

Magic Marker: A Color Analytics Interface for Image Annotation

Supriya Garg, Kshitij Padalkar, Klaus Mueller

Stony Brook University

Abstract. This paper presents a system that helps users by suggesting appropriate colors for inserting text and symbols into an image. The color distribution in the image regions surrounding the annotation area determines the colors that make a good choice – i.e. capture a viewer’s attention, while remaining legible. Each point in the color-space is assigned a distance-map value, where colors with higher values are better choices. This tool works like a “Magic Marker” giving users the power to automatically choose a good annotation color, which can be varied based on their personal preferences.

Keywords: Human-computer interaction, Color-space

1 Introduction

Annotation of an image with text is a commonly encountered task. It is a process people use to create art, advertisements, presentations or educational tools. While using popular image processing tools like Photoshop people usually use a trial and error method before arriving at a suitable choice. None of these tools guide the users by pointing out the appropriate set of colors. The most common method to accomplish this task is to choose colors via the HSV color wheel.

This paper presents a solution to this annotation problem by designing a user interface which works as a guide to the user in this color choosing task. We build an intuitive user interface which can be easily used by a non-expert, and base our calculations on well-known color perception paradigms. Our framework captures known color design rules to form a grading scheme. This grading scheme along with user preferences can help derive appropriate colorizations.

The annotation problem includes problems like layering, highlighting and blending with the background. In our current work, we focus on selecting a color for text. Text has the interesting property that it has a high level of detail, but is familiar to people at the same time. We focus on legibility, i.e. each letter should be clearly visible, and recognizable on its own. This is different from letting people to be able to speed-read, since in that process users tend to use their mental model to complete illegible letters.

Studies on readability or legibility of text, given a foreground-background color combination show that the subjective opinion is frequently based on the aesthetic and stereotypic presumptions and may thus differ from the objectively measured performances [5]. This tells us that it is better to have an interactive tool which guides the user towards a good sub-space; the user has to make the final choice.

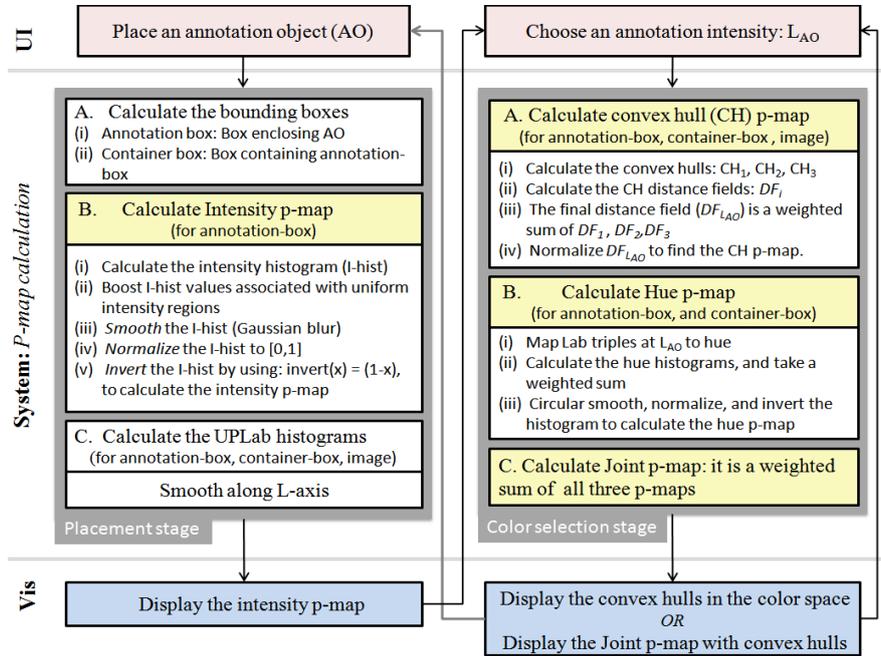


Fig. 1. The flowchart shows how the Magic-marker system calculates, and updates the preference map (p-map), once the user loads an image into it

The large amount of the work on legibility by web page designers uncovers a wealth of information regarding the importance of luminance contrast and chromatic contrast. A good review of the past studies on readability and legibility on posters and CRT displays is presented in the paper by Humar et al [6]. Most results suggest that a luminance contrast accounts for most of the variance in typical legibility experiments. Experiments by Travis et al [15] show that when the luminance contrast between text and background color was 0, a near perfect reading was still possible. This important finding means that purely chromatic differences may be sufficient for the visual system to maintain word identification. They explained the previous results by saying that typical displays produce much larger multiples of threshold luminance-contrast than threshold chromatic-contrast.

Experiments on reading speed on a color monitor [8] found that when both color and luminance contrast are present, there is no sign of additive interaction, and performance is determined by the form of contrast yielding the highest reading rate. Studies done on legibility on multi-color CRT displays [13] confirm this to some extent by saying that chromaticity contrast and luminance contrast are additive only under specific conditions. However, their results give more importance to luminance contrast by saying that chromaticity contrast can neither improve legibility if an acceptable level of luminance contrast is already present, nor substitute for luminance contrast.

Even today color difference (ΔE) equations assume that luminance and chromatic differences are additive — they usually use a weighted Euclidean distance approach. This clearly shows the lack of conclusion on the combined influence of luminance

contrast and chromatic contrast on legibility. Our tool is therefore meant to be used interactively to quickly get a choice of optimal label colors from which users can choose one according to their preferences. We encode rules from color perception, and legibility studies to arrive at the optimal colors.

The rest of the paper is divided as follows. Section 2 looks at some related work. We discuss the contributions of our work in section 3. In section 4, we discuss the theory behind our distance calculation and interface design. In section 5, we discuss the working details of the magic marker. In section 6 we look at some results. Finally we conclude in section 7, and discuss ways to apply this tool in different applications.

2 Related Work

In this section we look at some of the related work. First we look at how color can be represented to closely imitate the way humans perceive them. Next, we look at some methods that study the interaction of color. These papers concentrate on two aspects – generating color maps for mapping data to color, and designing interfaces where users can directly manipulate the colors in their visualization.

The Munsell color chart is used to evaluate the perceptual qualities of color spaces. Munsell describes color in terms of hue, value and chroma; hue corresponds to dominant wavelength, value to brightness and chroma to colorfulness. Unlike saturation, which is a statement of colorfulness as a fraction of the maximum possible for a given hue and value, chroma is an absolute measure of colorfulness. The maximum possible chroma differs among hues – for example, the maximum chroma for red is much greater than for green.

The landmark texts by Itten [7] and Wong [19] provide great insight on the human perception of color and its aesthetic aspects. Much information is also available in books by Stone [14] and Ware [17]. Color mapping is the well-studied topic of mapping data points to color based on human perception, cognition, and color theory. PRAVDAColor [2] helps users select color maps for mapping data points to color in scientific visualization. The Color Brewer [3] contains expert designed color palettes for mapping cartographic scalar data. These tools either take a lot of tweaking to come up with a suitable palette, or are pre-designed by experts. The work presented in [18] takes the expert out of the loop, and generalizes this process. Relevant also is the interactive color palette tool proposed by Meier et al [10], designed to support creative interactions for graphics designers.

While designing color, we often want to ensure that the final image looks natural, and aesthetically pleasing to the users. In [12] it was demonstrated that color maps should preserve a monotonic mapping in luminance since they are perceived as more natural by the human observers.

One popular design aspect is color harmony, which defines sets of colors that are aesthetically pleasing to human perception. In search of an intuitive 2D representation for visual designers, Itten then arranged the harmonic colors into a color wheel, which flattened this color space to mostly variations in hue. This system was used by Cohen et al [4] to quantify the color harmony of an image and shift the hues towards a harmonic setting. Wang et al [16] also use color harmony based rules while creating their framework to help users select colors in scene objects. The users can specify

their hues of choice, while the system assists by making suggestions based on aesthetics, and by optimizing the luminance and contrast channels. Neophytou and Mueller [11] develop a framework that allows users to manipulate colors directly in a 3D perceptual color space, instead of using multiple iterations using 2D color manipulation tools. This system assists users in object highlighting and annotation in color-rich scenes.

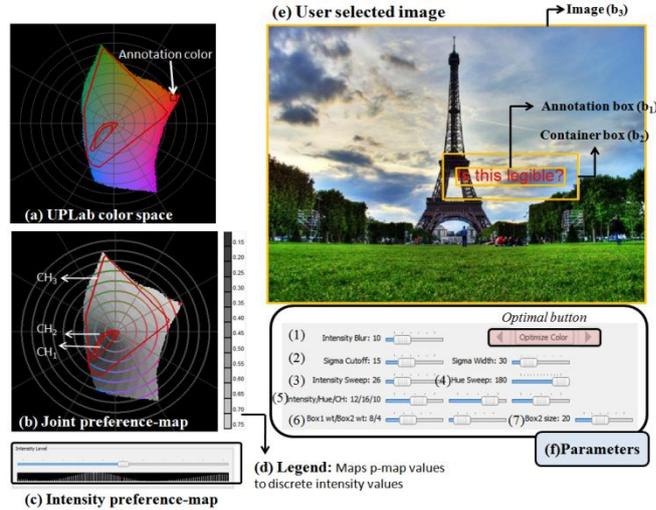


Fig. 2. Illustrates the different components of the user-interface

3 Our Approach

In this section, we discuss some of the shortcomings of the current state of art, and how we address these issues in our work.

Global vs. local effects: The work by Bauer tells us that as long as we pick a color outside the convex hull formed by the colors of a set of displayed data points, we can easily spot this target color. However, when we consider the case of a photographic image, an annotation occupies only a small portion of it. Therefore, the annotation color is highly influenced by the colors in the background surrounding it. For e.g. if the person standing next to you wears a combination of red and green, it will strike as being mismatched. However, if there are two people standing next to each other – one in red, and the other in green, the color clash reduces, and as they move further away, this clash slowly disappears. Also, when you look at Figure 2 (b), the outermost convex-hull (CH_3) occupies almost the entire color-space, leaving little choice for the users. However CH_1 , the convex-hull formed by the colors surrounding the text is much smaller, and is the set which truly represents the colors we must avoid. We take this local vs. global effect into consideration by giving different weights to different regions of the image depending on their distance from the annotation.

Conflicts in color mixing: The studies on legibility and readability of web-pages have shown us that there is no clear consensus on the interaction of luminance and chromatic contrast. Our system has an intuitive interface to assist users where they are provided with informed choices to help them make the final decision.

Color Space: Human vision is designed such that it is natural to describe colors as locations in a 3D space. The tristimulus values of a color are the amounts of three

primary colors needed to match that test color. In the CIE 1931 color space, tristimulus values provide a complete color description; however they do not correspond to the way humans perceive colors. Distances between tristimulus triples do not reflect the degree to which we perceive the colors to differ. Therefore, the CIE introduced two perceptual color spaces in 1976 – the CIELUV and CIELAB spaces. The CIELUV space was used in [18] to design their color palettes. Though these spaces are designed so that the distance in the color space is proportional to the difference perceived by humans, they do not contain the colors arranged uniformly along the hue, and chromatic channels. The Munsell data set, as discussed in section 2 is an example of a color space which separates luminance, hue and chroma into 3 orthogonal axes. Therefore, for our calculations, and user interface, we use a modified CIELAB space, such that it satisfies the properties of the Munsell color set.

4 Overview

The main task for our system is to find rules which assign legibility scores to each point in our 3D color space. In this section, we look at how these rules are derived.

Minimum contrast for legibility: Web-page designers have long studied the ways to choose a good foreground-background combination for maximum legibility. Though placing text on an image is far tougher since the background color is rarely uniform, we can still learn from their work. Since results from their studies show that a certain luminance contrast assures legibility, we must make sure that we achieve the minimum contrast from most colors in the background.

According to the studies by Maureen Stone [14], a contrast (ΔL) of 20 is legible for text; contrast of 30 is easily readable; and contrast of 60 is robustly readable. This result helps us assign scores to intensity levels based on luminance contrast.

Perceptually uniform CIE-LAB space: We require a perceptually uniform color space for our system since this would lend to an intuitive system for a user. In the CIE-Lab, colors at similar distance have similar perceptual distances; however the space itself is not a perceptually uniform space. This is because colors sharing the same Lab hue angle do always not share the same apparent hue. For example if you move inwards along a line joining a bright blue to the origin, the apparent hue changes to purple: this is the well-known “blue turns purple” problem in gamut mapping. Furthermore, the angles between hues are not equal, and colors of constant chroma are not equidistant from the neutral axis. To rectify this, we use the non-linear mapping ICC profile provided by Bruce Lindbloom [9].

The color space ICC profile performs a nonlinear mapping of Lab so that all properties of the Munsell color set are met. This means that when CIE-Lab colors are transformed through this profile, the resulting Uniform Perceptual Lab (or "UP Lab") colors have these properties:

- All colors of constant Munsell hue have the same UP Lab hue angle.
- All Munsell hues are evenly distributed around the hue wheel.
- All colors of constant Munsell chroma (saturation) lie the same distance from the neutral, i.e. chroma rings are perfect circles, centered on neutral.

- All chroma rings are equally spaced.

During this transformation, the L^* channel and the neutral colors remain unaffected. An illustration of the transformation is shown in Figure 3.

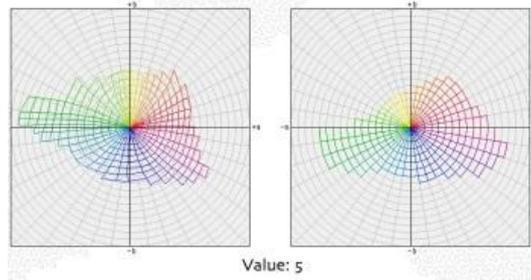


Fig. 3. The transform of the CIE-Lab space to UP-Lab space for the Munsell value 5.

Convex hulls: Bauer et al in [1] have shown that given a bunch of colored points laid out as a scatterplot, it is easier to find a target color with a color outside the convex hull (in the CIE-Lab space) of the source colors than inside the convex hull. This holds both when the target was linearly separable in chromaticity only, or in a combination of luminance and chromaticity.

This result gives us a metric which calculates the score for each point in the CIE-Lab space based on their distance from the convex hull.

Theory: Based on the background work done in color perception, and legibility of text on multi-color screens, we form some ground rules which help us in designing the Magic Marker user interface. We use three orthogonal preference-maps (p-maps) to find the colors which lead to good legibility. Finally, we combine these values by taking a weighted sum, to find the final preference-map value – the joint p-map.

Luminance Contrast: We need to make sure that there is a minimum amount of luminance contrast between the text and background. Since the background is not uniform, we calculate the contrast based on the intensity histogram of the background. Our thumb-rule is: $\Delta L > 30$ is sufficient for giving an ideal annotation color.

Hue based contrast: Luminance contrast alone is not sufficient to make a good labeling color. We need to make sure that the hue is also sufficiently distinct. For this purpose, we use the UP-Lab space to calculate our hue-value. The hue is determined by the angle formed with the origin at a given intensity: $\text{hue} = \tan^{-1}(b/a)$. This is similar to the LCH_{Lab} space as defined by Bruce Lindbloom. The further away a hue is from the hues in the background, the better it performs.

Convex hulls: As we learn from Bauer et al [1], it is easier to find a target color when it lies outside the convex hull formed by the distractor colors. This makes sure that we avoid the hue and chroma combinations present in the background. Also, since this space covers chroma, we do not create a separate preference-map for it. To make sure that we give higher importance to local features, we divide the image into three nested boxes (see Figure 2 (e)) – the annotation box (b_1) which is the bounding box for the annotation, the container box (b_2) that extends a small amount beyond b_1 , and finally the whole image (b_3).

Interface design: While designing the interface, we have to make sure that a user can explore the color space in the intuitive manner. Further, he should be able to make small changes to the annotation color in a controlled fashion, i.e. change the luminance, chroma or hue in small steps. So, for each intensity level, we display the corresponding colors, and to guide the users, we overlay circles to represent chroma,

and sectors to represent hues. The users can move along the sectors to get more saturated/de-saturated colors, and move along the circles to gradually change the hue.



Fig. 4. Sliding window. We show how uniform intensity is determined by passing a window less than the size of a

5 Implementation details

A user can upload a picture of choice into the interface, and the system gives a score to each point in the 3-D color space, based on its legibility potential at a given location in the image. We work in the UP

Lab color space for the score calculation, and the visual interface. To calculate the scores, we create a preference-map (p-map), which takes into accounts three values: (a) the convex hull p-map, (b) intensity based p-map, and (c) hue based p-map. For all of them, we work on the histogram of the image, in different dimensions. Also, to avoid sudden changes, and to take neighboring values into account, we apply the Gaussian filter at different steps.

An overview of our system is presented in the flowchart in Figure 1. Initially, bounding boxes are created for the region surrounding the text, and the intensity p-map is calculated based on the colors present in these regions. The user can then select the appropriate intensity levels, and for the selected intensity, the hue based and convex hull based p-maps are created to display the joint p-map which is a weighted sum of all three p-maps. The rest of the section will discuss these steps in detail.

Calculating the preference-map (p-map): Here we discuss how a color’s hue, intensity, and location with respect to the convex hulls defines its p-map value.

Intensity based p-map: The intensity based p-map assigns each intensity level in the range $[0,100]$, a value based on how well colors in that level will work from the point of view of luminance contrast with the background. As we mentioned in section 4.1, a contrast of 30 will make text easily legible. Here contrast is the absolute difference between the foreground and background intensities. Intensities that are farther away from the ones present in the background are given a higher value. A simple way is to calculate the intensity histogram, reverse it, and normalize it to the range $[0,1]$. Also, applying a Gaussian filter will help to give a lower value to intensities in close proximity to those in the background. However, we found that this didn’t work well enough in cases where there is a small patch with an almost uniform intensity (see Figure 4). This is because they do not have sufficiently large presence in the histogram to be penalized heavily; however since they form a continuous block, using a similar intensity works poorly. So we add a module which passes a sliding block over the text area to find out regions with small variations in intensity. If any such region is found, then the mean intensity, and intensities within a certain width (sigma width) are given a low value in the final histogram. These low values start from zero at the mean, and increase linearly towards the ends.

Hue based p-map: In the previous section we assign each intensity level a value. Within an intensity level, we would like to avoid the hues which are present in the background, especially those with similar intensity levels. We first find the hue values

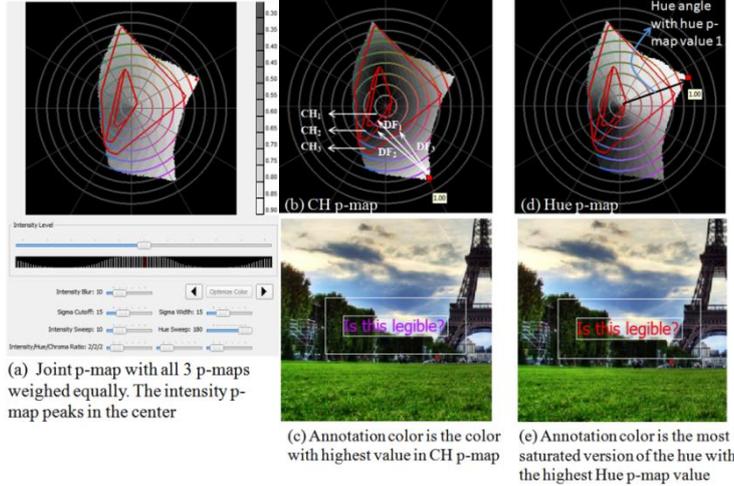


Fig. 5. (a) Shows the interface, including the joint p-map at intensity level 50. Next we look (b) CH p-maps, and (d) Hue p-maps look separately. (c),(e) show the image annotated with the best p-map value restricted to CH, and hue respectively.

for each point in our current luminance space. Then we calculate the hue histogram for the annotation box, and the container box. This is followed by smoothing using a circular Gaussian filter since hues wrap around in the UPLab space. Then we normalize the data to the

range $[0,1]$. Finally, the invert function is applied to get the hue p-map p_H .

Convex hull based p-map: The global convex hull method mentioned in section 3.3 appears overly restrictive for annotations which are local in nature, i.e., the color content of a distant image region may not be a distractor here. Further, they work in 2-D space, not taking the intensity into account. We therefore extend this idea by:

(a) Calculating convex hulls separately for each intensity level. In order to prevent sudden changes in the shape of the convex hulls, and also to take the occurrence of colors in neighboring intensities into account, we apply a 1-D Gaussian filter to the histogram of each chroma value.

(b) Creating separate convex hulls for three nested boxes in the image. Since the local effect is expected to be greater than the effect of the image as a whole, we give different weights to the boxes. The parameter window-size tells us the distance by which b_2 extends beyond b_1 (see Figure 2 (e)). Also, we give these 3 boxes weights w_1, w_2 and w_3 , with $w_1 > w_2 > w_3$. Since weights are really ratios, we have 2 more parameters w_1 and w_2 , with $w_3 = 1$. We then calculate the distance field $DF(a, b)$ of all possible $a*b$ pairs from the convex hulls as follows: $\sum_i (w_i \times DF_i(a, b))$, where $DF_i(a, b)$ is the distance of point (a, b) from the i th convex hull, CH_i (see Figure 5 (b)). These distance field values are normalized to the range $[0,1]$ to give us the convex hull p-map values p_{CH} .

The joint p-map is calculated as a weighted sum of the scores calculated in the previous three subsections, i.e. $(w_I * p_I + w_H * p_H + w_{CH} * p_{CH}) / (w_I + w_H + w_{CH})$.

Visual Interface: The visual interface as shown in Figure 2 helps the user explore the UP Lab space and find colors which have good p-map values for a given image and location. The intensity map is shown as a histogram at the bottom to guide the user towards intensities providing good luminance contrast. Next, we see a legend at the



Fig. 6. This shows some sub-optimal results obtained when the sigma-cutoff parameter is changed to 0. The first two images come from the higher ends of the intensity range, and the third image is from a lower intensity.

right, which shows the range of the values present in the p-map at the current intensity level. Finally, we see the color space, along with the convex

hulls (in red). We also overlay sectors and circles on the color space, so that a user can easily select colors with the same hue or saturation.

The user can toggle the display to see the p-map instead. In this view, we discretize the p-map values into 20 levels, and assign each point an intensity value based on these levels. This helps the user follow the patterns in the p-map much better. The user can hover the mouse over a point to see the exact p-map value. We retain the circles and sectors from the color space display, and fill them with the colors they represent to guide the user as he makes selections on the basis of the p-map. When an expert user is using the system, we can expose the parameters of the system, so that he can tweak them to get better results.

We provide an optimal button, which finds all the colors with the highest p-map values. This requires us to find the joint p-map values at all intensity levels. Next we find the maximum p-map value $\max_{p\text{-map}}$, and return all the colors with the property:

$$p\text{-map}(l, a, b) > \max_{p\text{-map}} \times \text{threshold}_{\text{opt}}$$

The threshold value $\text{threshold}_{\text{opt}}$ is a real number between 0 and 1. We usually use a value of 0.95 in our interactive system. Once these optimal colors have been found, the user can easily browse through them using forward/backward buttons. A minor tweak in one of the optimal colors often gives acceptable results.

6 Results

The Magic-marker tool is very simple to use for even novice users, and responds interactively for most operations. Let us look at a running example to see how the system works.

Case Study: The tool starts up with some initialization steps. The mapping of the CIE-Lab space to UPLab space using the look-up table takes a relatively large amount of time (~1-2 minutes). After the system starts running, a user can load any image of his choice. At this point, a default setting is already initialized, along with the default intensity level setting of 50. Once the image has been read in, we can see the intensity p-map values as a histogram, as well as the joint p-map values of the currently selected intensity in a 2-D representation as shown in Figure 5(a). The intensity p-map guides the user towards the intensity levels with higher p-maps. In the present case, we can clearly see that the intensity p-map is highest in the middle region – around the intensity level 50. Next, we look at the hue p-map and convex-hull p-map separately to understand how the p-map calculation works in those spaces. Convex hull p-maps, as shown in Figure 5(b) have the lowest p-map values inside CH_1 (where

it is always zero), and they progressively increase as we move outwards from CH_1 . Finally, a p-map value of 1 is reached at at-least one of the edge locations. In this example, the maximum is reached at purple, on the lower left corner.

The histogram p-map calculation takes a hue sweep value of 180. This means that if there is exactly one color present in the background, the hue p-map values will increase starting from 0, as we move away from the background hue, reaching a maximum value of 1 at the diametrically opposite hue value. However, since the hue p-map is already normalized, we will definitely see a hue value with the p-value 1 throughout. In Figure 5(d), we can see the hue value with p-map 1 highlighted – let us call it huepref-max. The text in Figure 5(e) is annotated with a color which is a combination of the current intensity level (50), hue value huepref-max, and the maximum possible chroma for this hue and intensity combination.

This analysis of the individual p-maps shows us that due to the way the CH p-map is calculated, the best colors will always lie on the edge of the color space, i.e. those colors that have the maximum possible chroma for a hue and intensity combination. Intensity maps and hue maps on the other hand restrict the best colors to a certain intensity level or hue value – thus any intensity or hue can be given the highest p-map value. When we combine these p-maps to form the joint p-map, the restriction from CH p-maps will remain.

The best colors as suggested by the hue p-map and CH p-map are clearly distinct – red and purple respectively. However, they are both perceived to perform equally well. We transfer our attention back to the joint p-map, with all three p-maps being weighed equally. Next we press the optimize button, which indeed returns variations of the red and purple annotation colors seen in Figure 5.

Our default setting had the sigma cutoff value set at 15. If we change it to zero, we end up with the intensity p-map at the bottom of Figure 5(a). This intensity p-map gives a good preference value to a much larger number of intensities. This makes it difficult for a user to explore the space, and find an appropriate color. Even the optimal button returns a lot of colors which do not lend themselves to a legible annotation. Some of them are shown in Figure 6. The first two have intensities in the higher intensity range (about 60-65), whereas the blue comes from intensity level 23. All these intensities have a low to mid-range score in the first intensity p-map, and are never considered as optimal color ranges. This shows us that identifying regions of uniform intensity in the background is a highly necessary process.

7 Conclusion

In this paper we present an interactive framework for users to choose appropriate colors for annotating their images. This works better than a static program which can suggest the best colors, since the field of legibility is still under study, and has no fixed conclusions on the interaction between luminance contrast and chroma contrast. Our system is fairly general, and can be easily modified to incorporate any newly acquired knowledge about color perception and luminance/chromatic contrast.

Future work includes extending this system to support highlighting elements in volume datasets. Further, we can incorporate aesthetics into our system by giving scores to colors based on color harmony.

References

1. Bauer, B., Jolicoeur, P., and Cowan, W.B.: Distractor heterogeneity versus linear separability in colour visual search. *Perception*(London. Print). 25(11), 1281–1293, (1996)
2. Bergman, L.D., Rogowitz, B.E., and Treinish, L.A.: A rule-based tool for assisting colormap selection. In: *Proc. of IEEE Visualization'95*, pp. 118-125, (1995)
3. Brewer, C.A.: Color use guidelines for data representation. In: *Proceedings of the Section on Statistical Graphics, American Statistical Association*, pp. 55–60, (1999)
4. Cohen-Or, D., Sorkine, O., Gal, R., Leyvand, T., and Xu, Y.Q.: Color harmonization. *Proceedings of ACM SIGGRAPH 2006*. 25(3), 624-630, (2006)
5. Hall, R. and Hanna, P.: The impact of web page text-background colour combinations on readability, retention, aesthetics and behavioural intention. *Behaviour and Information Technology*. 23(3), 183–195, (2004)
6. Humar, I., Gradisar, M., and Turk, T.: The impact of color combinations on the legibility of a web page text presented on CRT displays. *International Journal of Industrial Ergonomics*. 38, 885-899, (2008)
7. Itten, J.: *The art of color*. Van Nostrand Reinhold New York.(1969)
8. Legge, G.E., Parish, D.H., Luebker, A., and Wurm, L.H.: Psychophysics of reading. XI. Comparing color contrast and luminance contrast. *Journal of the Optical Society of America. A, Optics and Image Science*. 7(10), 2002-2010, (1990)
9. Lindbloom, B.J.: CIE Lab to Uniform Perceptual Lab profile is copyright © 2003 Bruce Justin Lindbloom, (2003)
10. Meier, B.J., Spalter, A.M., and Karelitz, D.B.: Interactive color palette tools. *IEEE Computer Graphics and Applications*. 24(3), 64–72, (2004)
11. Neophytou, N. and Mueller, K.: Color-Space CAD: Direct Gamut Editing in 3D. *IEEE Computer Graphics and Applications*. 28(3), 88–98, (2008)
12. Rogowitz, B.E. and Kalvin, A.D.: The “Which Blair Project”: a quick visual method for evaluating perceptual color maps. In: *Proc. of IEEE Visualization '01*, pp. 183-190, (2001)
13. Spenkelink, G. and Besuijen, J.: Chromaticity contrast, luminance contrast, and legibility of text. *Journal of the Society for Information Display*. 4(3), 135-144, (1996)
14. Stone, M.C.: *A field guide to digital color*. AK Peters, Ltd.(2003)
15. Travis, D.S., Bowles, S., Seton, J., and Peppe, R.: Reading from color displays: A psychophysical model. *Human Factors*. 32(2), 147–156, (1990)
16. Wang, L., Giesen, J., McDonnell, K.T., Zolliker, P., and Mueller, K.: Color Design for Illustrative Visualization. *IEEE TVCG*. 14(6), 1739-1754, (2008)
17. Ware, C.: *Information visualization: perception for design*. Morgan Kaufmann.(2004)
18. Wijffelaars, M., Vliegen, R., van Wijk, J.J., and van der Linden, E.J.: Generating Color Palettes using Intuitive Parameters. In: *Computer Graphics Forum*, pp. 743-750, (2008)
19. Wong, W.: *Principles of color design*. Wiley.(1996)