# Colorific: A Mixed Initiative Model for Choosing the Right Color

*Chinmay Kulkarni*
Stanford University HCI Group
Computer Science Department
Stanford, CA 94305
chinmay@cs.stanford.edu

*Julie Fortuna*
Stanford University HCI Group
Computer Science Department
Stanford, CA 94305
jfortuna@stanford.edu

**ABSTRACT**
Is it possible to automatically create a color palette relevant to a topic? Could such a palette be used to guide color choices while visualizing data? We envision a tool that automatically creates aesthetically pleasing and topic-relevant palettes for a large class of topics. In order to do this, we must first extract palettes from color pixel values of images from Google Images via clustering and topic models.

**ACM Classification:** H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

**General terms:** Design, Human Factors, Experimentation

**Keywords:** Information visualization, colors, crowdsourcing, user study, mixed initiative

## INTRODUCTION

TODO Julie/Chinmay. 300 words. Example based– Stanford Cardinal? Anger? Sadness? Understanding is aided, if supported

## RELATED WORK

CEK 400 words. One para each: Color, mixed initiative, emotions and color ( http://socrates.berkeley.edu/ plab/).

Prior work exists on automatic creation of color palettes. This work falls broadly in two categories. The first focuses on finding representative colors from images, that can be used as color palettes. The most recent of these is [8]. This line of research has so far focused only on extracting colors from a single image. This project extends this work by extracting colors from multiple, related images. I believe that some of the techniques used by [8], such as a weighted histogram that uses color saturation and neighborhood color coherence, can be adapted for multiple images too. Depending on constraints of time, I plan to explore some of these techniques.

The second category of research on palette generation focuses on optimizing visual properties, such as color saliency and perceptive color distance, both manual or rule-based, as pioneered by Brewer [3]; and with varying degrees of automation [6, 9]. I believe most such optimization research is complementary to this project, and can be used as a post-extraction step to optimize the colors chosen. Statistical work on color saliency is valuable, even if it hasn't been directly applied as a optimization objective; color saliency in the context outside data-visualization in [4, 1].

Topic models have been shown to be effective in information retrieval. Latent semantic analysis (and later, LDA), for instance, has been used to find "latent" similarities between concepts [5, 2]. Similar similarity-measures have been computed for nodes in a graph [7]. While these similarity measures may help to better cluster color-values, they don't target the domain of color recommendations directly.

## SYSTEM DESCRIPTION

550 Words

### Images as a source of color data

Colorific obtains colors for a topic from images that are labeled to be related to the topic. The assumption here is that images that are related to a topic will contain the topic's characteristic colors. Therefore, the first step in the Colorific pipeline is to obtain a set of images that relate to a topic.

In order to do this, Colorific uses a labeled corpus that contains images along with "tags" or topics that it is related to. Several labeled corpora exist on the Internet: for example, Flickr contains primarily photographs that have been tagged manually. ImageNet contains a taxonomy of images. Google Images, and other search engines, while not an explicit tagged corpus, can also provide images relevant to a topic (through search). Colorific uses Google Images as its image source, because of the large number of images it indexes (unlike, Flickr, which consists primarily of photographs), and because it does not images to be tagged explicitly (unlike ImageNet). This increased diversity and quantity in the corpus comes at a price, however– images vary in quality, size and relevance to topic. However, Colorific is largely robust to these these problems, as described below. The number of images that can be obtained is also limited by the API (to 32).

### Sampling Images

Since Colorific considers topic-related images as a proxy to topic-related colors, sampling pixels from these images is approximately equivalent to sampling color values from the topic's color space.

Therefore, given a set of images related to a topic (from Step 1 above), Colorific then randomly samples pixels from these images. Sampling could be performed in a variety of ways– it could be purely random ("population sampling"), which would count more frequent color values more often. Or, one could consider the "natural" distribution of colors for images, and weight color values that occur less frequently in the "natural distribution" higher. One could also consider more complex schemes which weight color values differently based on how close they are to edges in the image etc.

Colorific uses simple population sampling. Unlike other sampling schemes, this does not require us to know the "natural" distribution of colors in images, nor do we require image processing such as edge-detection. However, population sampling results in over-weighting of color values that occur frequently in general. We handle this problem with query expansion (Section ).

Colorific uniformly samples a fixed number of pixels from each image. This ensures that larger images don't monopolize the obtained sample. The result of this step is a sample of pixels/color-values from the images.

### Query expansion

Population sampling results in frequent colors being sampled more often. However, many frequent colors may not be indicative of the topic, and be merely a result of the natural distribution of color values in images. While one could build a sophisticated model for such a natural distribution, Colorific solves it in a different way.

Given a topic $t$, Colorific also queries Google Images for a set of topics similar to it (say $T'$), and finds their population samples ($S(T)$. Since the topics are similar, we expect their color distributions to be similar too. By "subtracting" color distributions of $T'$, Colorific finds a color distribution that is more specific to $t$.

*Subtracting distributions* There are several possible methods of subtracting distributions. The specific color distribution of a topic ($C(t)$) can be considered a hidden variable in a bayesian network such as in Fig TODO, and the observed values of the noisy distributions of other related topics could be used to obtain its value (using an expectation maximization algorithm). Bayesian models train slowly, however, and require large amounts of training data to move away from their priors (which is unavailable, due to limitations of the Image API).

Therefore, we make the stronger assumption that the observed frequency of a color ($Obs(t)$) is a linear combination of the specific distribution, and the distribution for other similar topics in $T'$. Since such a linear combination may lead to a negative value for $C(t)$, we clamp this value at zero. For our prototype, we used $\alpha = 0.15$.

$$Obs(t) \approx \alpha * C(T') + (1-\alpha) * C(t) \tag{1}$$

$$\implies C(t) \approx max\{0, \frac{Obs(t) - \alpha * C(T')}{(1-\alpha)}\} \tag{2}$$

While colors may perceptually be identical, they may have slightly different color values. Therefore, instead of subtracting raw frequencies, we bin color-values in LAB space, and subtract bin-frequencies. LAB is a color space in which the Euclidian distance between two color coordinates approximates the perceptual difference.

For binning, we used a bin size that was twice the just-noticeable difference in each dimension. After binning, frequencies of individual color values are meaningless, and are discarded. However, to ensure we don't introduce colors that weren't present in the images, we set the color value of the bin to the the color-value in the sample that is closest to its centroid. We denote the binning operator by $B$, and modify Equation 1 to:

$$B(C(t)) \approx max\{0, \frac{B(Obs(t)) - \alpha * B(C(T'))}{(1-\alpha)}\} \tag{3}$$

The result of this step is a set of bin-frequencies $B(C(t))$.

### Clustering color values

While $B(C(t))$ is an approximation to the color-distribution of the topic, we need to obtain individual colors that best represent the topic. This is equivalent to finding points in the color distribution around which the probability-density is highest.

These points can be considered as means of components in a mixture model [cite], where the color distribution is seen as a mixture of several (say $n$) independent component distributions that are chosen from with known probabilities. We tried three approaches. First, we fit a general gaussian mixture model [cite] to the binned-color-distribution. Since the LAB color space is small (and the space of perceptually valid colors smaller), this often fails for $n > 3$ gaussians. We also tried using gaussian mixtures with shared covariances (so all the Gaussians have the same shape, but may differ in size), which perform better for larger values of $n$. Lastly, we tried using K-Means clustering [cite], which is equivalent to a gaussian mixture with spherical Gaussians (so, all components are shaped as spheres). In our preliminary evaluation, we found that K-Means and shared-covariance Gaussians performed equally well, but K-Means was much faster.

Therefore, Colorific uses K-Means clustering. As with binning, clustering is done in LAB space.

K-Means clustering results in a number of clusters in the color space. Bins are assigned to the cluster whose centroid they are closest to. These centroids are the means of the spherical Gaussians. As explained above, these centroids can then be considered the colors that best represent the topic.

*Clustering quality* Since Colorific considers the centroid of each cluster as a candidate color for the topic, the quality of the clustering affects results significantly (in general, K-Means clustering is seeded randomly, and can converge on different clusters each time it is run).

We do not use any existing results for topic-relevant colors, and so it is not possible to evaluate if the clustering was done "correctly". Therefore, instead of finding the "correct" clustering, we try to maximize attributes that all correct clusters should have. *First*, clusters should be dense, so we aren't mixing colors that are perceptually distant. *Second*, cluster-centroids must be widely separated, so the colors obtained are different perceptually.

The Davies-Boudin index is a metric that tries to balance these two criteria.

$$DB = \frac{1}{n} \sum_{i=1, i \neq j}^{n} max\left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right) \quad (4)$$

$\sigma_i, \sigma_j$ are the average distance of points in cluster $i$ and $j$ from the respective clusters, and $d(c_i, c_j)$ is the distance between cluster centroids. A lower value of DB implies quality of the clustering.

However, the LAB color space is only finite, which limits cluster separation. Therefore, for Colorific , the density of the cluster is more important than cluster separation, and we use a modified version of the Davies-Boudin index as below.

$$DB' = \frac{1}{n} \sum_{i=1, i \neq j}^{n} max\left(\frac{\sigma_i^2 + \sigma_j^2}{d(c_i, c_j)}\right) \quad (5)$$

Colorific runs K-Means clustering several times and picks centroids that minimize $DB'$.

The overall goal of Colorific is to find colors that are relevant to a given topic. To do this, Colorific assumes that images related to a topic contain colors that are relevant to the topic. In particular, it assumes that there is a mapping, say $\mathcal{C}(t)$, from the set of images for a topic $t$, $I(t)$, and the set of colors, $C(t)$ that are relevant to it.

$$\mathcal{C}(t) : I(t) \rightarrow C(t)$$

Therefore, the main contribution of Colorific is to find an efficient way to compute $\mathcal{C}(t)$. We use Google Images as a means to find images related to a given topic.

Colorific then randomly samples a number of pixels from such images. Given our assumption that images are a good source of color for topics, random sampling from a set of images is used as a way to randomly sample from the topic-space.

### Query System
Google Images is queried for images from the category.

### Statistical summarization
We assume that the images from the category are a random sampling from the concept-space of the category. Taking this assumption further, we look at the *average* frequencies of the different colors as a metric of how concepts are shared across the values in a category.

Since we are interested in the colors specific to a category value, we substract a fraction of the average color frequency.

### Clustering
We cluster the result to get relevant colors in LAB space. We found that low saturation colors are less likely to be relevant, so we reweight more saturated colors to be more relevant.

## SYSTEM EVALUATION
TODO Julie. NOTE: Use APA format. Relevance, Likability (and Concrete/abstract)

We conducted surveys on Amazon Mechanical Turk to judge the quality of colors and palettes generated by Colorific. The study evaluated the following hypotheses:

**H1:** Machine-generated colors are perceived to be more topic-relevant than colors chosen randomly from the Protovis palette, a palette available in an industry standard data visualization toolkit.

**H2:** Machine-generated palettes are perceived to be more likable and aesthetically pleasing than palettes composed of random Protovis colors.

### Method
To test *topic relevance*, we showed 50 participants from Mechanical Turk six options for each topic. Four of them were colors generated by Colorific. One was a random color picked from the Protovis palette, an industry standard in data visualization. The last choice was none of the above. Participants picked one choice that best suited the topic.

To test *palette likability*, we wanted to test if we can combine colors from different football teams, like Cardinal, Cal Bears, and others to create a single, cohesive palette. We hypothesized that the algorithm can choose palettes that are liked better than random palettes. We werent sure of the best way to pick one color, so we tried four variations. One tried to pick the most frequent color, another optimized for saturation, one maximized perceptual distance between the colors, and the last picked colors from the topic at random. We asked 49 participants from Mechanical Turk to rate our four variations and one randomly generated palette, from Protovis on a scale of 1-7.

### Results
*Topic Relevance* We found that the Colorific colors were preferred in an overwhelming majority ($\chi^2$ = 83.7562, df = 2, p $<$ 0.001). 947 times out of 1200, Turkers picked a Colorific color. So, Colorific is able to pick at least some good colors for a topic.

*Palette Likability*

## DISCUSSION
JMF Quotes and stuff. Include pilot if any. (500-600 words)

At the end of the test, participants filled an optional feedback form. Like we predicted, some of them had trouble picking the best color where more than one color was acceptable. I thought this was HARD! Some of the time I wanted to pick more than one color (for CAL there was a gold and a blue option). It definitely forced to me to think in a way I don't normally think. This was a very interesting task and it really made me think about which colors are associated with certain items.

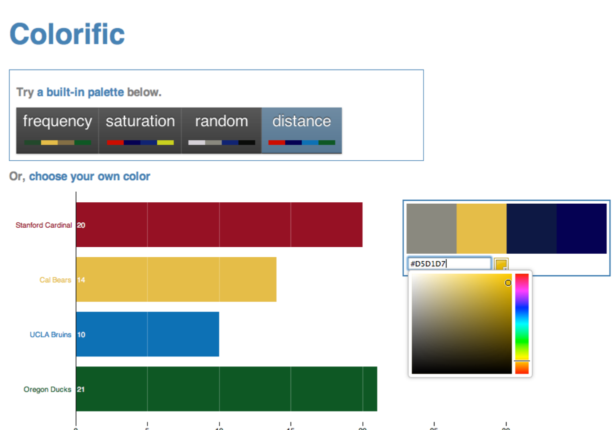
Apparently, while Colorific got the right set of colors, it failed to pick the best one for the palette. I thought it was interesting. I wish I could have chosen my own colors for a few things. It was sometimes difficult to decide because usually none of the color palettes were colors I would have chosen.

So, we thought we could use a mixed initiative strategy in which the computer offers the set of appropriate colors, and the human chooses the final color for the visualization.

## SCENARIO: DESIGNING WITH COLORIFIC

Colorific is designed to help novices with little experience in designing information visualizations quickly choose appropriate and compelling colors for their visualizations. Let's examine David, a football fan, as he makes a bar chart of the number of games won by a selection of Pac-10 football teams.

David begins making his chart in Microsoft Excel, but Excel has automatically colored each bar from a random set of colors. David wants to color each bar a representative color for the team. However, both the Cal Bears and the UCLA Bruins have the colors blue and gold. Which should be blue, and which should be gold? He loads Colorific to find out. David enters the specific topics he is plotting, such as the Stanford Cardinal and the Cal Bears, as well as their values. Colorific plots his values on a bar chart and automatically generates a selection of four potential palettes he can choose from. David selects one automatically generated palette, which looks almost the way he wants it. However, after seeing the results in context, he thinks the Cal Bears bar should be gold instead of blue. He clicks on the bar and is presented with a selection of other Colorific generated colors for the Cal Bears, as well as a color picker that allows him to tweak Colorific colors or select his own.




## DESIGN SPACE

Colorific provides one point in the space of many data visualization tools. We discuss limitations in our design and propose avenues for future work.

### Task Type

Creating data visualizations requires many component tasks. People must clean the data, then plot the data in a reasonable chart. Finally, aesthetic changes to the chart must be made, including coloring the chart appropriately. Colorific currently only supports the task of coloring a given chart. In the future, it would be beneficial to incorporate Colorific into a larger data visualization tool, so that people could clean, plot, and color their data all from within one tool.

### Expertise

Colorific is designed to be beneficial to both novice and expert data visualization designers. For novices, the main advantage of Colorific is the pre-made palettes, which can be tweaked and edited while giving the designer real-time feedback on what the color will look like in context of the visualization. For experts, Colorific's main advantage is saving time and effort. Trying to find the exact right color via a color picker is difficult and time-consuming. Colorific proposes appropriate colors, and makes it easy to make slight adjustments to a color to fit a theme.

### Time Scale

blah blah blah time

## CONCLUSION

CEK Make up stuff

## ACKNOWLEDGMENTS

CEK Thank Jeff, Scott and Jesse. (BOTH together 150 words)

## REFERENCES

1. R. Benavente, F. Tous, R. Baldrich, and M. Vanrell. Statistical modelling of a colour naming space. In *Proceedings of the 1st European Conference on Colour in Graphics, Imaging, and Vision (CGIV2002)*, pages 406–411.

2. D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

3. C. Brewer. Color use guidelines for data representation. In *Proceedings of the Section on Statistical Graphics, American Statistical Association*, pages 55–60, 1999.

4. J. Chuang, M. Stone, P. Hanrahan, and S. Consulting. A Probabilistic Model of the Categorical Association Between Colors. 2008.

5. S. Dumais, G. Furnas, T. Landauer, S. Deerwester, and R. Harshman. Using latent semantic analysis to improve access to textual information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 281–285. ACM, 1988.

6. C. Healey. Choosing effective colours for data visualization. In *Visualization'96. Proceedings.*, pages 263–270. IEEE, 1996.

7. G. Jeh and J. Widom. SimRank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 538–543. ACM, 2002.

8. B. Morse, D. Thornton, Q. Xia, and J. Uibel. Image-Based Color Schemes. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, volume 3, pages III–497. IEEE, 2007.

9. A. Zeileis, K. Hornik, and P. Murrell. Escaping RGB-land: Selecting colors for statistical graphics. *Computational Statistics & Data Analysis*, 53:3259–3270, 2009.