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

Assumptions

Colorific obtains colors for a topic from images that are labeled to be related to the topic.

Assumption 1: Images related to a topic will contain the topic's characteristic colors. Therefore, sampling pixels from these images is equivalent to sampling color values from the topic's color space.

$$S(I(t)) \approx S(C(t)) \tag{1}$$

where $I(t)$ is the set of relevant images to a topic, and $C(t)$ is its color distribution. S is a sampling function.

Assumption 2: Colorific also makes the assumption that similar topics will have a similar color distribution. i.e,

$$\begin{aligned} &\text{If } t \approx t' &&\text{for } t, t' \in T \\ \implies\ &C(t) \approx C(t') \end{aligned}$$

where T is the set of all topics.

Images as a source of color data

The first step in the Colorific pipeline is to obtain a set of topic-related images. Several corpora on the Internet allow one to find a set of images that related to a given topic. For example, Flickr contains primarily photographs that have been tagged manually. ImageNet contains a taxonomy of images. Google Images, and other search engines, do not use a manually tagged corpus, but allow the corpus to be search to find relevant images. 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 require images that are tagged explicitly (unlike ImageNet), which increases the diversity of the corpus. However, the indexed images vary in quality, size and topic relevance. The number of images per search is also limited by the API (to 32).Colorific is largely robust to these these problems, as described below.

Sampling Images

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 several ways– it could be purely random, which would count more frequent color values more often ("population sampling"). 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 and uniformly samples a fixed number of pixels from each image (S_P). Unlike other sampling schemes, this requires no knowledge of the "natural" distribution of colors in images, nor are pre-processing steps like edge-detection. Population sampling results in over-weighting of color values that occur frequently in general, which we handle by query expansion (Section).

Query expansion

Population sampling results in frequent colors being sampled more often. However, frequent colors may not be indicative of the topic, and merely be an artifact of the natural distribution of colors in images.

Given a topic t, Colorific also queries Google Images for a set of topics similar to it (say T'), and finds ($S_P(I(T'))$). Since the topics are similar, we expect their color distributions to be similar too (Assumption 2). By "subtracting" color distributions of $S_P(I(T'))$, 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 modeled as 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 infer its value. Bayesian models train slowly, however, and require large amounts of training data (which is unavailable, due to limitations of the Image API).

Therefore, we make the stronger assumption that the observed frequencies of color for a topic ($Obs(t)$) are a linear combination of the topic-specific distribution, and the distribution for similar topics T'. Since $C(t)$ is non-negative, we clamp this value at zero. For our prototype, we used $\alpha = 0.15$.

$$Obs(t) \approx \alpha * C(T') + (1 - \alpha) * C(t) \tag{2}$$

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

Lastly, while colors may be perceptually very similar, their color values may differ. Therefore, instead of subtracting raw frequencies, we bin color-values in LAB space, and subtract bin-frequencies (the LAB color space is designed such that 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, we set the color value of the bin to the the color-value in the sample that is closest to its centroid to ensure we don't introduce colors that weren't present in the images. We denote the binning operator by B, and modify Equation 2 to:

$$B(C(t)) \approx max\{0, \frac{B(Obs(t)) - \alpha * B(C(T'))}{(1 - \alpha)}\} \tag{4}$$

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.

The color distribution for a topic can be considered as a mixture model [cite], where the final distribution comes from one of several (say n) independent component distributions that are chosen from with known probabilities. In such a model, the representative colors will be the means of the components.

We tried three approaches. First, we fit a general gaussian mixture model [cite] to $B(C(t))$. The perceptually valid region of the LAB color space is small and GMMs often fail for $n > 3$ gaussians. Second, we tried using gaussian mixtures with shared covariances (so components have the same shape, but may differ in size) which work with larger values of n. Third, we tried using K-Means clustering [cite], which is equivalent to a gaussian mixture with spherical Gaussian components.

In our preliminary evaluation, we observed 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. Cluster centroids are means of the spherical Gaussians.

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 (K-Means clustering is seeded randomly, and can converge to different clusters each time it is run).

Colorific does not use existing data about which colors are topic-relevant, and so cannot evaluate if clustering was done "correctly". Instead, we try to maximize attributes associated with correct clusters. *First*, clusters should be dense, so each represents a perceptually coherent color. *Second*, cluster-centroids must be widely separated, so the colors obtained aren't just variations of each other.

The Davies-Boudin index tries to balance these two criteria [cite]. A lower value of DB indicates better clustering.

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

σ_i, σ_j are the average distances of points in cluster i and j from the respective centroids, $d(c_i, c_j)$ is the distance between cluster centroids.

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

$$DB' = \frac{1}{n} \sum_{i=1, i \neq j}^{n} max\left(\frac{\sigma_i^2 + \sigma_j^2}{d(c_i, c_j)}\right) \tag{6}$$

Colorific runs K-Means clustering several times and picks centroids that minimize DB'. The resulting centroids are displayed as candidate colors.

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 (χ^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.

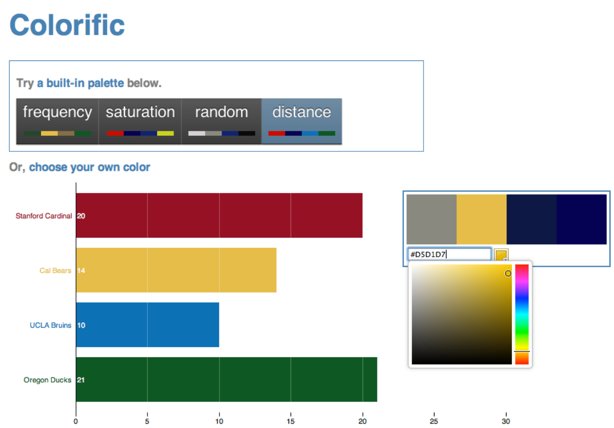
Apparently, while Colorific got the right set of colors, it failed to pick the best one for the palette. I thought it was interesting. I wish I could have chosen my own colors for a few things. It was sometimes difficult to decide because usually none of the color palettes were colors I would have chosen.

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 RGBland: Selecting colors for statistical graphics. *Computational Statistics & Data Analysis*, 53:3259–3270, 2009.