数字图像处理
Digital Image Processing

刘利刚
lgliu@ustc.edu.cn
http://staff.ustc.edu.cn/~lgliu

Graphics&Geometric Computing Lab
@USTC

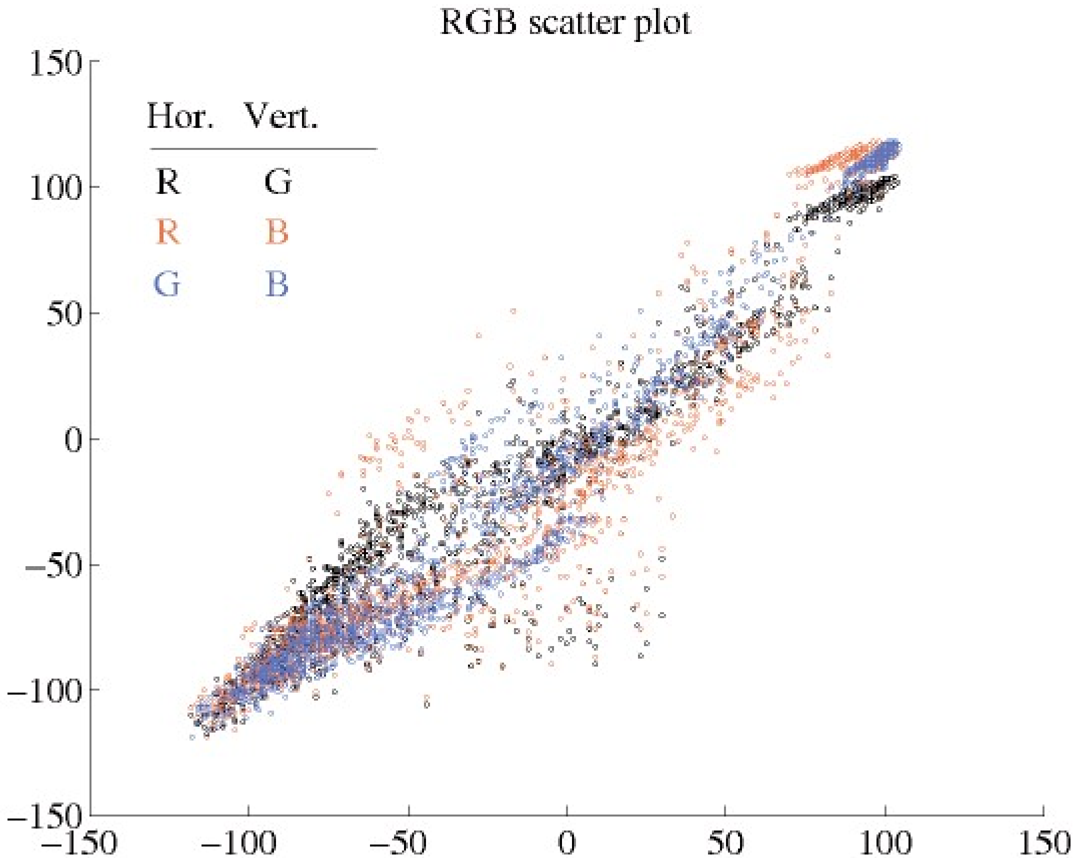
Image as Points

Point Processing of Images

Point Processing of Images

- In a digital image, point = pixel.
- Point processing transforms a pixel's value as function of its value alone;
- it does not depend on the values of the pixel's neighbors.

图像：3D线性空间的点集



Color Spaces: Different Basis

- RGB
- HSV/HSL
- CMY
- CIE XYZ
- $l\alpha\beta$

Example: RGB→ $l\alpha\beta$

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{aligned} \mathbf{L} &= \log L \\ \mathbf{M} &= \log M \\ \mathbf{S} &= \log S \end{aligned}$$

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{L} \\ \mathbf{M} \\ \mathbf{S} \end{bmatrix}$$

Example: RGB→HSV

The R,G,B values are divided by 255 to change the range from 0..255 to 0..1:

R' = R/255

G' = G/255

B' = B/255

Cmax = max(R', G', B')

Cmin = min(R', G', B')

Δ = Cmax - Cmin

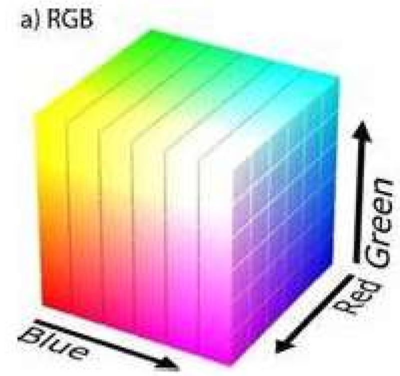
Hue calculation:

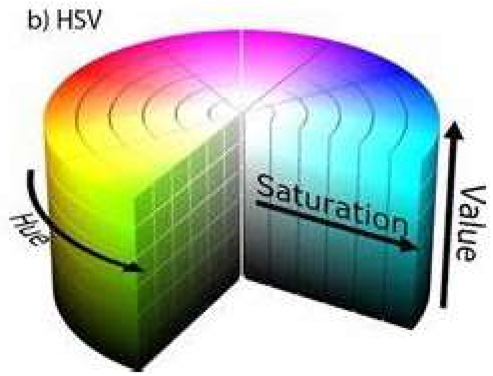
$$H = \begin{cases} 60^\circ \times \left(\frac{G'-B'}{\Delta} mod 6\right) & , Cmax = R' \\ 60^\circ \times \left(\frac{B'-R'}{\Delta} + 2\right) & , Cmax = G' \\ 60^\circ \times \left(\frac{R'-G'}{\Delta} + 4\right) & , Cmax = B' \end{cases}$$

Saturation calculation:

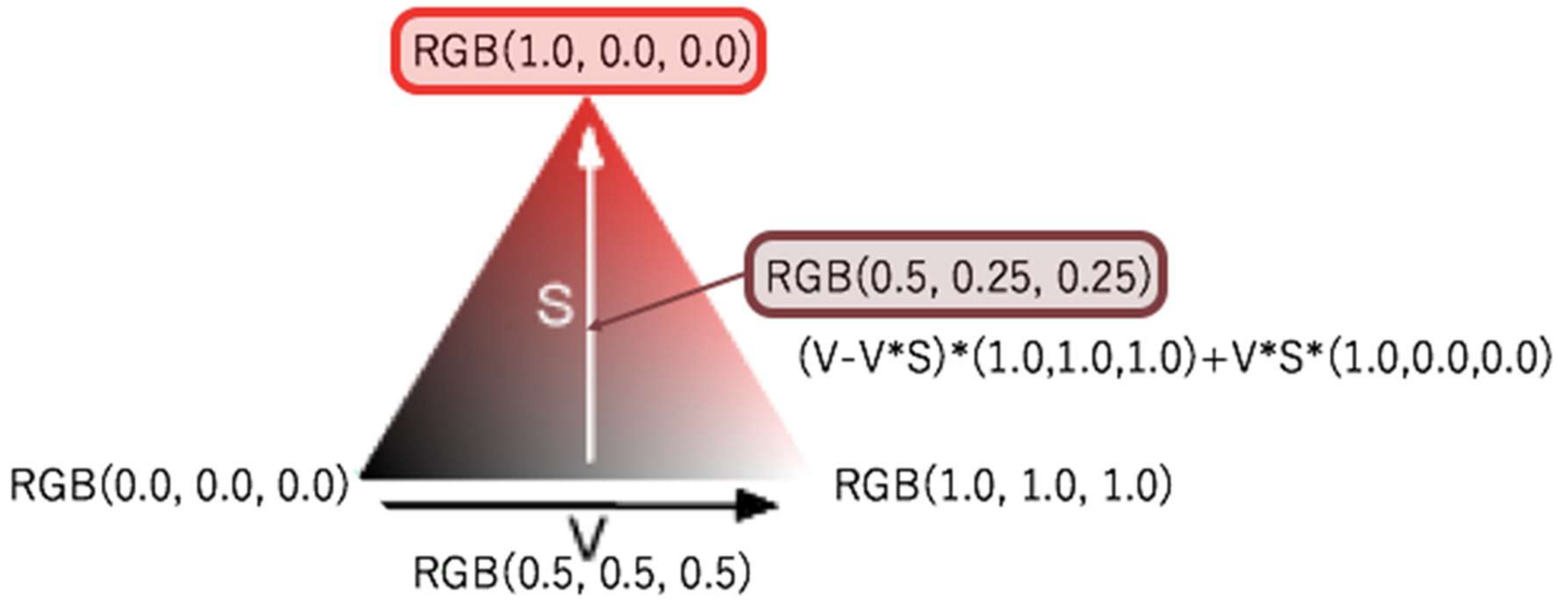
$$S = \begin{cases} 0 & , C_{max} = 0 \\ \frac{\Delta}{C_{max}} & , C_{max} \neq 0 \end{cases}$$

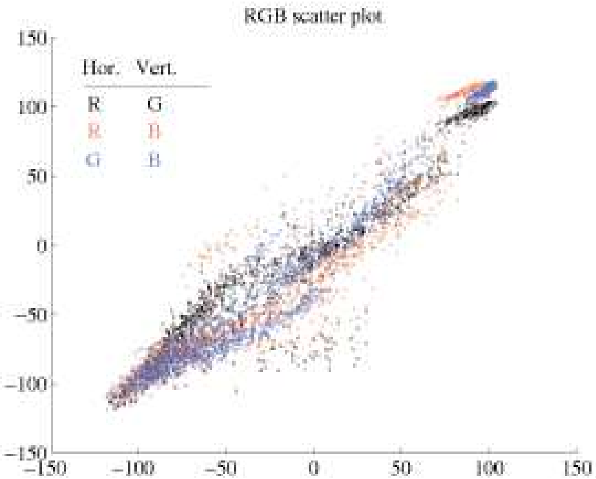
Value calculation: V = Cmax



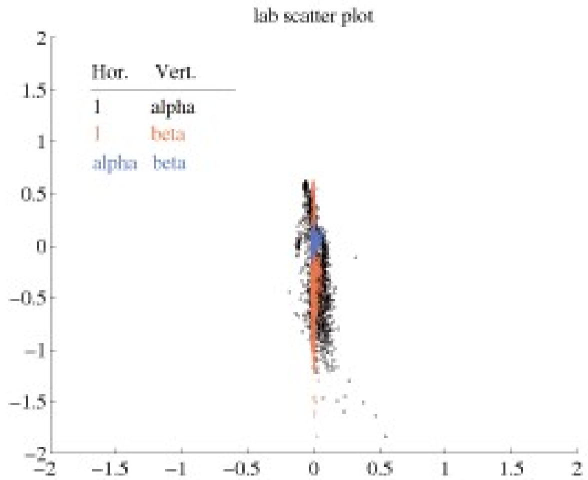


Example: HSV→RGB

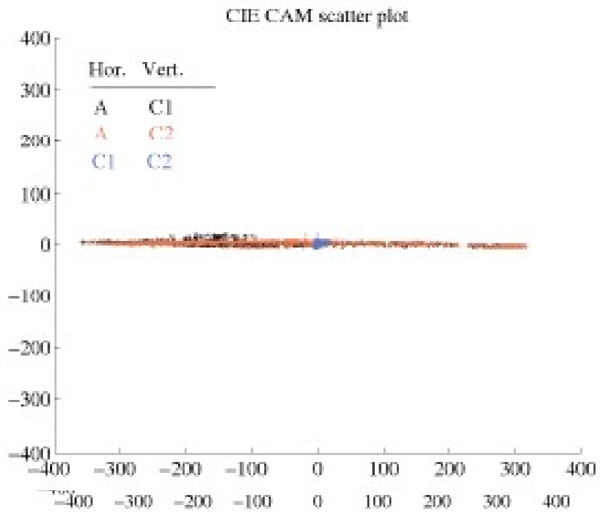




RGB



$l\alpha\beta$



CIE

Point Processing


- gamma


- brightness


original


+ brightness


+ gamma


histogram mod


- contrast


original

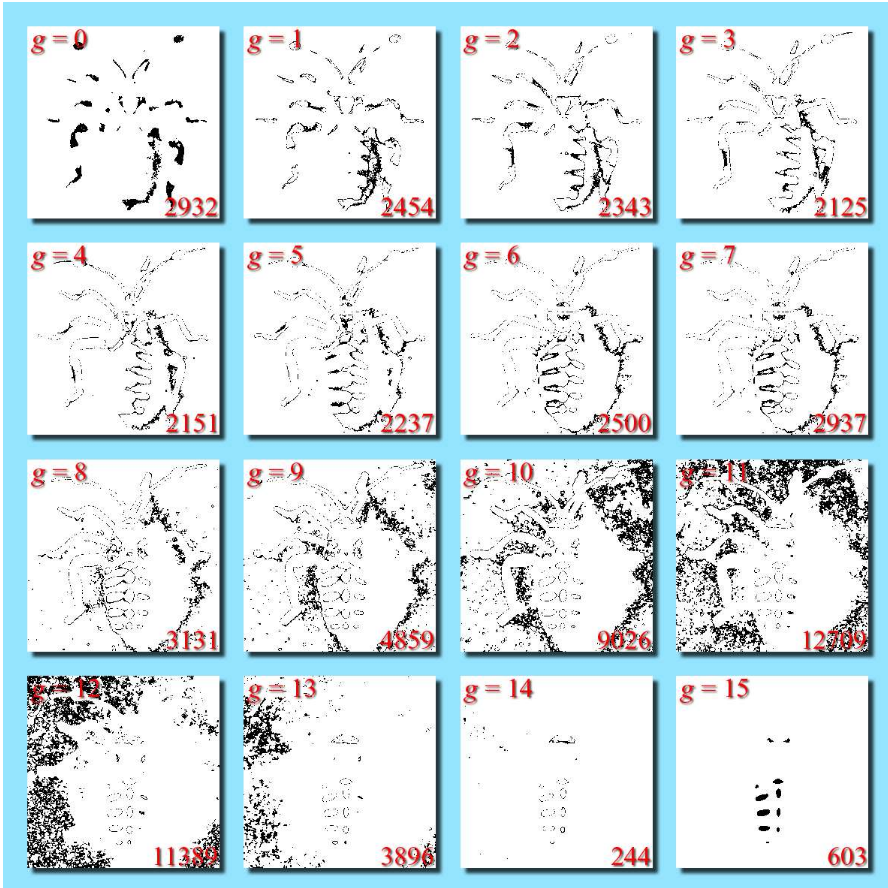

+ contrast


histogram EQ

The Histogram of a Grayscale Image

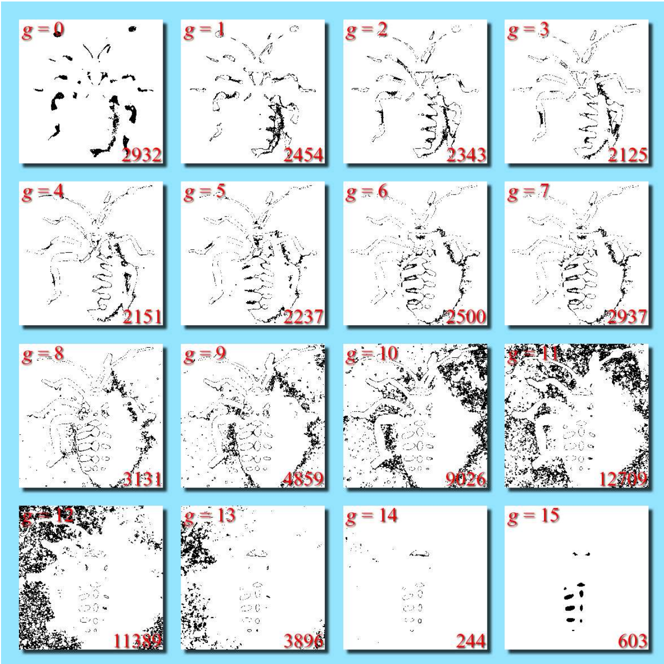
16-level (4-bit) image

lower RHC: number of pixels with intensity g



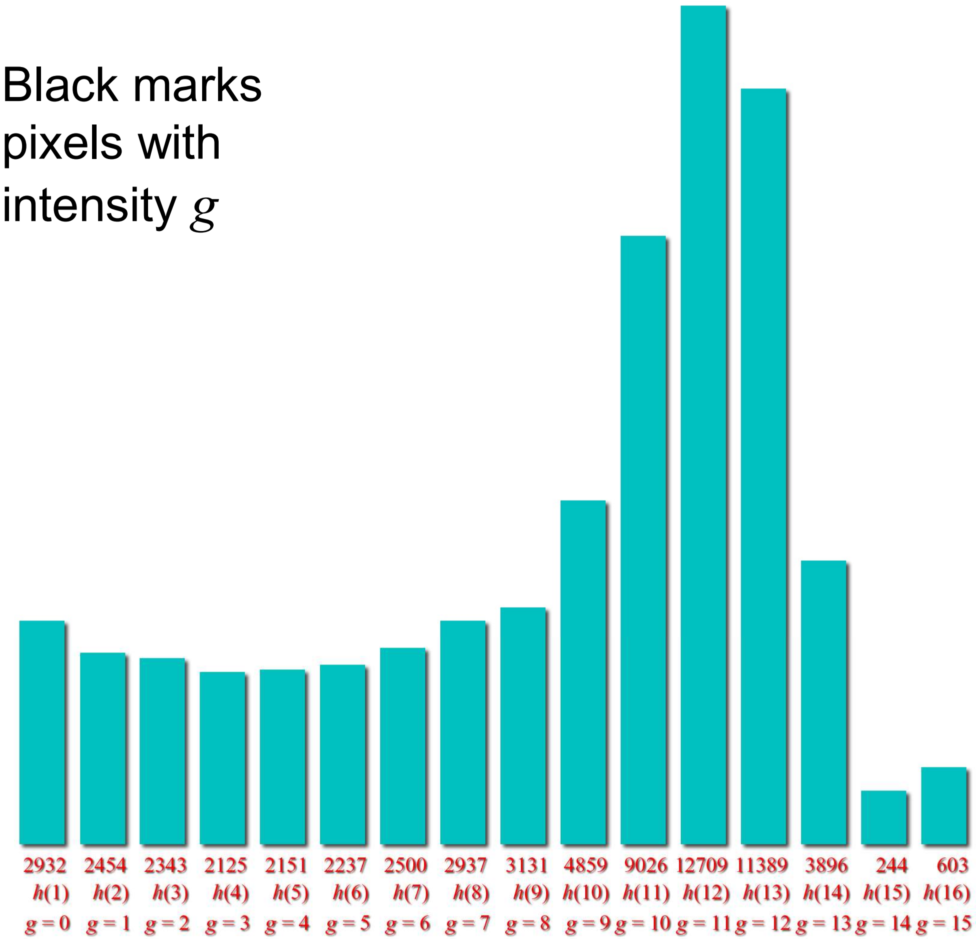
black marks pixels with intensity g

The Histogram of a Grayscale Image

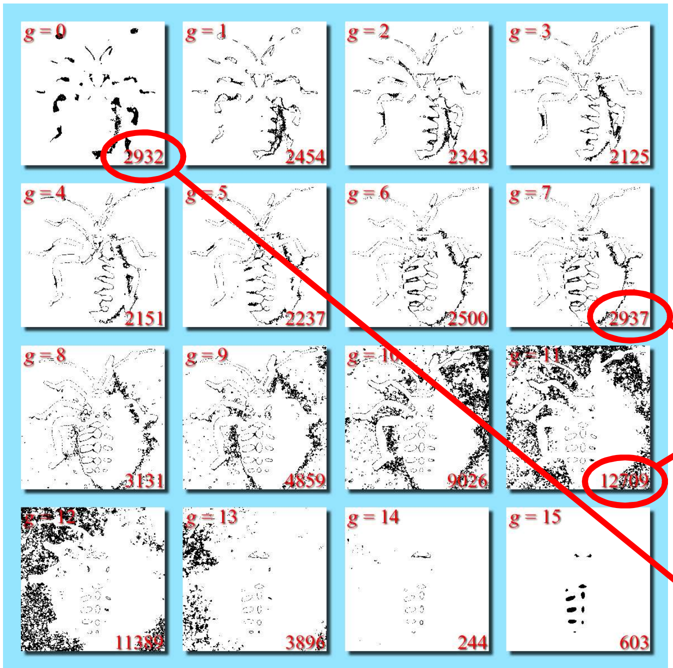


Black marks pixels with intensity g

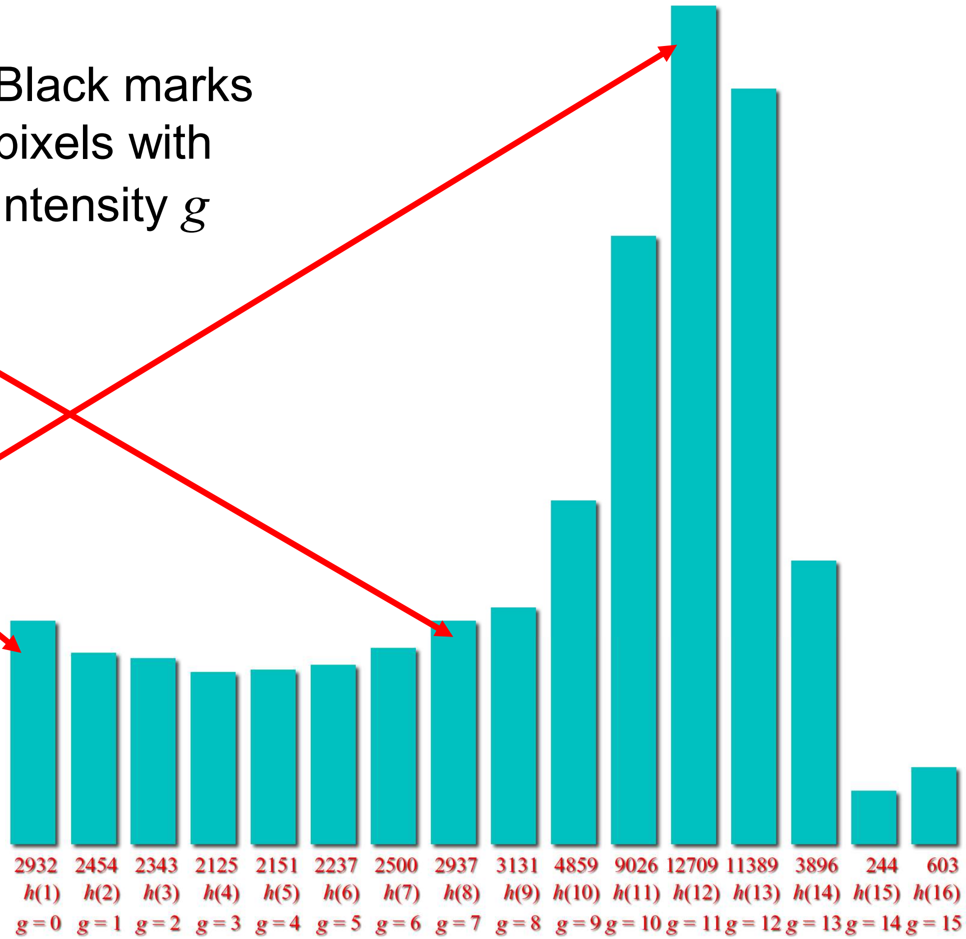
Plot of histogram:
number of pixels with intensity g



The Histogram of a Grayscale Image



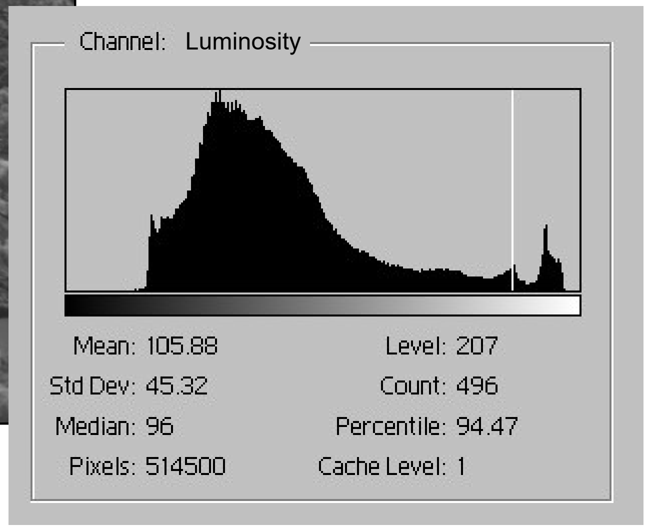
Black marks pixels with intensity g



Plot of histogram:
number of pixels with intensity g

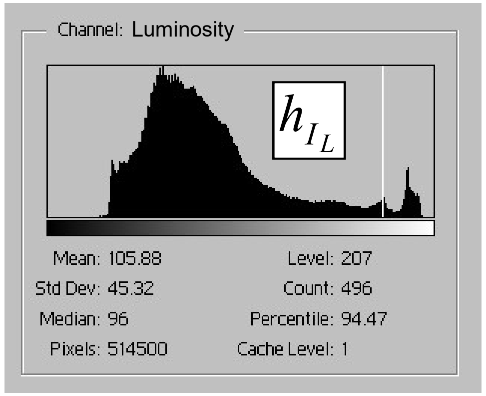
The Histogram of a Grayscale Image

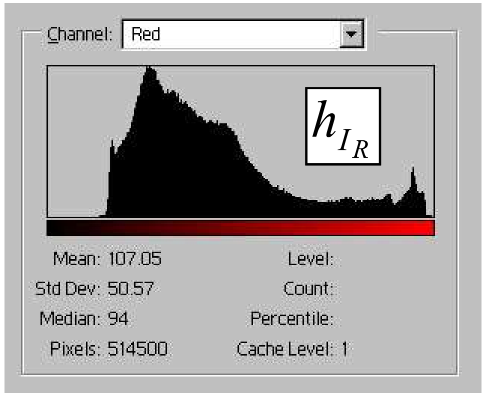
$h_{\mathbf{I}}(g+1) =$ the number of pixels in $\mathbf{I}$ with graylevel g.

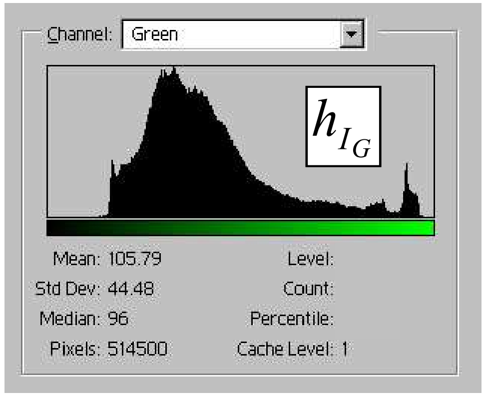


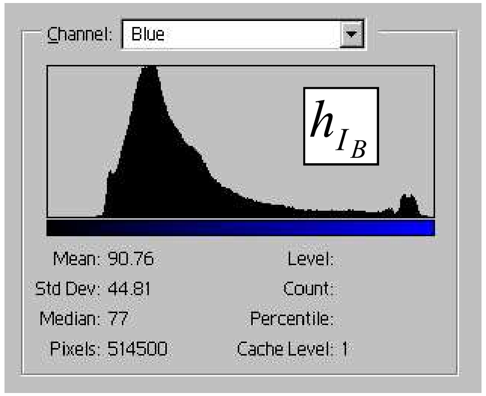
The Histogram of a Color Image

There is one histogram per color band R, G, & B. Luminosity histogram is from 1 band = (R+G+B)/3









Value or Luminance Histograms

The value histogram of a 3-band (truecolor) image, **I**, is the histogram of the value image,

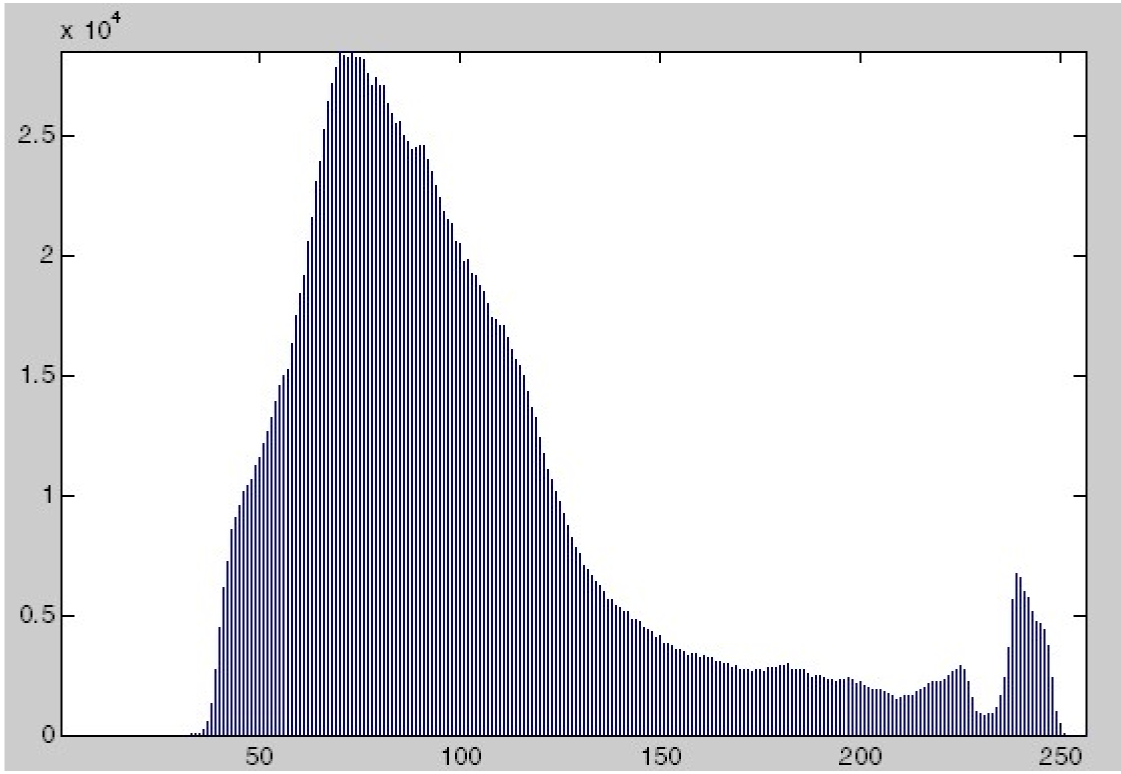
$$\mathbf{V}(r,c) = \frac{1}{3}[\mathbf{R}(r,c) + \mathbf{G}(r,c) + \mathbf{B}(r,c)]$$

Where **R**, **G**, and **B** are the red, green, and blue bands of **I**.
The luminance histogram of **I** is the histogram of the luminance image,

$$\mathbf{L}(r,c) = 0.299 \cdot \mathbf{R}(r,c) + 0.587 \cdot \mathbf{G}(r,c) + 0.114 \cdot \mathbf{B}(r,c)$$

Value Histogram

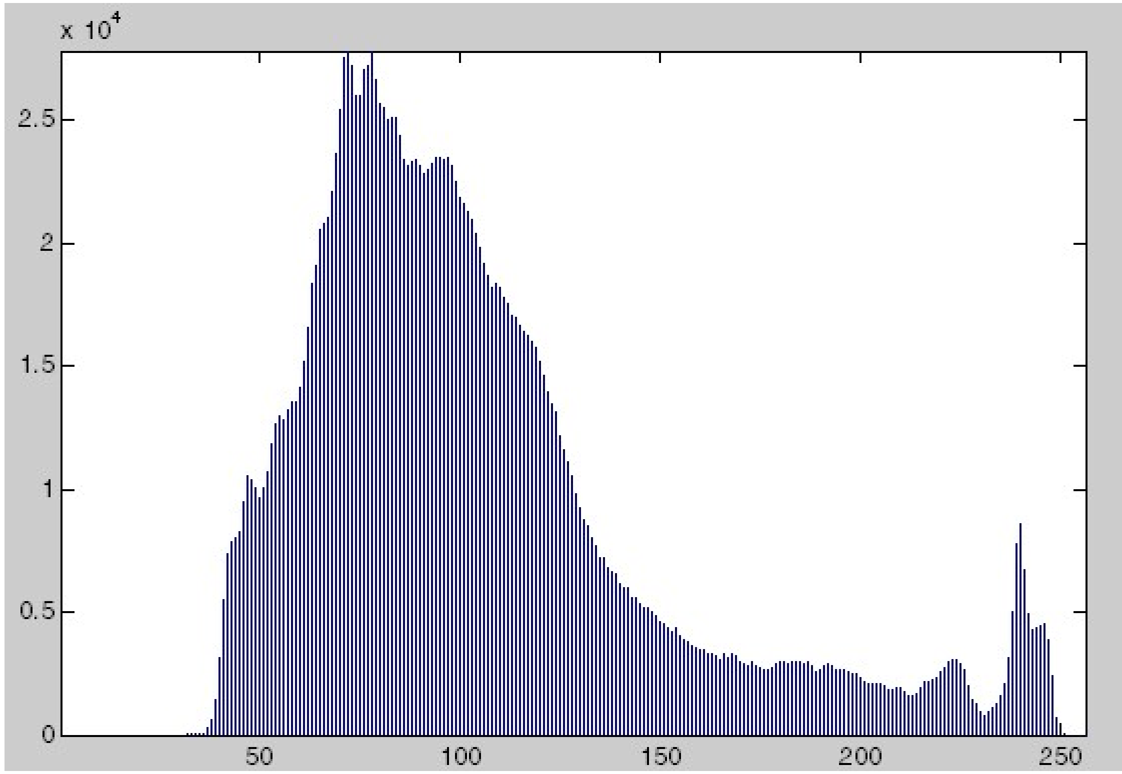
Value image, **V**.



Histogram of the value image.

Luminance Histogram

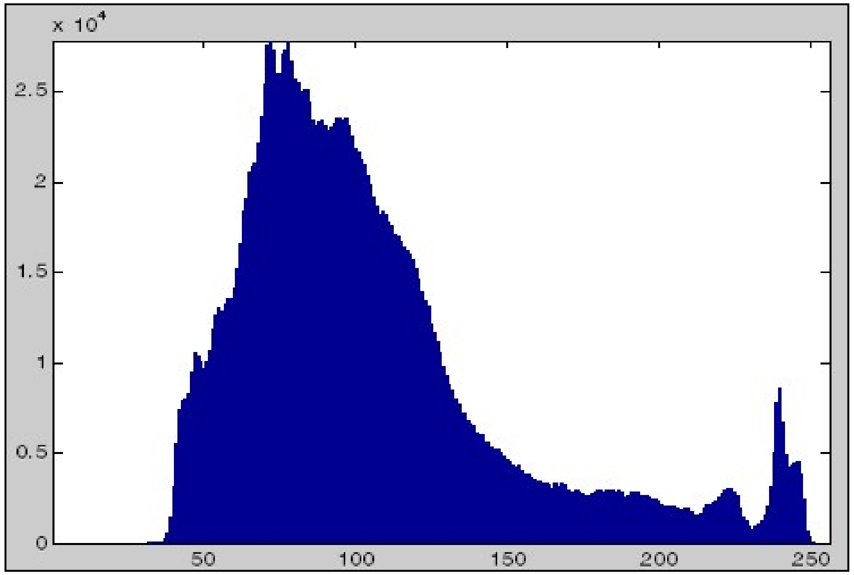
Luminance image, **L**.



Histogram of the luminance image.

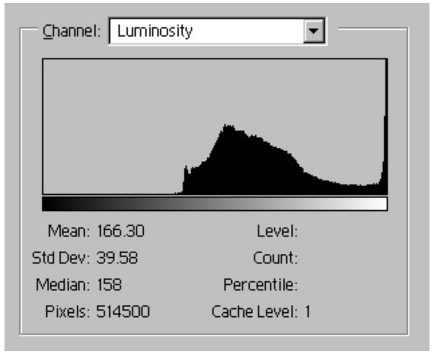
Point Processes: Original Image

Kinkaku-ji (Temple of the Golden Pavilion), also known as Rokuon-ji (Deer Garden Temple), is a Zen Buddhist temple in Kyoto, Japan.



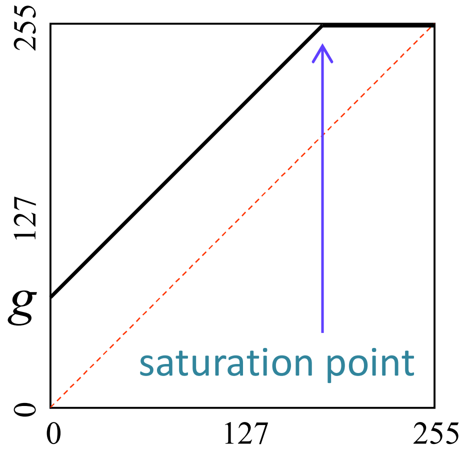
Luminance Histogram

Point Processes: Increase Brightness



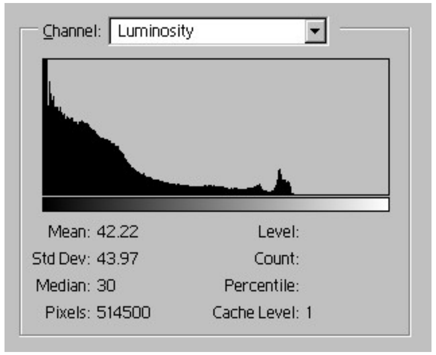
$$\mathbf{J}(r,c,b) = \begin{cases} \mathbf{I}(r,c,b) + g, & \text{if } \mathbf{I}(r,c,b) + g < 256 \\ 255, & \text{if } \mathbf{I}(r,c,b) + g > 255 \end{cases}$$

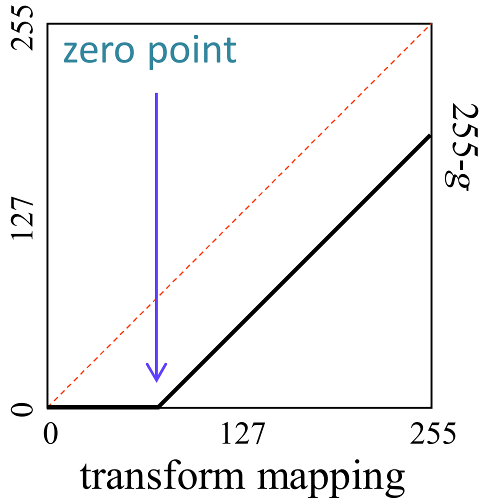
$g \geq 0$ and $b \in \{1,2,3\}$ is the band index.



transform mapping

Point Processes: Decrease Brightness

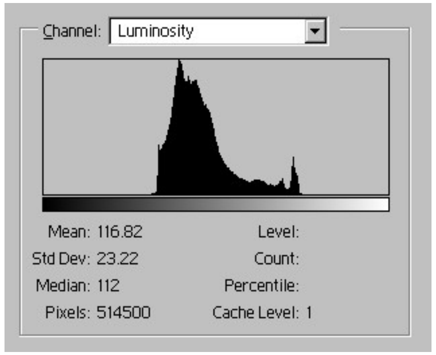




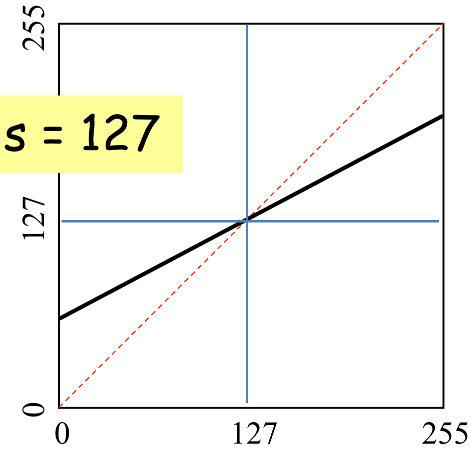
$$\mathbf{J}(r,c,b) = \begin{cases} 0, & \text{if } \mathbf{I}(r,c,b) - g < 0 \\ \mathbf{I}(r,c,b) - g, & \text{if } \mathbf{I}(r,c,b) - g > 0 \end{cases}$$

$g \geq 0$ and $b \in \{1,2,3\}$ is the band index.

Point Processes: Decrease Contrast



Here, s = 127



transform mapping

$$\mathbf{T}(r,c,b) = a\left[\mathbf{I}(r,c,b) - s\right] + s,$$

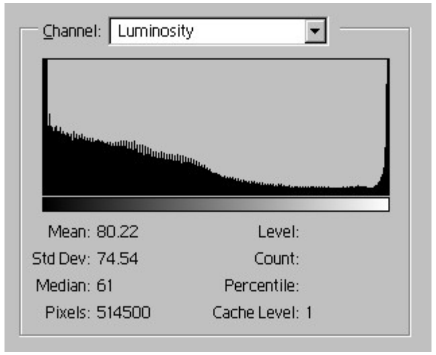
where $0 \le a < 1.0$,

$s \in \{0,1,2,\ldots,255\}$, and

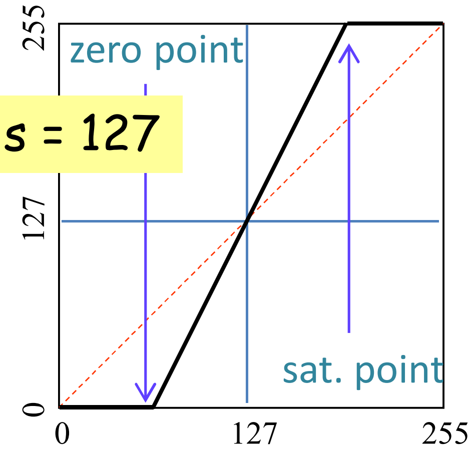
$b \in \{1,2,3\}$.

s is the center of the contrast function.

Point Processes: Increase Contrast



Here, s = 127



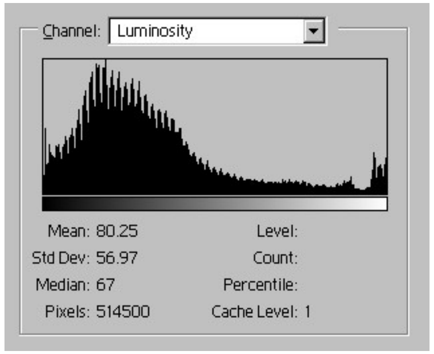
transform mapping

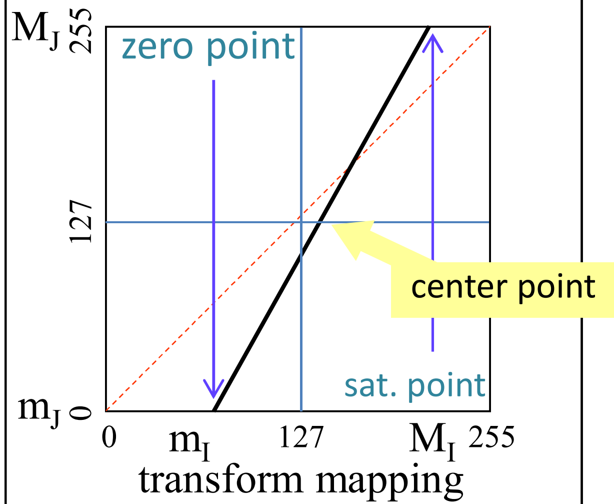
$$\mathbf{T}(r,c,b) = a[\mathbf{I}(r,c,b) - s] + s$$

$$\mathbf{J}(r,c,b) = \begin{cases} 0, & \text{if } \mathbf{T}(r,c,b) < 0, \\ \mathbf{T}(r,c,b), & \text{if } 0 \le \mathbf{T}(r,c,b) \le 255, \\ 255, & \text{if } \mathbf{T}(r,c,b) > 255. \end{cases}$$

$$a > 1, \quad s \in \{0, \ldots, 255\}, \quad b \in \{1,2,3\}$$

Point Processes: Contrast Stretch





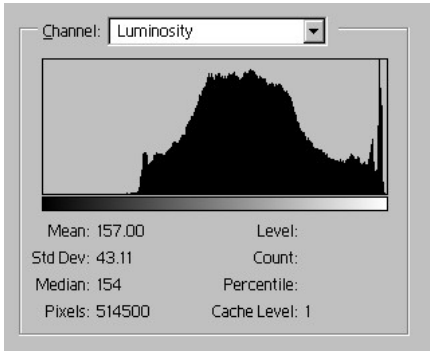
Let $m_{\mathbf{I}} = \min[\mathbf{I}(r,c)], \quad M_{\mathbf{I}} = \max[\mathbf{I}(r,c)]$

$m_{\mathbf{J}} = \min[\mathbf{J}(r,c)], \quad M_{\mathbf{J}} = \max[\mathbf{J}(r,c)]$

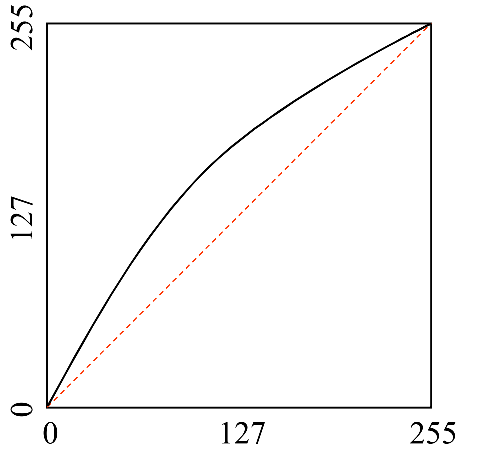
Then,

$$\mathbf{J}(r,c) = (M_{\mathbf{J}} - m_{\mathbf{J}})\frac{\mathbf{I}(r,c) - m_{\mathbf{I}}}{M_{\mathbf{I}} - m_{\mathbf{I}}} + m_{\mathbf{J}}.$$

Point Processes: Increased Gamma

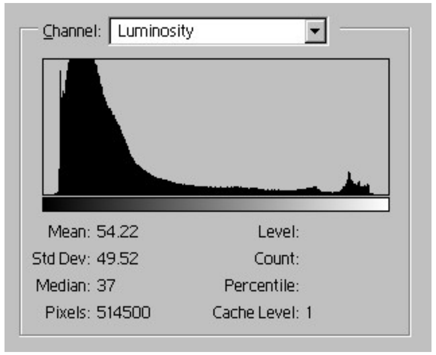


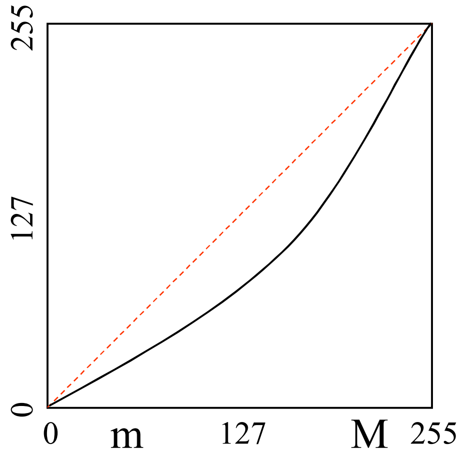
$$\mathbf{J}(r,c) = 255 \cdot \left[\frac{\mathbf{I}(r,c)}{255} \right]^{\frac{1}{\gamma}} \quad \text{for} \quad \gamma > 1.0$$



transform mapping

Point Processes: Decreased Gamma



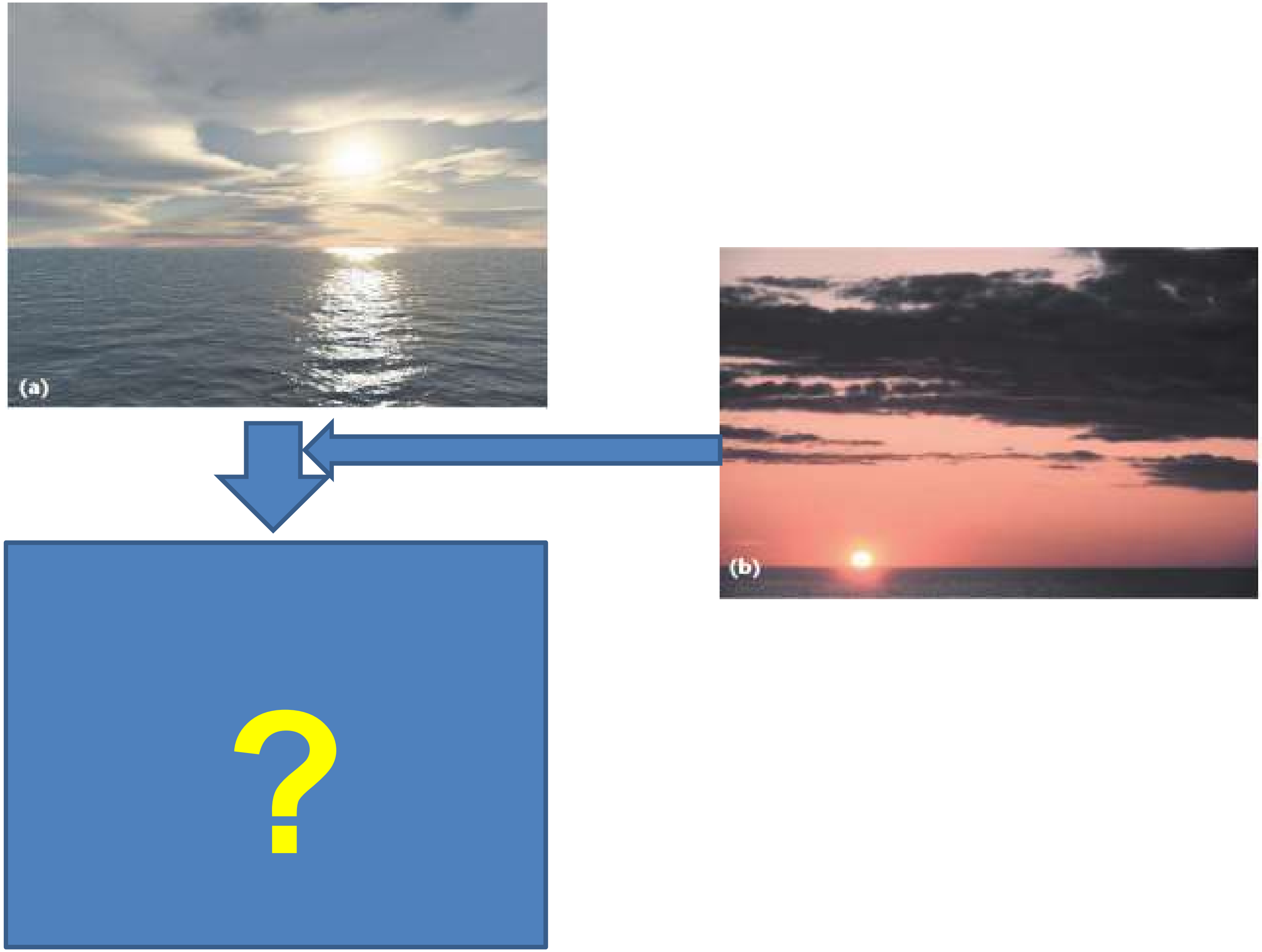


transform mapping

$$\mathbf{J}(r,c) = 255 \cdot \left[\frac{\mathbf{I}(r,c)}{255} \right]^{\frac{1}{\gamma}} \quad \text{for} \quad \gamma < 1.0$$

Applications

Example 1: Color Transfer



Erik Reinhard et al. Color Transfer between Images. IEEE CG&A, 2001.

Algorithm

- 使点云中心、形状相似（均值和方差变换）

$$l^* = l - \langle l \rangle$$
$$\alpha^* = \alpha - \langle \alpha \rangle$$
$$\beta^* = \beta - \langle \beta \rangle$$

$$l' = \frac{\sigma_t^l}{\sigma_s^l} l^*$$
$$\alpha' = \frac{\sigma_t^\alpha}{\sigma_s^\alpha} \alpha^*$$
$$\beta' = \frac{\sigma_t^\beta}{\sigma_s^\beta} \beta^*$$

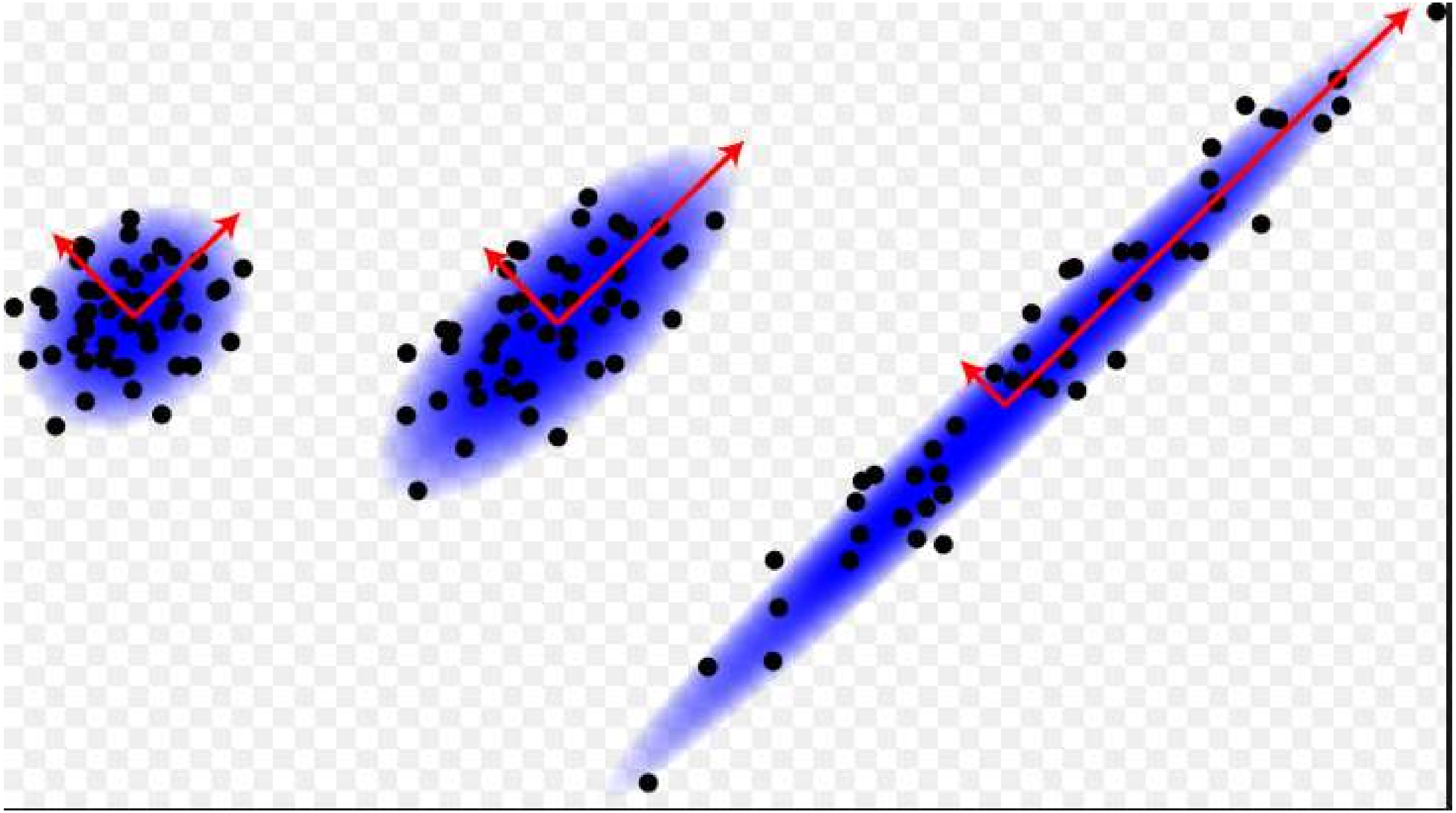
Example 2: Color2Grey

Color

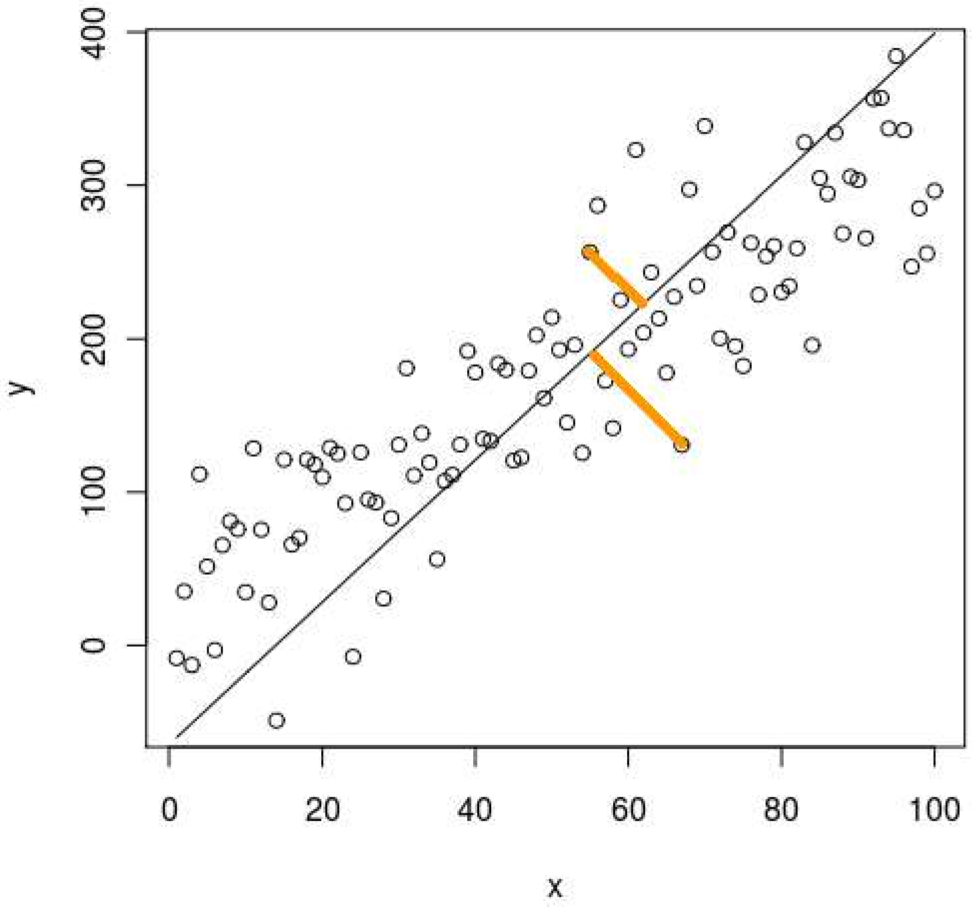
Grayscale

Amy A. Gooch et al. Color2Gray: Salience-Preserving Color Removal. Siggraph 2005.
https://users.cs.northwestern.edu/~ago820/color2gray/

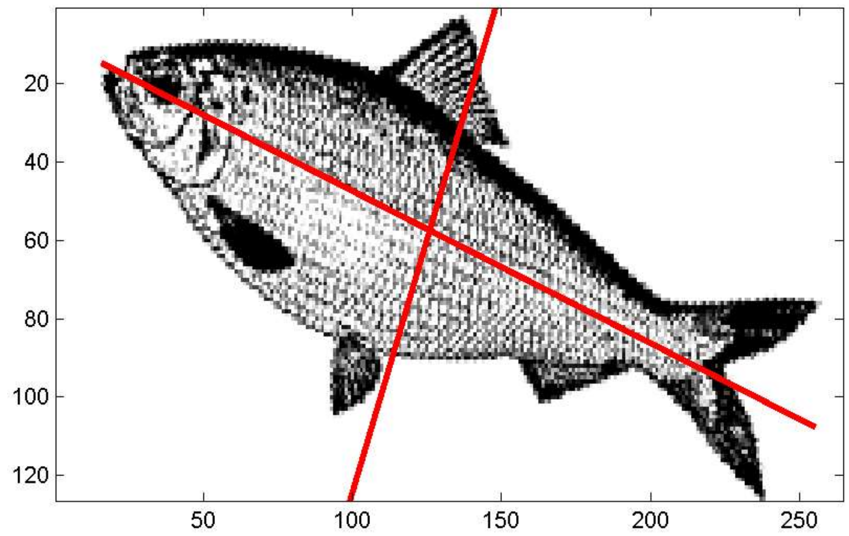
Problem: Dimension reduction

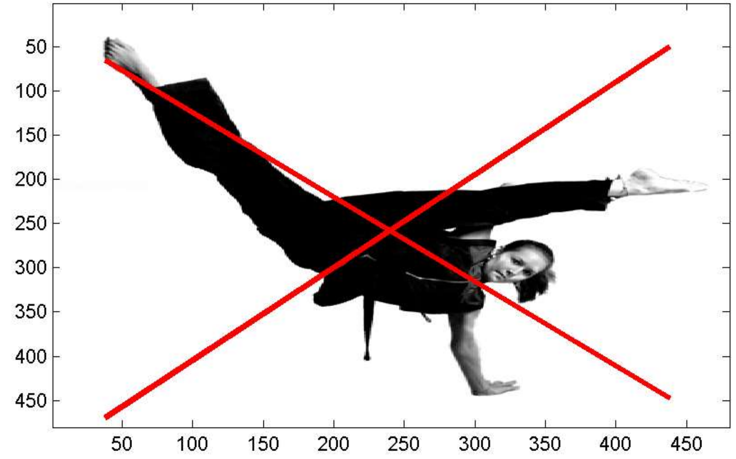


3D → 1D



Principal Component Analysis (PCA)





Principal Component Analysis

- PCA is the most commonly used dimension reduction technique.
- (Also called the Karhunen-Loeve transform).
- PCA – data samples $x_1, , x_N$.
- Compute the mean $\mu = (1/N)\sum_{i=1}^{N} x_i$.
- Computer the covariance:

$$K = (1/N)\sum_{i=1}^{N}(x_i - \mu)(x_i - \mu)^T.$$

Principal Component Analysis

- Compute the eigenvalues λ
  and eigenvectors e of the matrix K.
- Solve $Ke = \lambda e$.
- Order them by magnitude:

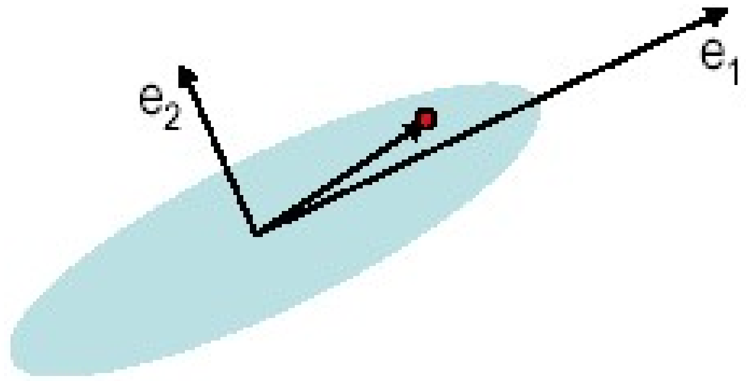
$$\lambda_1 \geq \lambda_2 \geq .\lambda_N.$$

- PCA reduces the dimension by keeping direction e such that $\lambda < T$.

Principal Component Analysis

- For many datasets, most of the eigenvalues \lambda are negligible and can be discarded.

The eigenvalue λ measures the variation
In the direction e



Example:
$$\lambda_1 \neq 0, \lambda_2 = 0.$$

Principal Component Analysis

- Project the data onto the selected eigenvectors: $$x \mapsto \mu + \sum_{i=1}^{M} a_i e_i$$
- Where $$a_i = (x - \mu) \cdot e_i.$$

$$\frac{\sum_{i=1}^{M} \lambda_i}{\sum_{i=1}^{N} \lambda_i}$$

- is the proportion of data covered by the first M eigenvalues.

Color2Gray

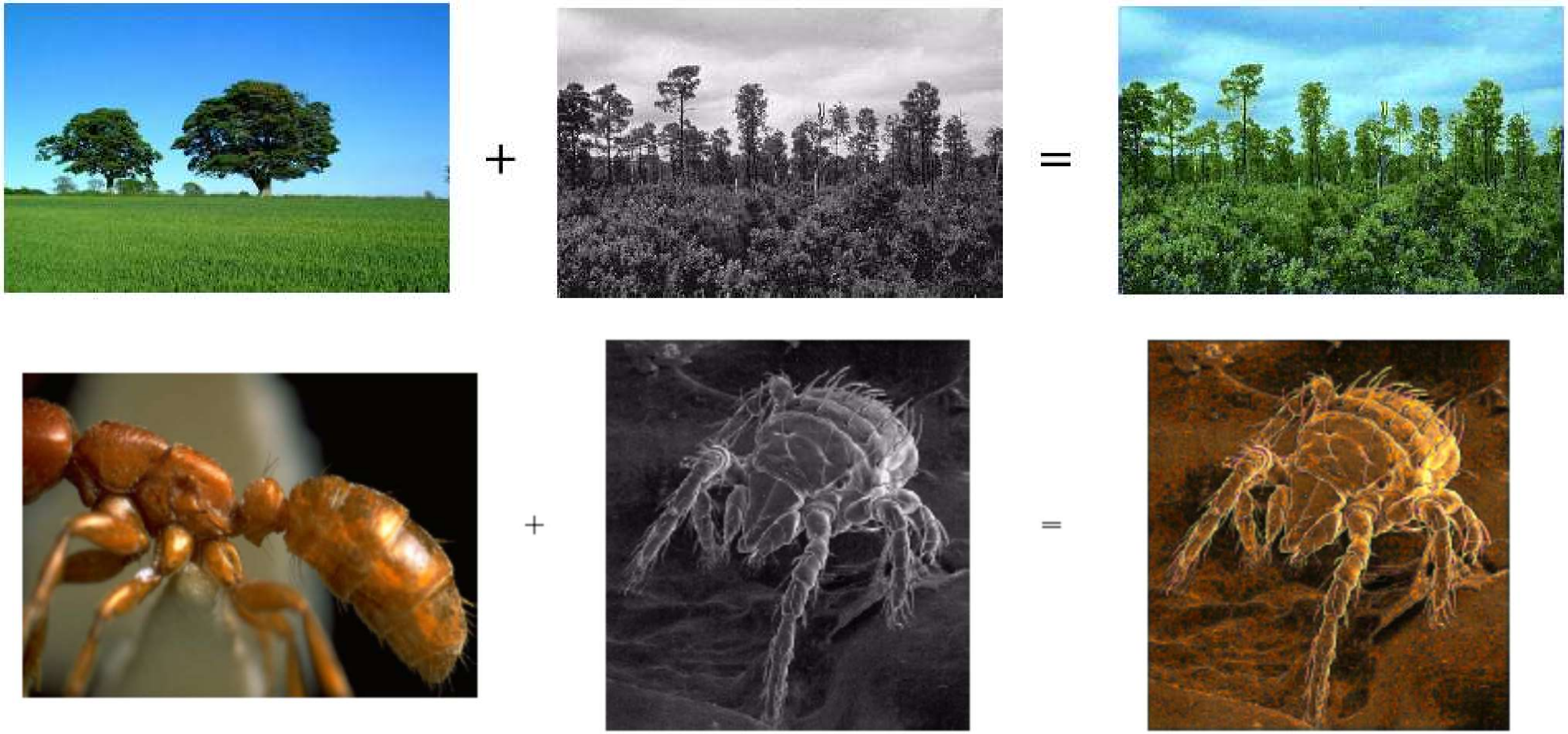
Color

Grayscale

New Algorithm

Example 3: Colorization

- Assign color to gray images



Anat Levin et al. Colorization using optimization. Siggraph 2004.

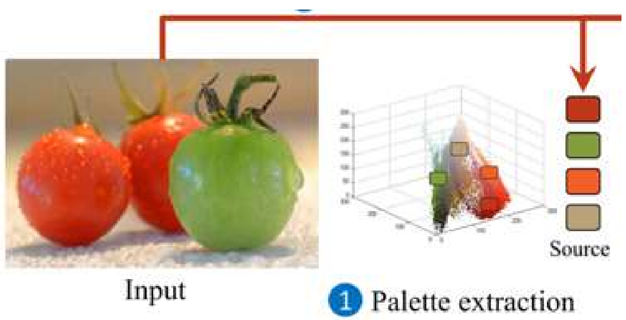
Example 4: Color Harmonization

original image

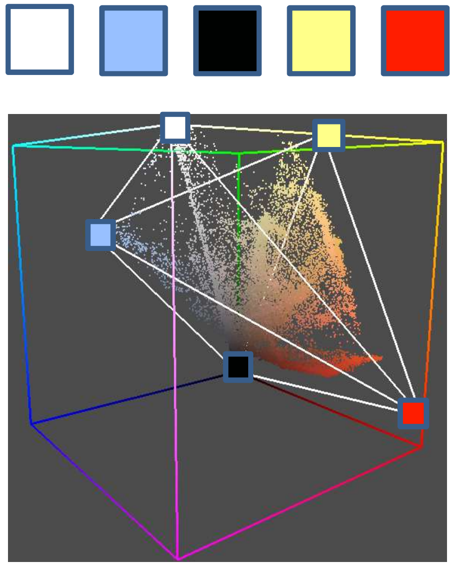
harmonized image

Daniel Cohen-Or et al. Color Harmonization. Siggraph 2006.

Example 5: 基于调色板的重着色



输入图像



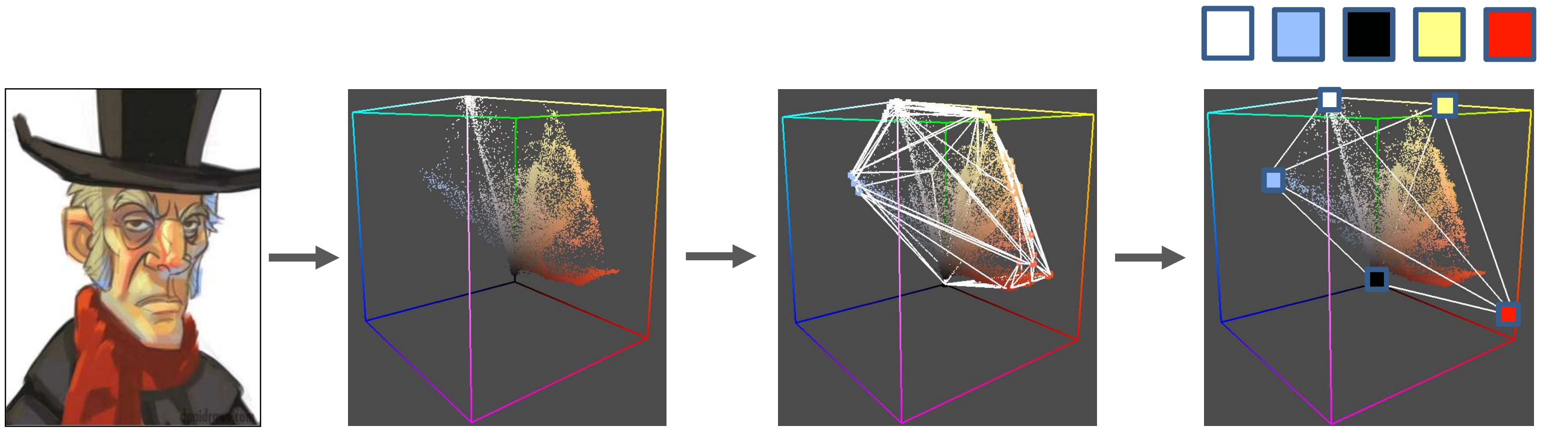
简化的 RGB 凸包
(调色板)

Jianchao Tan et al. Palette-based image decomposition, harmonization, and color transfer. arXiv preprint arXiv:1804.01225 (2018).
Jianchao Tan et al. Efficient palette-based decomposition and recoloring of images via RGBXY-space geometry. Siggraph Asia 2018.
Yili Wang et al. An Improved Geometric Approach for Palettebased Image Decomposition and Recoloring. Computer Graphics Forum, 2019.

基于几何凸包的调色板提取

[Tan et al. 2017; Tan et al. 2018; Wang et al. 2019]

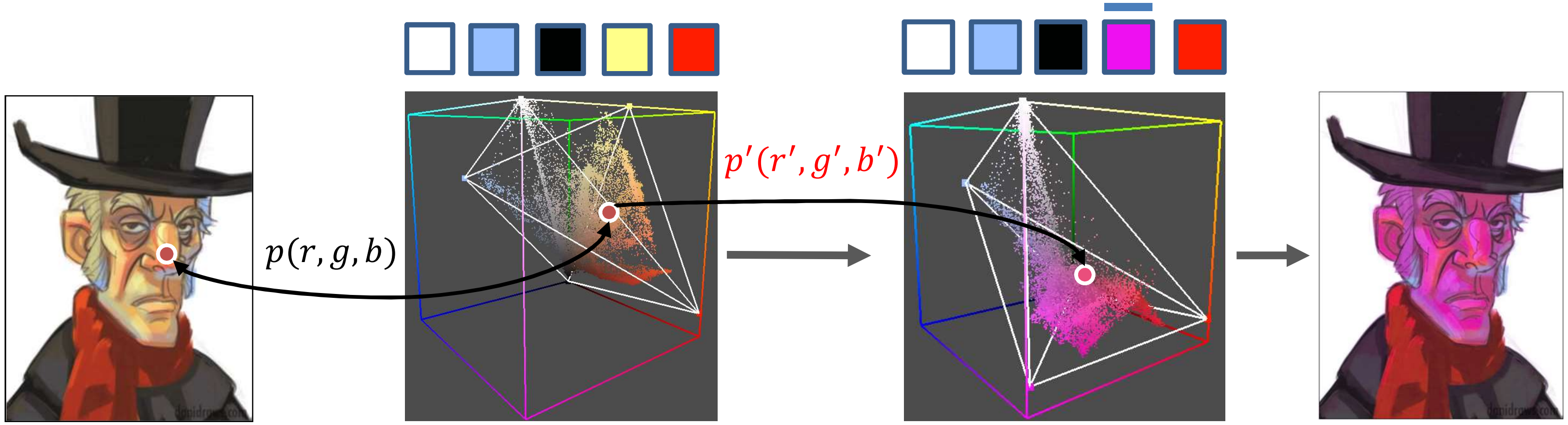
- 调色板提取方法
  - 求所有点的RGB 3D凸包
  - 凸包简化：减少顶点数



输入图像　　RGB颜色空间　　RGB颜色空间的凸包　　简化的 RGB 凸包 (调色板)

基于几何凸包的图像分解和重着色

- 图像分解和重着色
  - 类似于几何/动画中的插值系数和几何变形



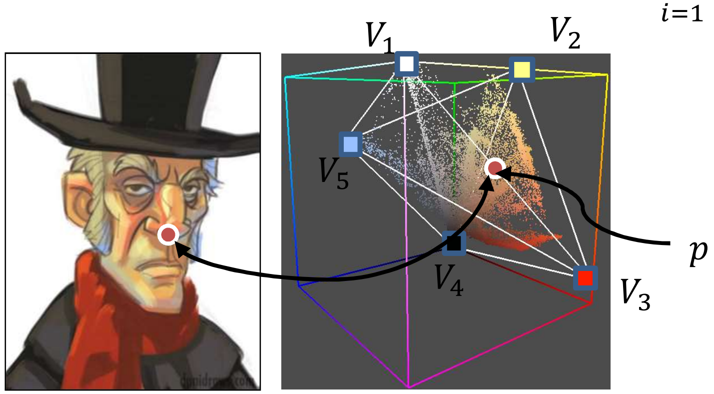
输入图像　　调色板(凸包) V　　调色板(凸包) V'　　重着色效果

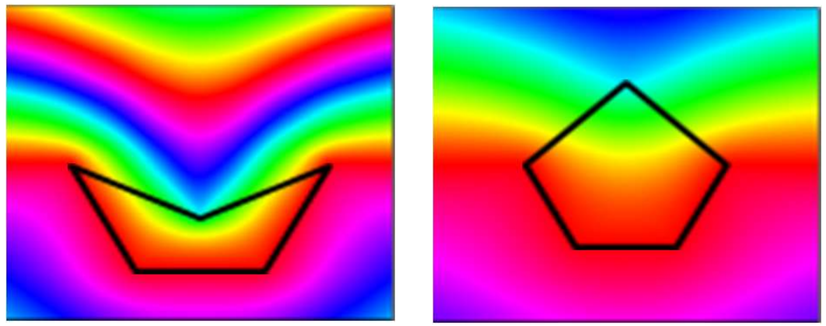
- Mixing weights: $p = \sum_{i=1}^{|V|} w_i V_i \quad 1 = \sum_{i=1}^{|V|} w_i$　　Recoloring: $p' = \sum_{i=1}^{|V|} w_i V_i'$

基于几何凸包的图像分解和重着色

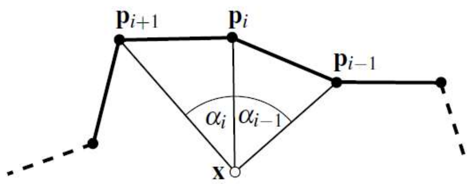
- Mixing weights 应满足

$$p = \sum_{i=1}^{|V|} w_i V_i \qquad 1 = \sum_{i=1}^{|V|} w_i$$

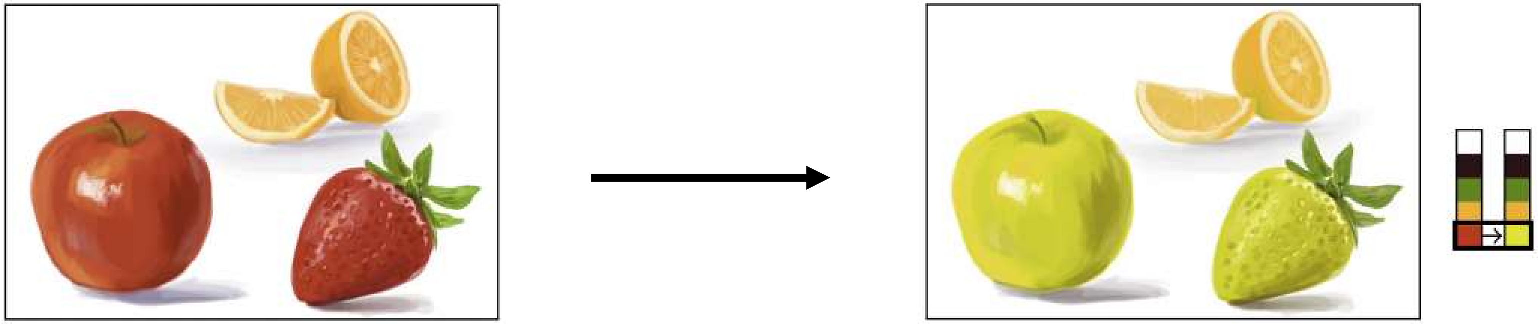


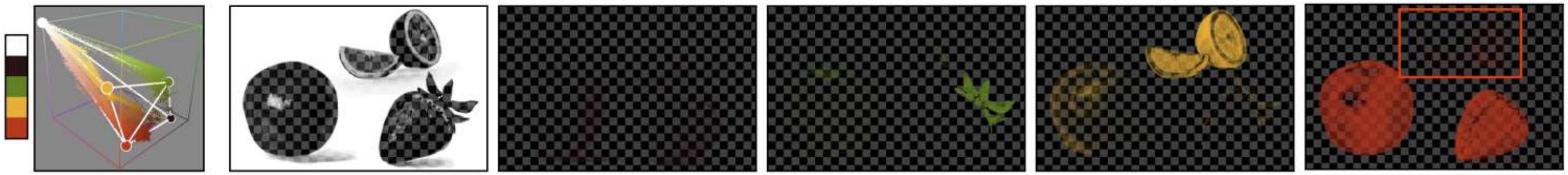


- 使用MVC (Mean Value Coordinates)
  - 二维情况： $w_i = \frac{u_i}{\sum_{i=1}^{m} u_i}, u_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right)+\tan\left(\frac{\alpha_i}{2}\right)}{\|p_i - x\|}$
  - 非常光滑：内部处处可导，外部处处连续



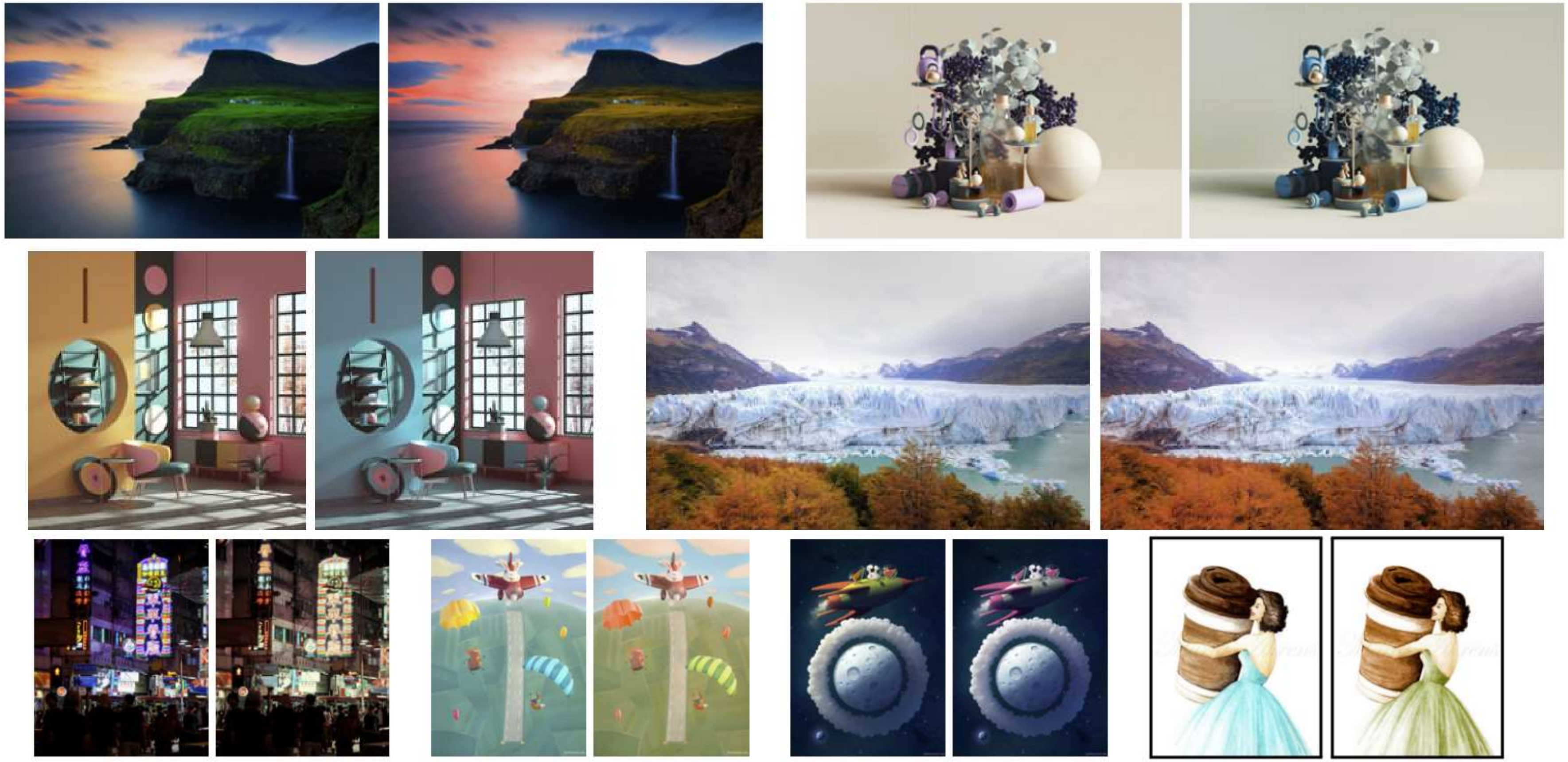
实验结果





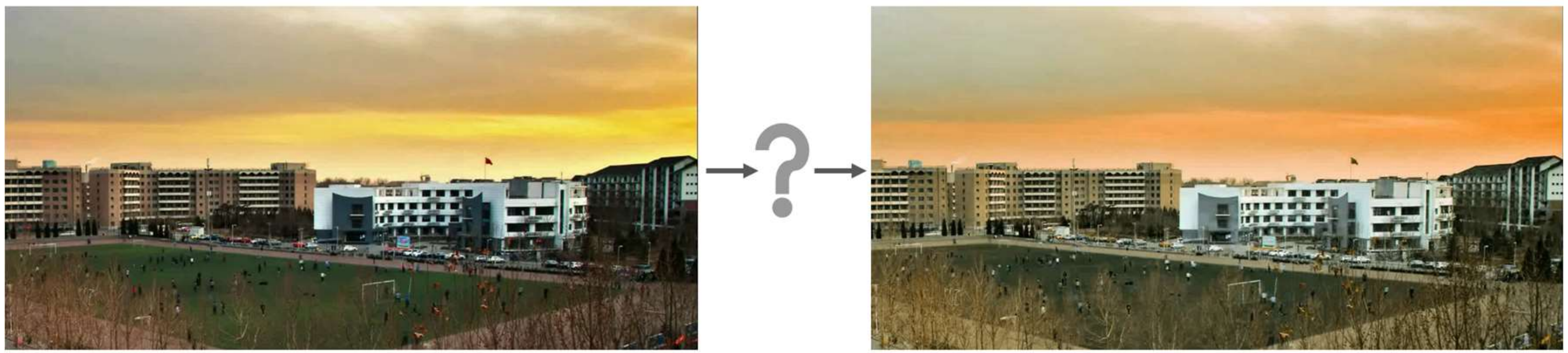
实验结果

更多实验结果



基于调色板的视频编辑重着色

- 挑战
  - 视频颜色是随时间变化的，用户编辑也可以是随时间变化的
- 视频调色板的预期性质
  - 具有较少数量的控制顶点、较低的重建误差、较好的紧凑性
  - 具有较好的帧间连续性，能处理复杂的颜色时间变化

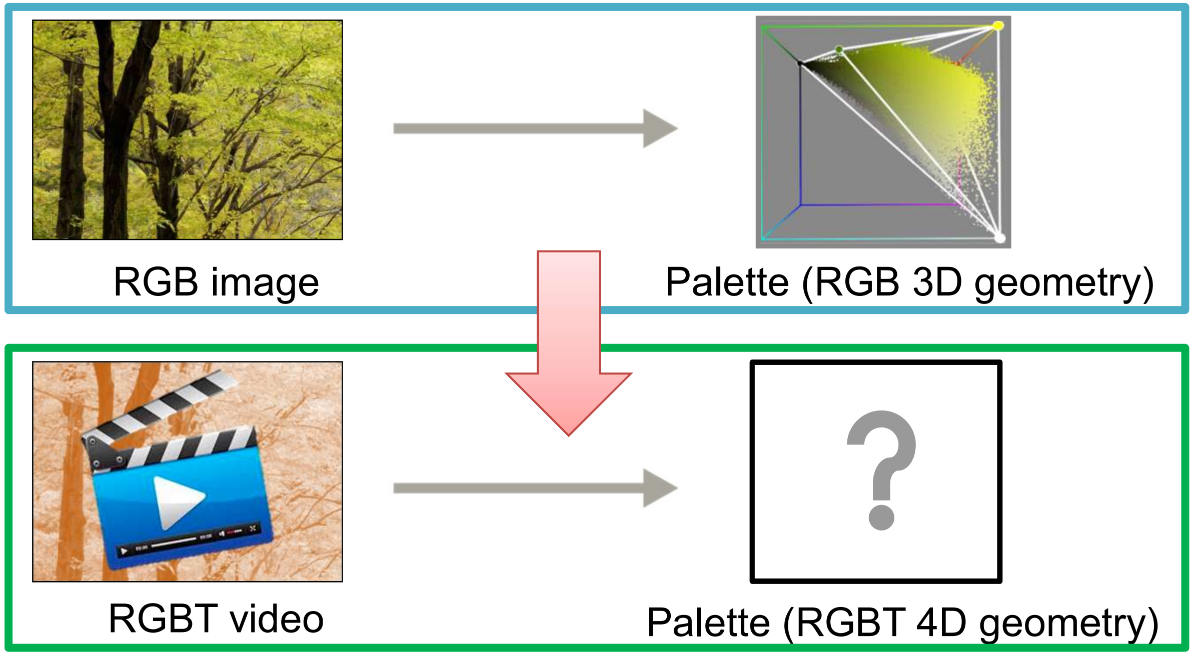


Input video　　Palette　　Recolored video

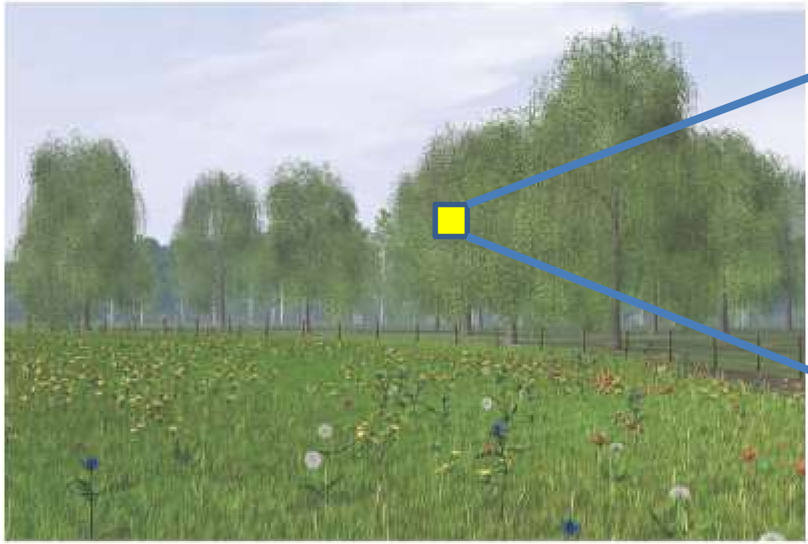
Du et al. Video Recoloring via Spatial-Temporal Geometric Palettes. Siggraph 2021.

基于调色板的视频编辑重着色

- 主要思想
  - 视频是随时间变化的RGB 3D点云 (或RGBT 4D点云)
  - 视频调色板表示用 RGBT 4D几何
  - 任一帧的3D RGB调色板通过切割4D几何得到



Feature Space?



$$(r, g, b)$$

$$(r, g, b, u, v)$$

Thanks!