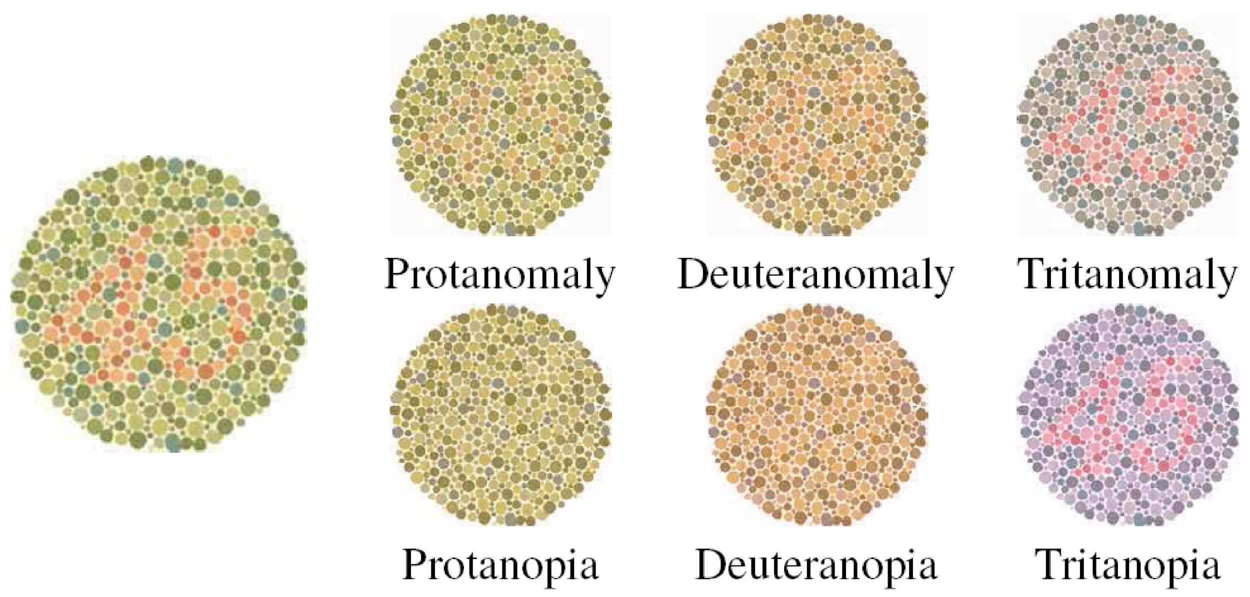
Image Recolorization for the Colorblind

Jia-Bin Huang[1], Chu-Song Chen[1], Tzu-Cheng Jen[2], Sheng-Jyh Wang[2]

[1] Institute of Information Science, Academia Sinica, Taipei, Taiwan
[2] Institute of Electronics Engineering, National Chiao Tung University, Hsinchu, Taiwan

Problem

- People with color vision deficiency (CVD) have difficulty in distinguishing between some colors.
- How the colorblind perceive colors?



Background

- CVD results from partial or complete loss of function of one or more types of cone cells.
- Simulation of color perception of people with CVD [1]
- Related works addressing CVD accessibility:
  - Guidelines for designers to avoid ambiguous color combinations
  - (Semi-)automatic methods for recoloring images

Contributions

- Generalize the concept of key colors in a image
- Propose to measure the contrast between two key colors by the symmetric KL divergence
- Interpolate colors to ensure local smoothness with a few key colors

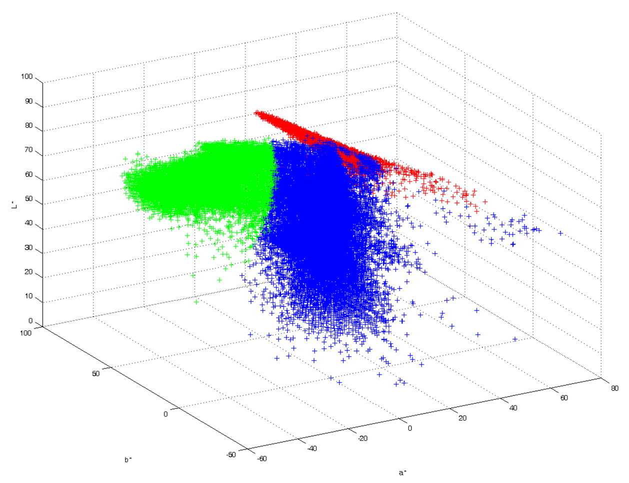
Reference

1 H. Brettel, F. Vienot, and J.D. Mollon, "Computerized simulation of color appearance for dichromats", J. Optic. Soc. Amer. A, 1997.

2 L. Jefferson and R. Harvey, "Accommodating color blind computer users," in ACM SIGACCESS, 2006.

The Proposed Algorithm

- **Image Representation via Gaussian Mixture Modeling**
  - Use the CIEL*a*b* color space
  - Approximate the distribution by K Gaussians: $p(x|\Theta) = \sum_{i=1}^{K} \omega_i G_i(x|\theta_i)$
  - Learn the parameters by the Expectation-Maximization (EM) algorithm
  - Select the optimal number of K by the Minimum Description Length (MDL) principle



- **Target Distance**
  - Generalize the concept of key colors from "point" to "cluster".
  - Use symmetric Kullback-Leibler (KL) divergence as our dissimilarity measure The symmetric KL divergence: $D_{sKL}(G_i, G_j) = D_{KL}(G_i||G_j) + D_{KL}(G_j||G_i)$
  - For Gaussians, analytical solutions exists

$$D_{sKL}(G_i, G_j) = (\mu_i - \mu_j)^T(\Sigma_i^{-1} + \Sigma_j^{-1})(\mu_i - \mu_j) + tr(\Sigma_i\Sigma_j^{-1} + \Sigma_i^{-1}\Sigma_j - 2I)$$

- **Optimization**
  - Define the color mapping functions $M_i(\cdot), i = 1, \ldots, K$
  - The error introduced by the i_{th} and j_{th} key colors:

$$E_{i,j} = [D_{sKL}(G_i, G_j) - D_{sKL}(Sim(M_i(G_i)), Sim(M_j(G_j)))]^2$$

  - Introduce weights for each color values: $\alpha_j = ||x_j - Sim(x_j)||$
  - Obtain weight for each cluster:

$$\lambda_i = \frac{\sum_{j=1}^{N} \alpha_j p(i|x_j, \Theta)}{\sum_{i=1}^{K}\sum_{j=1}^{N} \alpha_j p(i|x_j, \Theta)}$$

  - Rewrite the objective function

$$E = \sum_{i=1}^{i=K} \sum_{j=i+1}^{j=K} (\lambda_i + \lambda_j) E_{i,j}.$$

  - Minimize the objective function via direct search optimization method.
- **Gaussian Mapping for Interpolation**
  - Compute the transformed colors using

$$T(x_j)^H = x_j^H + \sum_{i=1}^{K} p(i|x_j, \Theta)(M_i(\mu_i)^H - \mu_i^H)$$

Future Directions

- Relax the assumption of Gaussian distribution (e.g., use non-parametric modeling)
- More principled optimization procedure
- Subjective evaluation

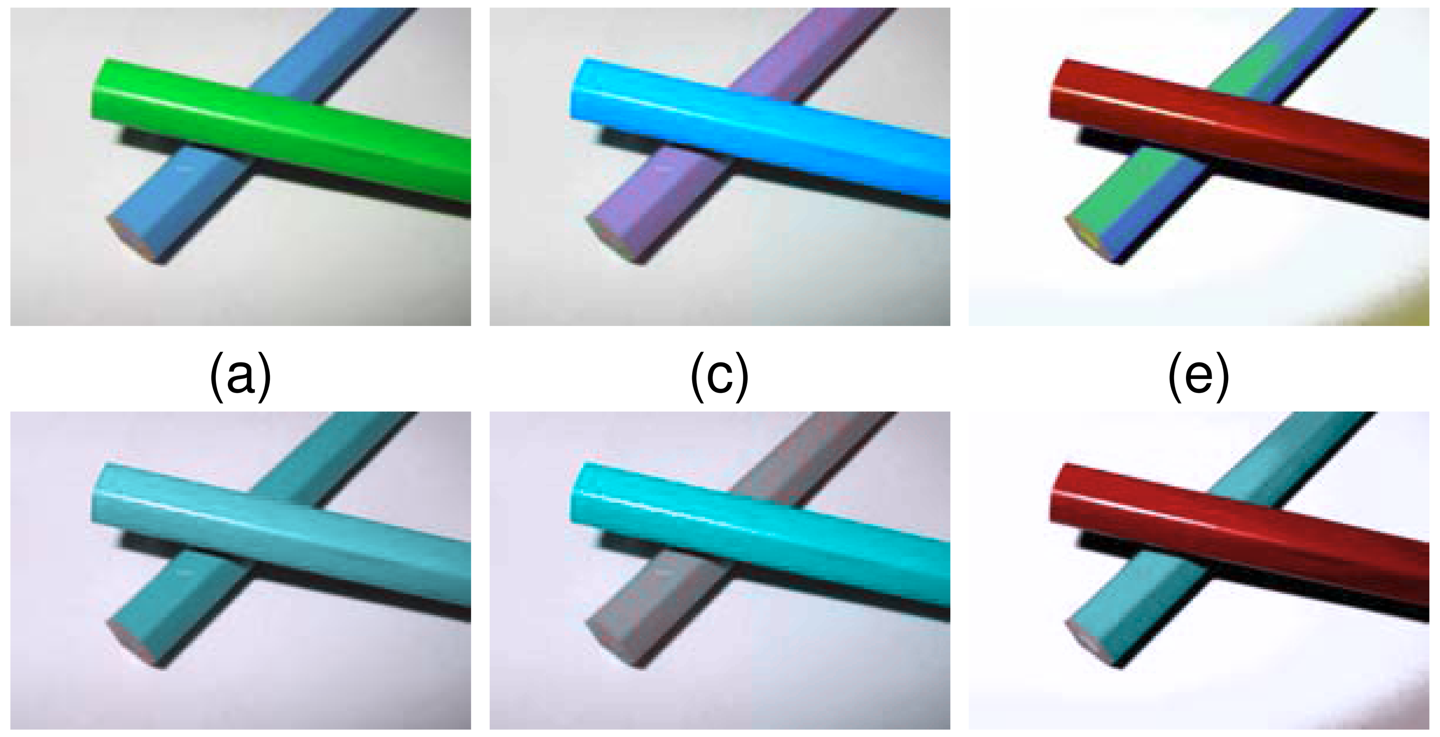
Experimental Results

- Sample results

(a) (b) (c)

Figure: (a) Original images. (b) Simulated views of the original images for protanopia (first row), deuteranopia (second row), and tritanopia (third row). (c) Simulation results of the re-colored images.

- Comparison with [2]



(a) (c) (e)

(b) (d) (f)

Figure: Comparison with [2]. (a) The original image. (b) The simulated view of (a) for tritanopia. (c)(d) The re-colored result by the proposed method and its simulated view. (e)(f) The re-colored result by [2] and its simulated view.