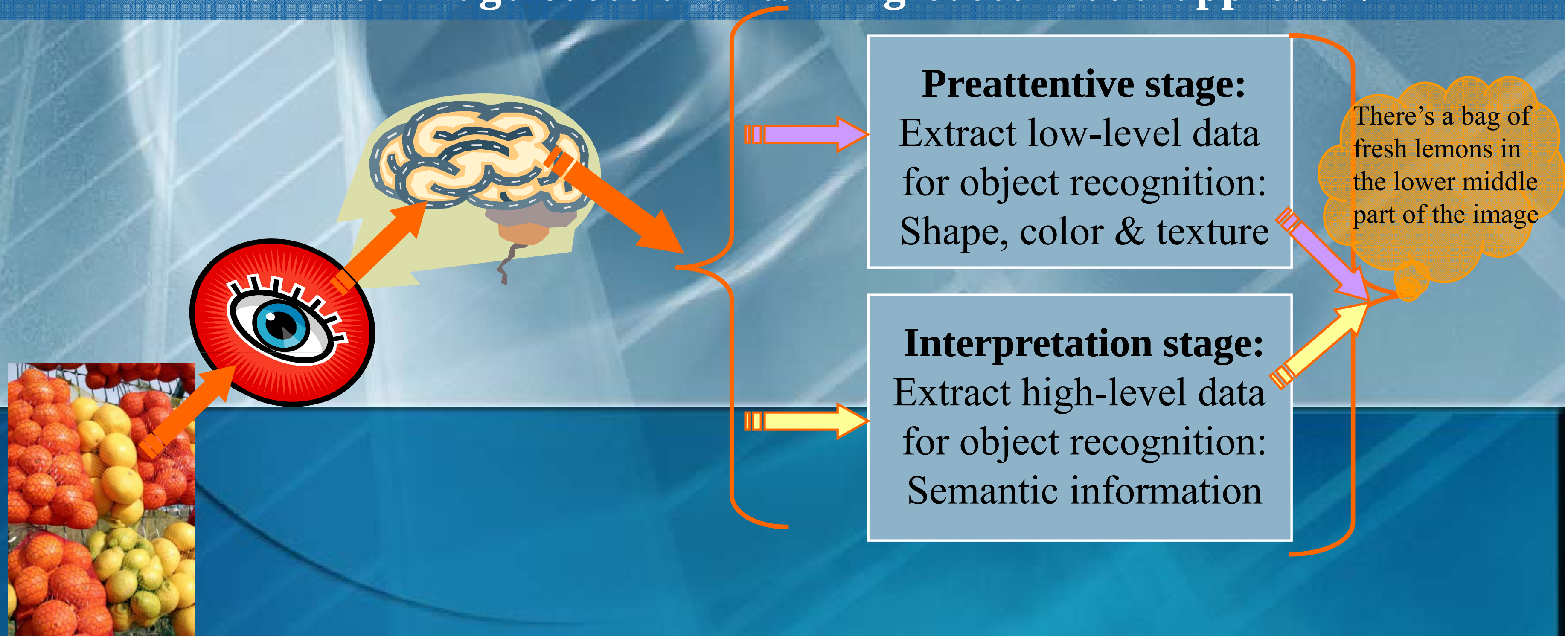
- **Spatial and temporal perception:**
  - **Visual info – simultaneously processed in several "visual channels":**
    - **high frequency active channels** (P-channels): perception of details
    - **medium frequency active channels**: shape recognition
    - **low frequency active channels** (M-channels): perception of motion

=> The simultaneous results of the 3 channels, achromatic & chromatic,

- filtered by specific spatial and temporal contrast sensitivity functions (CSFs); achromatic CSF > chromatic CSF
- combined further in the vision process

1. Human perception of color images (12)

- **Human color visual model – a point of view:**
  - Still an open research issue; gap between traditional computer vision and human vision sciences => **new human vision models needed**
  - **The mixed image-based and learning-based model approach:**



2. Color imaging applications - overview

I. Consumer imaging applications:

- **Mostly involves image processing, image enhancement**
- **Color management challenges** => achieve WYSIWYG concept, by color appearance models & color management methods – standardized
- **Basic applications fields**: graphics arts; HDTV; web; cinema; archiving, involving image/video restoration, colorization, image inpainting

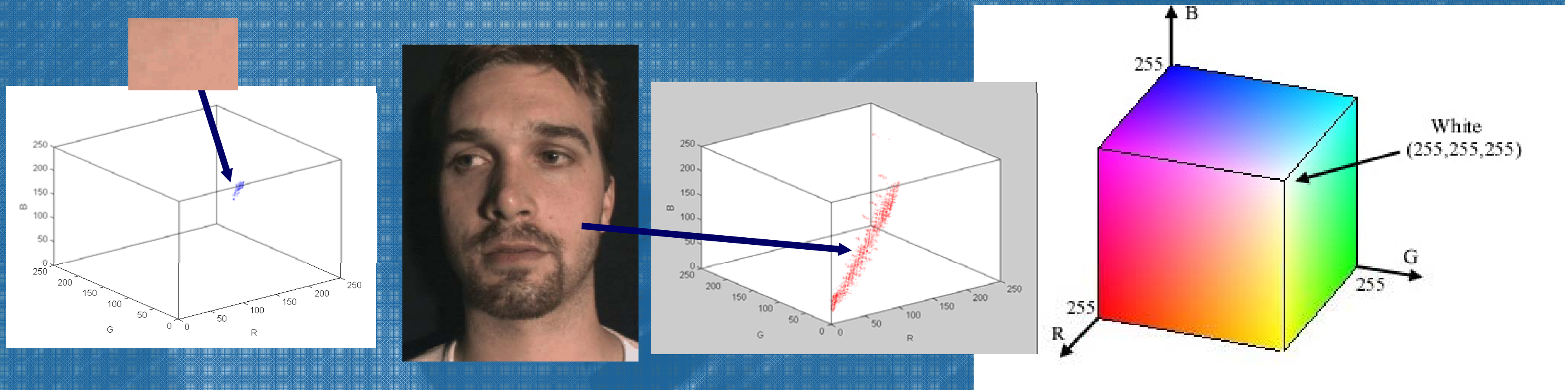
II. Medical imaging applications:

- **Mostly involves image analysis**
- **Challenges** => model image formation process & correlate image interpretation with physics based models;
  => analyze changes over time
- **Methods**: use low level features & add high level interpretation to assist diagnostic

III. Machine vision applications:

- Robot vision; industrial inspection => **image analysis & interpretation methods – similar to medical imaging**

3. Color spaces, properties, metrics (1)



- **Color spaces properties:**

- **P1. Completeness:**

  Def.1: A color space S_C is called ***visually complete*** iff includes all the colors perceived as distinct by the eye

  Def.2: A color space S_C is called ***mathematically complete*** iff includes all the colors possible to appear in the visible spectrum

3. Color spaces, properties, metrics (2)

- **P2. Compactness:**
  Def.: A color space S_C is called ***compact*** if any two points of the space $\mathbf{s_i}$, $\mathbf{s_j}$ are perceived as distinct colors
  - Note: One can obtain a compact color space from a mathematically complete color space through color space quantization (e.g.: vector quantization)
- **P3. Uniformity:**
  Def.1: A color space S_C is called ***uniform*** if a distance norm d_C over S_C can be defined so that: $d_C(\mathbf{s_i}, \mathbf{s_j}) \sim$ perceptual similarity of $\mathbf{s_i}$ and $\mathbf{s_j}$
  - **Note:** Usually, d_C = Euclidian distance
- **P4. Naturalness:**
  Def.: The color space S_C is called ***natural*** if its coordinates are directly correlated to the perceptual attributes of color.
  The perceptual attributes of color = the HVS specific attributes in the perception and description of a color: ***Brightness; Nuance (Hue); Saturation (Purity).***
  - **Note:** the RGB space (the primary color space) only satisfies completeness => the need to define other spaces for color representation.

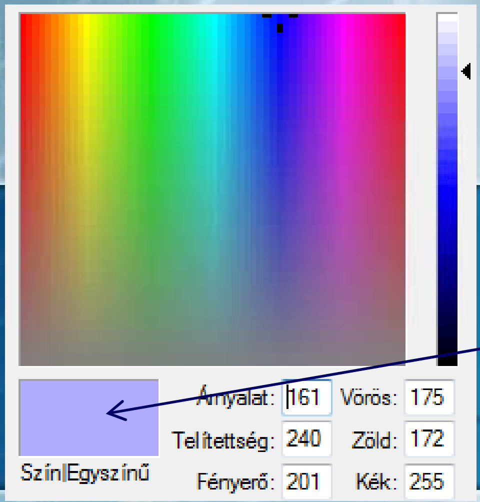
3. Color spaces, properties, metrics (3)

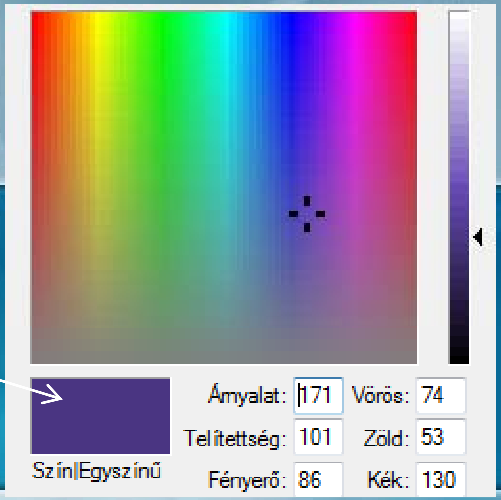
- **Description of the perceptual attributes of color :**

1. Brightness = how bright or dark a color looks;

2. Hue = the dominant wavelength of the color; 4 fundamental hues in the human description of colors: red, yellow, green, blue;

3. Saturation = measure of the purity of the color; if the spectrum of the radiation for the color is very narrow => ***high saturation***; the non-saturated colors = greys





3. Color spaces, properties, metrics (4)

- **Conventional color spaces:**
  - Reversible transforms of the primary (RGB) color space
  - Classified as linear and non-linear
  - **Linear transforms to obtain color spaces =** rotations and scalings of the RGB cube (OPP, YUV, YIQ, YCbCr, XYZ, Ohta $I_1I_2I_3$ ...)

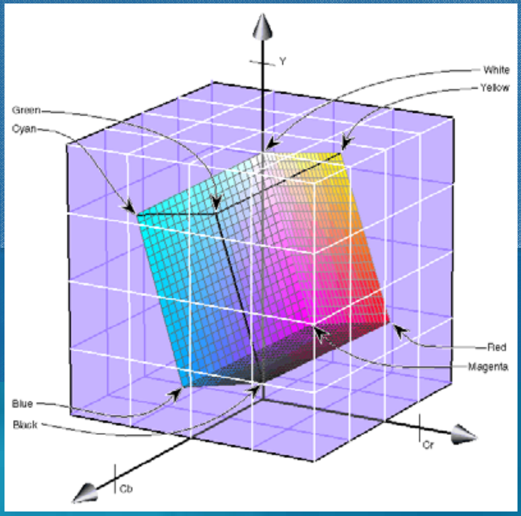
$$\mathbf{s}_{(C)} = \mathbf{T}^{(C)}\mathbf{s}, \quad \forall \mathbf{s} \in S_{(R,G,B)}.$$

$$\mathbf{s} = \left(\mathbf{T}^{(C)}\right)^{-1}\mathbf{s}_{(C)}, \quad \forall \mathbf{s}_{(C)} \in S^{(C)}.$$

$$\mathbf{T}^{(OPP)} = \begin{bmatrix} 1 & 1 & 1 \\ -1 & -1 & 2 \\ 1 & -2 & 1 \end{bmatrix}; \quad \mathbf{T}^{(YUV)} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.1 \end{bmatrix}; \quad \mathbf{T}^{(YIQ)} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix}.$$

$$\mathbf{T}^{(YCrCb)} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.5 & -0.4187 & -0.0813 \\ -0.1687 & -0.3313 & 0.5 \end{bmatrix}. \quad \mathbf{T}^{(XYZ)} = \begin{bmatrix} 0.49 & 0.31 & 0.2 \\ 0.177 & 0.812 & 0.011 \\ 0 & 0.01 & 0.99 \end{bmatrix}.$$

$$\mathbf{T}^{(I_1I_2I_3)} = \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/2 & 0 & -1/2 \\ -1/4 & 1/2 & -1/4 \end{bmatrix}.$$



3. Color spaces, properties, metrics (5)

- **Conventional color spaces (2):**
  - **Non-linear transforms to obtain color spaces =>** needed to match the perceptual color attributes by their coordinates (CIE L*a*b*, CIE L*u*v*, HSV, HLS, HSI, Munsell.

$$
\begin{aligned}
L^* &= 116 f\left(\frac{Y}{Y_0}\right) - 16\\
a^* &= 500\left[f\left(\frac{X}{X_0}\right) - f\left(\frac{Y}{Y_0}\right)\right]\\
b^* &= 200\left[f\left(\frac{Y}{Y_0}\right) - f\left(\frac{Z}{Z_0}\right)\right]
\end{aligned}
$$

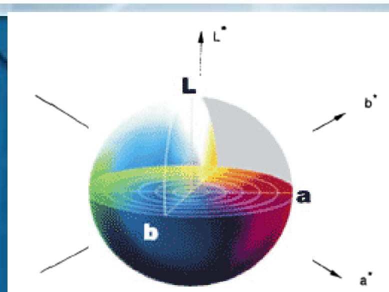
$$
f(x) = \begin{cases} x^{\frac{1}{3}}, & x > 0.008856\\ 7.787x + \frac{16}{116} & \text{otherwise,} \end{cases}
$$

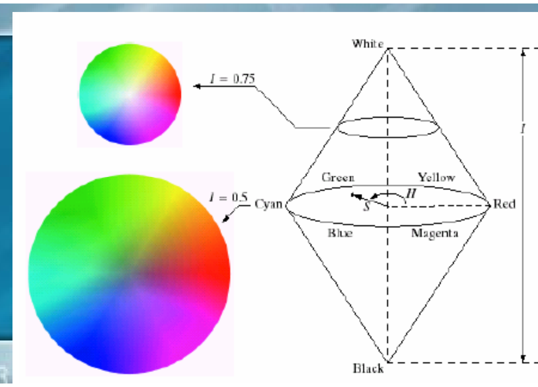
$$
H = \begin{cases} \theta & \text{if } B \le G\\ 360 - \theta & \text{if } B > G \end{cases} \quad \text{with } \theta = \cos^{-1}\left\{ \frac{\frac{1}{2}[(R-G)+(R-B)]}{\left[(R-G)^2 + (R-B)(G-B)\right]^{\frac{1}{2}}} \right\}
$$

$$
S = 1 - \frac{3}{(R+G+B)}[\min(R,G,B)]
$$

$$
I = \frac{1}{3}[R+G+B]
$$

RG sector ($0 \le H < 120$)	GB sector ($120 \le H < 240$)	BR sector ($240 \le H < 360$)
$B = I(1-S)$	$R = I(1-S)$	$G = I(1-S)$
$R = I\left[1 + \frac{S\cos H}{\cos(60-H)}\right]$	$G = I\left[1 + \frac{S\cos(H-120)}{\cos(60-(H-120))}\right]$	$B = I\left[1 + \frac{S\cos(H-240)}{\cos(60-(H-240))}\right]$
$G = 1-(R+B)$	$B = 1-(R+G)$	$R = 1-(G+B)$





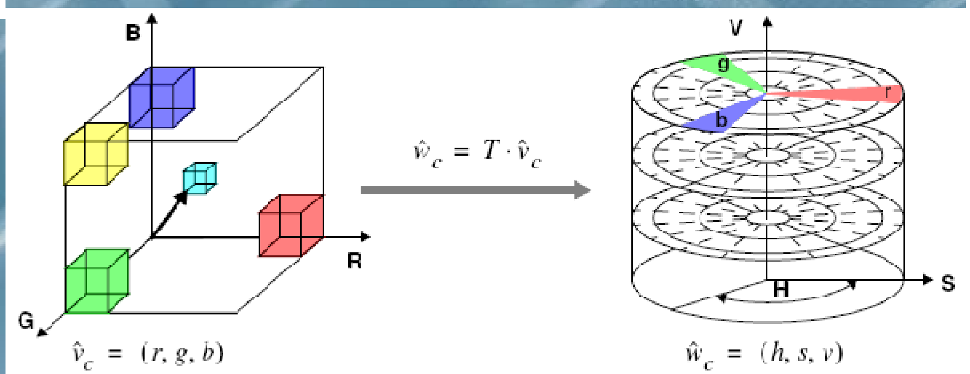
3. Color spaces, properties, metrics (6)

- Denote: r, g, b – color primaries normalized to [0;1]
  => HSV space transformations:

$$\mathbf{v} = \max(r, g, b), \qquad \mathbf{s} = \frac{\mathbf{v} - \min(r, b, g)}{\mathbf{v}}$$

$$\acute{r} = \frac{\mathbf{v} - r}{\mathbf{v} - \min(r, b, g)}, \quad \acute{g} = \frac{\mathbf{v} - g}{\mathbf{v} - \min(r, b, g)}, \quad \acute{b} = \frac{\mathbf{v} - b}{\mathbf{v} - \min(r, b, g)}$$

$$6\mathbf{h} = \begin{cases} 5 + \acute{b} & \text{if } r = \max(r, g, b) \text{ and } g = \min(r, b, g) \\ 1 - \acute{g} & \text{if } r = \max(r, g, b) \text{ and } g \neq \min(r, b, g) \\ 1 + \acute{r} & \text{if } g = \max(r, g, b) \text{ and } b = \min(r, b, g) \\ 3 - \acute{b} & \text{if } g = \max(r, g, b) \text{ and } b \neq \min(r, b, g) \\ 3 + \acute{g} & \text{if } b = \max(r, g, b) \text{ and } r = \min(r, b, g) \\ 5 - \acute{r} & \text{otherwise} \end{cases}$$



Reverse transform:

$$\alpha = 6\mathbf{h} - \mathrm{round}(6\mathbf{h}), \qquad \omega_1 = (1 - \mathbf{s}) * \mathbf{v},$$
$$\omega_2 = (1 - (\mathbf{s} * \alpha)) * \mathbf{v}, \qquad \omega_3 = (1 - (\mathbf{s} * (1 - \alpha))) * \mathbf{v}$$

$$\mathbf{r} = \begin{cases} \mathbf{v} & \text{if } \alpha = 0 \text{ or } \alpha = 5 \\ \omega_1 & \text{if } \alpha = 2 \text{ or } \alpha = 3 \\ \omega_2 & \text{if } \alpha = 1 \\ \omega_3 & \text{if } \alpha = 4 \end{cases} \qquad \mathbf{g} = \begin{cases} \mathbf{v} & \text{if } \alpha = 1 \text{ or } \alpha = 2 \\ \omega_1 & \text{if } \alpha = 4 \text{ or } \alpha = 5 \\ \omega_2 & \text{if } \alpha = 3 \\ \omega_3 & \text{if } \alpha = 0 \end{cases} \qquad \mathbf{b} = \begin{cases} \mathbf{v} & \text{if } \alpha = 3 \text{ or } \alpha = 4 \\ \omega_1 & \text{if } \alpha = 0 \text{ or } \alpha = 1 \\ \omega_2 & \text{if } \alpha = 5 \\ \omega_3 & \text{if } \alpha = 2 \end{cases}$$

3. Color spaces, properties, metrics (7)

- **Ad-hoc color spaces:**
  - Ideea: define the color space according to the ***most characteristic color components of a set of images*** ⇔ application-dependent
    => e.g. YST color space for human faces: Y – luminance; S – color average value from the set of faces; T – the orthogonal to Y and S
  - **Some basic approaches:**

    (1) For image segmentation:

    Fisher distance strategy to segment object-background (LDA generated color space)

    (2) For feature detection:

    Diversification principle strategy for selection & fusion of color components => automatically weight color components to balance between color invariance & discriminative power

    (3) For object tracking:

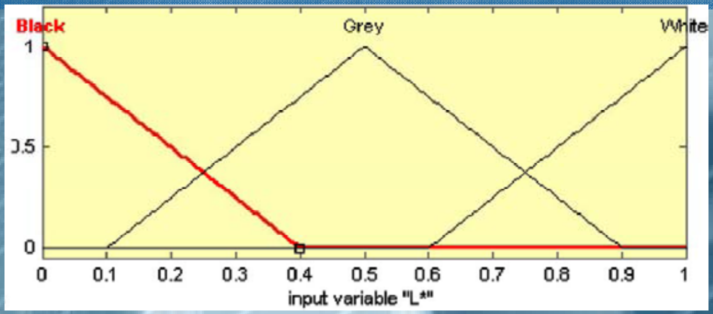
    Adaptive color space switching strategy => dinamically select the best color space for given environment lighting (from all conventional color spaces)

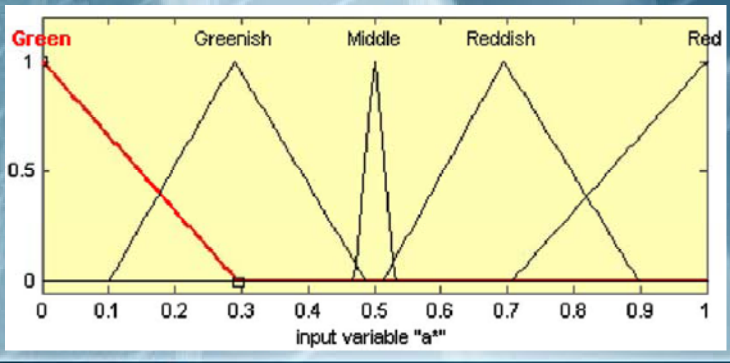
3. Color spaces, properties, metrics (8)

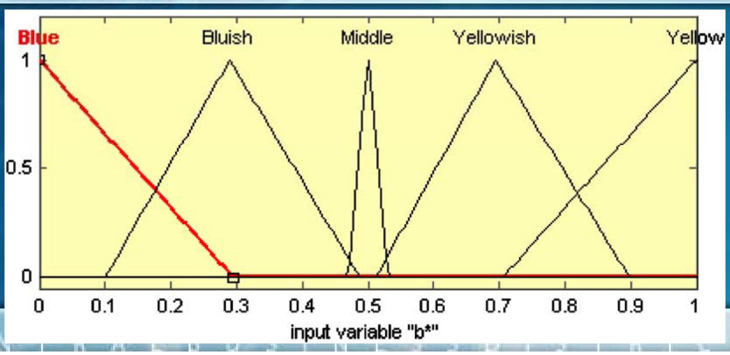
- **Color spaces defined on the human perceptual colors:**
  According to human psychology of color perception => 10 dominant perceived colors:
  (1) black, grey, white, red, brown, yellow, green, blue, cyan, magenta; or alternatively (other sources),
  (2) black, grey, white, red, brown, yellow, green, blue, pink, orange
- **Some approaches:**
  (1) Fuzzy logic based:

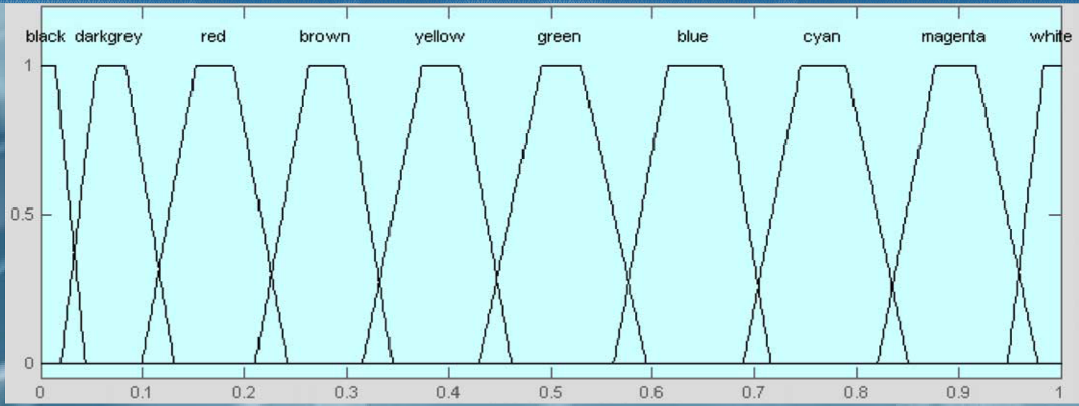
3. Color spaces, properties, metrics (9)

- **The knowledge base of the fuzzy system:**









Fuzzy rules (27 in total, see [Konstantinidis05]):

R1: If L is Black and a* is Middle and b* is Middle
then Perceptual Color is Black

R2: If L is White and a* is Middle and b* is Middle
then Perceptual Color is White

R3: If a* is Reddish and b* is Yellow
then Perceptual Color is Brown

…

3. Color spaces, properties, metrics (10)

(2) Color mixture based approach:

- Inspired by color mixing by painters;
- Any color can be obtained as an additive mixing of R, G, B in the right amount => decompose any color in 8 layers, $K_0 \ldots K_7$; $K_i=(R_i, G_i, B_i)$, i=0,1,...,7; i – one bit in the 8-bits representation per color primary
- Assign a scalar value v to any mixture of colors as follows:

$$v = \sum_{i=0}^{7} \frac{2^i (R_i 2^2 + G_i 2 + B_i)}{2^8 - 1}$$ => v=0 – black ... 7 - white

- Clasiffy any color according to its color mixture value v:

v	Classification of the Pixel Color
$0 \le v < 0.875$	Black
$0.875 \le v < 1.75$	Blue
$1.75 \le v < 2.625$	Green
$2.625 \le v < 3.5$	Cyan
$3.5 \le v < 4.375$	Red
$4.375 \le v < 5.25$	Pink
$5.25 \le v < 6.125$	Yellow
$6.125 \le v \le 7$	White

3. Color spaces, properties, metrics (11)

- **Color difference metrics in color spaces:**
  - **In linear transformed-based color spaces =>** Euclidian metric – common choice; other metrics are also possible, e.g. Mahalanobis
  - **In non-linear transformed based spaces =>** metrics should take into account what is linear and what is angular! (i.e. see hue! – an angle)
  - **Some basic metrics:**

    (1) Variants of Euclidian distance for linear spaces:

    Minkowski distance (q=1 – city-block; q=2 – Euclidian):

$$d(i,j)=\sqrt[q]{(|x_{i_1}-x_{j_1}|^q+|x_{i_2}-x_{j_2}|^q+\ldots+|x_{i_p}-x_{j_p}|^q)}$$

Mahalanobis distance:

$$mahalanobis(x,y)=\sqrt{(x-y)\Sigma^{-1}(x-y)^T}$$

3. Color spaces, properties, metrics (12)

- **Color difference metrics in color spaces – contnd.:**

(2) CIEDE2000:

defined for CIELAB space:

$$\Delta E_{00}(L_1^*, a_1^*, b_1^*; L_2^*, a_2^*, b_2^*) = \Delta E_{00}^{12} = \Delta E_{00}$$

$$\begin{aligned}
C_{i,ab}^* &= \sqrt{(a_i^*)^2 + (b_i^*)^2} \qquad i = 1,2 \\
\bar{C}_{ab}^* &= \frac{C_{1,ab}^* + C_{2,ab}^*}{2} \\
G &= 0.5\left(1 - \sqrt{\frac{\bar{C}_{ab}^{*\,7}}{\bar{C}_{ab}^{*\,7} + 25^7}}\right) \\
a_i' &= (1+G)a_i^* \qquad i = 1,2 \\
C_i' &= \sqrt{(a_i')^2 + (b_i^*)^2} \qquad i = 1,2 \\
h_i' &= \begin{cases} 0 & b_i^* = a_i' = 0 \\ \tan^{-1}(b_i^*, a_i') & \text{otherwise} \end{cases} \qquad i = 1,2
\end{aligned}$$

$$\begin{aligned}
\Delta L' &= L_2^* - L_1^* \\
\Delta C' &= C_2' - C_1' \\
\Delta h' &= \begin{cases} 0 & C_1'C_2' = 0 \\ h_2' - h_1' & C_1'C_2' \neq 0;\ |h_2' - h_1'| \leq 180^\circ \\ (h_2' - h_1') - 360 & C_1'C_2' \neq 0;\ (h_2' - h_1') > 180^\circ \\ (h_2' - h_1') + 360 & C_1'C_2' \neq 0;\ (h_2' - h_1') < -180^\circ \end{cases} \\
\Delta H' &= 2\sqrt{C_1'C_2'}\sin\left(\frac{\Delta h'}{2}\right) \\
\bar{L}' &= (L_1^* + L_2^*)/2 \\
\bar{C}' &= (C_1' + C_2')/2
\end{aligned}$$

3. Color spaces, properties, metrics (12)

$$
\begin{aligned}
\bar{h}' &= \begin{cases} \frac{h_1'+h_2'}{2} & |h_1'-h_2'| \leq 180^\circ; C_1'C_2' \neq 0 \\ \frac{h_1'+h_2'+360^\circ}{2} & |h_1'-h_2'| > 180^\circ; (h_1'+h_2') < 360^\circ; C_1'C_2' \neq 0 \\ \frac{h_1'+h_2'-360^\circ}{2} & |h_1'-h_2'| > 180^\circ; (h_1'+h_2') \geq 360^\circ; C_1'C_2' \neq 0 \\ (h_1'+h_2') & C_1'C_2' = 0 \end{cases} \\
T &= 1 - 0.17\cos(\bar{h}'-30^\circ) + 0.24\cos(2\bar{h}') \\
&\quad +0.32\cos(3\bar{h}'+6^\circ) - 0.20\cos(4\bar{h}'-63^\circ) \\
\Delta\theta &= 30\exp\left\{-\left[\frac{\bar{h}'-275^\circ}{25}\right]^2\right\} \\
R_C &= 2\sqrt{\frac{\bar{C}'^7}{\bar{C}'^7+25^7}} \\
S_L &= 1 + \frac{0.015(\bar{L}'-50)^2}{\sqrt{20+(\bar{L}'-50)^2}} \\
S_C &= 1 + 0.045\bar{C}' \\
S_H &= 1 + 0.015\bar{C}'T \\
R_T &= -\sin(2\Delta\theta)R_C \\
\Delta E_{00}^{12} &= \Delta E_{00}(L_1^*, a_1^*, b_1^*; L_2^*, a_2^*, b_2^*) \\
&= \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2 + R_T\left(\frac{\Delta C'}{k_C S_C}\right)\left(\frac{\Delta H'}{k_H S_H}\right)}
\end{aligned}
$$

4. Basic color image processing

- **Important note: Color image processing is not merely the processing of 3 monochrome channels!!!**

  … maybe 1 exception: Retinex algorithm

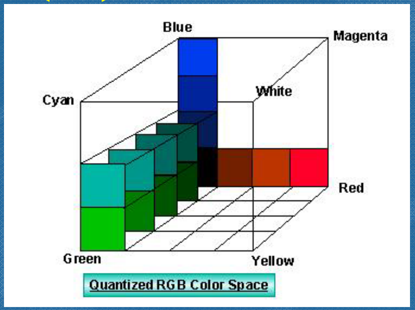
- **Yet => some generalizations and applications of monochrome (grey-level) image processing can be derived/used in color image processing and analysis:**

  - **Generalization of scalar algorithms to the vectors case (color space)**
  - **Processing of the luminance (brightness) component alone**
  - **Independent & different processing of each coordinate, after the color space transform** (linear or non-linear transform)

4.1. Color image quantization (1)

- **Goal of quantization:** build a reduced color space, with the smallest possible number of colors (the representative image colors), so that the perceived difference between the quantized image and original image → 0.
- **Open problem:** definition of “perceived difference”;
  - 1st approach: minimize the sum of distances between colors and the centers of color clusters resulting in the quantization process (⇔ minimize the sum of distances within each cluster)
  - 2nd approach: maximize the sum of distances between the colors in different clusters (⇔maximize the sum of distances between cluster pairs)
- **Typical approach for color space quantization: VQ**

4.1. Color image quantization (2)

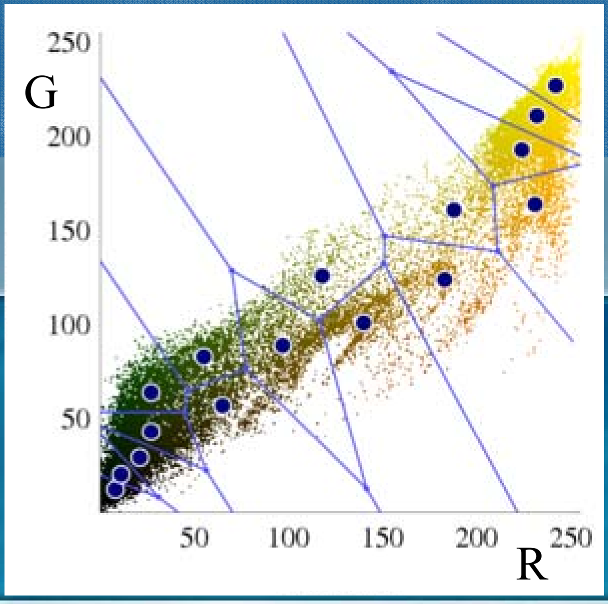


- **Vectorial Quantization (VQ) of the color space:**
  - Several versions; all based on LBG original algorithm
  - Motivation: reduce (usually drastically!) the number of colors in a group of images. How? **Cluster** similar colors together (color points = vectors => the name "***vector quantization***" = ***VQ***); determine the **cluster centers**; replace each image color with the closest cluster center

B=0 for all pixels

VQ codebook

(Voronoi diagram)



4.1. Color image quantization (3)

- **Vectorial Quantization (VQ) of the color space – basic algorithm:**
  - Let: N – # of colors in the (set of) image (s); M – target number of colors (M<<N); each color = $\mathbf{s}_i[3\times1]$ (e.g. $\mathbf{s}_i=[R\ G\ B]^T$), $i=1,2,\ldots,N$
  - **Algorithm:**

    1. **Initialization:** choose M "codewords", { $\mathbf{s}_{q1}$, $\mathbf{s}_{q2}$,…, $\mathbf{s}_{qM}$} lying in the color space chosen for quantization ⇔ *codeword initialization*
    2. **Codebook optimization:**

       2.1. For each $i=1,2,\ldots,N$, assign $\mathbf{s}_i$ to the cluster k that satisfies:

       $$k = \arg\min_{j=1,2,\ldots,M} d\left(\mathbf{s}_i, \mathbf{s}_{qj}\right)$$

       => The initial partition regions = the initial clusters B_1, B_2,…,B_M.

       2.2. Compute the overall distortion:

       $$D = \frac{1}{N}\sum_{j=1}^{M}\sum_{\mathbf{s}_i\in B_j} d\left(\mathbf{s}_i, \mathbf{s}_{qj}\right)$$

       2.3. If $D>\varepsilon$ => update codewords: $\mathbf{s}_{qj} = \frac{1}{card\left(B_j\right)}\sum_{\mathbf{s}_i\in B_j}\mathbf{s}_i$

       and go to step 2. Otherwise => convergence reached => final codebook and codewords.

4.2. Color image filtering (1)

- **Most popular filtering goal :** remove noise (color noise) from the original

- Why is noise disturbing?
    - **Perceptually:** image appearing visually unpleasant,...
    - **For analysis applications:** noise = high frequency => same as sharp edges...

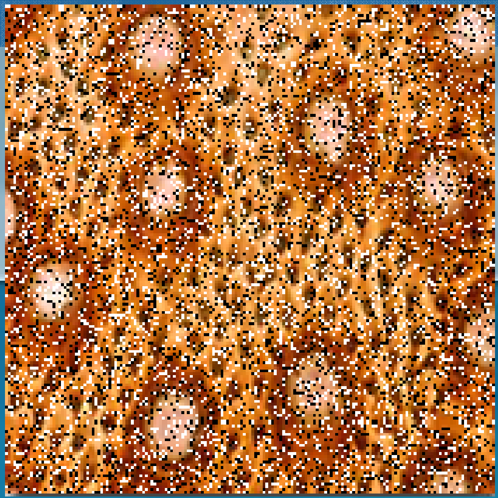
- **Noise filtering** algorithms for **color images**:
    - Most common types of noise: impulse noise; Gaussian noise; speckle noise; stripping noise
    - Several types of vector filtering operators derived in last 10 years
    - Important class of noise filtering operators for color images: rank vector filters
    - Open issues: **develop adaptive filters for color images,** to preserve fine details & reduce all types of noise efficiently (including additive!) ⇔ filters capable to **adapt to local image statistics!**
    - Other filtering approaches: morphological operators; wavelets; PDEs...

4.2. Color image filtering (2)

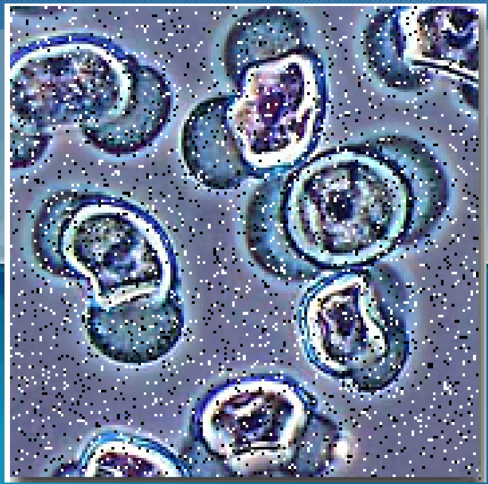
Original (noise-free)

- **Vector median filters for color images:**
  - Particular case of rank filters
  - Principle: for each pixel location (i, j):
    - take the brightness/color values in a window $W_{(i,\,j)}$
    - order the brightness/color values in increasing order
    - output: new brightness/color at (i, j) = middle of string
  - Very useful for impulse color noise:

Original (noise-free)



30% impulse noise



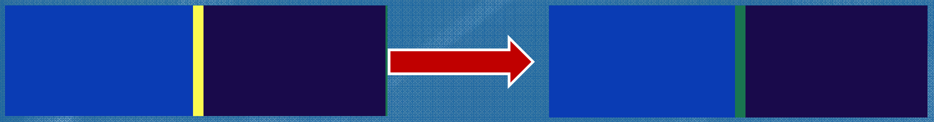
10% impulse noise

4.2. Color image filtering (3)

- **Vector median filters for color images** – some practical algorithms:

Note:

- o Biggest problem in vector median filtering generalization for color images: (1) **how** to define the ordering?; (2) **what** means "increasing color values"?
- o "Brute approach" (i.e. in RGB space => treat each channel independently, apply 3 median filters independently) **does not work!** (color distortion): 3×1 window, $\mathbf{s}_1$=[7 117 182], $\mathbf{s}_2$=[250 250 80], $\mathbf{s}_3$=[25 10 75]=> filter independently: $\mathbf{s}$=[25 117 80]...



=> solutions:

1. The Adaptive Scalar Median Filter:

- Consider 2 representations of the image: in RGB and HSI color space => denote: $\mathbf{s}_p=[R\ G\ B]^T$; $\mathbf{s}_h=[h\ s\ i]^T$
- W – any window around the current pixel (x,y); e.g. $W_{(x,y)}[3\times3]$.
- Let: r_m, g_m, b_m – average R, G, B values inside $W_{(x,y)}$;
- Compute: $[h_m\ s_m\ i_m]^T = \mathrm{HSI}([r_m\ g_m\ b_m]^T)$;

4.2. Color image filtering (4)

- Additional : (x_R,y_R) = pixel position in $W_{(x,y)}$ that satisfies:
  $R(x_R,y_R)=\text{median}\{R(x,y)|(x,y) \text{ in } W_{(x,y)}\}$
  (x_G,y_G) = pixel position in $W_{(x,y)}$ that satisfies:
  $G(x_G,y_G)=\text{median}\{G(x,y)|(x,y) \text{ in } W_{(x,y)}\}$
  (x_B,y_B) = pixel position in $W_{(x,y)}$ that satisfies:
  $B(x_B,y_B)=\text{median}\{B(x,y)|(x,y) \text{ in } W_{(x,y)}\}$

⇒ **We can now build a "median matrix" M**[3×3]:

$$\mathbf{M} = \begin{bmatrix} R(x_R, y_R) & R(x_G, y_G) & R(x_B, y_B) \\ G(x_R, y_R) & G(x_G, y_G) & G(x_B, y_B) \\ B(x_R, y_R) & B(x_G, y_G) & B(x_B, y_B) \end{bmatrix}$$

- ***Note:***
  - The diagonal of **M** – most likely to be the median color, ***but*** is a ***new color!!!***
  - Any column of **M** = an existing color , **but** not necessarily really the median!
  - **=> *Virtually*** one can select as filter's output ***any*** combination of RGB values
    => how do we know which one is ***optimal?***

4.2. Color image filtering (5)

- ***Selection criteria*** for the output color of the adaptive scalar median filter:

C1. The hue changes should be minimized
C2. The shift of saturation should be as small as possible.
C3. An increase in saturation is preferable to a decrease in saturation
C4. Maximize the relative luminance contrast.

$\Rightarrow$ ***In mathematical (algorithmical) form:***

1. Find (l,p,q) so that:

$$\left|HSI\left(\left[\mathbf{M}(1,l)\,\mathbf{M}(2,p)\,\mathbf{M}(3,q)\right]\right)(1)-h_m\right|=\left|h(l,p,q)-h_m\right|=\min_{(i,j,k)\in\{1,2,3\}^3}\left|h(i,j,k)-h_m\right|$$

2. If (l,p,q) is unique => output = [$\mathbf{M}(1,l)$ $\mathbf{M}(2,p)$ $\mathbf{M}(3,q)$]^T ; ***otherwise:*** on the subset of (l,p,q) candidates, find (l',p',q') so that:

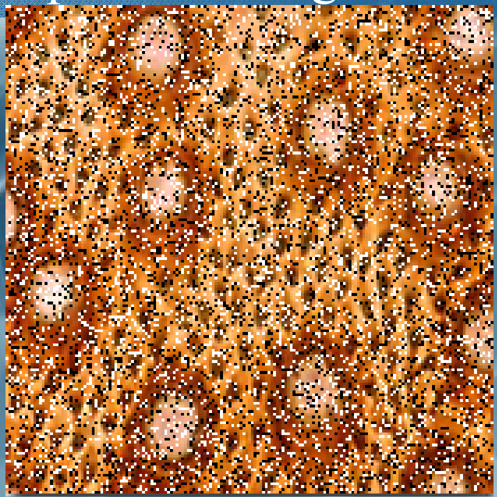
$$\left|HSI\left(\left[\mathbf{M}(1,l')\,\mathbf{M}(2,p')\,\mathbf{M}(3,q')\right]\right)(2)-s_m\right|=\left|s(l',p',q')-s_m\right|=\min_{(i,j,k)\in\{(l,p,q)\}}\left|s(i,j,k)-s_m\right|$$

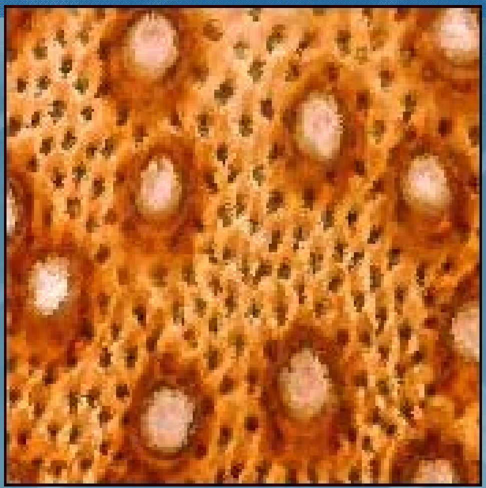
3. If (l',p',q') is unique => output = [$\mathbf{M}(1,l')$ $\mathbf{M}(2,p')$ $\mathbf{M}(3,q')$]^T ; ***otherwise:*** on the subset of (l',p',q') candidates, select the one with largest s and i.

4.2. Color image filtering (6)

- Results of scalar adaptive filtering:


original


noisy


filtered

4.2. Color image filtering (7)

2. The Vector Median Filter:

- Unlike the scalar adaptive median filter => it ***guarantees*** that its output = ***always a color that is present in the image window***
- Consider the RGB color space representation of the image, $\mathbf{s}=[R\ G\ B]^T$;
- W – any window around the current pixel (x,y); e.g. $W_{(x,y)}[3\times 3]$.
- $\|\bullet\|_L$ – some vector norm (e.g. Euclidian distance)
- Let: $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$= the colors inside $W_{(x,y)}$ => the ***vector median filter:***

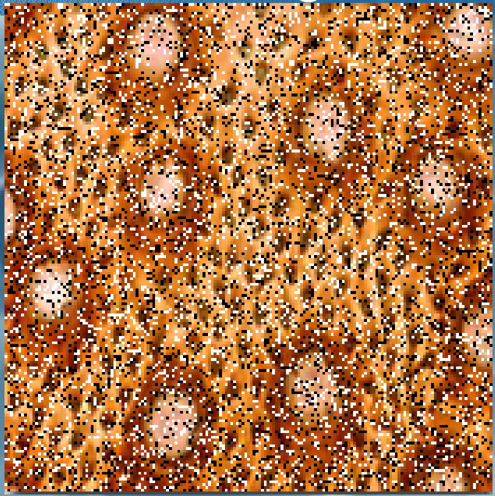
$$VM\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\} = \mathbf{s}_{VM}, \quad \mathbf{s}_{VM} \in \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$$

so that:

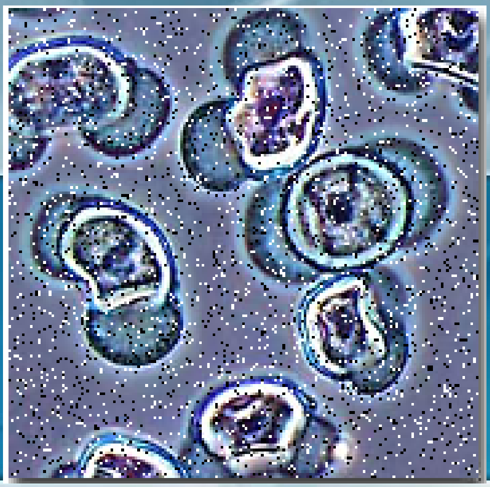
$$\sum_{i=1}^{N} \left\| \mathbf{s}_{VM} - \mathbf{s}_i \right\|_L \le \sum_{i=1}^{N} \left\| \mathbf{s}_j - \mathbf{s}_i \right\|_L, \ \forall j = 1,2,\ldots,N.$$

4.2. Color image filtering (8)

- Results of vector median filtering:


original noisy filtered


4.2. Color image filtering (9)

2. The Median Filter Based on Conditional Ordering in the HSV Space :

- Consider the representation of the image in HSV color space => denote: $\mathbf{s}=[h\ s\ v]^T$, h – angle, s,v – [0;1] valued
- W – any window around the current pixel (x,y); e.g. $W_{(x,y)}[3\times3]$.
- Principle of the conditional ordering based filter:
  - (1) select a-priori an *importance order* for the vectors' components
  - (2) order the vectors based on their components' relation in the predefined order

- In the HSV color space: conditional ordering based filtering principles:
  (1) sort the color vectors in W based on v: order from smallest to largest v
  (2) ***ordering colors with same v***: sort based on s: from largest to smallest s
  (3) ***ordering colors with same v and s:*** from smallest to largest h

4.2. Color image filtering (10)

⇒ **Define the operators:** $<_{hsv}$,$=_{hsv}$ **for color ordering in the HSV color space** as follows:

if: $\mathbf{s}_1 = [h_1 \quad s_1 \quad v_1]^T ; \mathbf{s}_2 = [h_2 \quad s_2 \quad v_2]^T$ = 2 colors in HSV,

then:

$$\mathbf{s}_1 <_{hsv} \mathbf{s}_2 \Leftrightarrow ((v_1 < v_2) \vee ((v_1 = v_2) \wedge (s_1 > s_2)) \vee ((v_1 = v_2) \wedge (s_1 = s_2) \wedge (h_1 < h_2)))$$

$$\mathbf{s}_1 =_{hsv} \mathbf{s}_2 \Leftrightarrow ((v_1 = v_2) \wedge (s_1 = s_2) \wedge (h_1 = h_2))$$

⇒ Let: $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$= the colors inside $W_{(x,y)}$=>the HSV conditional ordering median filter algorithm:

1. *Order* $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$ increasingly in respect to $<_{hsv}$:

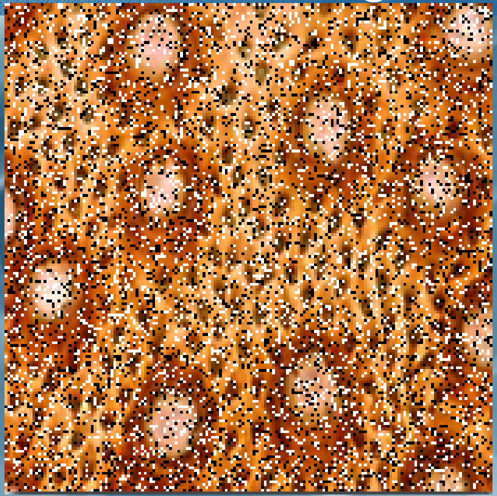
$$\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\} \rightarrow \{\mathbf{s}'_1, \mathbf{s}'_2, \ldots, \mathbf{s}'_N\}, \forall \mathbf{s}'_i \in \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}, \mathbf{s}'_1 <_{hsv} \mathbf{s}'_2 \ldots <_{hsv} \mathbf{s}'_N$$

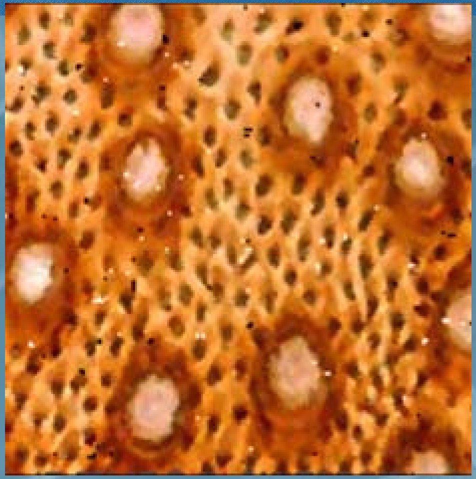
2. *Output* the color in the middle of the ordered strig: med$\{\mathbf{s}'_1, \mathbf{s}'_2, \ldots, \mathbf{s}'_N\}$

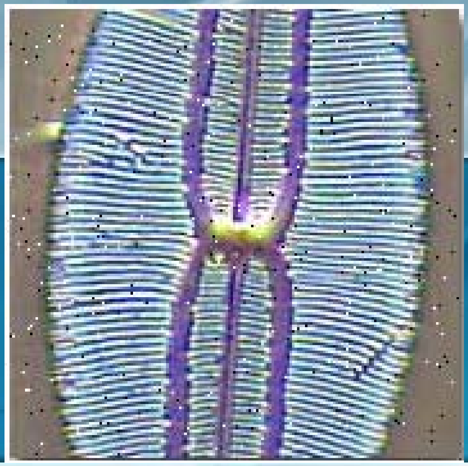
4.2. Color image filtering (11)

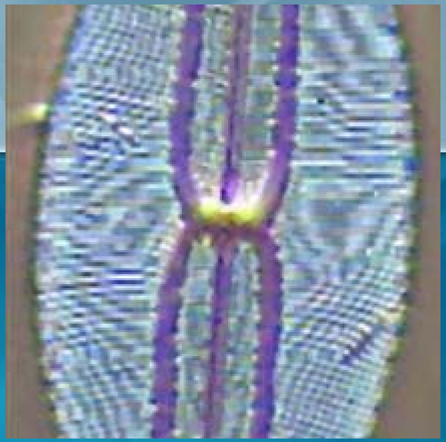
- Results of HSV conditional ordering median filter:


original


noisy


filtered





4.3. Color image enhancement (1)

- **Can have various goals** (more than grey level image enhancement) ; some typical:

1. Image contrast enhancement
2. Color enhancement ⇔ increase of color saturation, illuminant lighting compensation, etc.

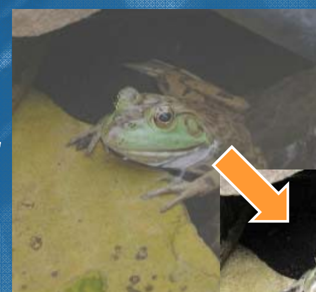
... and others....

pointwise operations

3. Image de-blurring
4. Edge enhancement

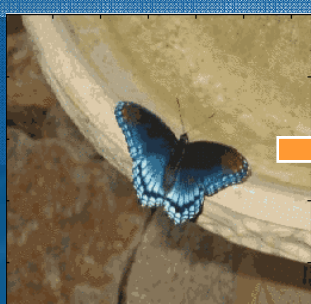
... and others...

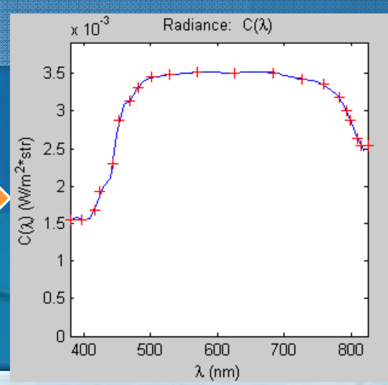
spatial operations



contrast

saturation





Changing illuminant

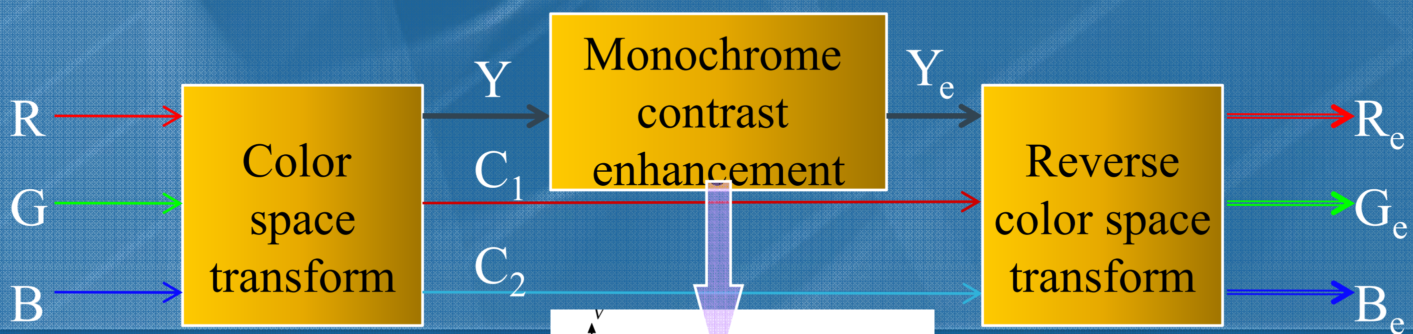
4.3. Color image enhancement (2)

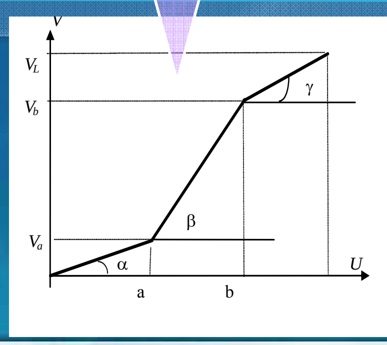
- **E.g. Contrast enhancement in color images:**
  - **Basic (popular) approach:**

    ⇒ human eye - 5× more sensitive to brightness contrast then color contrast

    ⇒ can achieve good contrast enhancement on brightness component alone!

    ⇒ typically:

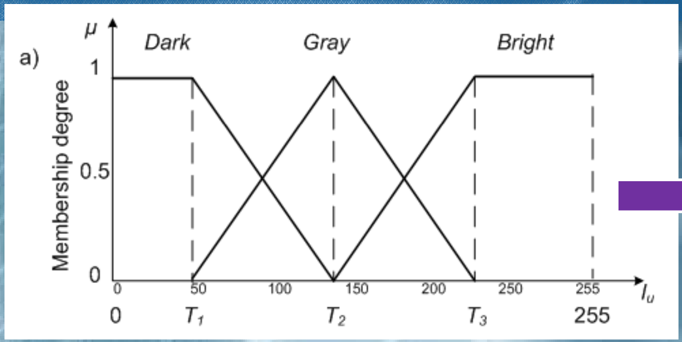




$$v=\begin{cases} mu & ,0\le u\le a\,,m=tg\alpha \\ n(u-a)+v_a & ,a\le u\le b\,,n=tg\beta \\ p(u-b)+v_b & ,b\le u\le L\,,p=tg\gamma \end{cases}$$

4.3. Color image enhancement (3)

- **A simple approach: fuzzy rule-based contrast enhancement:**

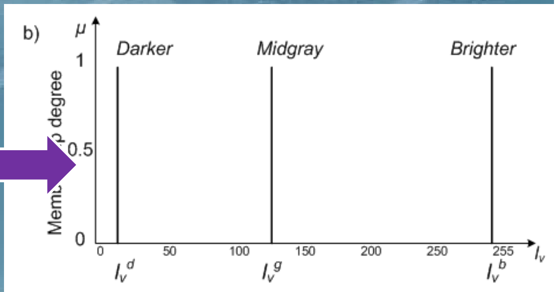


Fuzzy rules:

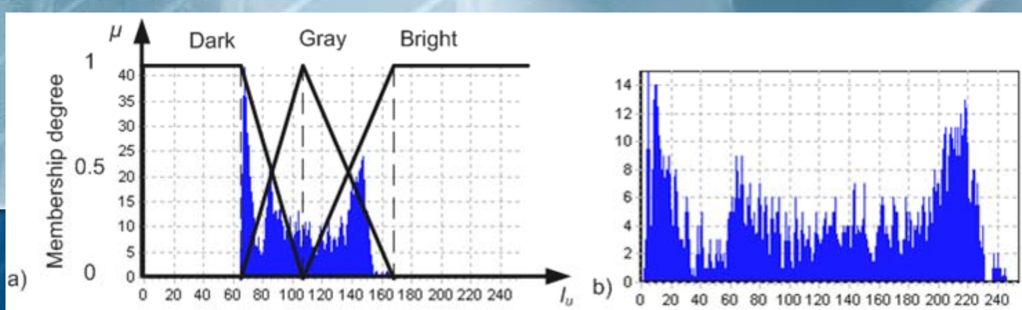
If Y is Dark => Y_e is Darker

If Y is Gray => Y_e is Midgray

If Y is Bright => Y_e is Brighter



4.3. Color image enhancement (4)



5. Color image segmentation

- **Segmentation =** partition the image in disjoint homogeneous regions
- ***"Good segmentation"*** (Haralick & Shapiro) ⇔:
  - Uniform + homogeneous regions in respect to some visual features
  - Regions interiors – simple, without many small holes
  - Adjacent regions – significantly different visual feature values
  - Region boundaries – simple, smooth, spatially accurate

- ***Formal definition***: **I** – image set of pixels => segmentation of **I** = the partition P of N subsets R_k ; H – some homogeneity predicate =>:

$$\bigcup_{k=1}^{N} R_k = \mathbf{I}; \quad R_k \bigcap R_l = \Phi, \forall k \neq l; \quad H(R_k) = true, \forall k; \quad H\left(R_k \bigcup R_l\right) = false, \forall k \neq l \quad adjacent.$$

- **Color & texture – basic homogeneity attributes for segmentation**
- **Main color image segmentation categories:**
  1. Feature space based methods => no spatial neighborhood constraints
  2. Image domain based methods => spatial neighborhood constraints
  3. Physics based methods => special class; not found on grey scale methods

5.1. Feature space color image segmentation (1)

- **"Generalizations" of classical grey scale image segmentation strategies**
- **Two main approaches:**
  1. Color clustering
  2. Histogram thresholding

- Main issue: what color features are the most suitable for clustering/histogram analysis? => application/image content dependent!
- Segmentation strategies => still research/open issues, since ***good segmentation = "basic ingredient" for good image analysis***

- ***Current state-of-the art trends:***
  - to combine the use of low level, intermediate level and high level features;
  - to use learning => supervised segmentation (model-based)
  - describe and make "clever" use of a-priori info!

5.1. Feature space color image segmentation (2)

1. Color clustering:

= Non-supervised classification of objects/pixels ⇔ algorithms that **generate classes/partitions without any a-priori knowledge**

=>All basic methods for **any feature vectors clustering** can be applied; any color space can be used => feature space = the color space; most common:

- ***K-means:*** (iterative procedure)

K – number of clusters (user–defined); $\mathbf{S}=\{\mathbf{s}_1, \mathbf{s}_2,\ldots, \mathbf{s}_N\}$ – pixels' colors; $V=\{v_1,\ldots,v_K\}$ – an initial random set of color prototypes; $\|.\|$ – a distance norm in the color space

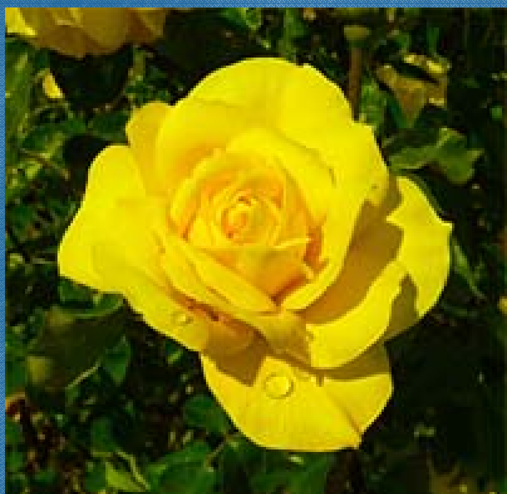
$U[K\times N]$ = membership degrees matrix for the N colors in **S** to the K classes: $U=\{u_{ji}\}$, $j=1,2,\ldots,K$; $i=1,2,\ldots,N$:

$$u_{ji} \in \{0;1\}, \quad u_{ji} = \begin{cases} 1, & \|\mathbf{s}_i - \mathbf{v}_j\|^2 = \min\limits_{k=1,2,\ldots,K} \|\mathbf{s}_i - \mathbf{v}_k\|^2 \\ 0, & otherwise \end{cases}.$$

Clustering objective: find **U**, V that minimize the cost function:

$$J(\mathbf{U},\mathbf{V}) = \sum_{j=1}^{K}\sum_{i=1}^{N} u_{ji}\|\mathbf{s}_i - \mathbf{v}_j\|^2.$$

K-means example



= Color image, B=0;

⇒ Extract 6 significant colors from the image; cluster them in 3 classes.

⇒ N=6; K=3; feature space: RGB; as B=0 => enough to use (R,G) => 2-D space

⇒ Data points: $\mathbf{s}_i[2\times 1]$, i=1,2,…,6,

$\mathbf{s}_1=[21\ 38]^T$; $\mathbf{s}_2=[20\ 40]^T$; $\mathbf{s}_3=[8\ 28]^T$; $\mathbf{s}_4=[8\ 25]^T$;

$\mathbf{s}_5=[246\ 185]^T$; $\mathbf{s}_6=[242\ 181]^T$

⇒ Goal: cluster the colors in 3 clases, K-means

⇒ Distance norm: Euclidian

⇒ Initialization: select the class centers;

e.g. $\mathbf{v}_1=\mathbf{s}_1$; $\mathbf{v}_2=\mathbf{s}_2$; $\mathbf{v}_3=\mathbf{s}_3$

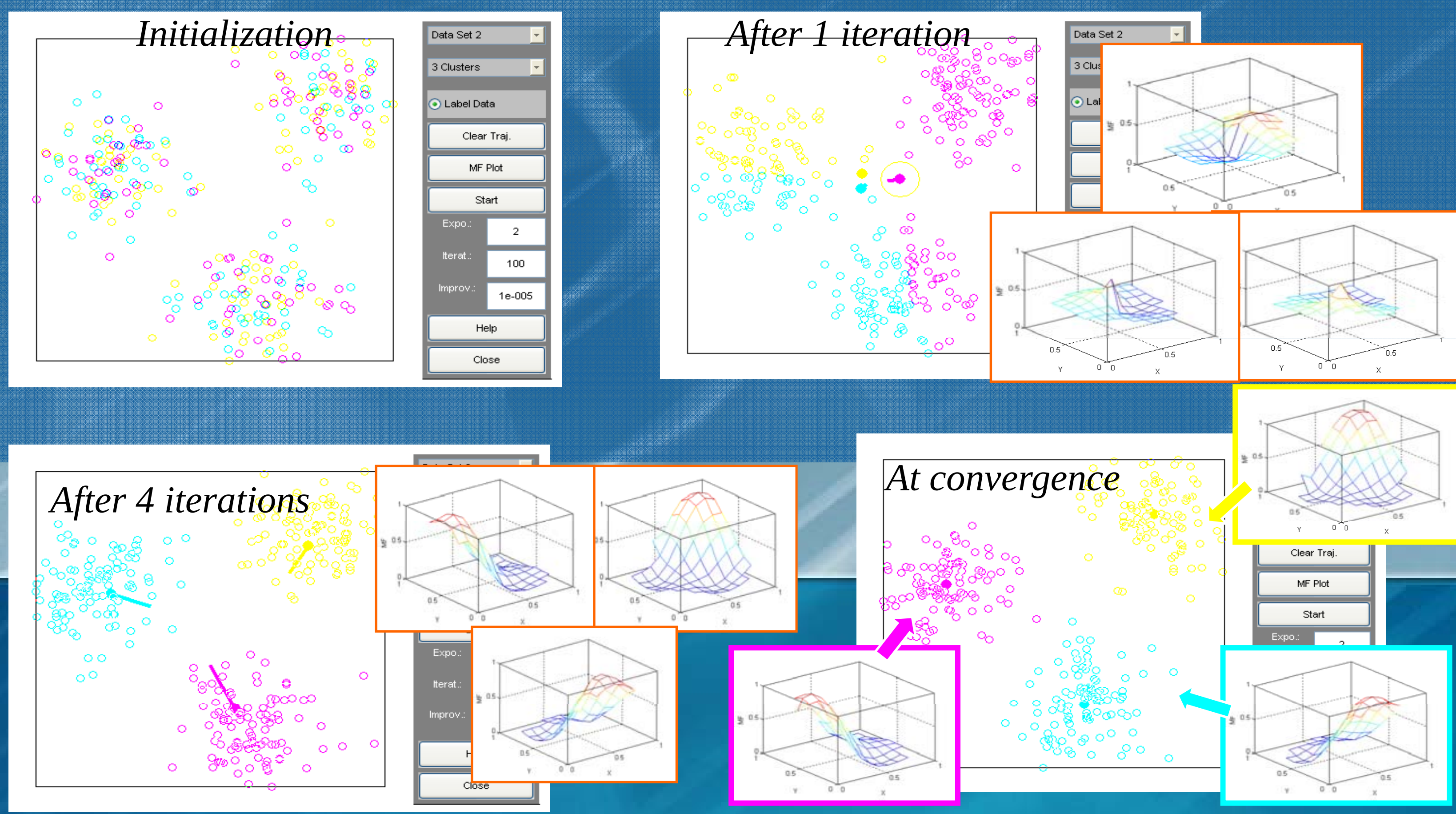
⇒ Compute the initial matrix $\mathbf{U}[3\times 6]$:

$$\mathbf{U}^0=\begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&1&1\\0&0&1&1&0&0\end{bmatrix}$$

Continue till convergence

5.1. Feature space color image segmentation (3)

- ***Fuzzy K-means*** (⇔ fuzzy c-means): the "soft version" of K-means



Fuzzy K-means algorithm:

Step 1. Initialization: specify K, U^0 – ***fuzzy partition!***; m; e (=convergence error); p=0 (iteration step).

Step 2. p=p+1

Compute V^p from $\mathbf{U}^{p-1}$ using:

$$\mathbf{v}_j^p = \frac{\sum_{i=1}^{N} \left(u_{ji}^{p-1}\right)^m \mathbf{x}_i}{\sum_{i=1}^{N} \left(u_{ji}^{p-1}\right)^m}, \quad \forall j = 1,2,\ldots,K.$$

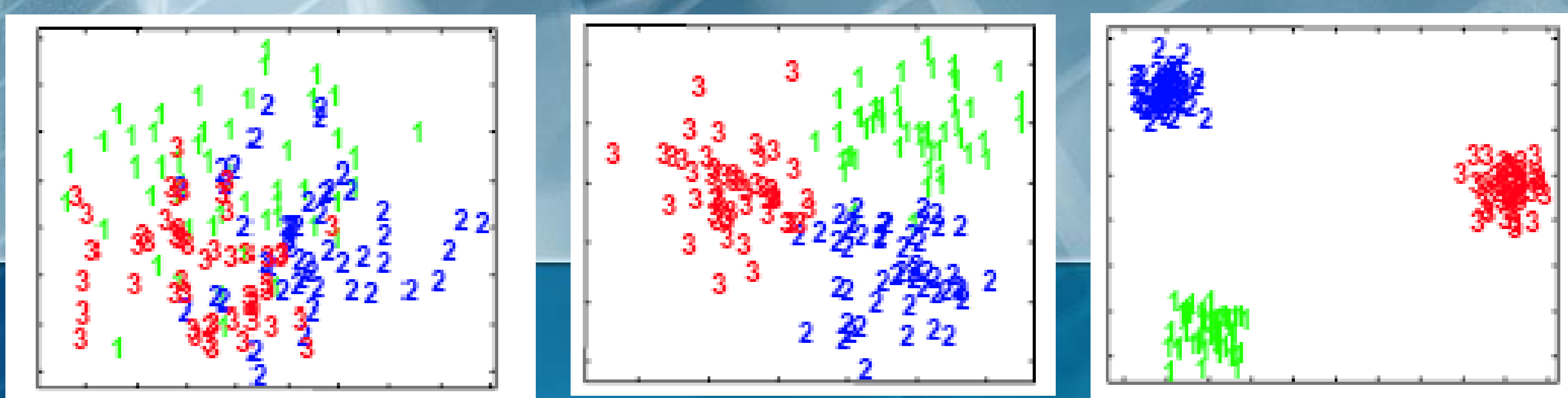
Compute $\mathbf{U}^p$ from V^p using:

$$u_{ji}^p = \left(\sum_{k=1}^{K} \left(\frac{\left\| \mathbf{x}_i - \mathbf{v}_j^p \right\|}{\left\| \mathbf{x}_i - \mathbf{v}_k^p \right\|} \right)^{\frac{2}{m-1}} \right)^{-1}, \quad \forall j = 1,2,\ldots,K, \forall i = 1,2,\ldots,N.$$

Step 3. If $\max_{j=1}^{K} \max_{i=1}^{N} \left(u_{ji}^p - u_{ji}^{p-1} \right) < e$ => STOP. Otherwise => go to Step 2.

5.1. Feature space color image segmentation (4)

- Many other clustering methods: ISODATA, mean shift, constrained gravitational clustering, graph partitioning, adaptive k-means, and supervised methods (***Bayesian color models, Kohonen maps, elipsoidal constrained color clusters***)
- ***Note:*** selection of the color space – application dependent; controls the success of correct clustering => quality of the segmentation!



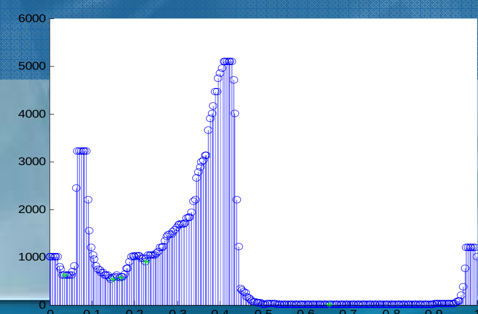
5.1. Feature space color image segmentation (5)

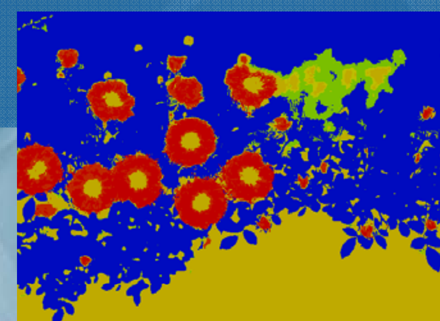
2. Histogram thresholding:

- Very popular for grey scale images: peaks & valleys detection; peaks = significant clusters; valleys = boundaries between clusters
- ***Main problem*** in generalization to color image segmentation: ***histogram = 3-D support function*** => unlike the 1-D support function for grey scales

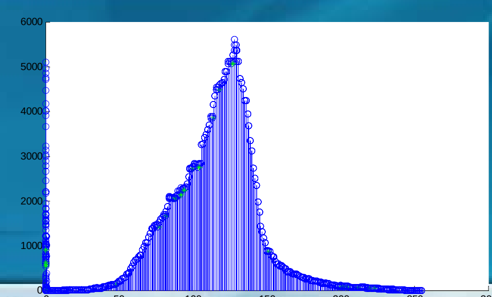
=> (1) Attempt to find the ***most relevant color feature*** to have a 1-D histogram in the color space case; commonly – use the hue H

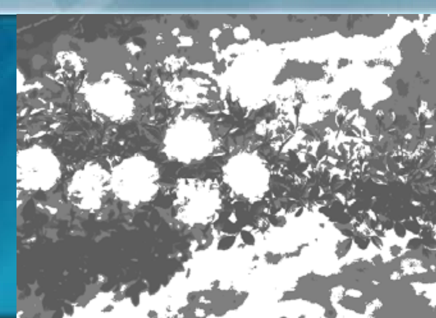
Hue





Brightness

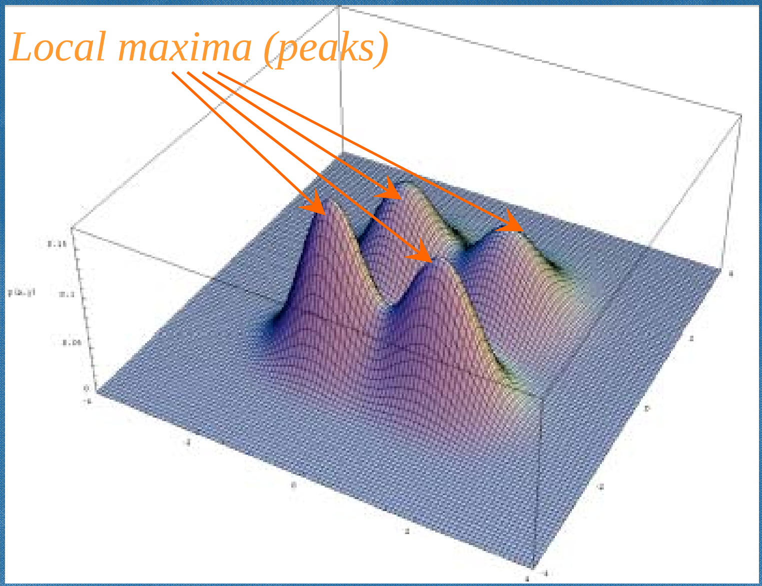




5.1. Feature space color image segmentation (6)

(2) Independently threshold the 3 color features histograms (in some color space) + use logical predicates to combine segmentation results

(3) Use pairwise features: e.g. (H,S)
=> 3-D surfaces as histograms
=> find peaks and valleys
=> segmentation



(4) Histograms modeling by Gaussian pdfs on each component in a decorrelated color space

.... Etc....

5.2. Image domain segmentation of color images

- **Previous techniques *don't guarantee spatial compactness of regions***

⇒ Image domain segmentation techniques add spatial constraints to improve segmentation (wrt compactness)

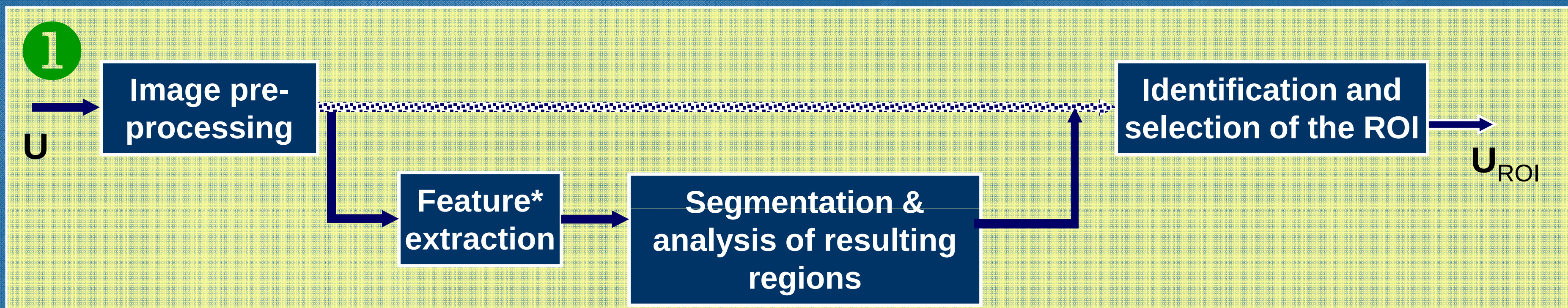
- **Two main approaches** (as in grey scale):
    1. Split – and – merge; e.g. most typical: quad-trees decomposition + merging
    2. Region growing; as in grey level case => need solutions to find good seeds

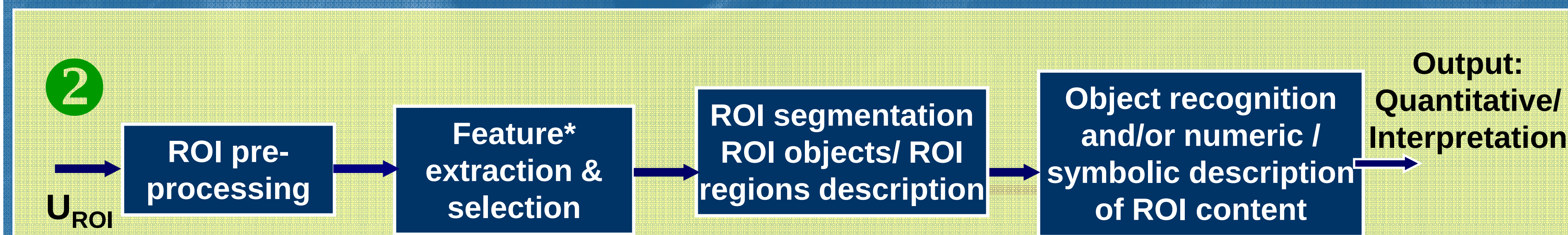
- **Main issue:** the ***similarity concept*** must be expressed in 3-D space! (distance measures ⇔ similarity measures between colors, not between grey levels) (E.g. use RGB and Euclidian distance as measure of "closeness" of colors)
- Some approaches use subsets of color features - i.e. H, S or H, V

- ***Note:*** edges can be also used; either on brightness, or the generalized 3-D gradient

6. Color image analysis

- **Analysis** – image content interpretation, far beyond processing & segmentation:





* Several studies say: ***color = the most expressive visual feature***

- ***Main challenges in color image analysis*** (esp. image retrieval, object recognition):
  (1) *develop high-level features for semantic modeling the image content;*
  (2) *fill the gap between existing (low-level, intermediate-level features) and high level features + variety of features that can be described by an observer*

6.1. Color features

- **What are they?**
  - Everything that can be extracted from color spaces
- **How can they be used?**
  - in color image indexing/retrieval: used to match objects by color similarity
  - in medical analysis, aerial imaging: used to classify color regions & to recognize specifically colored objects
  - Classical object matching applications (using color): color template matching; color histogram matching; hybrid models
  - More advanced use of color features: embed information about the ***spatial organization of colors*** (=intermediate level feature) & pixel independence relationships; => compare images with EMD (Earth Mover Distance)
- **Open issues?**
  - Usually – color features vary under various illuminant condition => suggested: define high-order invariant color features & entropy-based similarity measure
- **Standardizations:**
  - MPEG-7 color descriptors: color space; color quantization; dominant colors; scalable color; color layout; color structure; GOF/GOP color; room for more...

6.2. Color-based object tracking

- Many applications: surveillance; video analysis; robotics; videos coding; human-computer interaction; etc.

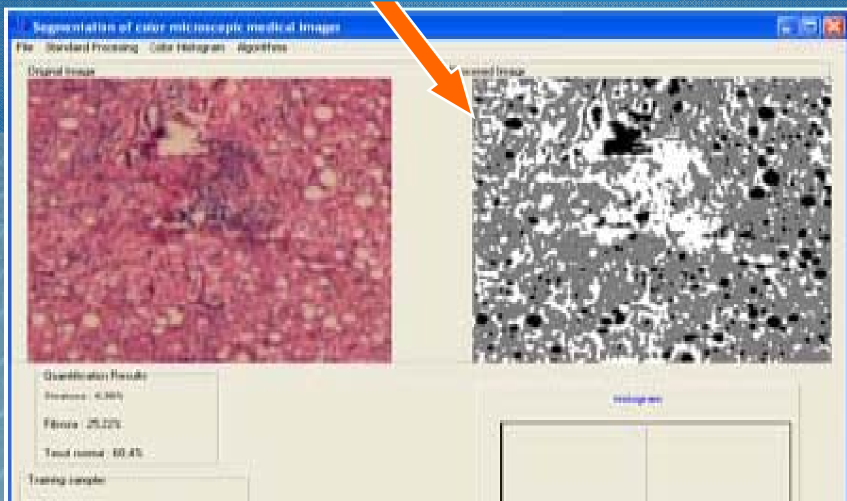
- Why is the color so useful for such applications?
  - robust in partial occlusion cases
  - robust against shape deformation & field of view changing

- **Main approaches: color models based:**
  - Semi-parametric models: mixtures of Gaussians (MoG) ; combined with EM
  - Non-parametric models: color histograms; combined with Bhattacharrya distance, mean-shift algorithm

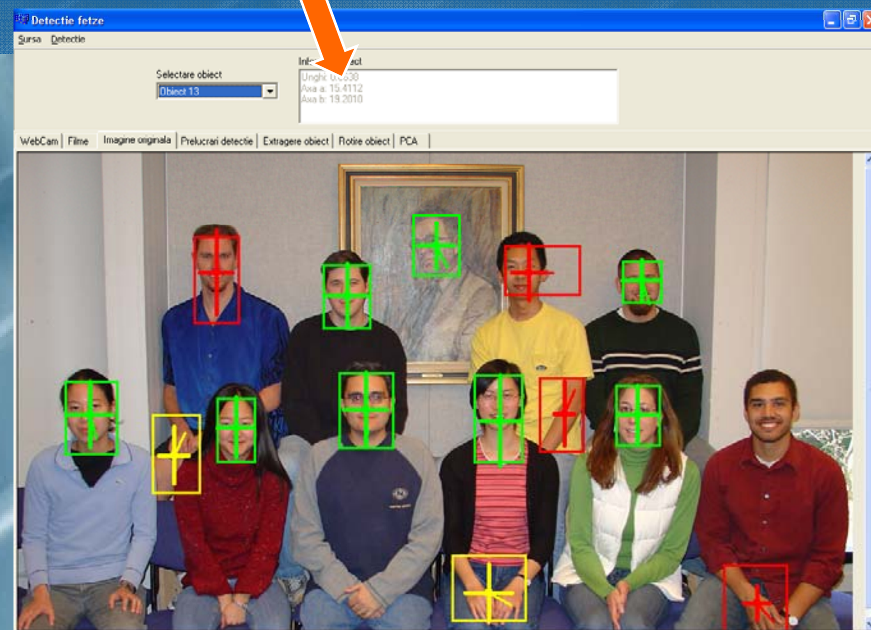
- **Other approaches:** stereo vision + color; active color appearance models

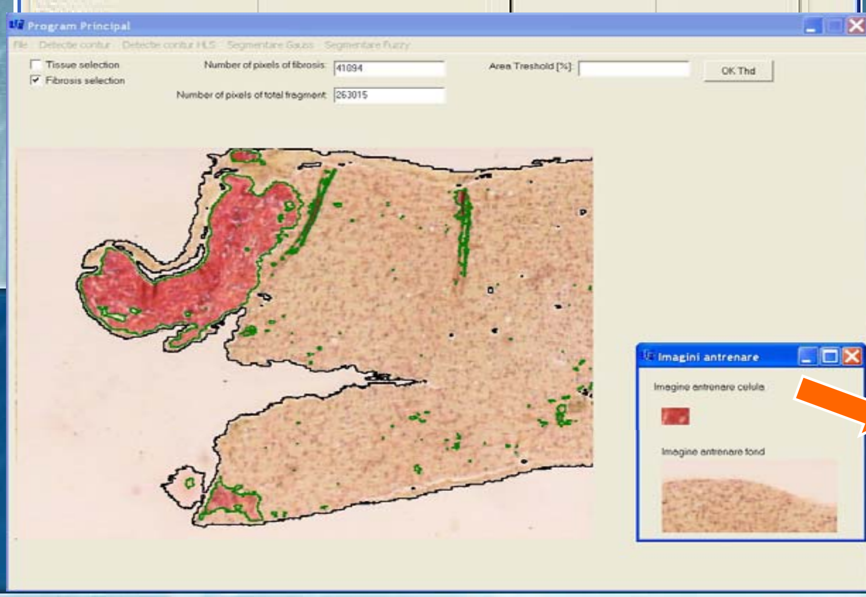
6.3. Some analysis examples

Cell counting



Face detection & localization





Liver biopsy morphometry

6.4. Some open issues: color saliency; color constancy

- **Color saliency:**
  - ***Color saliency models*** = model how HVS ***perceives*** color based on its spatial organization
  - ***Theory:*** HVS => ROI selection guided by neurological + cognitive processes
    - ***Neurological selection:*** by bottom-up (stimuli-based) info
    - ***Cognitive selection:*** by top-down (task-dependent) cues
  - ***Currently =>*** color models don't use color saliency info satisfactory (some saliency maps exist only from RGB data, not spatial info);

    => e.g. don't use HVS learned knowledge as: more attention given to color details than uniform large patches; color perception is depending on the surrounding colors

    => future research needed ***on developing perceptual multiscale saliency maps based on competition between bottom-up cues*** (color, intensity, orientation, location, motion)

6.4. Some open issues: color saliency; color constancy

- **Color constancy:**
  - In HVS: Color constancy = the subconscious ability to separate the illuminant spectral distribution from spectral surface reflectance function
    ⇔ to recognize the color appearance of an object invariant to illuminant
  - ⇒ In machines: Color constancy = ability to measure colors independent on the color of the light source (illuminant)
  - ⇒ Important goal, but ***very difficult to achieve; open research issue***

  - Some approaches:
    - Illuminant estimation algorithms: max-RGB, gray-world, gammut mapping, Bayesian models, neural networks;
    - Use high-level visual information for illuminant estimation: model objects by semantic info (i.e. green grass, blue sky) + add color knowledge
    - Use physics scenarios => but don't always match the real illuminant source mixture...