Data driven webpage color design

Zhenyu Gu [*], Jian Lou

Shanghai Jiaotong University, China

ABSTRACT

This paper presents a design framework for automatic webpage coloring regarding several fundamental design objectives: proper visual contrasts, multi-color compatibility and semantic associations. The objective functions are formulated with data-driven probabilistic models: the Color Contrast model concerning visual saliencies is trained on 52,000 basic components parsed from 500 popular webpages. Color Compatibility and Semantics are modeled from a dataset of manually tagged and rated color schemes from Adobe Kuler. To incorporate the multi-objectives in optimization, the framework adopts a lexicographic strategy, which determines the best choices by optimizing the objectives one by one in a user specified sequence. We demonstrate the effectiveness of the models and the flexibility of the framework in two typical web color design scenarios: fine tuning a colored page and recoloring a page with a specified palette. Independent perception experiments verify that the system-generated designs are preferable to those generated by nonprofessionals.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

Color is a primitive modularity of a vision system, triggering instant aesthetic sensation [1]. The first impression of a webpage on an audience only requires an initial 50 ms of glimpse [2]. While many people can easily distinguish good and bad color designs, creating a pleasing color design requires much more time and effort [3]. Webpage color design usually requires considerable experimentation and guesswork. It is highly difficult to systematically investigate the world of all color constellations. Even permutations of several colors on a webpage are usually astronomical. (Human beings can distinguish approximately 128 different hues, 130 different tints, and 23 shades, reaching a total of approximately 82,720 different colors [4].)

Is it possible that computers complete the coloring process by automatically evaluating and screening this huge amount of design alternatives? The main challenges in automatic color design are: how to determine a coloring that is aesthetically pleasing yet also contributes to the functionality of the design, and how to use color to convey specific moods to the audience.

In this paper, we begin with computational encoding of the properties that make a webpage coloring agreeable. A data driven framework can determine an optimum coloring effect for the webpage with minimal user interactions and specifications.

A webpage coloring must consider at least three fundamental design objectives [5]: functionality (usability) for visual communication, i.e., legibility and proper visual saliency allocation; multi-color compatibility, i.e., harmony; and sematic associations with the specific moods.

Three objective functions are subsequently formulated with probabilistic models on datasets of design examples:

Color contrast. Color contrast not only makes a form distinguishable, but research suggests that proper distribution of color contrasts yields a hierarchy of visual saliencies [6], allocating our resources of attention and reducing our cognitive loads. Color contrast is also regarded as a foundation of harmony [7]. To determine proper edge contrasts of color patches on a webpage, a probabilistic model is trained on 52,000 visual components parsed from 500 webpages with their edge contrast properties and selected predictive features.

Color harmony. Harmony is the most addressed objective of color design. The framework employs a Color Compatibility model, developed by O'Donovan et al. [8]. The model is a regression of user preferences on a large dataset of color schemes from Adobe Kuler and Amazon Mturk samples. A color scheme is an ordered row of five colors.

Color semantics. Note that contrast and harmony are general qualities that cannot capture the specific color themes. To express specific color moods, we introduce a probabilistic model, driven by the dataset of tagged color schemes from Adobe Kuler, to fine-tune a webpage according to specified keywords.

---

* Corresponding author.
  E-mail address: zygu@sjtu.edu.cn (Z. Gu).

Finally, to incorporate the multi-objective functions in the coloring process, the framework adopts a lexicographic strategy [9], which applies the objective functions sequentially, not simultaneously.

The main contribution of this paper is an attempt to automate webpage coloring through a data driven optimization. Fundamental objectives of webpage color design are addressed and formulated with probabilistic models, to capture desirable properties of webpage coloring underlying collected design data. We demonstrate that cooperation of the objectives could lead to acceptable converged states of results. Independent perception experiments verify that the system-generated designs are preferable to designs generated by nonprofessionals.

2. Background

Color design supporting tools are closely related to studies of color perception and aesthetic computing.

2.1. Color design supporting tools and automatic coloring

There are generally two approaches to color design automation: rule-based and data-driven. A number of quantitative models in color harmony theory have been put forward in the past two centuries to help designers create pleasant color combinations [7, 10–12].

Itten's color contrasts [7] and Matsuda's harmonious templates [13] are the most widely cited models. Color harmony is quantified by certain geometric structures in the color space. The models have been widely applied to graphic design and image processing systems [14–19]. The rule-based models, however, seem rigorous but oversimplified. Ou et al. studied people's preferences for monochrome and color pairs, indicating that colors selected according to these rules do not necessarily lead to harmony [20].

To complete webpage coloring, this paper uses data-driven approaches, which predict the performances of new color designs in real contexts of layouts based on existing sample data. O'Donovan et al. [8] used a regression model to predict the compatibility of a color scheme based on online color scheme datasets. The model has been integrated into a clothing optimization system [21] and a generative pattern coloring system [3]. The data-driven approaches are expected to be more robust to predict colorings in real contexts of uses than the classical deterministic rules. Moreover, early rule-based systems focus on color harmonization, which is just one of the aspects of color design. It is suggested that having only color compatibility cannot ensure a successful coloring result in design [3]. We believe that the data-driven approach is not only suitable for color compatibility but also useful for optimizing color contrasts and semantics. It is expected that with well defined objectives and proper constraints, the optimization may lead to a stably converged coloring result.

2.2. Coloring in a context of usage

Some studies focus more on evaluating colors within their contexts of use. Zhang et al. developed a color harmony measure model with shape information [22]. Wang et al. applied color scheme to enhance images while complying with common sense of the natural world colors by taking color-texture joint probability as a constraint [23]. Lin implemented a generative auto-coloring system for 2D patterns [3]. They collected learning cases of 8000 designs of color patterns to establish a mapping between the color properties and spatial features of shapes.

Webpage coloring has functionality for visual communication, which is slightly different from decorative patterns. Coloring a webpage must ensure accessibility. W3C's WCAG2.0 gives

Fig. 1. Due to the improper contrast of the wireframe, the top table seems incongruous as opposed to the bottom. This example was created by designer Tyler Tate (www.smashingmagazine.com/2009/10/07/minimizing-complexity-in-user-interfaces/).

some recommended Legibility standards of minimum luminance contrast ratios against backgrounds for both small and large fonts, regarding peoples with low vision or color vision deficiency [24]. Our system uses those guidelines as the hard constraints in coloring processes.

Poor color contrast choices can induce eyestrains and headaches (see Fig. 1). Saliencies typically arise from contrasts between components and their surroundings [25]. The proper distribution of saliencies allocates viewer limited resources of attention, which is an essential measure of the quality of the visualization [26]. Color blocks on the webpage with different usages often need different visual saliencies. A vivid color patch (high saturation against the background, sometimes called the accent color) produces a pop-out effect, enabling humans to detect targets such as navigation bars or link buttons very rapidly, even without any consciousness. A psychological phenomenon of simultaneous contrast demonstrates that color contrast may bias our perceptions of color [27].

To predict web components' edge contrasts, our data-driven model captures the joint probabilities between the web components' edge contrasts and other traits embodying their functionalities. Kumar's study in building a webpage design repository shows that webpage modules and elements have obvious categories of forms and functions [28].

2.3. The meaning of colors

Color semantics, sometimes known as the meaning of colors [5], attract interest from researchers across many disciplines, including psychology, design and computing. The variations of color, as an emotion messenger, can be associated with different moods, such as cheerful or quiet, hard or soft, warm or cool, bitter or delicious. Lin et al. used the term semantically resonant to refer to color choices that are evocative of a given concept [29].

Semantic Differential of Kansei Engineering is a well-established method for designers to capture semantic feelings of visual forms [30]. Through a user perception study, then multivariate factor analysis, Ou et al. suggested three semantic dimensions: activity, weight and heat, with corresponding empirical functions to evaluate any input unary and dual colors in a CIElab space [31,32].

Color constellations can evoke more rich and complex imagery. Semantics for multi-color combinations (color schemes) are generally category based. Kobayashi's Color Image Scale [33] provides 1170 three-color combinations grouped into 15 subsets and labeled with 180 adjectives on two semantic scales. Pantone Color Scheme [34] defined 27 adjective clusters. Each cluster

contains 24 three-color combinations. Moreover, Adobe Kuler and Colourlover communities have accumulated a large number of color schemes with semantic tags for retrieval. Compared with the Kansei method, the category-based method is more simple to use in design practices.

Category-based tools so far only support keyword-based color scheme retrieval. It is possible and more useful to generalize and predict beyond the example data. Our data driven approach provides a probabilistic regression model to interpolate the subtle mood changes in color schemes based on some discriminative features of semantic categories.

3. Data-driven approach and system framework

The framework is illustrated in Fig. 2. The framework mainly consists of a *color block/text parser*, a *generative design engine*, three objective models, and two datasets. The three data driven objective models are prepared offline, to statistically capture the properties of desired colorings with the collected datasets. Details of the two datasets are given in Section 4.

In runtime, the pipeline takes a webpage's fixed layout and a few keywords describing certain moods as inputs, and outputs sorted optimum colorings of the webpage.

An input page layout specifies which segments on the page can be colored in and which segments are assigned with the same color. For instance, the background of the sidebars may have the same color as the body texts, and all hyperlink texts must be the same color.

The *color block/text parser* processes the input webpage into a set of elementary color patches and text boxes, which is then encoded in the *generative design engine* for generating alternative page colorings. Moreover, it extracts all the components' spatial features and properties of the Document Object Model (DOM), which are necessary predictive information for some context relevant evaluations later on. Details of the Parser are given in Section 4.1. The procedure of the page parsing and feature extraction must keep same as in preparing the *edge contrast training set* for the contrast model.

The *generative design engine* provides the infrastructure for regenerating and screening new designs by calling the three objective models.

3.1. Two phase design generation: palette preparation and page coloring

Coloring a webpage can be modeled as a spatial permutation problem. Design solutions are encoded as a string of RGB values. Changing the sequence and values generates a new alternative coloring of the webpage. A webpage may have hundreds of blocks and text boxes. Most segments are colored in groups. We refer to a set of segments that map to the same color as a color group, and a webpage normally uses less than ten colors. Suppose a webpage has k segments assigned with n ($n \leq k$) colors picked from a set of m colors ($m \in [n \ldots 256^3]$). There are a total of $C_m^n P_n^k$ solutions (C means combination and P means permutation). If we pick colors from the full color set of RGB, even if the page has only a few color groups and the regions of color groups are fixed, the solution set is still enormous.

To reduce the complexity, the coloring process in the *generative design engine* has two phases: palette (theme) generation and page coloring. The framework supports automatic color palette (theme) retrieval or selection from the Kular dataset in terms of user input keywords. A palette is a considerably smaller color set generated in advance to page coloring. It is an intermediate form that enables pre-evaluation of color harmony and color semantics without taking into account the spatial context of the page. This

is a common practice in real color design. The palette can also be generated by color quantization of a mood board [35,36] or inquiring online databases, such as Adobe Kuler. A number of tools have been developed to manually or automatically generate harmonious color themes [10–13]. Since the palette generation is a well addressed topic, we focus our later discussion and demonstrations more on how the system evaluates and optimizes the coloring effects in the spatial contexts of webpages, which is believed the bottleneck of the pipeline.

3.2. Data-driven design evaluation

To evaluate the new designs, the framework uses three data driven models: the contrast, harmony and semantic models.

3.2.1. Color contrast

The edge contrasts of the components parsed from the input webpage must be evaluated in their spatial contexts. The contrast model predicts each components edge contrast with its spatial features and DOM properties. The details of the model are given in Section 5.1. The model is trained offline on the *contrast training set*, which has 52 000 elementary color patches (visible DOM nodes and their render-time color properties) parsed from the 500 positive webpage examples in the *web design repository* collected with a *design crawler*.

3.2.2. Color harmony

Color harmony is evaluated using the data-driven model introduced by O'Donovan et al. [8]. To evaluate a palette more than five colors, we extract representative five colors from the palette using k-means ($k = 5$) in CIElab space. The colors selected are the closest ones to the cluster centers. Their spatial arrangement retains their original sequential order on the palette. In general, low-scoring schemes are rated lower than high-scoring schemes regardless of ordering [3].

To evaluate a colored page, the five largest color groups on the page are reordered linearly. The five color groups can be modeled as a node graph with connections that represent their spatial closeness. Next, we transform the graph into a line by solving a traveling salesman problem (TSP) to find the shortest path passing all five nodes. If a coloring solution contains fewer than five colors, we repeat colors in order of size to fill the remaining swatches.

3.2.3. Color semantics

The semantic model needs a few user specified keywords as an extra input to complete its setup. Users can specify an expected mood with one or multiple keywords, for instance:"Business","Clean", "Playful" and so on. Each keyword retrieves a set of color schemes from Adobe color theme database. With identified discriminative features to semantic categories of color schemes, which were firstly introduced by Csurka et al. [37], our approach is to use Probability Density Functions to infer the high probability areas in the feature space based on the collected samples. The method is detailed in Section 5.2. The approach provides a regression to capture the subtle mood changes in between the color scheme samples.

It is necessary to evaluate color semantics once again after assigning colors from the palette to the page context, because the member colors and their areas might have changed in the spatial permutation.
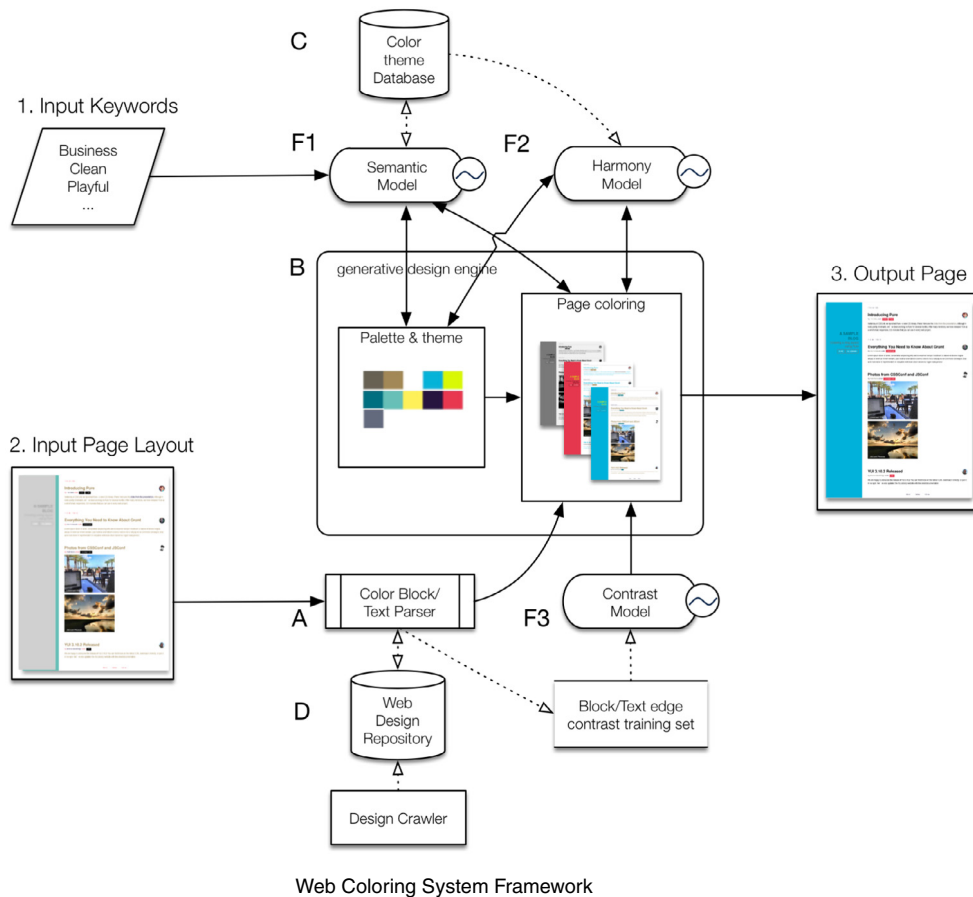
Web Coloring System Framework

Fig. 2. The framework mainly consists of a color block/text parser (A), a generative design engine (B), two databases (C, D), and three predictive models (F1 ~ F3). The pipeline takes a few keywords (1) describing certain mood and a webpage's fixed layout (2) as inputs, and outputs sorted optimum colorings (3) of the webpage. The dashed connections represent the offline processes for the objective model trainings.

3.3. Multi-objective optimization utilizing user interaction

Color design can be modeled as a multi-objective optimization problem. The optimization usually requires integrating all of the objective functions as a weighted sum or product. The weight setting of multi-objective optimization is still a challenge [38]. Towards different design specifications, the adaptive weight learning is computationally intensive. In this paper, we adopt a lexicographic strategy, a more flexible approach to coordinating the multi-objectives utilizing user interaction, which determines the best choices by optimizing the objectives one by one in a user specified sequence. The strategy will be detailed in Section 6 and demonstrated in Section 7.

4. Data preparation

Two datasets are prepared offline. Data acquisition, especially the web design repository, needs some page segmentation and vision processing techniques.

4.1. Web design repository and contrast model training set

To establish the *web design repository*, we use a style crawler [28] to collect the top 300 designs on Alexa in 2013 and 200 popular designs from design communities. The crawler uses a layout engine, Webkit, to process retrieved HTML CSS into a DOM tree and render visual appearances. The Text/Block parser module traverses the DOM tree to extract their computed styles and node properties. Assisted with visual segmentation techniques, the

nodes' edge contrast and spatial shape features are extracted. The invisible nodes are discarded.

The DOM nodes with visible color properties are categorized as: text nodes (nodes that have "InnerText" properties) and non-textual nodes of color blocks. The edge contrasts of the texts (WCAG 2.0 legibility) are estimated first. Meanwhile, the text node's other DOM properties, i.e., fontSize, fontWeight, numLinks, numChildren, numImages, numSiblings, siblingOrder, textArea, wordCount, treeLevel, verticalSidedness, are saved together. Next, the remaining color blocks on the pages are segmented (see Fig. 3). The colors and edge contrasts of those blocks are therefore estimated statistically. Their extensive geometrical and spatial features are measured, including: each color block's contour length, relative area, the depth of nesting, bounding box, elongation, invariant moments, Normalized Discrete Compactness (a relationship between the color patch's boundary edges and its area [39]), the area of its background, and their semantic tags of DOM, such as: search, footer, header, image, logo, navigation, and so on.

The boxes/blocks (basic components) are saved in the dataset as elementary entries. Because there are usually many repetitive blocks on a single page, blocks with the same color and shape features are clustered (grouped) into one component. Each basic component therefore has its group properties, such as: the number of siblings, *xy* frequency, *xy* interval, group moment and centroid, density. Each basic component has 46 features.

After parsing the 500 webpages, we obtain a sample of approximately 52 000 components, including 45 300 text nodes and 6800 nodes of color blocks. The parsing process coalesces each component's visual, semantic, and render-time features into
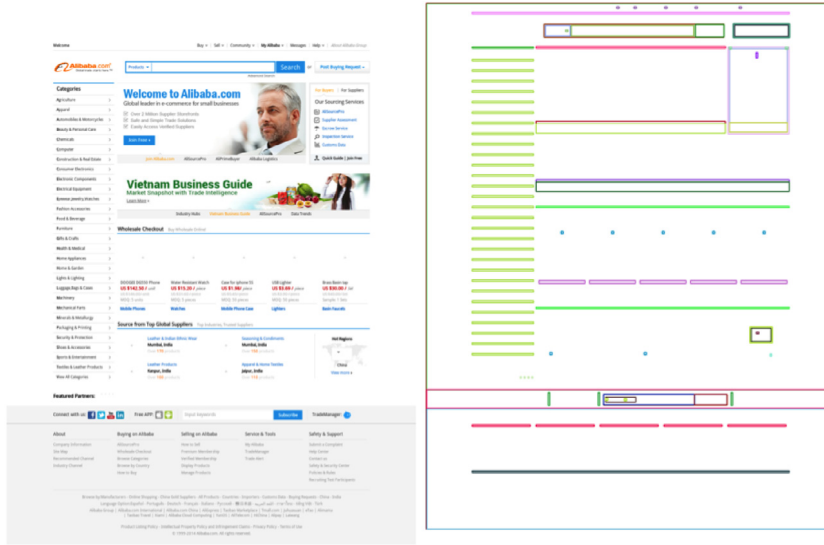
Fig. 3. Color segmentation of webpage. The left is the original webpage. The right is segmented, and grouped blocks are marked with color silhouettes. The text and images have been filtered out in advance.

a normalized vector descriptor consisting of 6 color contrast properties, 27 predictive DOM features of texts, and 46 spatial and shape properties of color blocks, exposing the properties in a convenient form for future model training and feature selection.

The original color properties of RGB are also transformed into the HSV and CIElab space, which are closer to the nature of human perception. Fig. 4 visualizes the distribution of samples of texts and blocks on the plane of edge contrasts (on saturation and value).

4.2. Color scheme database

The Kuler dataset comprises more than 100,000 color schemes created by visitors to the kuler.adobe.com. Most of the color schemes from Kuler consist of RGB raw data and have been tagged with several keywords. This approach allows users to retrieve the wanted color schemes quickly. Color schemes with different semantic tags proved to be distributed in different areas and clusters [37]. Fig. 5 illustrates the distributions of the color schemes with the semantic tags of "business" and "fresh" separately.

5. Objective models

We now describe the methods for formulating the objectives. The models of edge contrasts and color semantics are generally based on the idea of probability density estimation. To evaluate the color harmony, we use the MATLAB implementation provided by ODonovan et al. [8].

5.1. Probability estimation of color contrasts

The Contrast Model is expected to evaluate the edge contrasts of basic components parsed from an input layout. With the training set introduced in Section 4.1, our method is to estimate the probability distribution on the edge contrast of each basic component when given its known features:

$$P(C_{cp}|F_{cp}). \tag{1}$$

For a component cp, C is its edge contrast and F is its known spatial and morphological features. To obtain the distribution, we first discretize the space of color contrast by clustering (k-means) the training sample into a number of groups in terms of their edge contrast values (here, we use the differences on the 3 dimensions of the HSV space with 6 kernels found in the clustering).

Next, we use the clustered sample to train a classifier. The classifier is multinomial logistic regression (MNR). The classifier can predict, when given a Feature vector F_{cp}, the probabilities of it falling into each cluster. Then, we place a Gaussian Kernel at each cluster's center C_j with the corresponding probability as its density and set the widths in terms of the cluster's standard deviations σ. Through the construction of the Gaussian Mixture Model (GMM), we obtain an estimation of $P(C_{cp}|F_{cp})$, which is a continuous probability distribution of variable C_{cp} depending on the feature vector F_{cp}.

$$P(C_{cp}|F_{cp}) = \sum_{j=6} \exp\left(\frac{-\|C_{cp} - C_j\|^2}{2\sigma^2}\right) \cdot P(C_j|F_{cp}). \tag{2}$$

Fig. 6 visualizes the estimated probability distributions of brightness and saturation contrasts for different components on a page layout.

The method was firstly introduced by Charpiat et al. [40], who built multimodal distributions of colors given local texture features for gray scale image colorization. Later, Wang referred to this method in image color theme enhancement [23] and Lin in her generative coloring of patterns [3].

We encode color solution $X = (x_1, x_2, \ldots, x_k)$, $X \in S$. x_k represents the alterable color attribute of the kth color group on a page, and S is the set of all solutions. Suppose there are N basic components parsed from the page, then, its overall contrast settings are then evaluated with a weighted sum of local probabilities:

$$f_c(X) = \sum_{i=1}^{N} P(C_i(x_{fg}, x_{bg}|F_i))A_i,$$

$$x_{fg}, x_{bg} \in (x_1, x_2, \ldots, x_k) \tag{3}$$

where $C_i(x_{fg}, x_{bg})$ is the edge contrast of the ith component with corresponding foreground color group x_{fg} and background color group x_{bg}. F_i is the predictive feature vector of the component's edge. A_i is the area proportion of the ith component. It roughly represents the contribution of visual impact of the component. This equation approximately represents the contribution of visual impact of the component.

The vector F_i has been described in the last section. In the training, it is reduced to 32 dimensions with feature selection techniques from 46 initial dimensions. The Sequential Feature
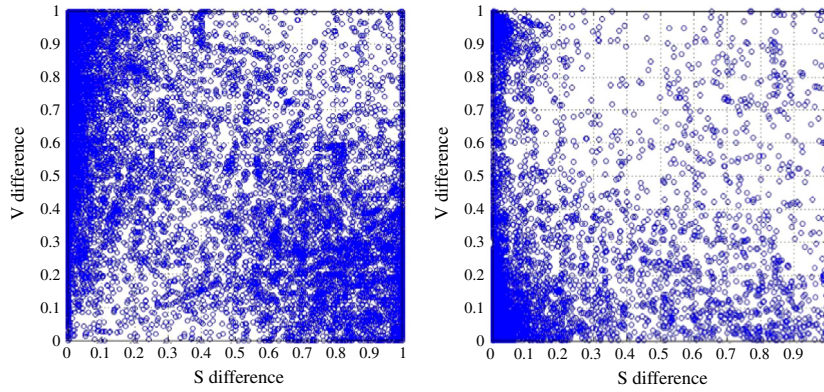
Fig. 4. The left image is the distribution of text nodes on the dimensions of saturation contrast and volume contrast. The right is of color blocks. The images indicate that the samples belong to manifolds or clusters in the feature space rather than random distributions.
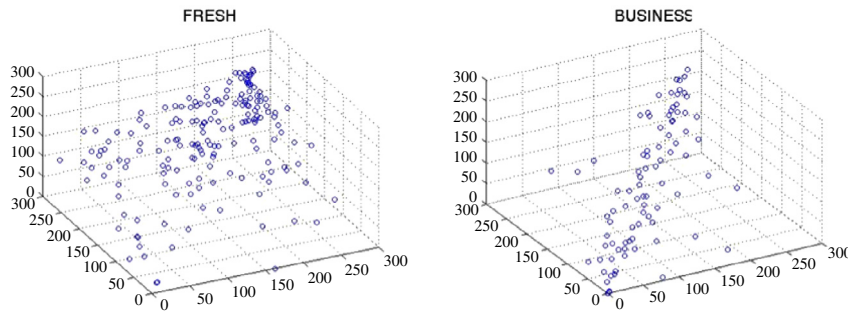

Fig. 5. Color distributions of the color schemes retrieved for different keywords in the RGB space. The member colors of the schemes tagged with "BUSINESS" are more closely and evenly distributed on the diagonal line of gray, while the "FRESH"'s are more scattered and have an apparent convergence point near the white point (255, 255, 255).

Selection technique (SFS) [41] adds features one by one to a sub-feature set while cross-validating and evaluating misclassification rates of the MNR classifier. The decreases of misclassification rates are actually the indexes of information gains of the added features. Those features that have very low contribution to the classification rate (<0.001) are discarded.

5.2. Regression models for color semantics

It is supposed that audiences' subjective evaluations of a design X regarding a specific mood being aroused is a normal distribution and could be estimated by

$$f_s(X) = E(y|X) = \int yP(y|X)dy \tag{4}$$

where y is the rating of the mood strength. According to Bayes' theorem

$$f_s(X) = \int y \frac{P(X,y)}{P(X)} dy = \frac{\int yP(X,y)dy}{\int P(X,y)dy}. \tag{5}$$

With a set of sample (y_i, X_i), $i \in \{1, 2 \ldots, n\}$, collected for a specified mood, the $P(X, y)$ and $P(X)$ can be approximated using GMM with n kernels.

$$P(X,y) = \frac{1}{(2\pi)^{(k+1)/2\sigma^{(k+1)}}} \cdot \frac{1}{n}$$

$$\times \sum_{i=1}^{n} \exp\left(\frac{-\|X - X_i\|^2}{2\sigma^2}\right) \cdot \exp\left(\frac{-\|y - y_i\|^2}{2\sigma^2}\right) \tag{6}$$

where X_i is the ith design (or ith cluster mean) in the sample set. y_i is the rating of the mood given by people. k is the dimension of X. σ is the diameter of the Gaussian kernels. After combining (5) and (6), then performing integrations we get:

$$f_s(X) = \frac{\sum\limits_{i=1}^{n} y_i \exp(-\|X - X_i\|^2/2\sigma^2)}{\sum\limits_{i=1}^{n} \exp(-\|X - X_i\|^2/2\sigma^2)}. \tag{7}$$

It reveals that for any new design X input, $f_s(X)$ can be visualized as a weighted mean of y_i. The weights are the exponential function of the L_2 distance between X and X_i.

It is impractical to manually rate each X_i with a scale value y_i in terms of the specified mood. The training data available is a large set of color schemes tagged with all kinds of keywords in the Adobe Kuler database. Users can specify an expected color mood with one or several weighted keywords, for instance, "energetic and young but not so hot". Then, as an approximation, the color schemes retrieved with keywords "energetic" and "young" are rated +1. And the "hot" schemes are rated 0. Some randomly sampled color schemes from the rest are rated 0.5. Usually there are thousands and hundreds color schemes for each keyword in Adobe dataset. The reason to use this probability density model is that its non-parametric nature allows users to avoid re-training for any new combination/weighting of mood keywords.

Any solution X needs to be transformed into a k-dimension semantic feature vector before evaluation. We initially selected total 82 high-level features: in addition to raw features of five member colors' RGB and HSV values, many statistical global features are calculated, including mean, standard deviation, max
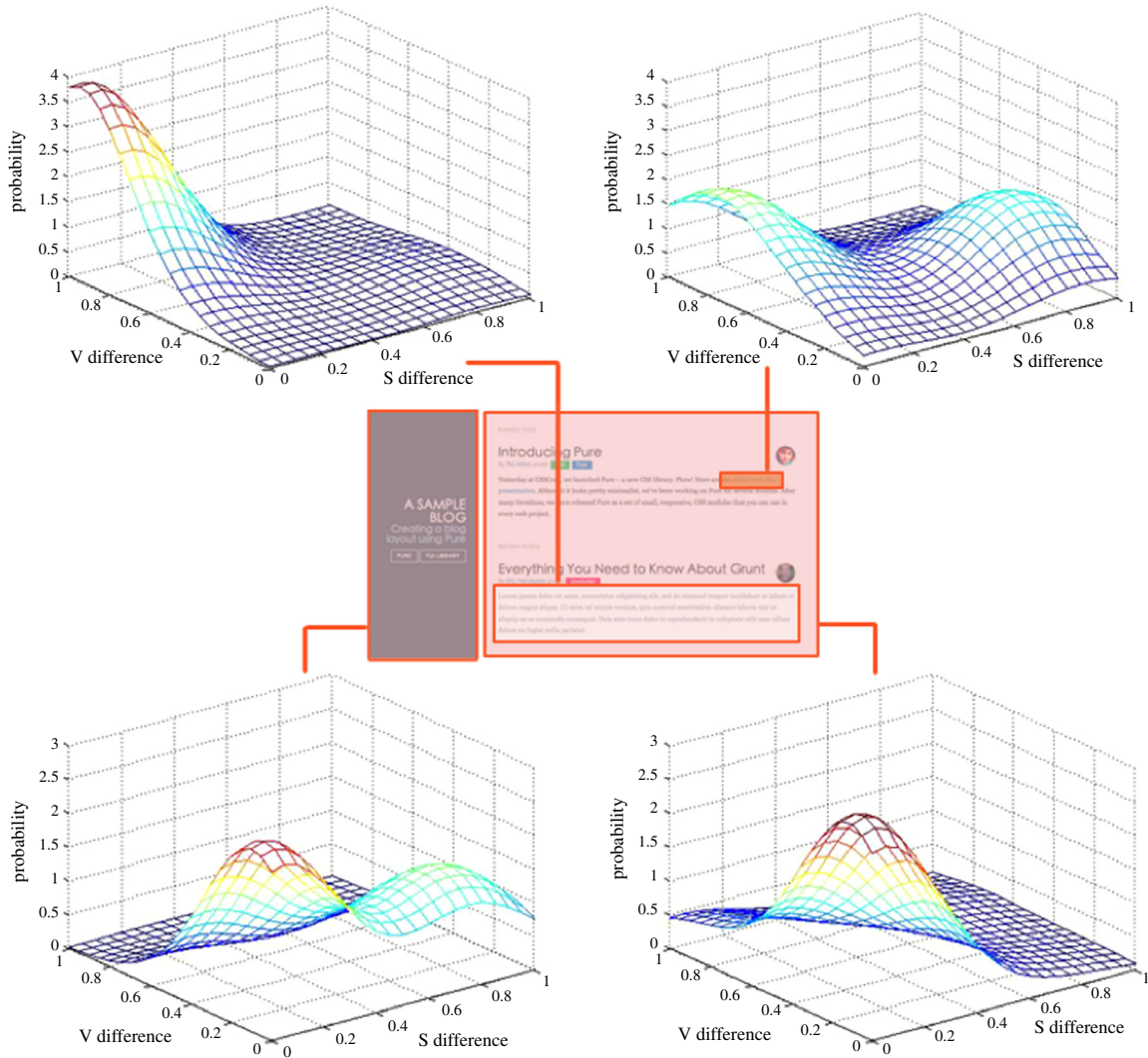
Fig. 6. The model estimates the probability distributions of edge contrasts (here, only value and saturation differences are visualized) of different color regions (connected with redlines) on the webpage. The top left is the distribution of the contrast of the body text. The top right is the link. The bottom left is the sidebar. The bottom right is the background. We can see that it is highly possible for the body text on the page to use high value difference and low saturation contrast, while the background prefers both low value and saturation contrasts in this context.

and min of each channel of HSV, RGB, Lab, and edge contrasts of the channels; dominant color (the member color closest to the mean); accent color (the member color that has the largest difference from the dominant color); bag of visual word (BOV) and Fisher vectors based on the Kuler dataset, and hue entropy [8]. O'Donovan has investigated most of these features in his research of color scheme compatibility with weight learning. We expect them to also be effective for color semantics. Csurka et al., through the error rates of cross validations of classifications, proved that the BOV and Fisher Vector are more discriminative than the raw RGB feature in semantic categories [37].

With the semantic features, the model could capture a subtle mixed mood expressed with any combination of multiple keywords, which may not be available by direct keyword searching in the dataset. For instance, the dataset has 156 "soft + fresh" and 1582 "Business" but returned nothing for "business + soft + fresh". Utilizing the model, users could infer a mood specified with multi-words and induce the most representative sample in a set. An experiment on the Kuler set illustrates this point (Fig. 7). The model may also allow the users to filter out an original, vast color palette into a smaller amount of colors.

6. Lexicographic strategy of optimization

Color design is used to find the optimum permutation X by maximizing the contrast, semantic and harmony objectives:

$$\max \ (f_c(X), f_s(X), f_h(X)), \quad \text{s.t.} \ g(X) \leq 0, \ X \in S \quad (8)$$

$g(X)$ is the constraint, such as the W3C web standard of legibility.

The lexicographic strategy progressively reduces the number of candidate solutions by applying selection criteria in sequence. The optimization starts from seeking one of the objectives:

$$\max f_{1st}(X), \quad X \in S.$$

We next obtain the optimum f_{1st}^* and a reduced solution set S_{1st}

$$S_{1st} = \left[X | f_{1st}(X) \geq \left(1 - \frac{\delta_{1st}}{100} \right) f_{1st}^* \right] \cap S, \quad \delta \in [0, 100]. \quad (9)$$

Next, we continue to reduce the set S by conducting such an operation using other objective functions until the final optimal solution set in a lexicographic sense. δ represents the relaxation factor, adjusting the proportion of surviving solutions in each stage. The method takes effect on the condition that the reduced S must have more than one element every time. Color design usually has
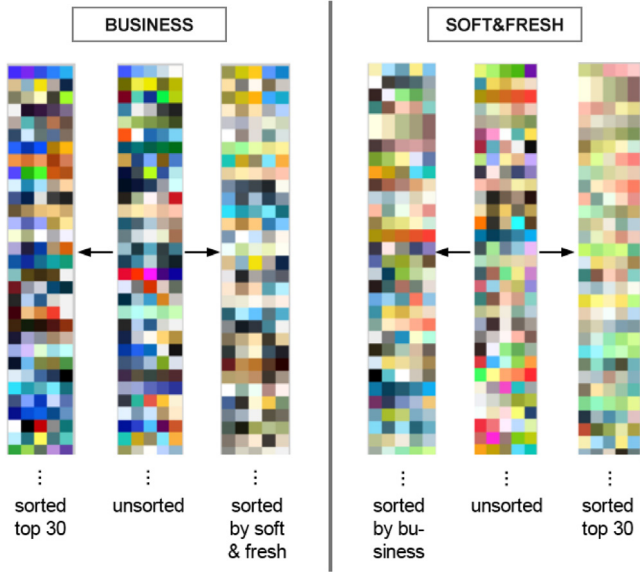
Fig. 7. The semantic model can capture subtle changes of color tones. The left 3 columns are color themes tagged with "Business" in the dataset. The right 3 are tagged with "Fresh + Soft". The first column from the left is the sorted top 30 "Business" out from the second column, which is unsorted 1582 themes of "Business". The third column is the "Business" sorted by "Fresh + Soft". The right 3 columns are treated the same as the left symmetrically. In this experiment, we use 46 semantic features and 50 Gaussian kernels with = 0.2, and the MSE of the approximation is less than 0.09.

Fig. 8. A palette with 11 colors.

quite a number of optimum solutions rather than an absolutely unique one. It need not be the strict optimal before starting the next stage of operation.

7. Applications

We now demonstrate how the models and strategy may be useful when applied to two typical web-coloring scenarios. One is coloring a webpage with a prepared color palette. Another is fine-tuning a colored webpage within specified ranges of changes. Both the situations take the webpages' layouts as inputs.

Because the areas of the components and color groups are fixed during the optimization, the spatial and morphological features of the components and color groups can be pre-calculated in a single run offline. Therefore, the objective functions only need small fractions of recalculation for every update of the coloring on the page.

7.1. Coloring a page with a prepared palette

From color design communities, such as Adobe Kuler, Colorlover, Aviary Toucan and Daily Color Scheme, designers can search for wanted color schemes (color palette). Color schemes can be pre-evaluated by using the harmony model and semantic model. We have demonstrated the semantic fine-tuning in Fig. 7. Well-selected color schemes, however, do not assure an expected final coloring effect on page. Transferring colors from a color scheme to different regions of a webpage usually produces significantly different visual effects.

In this demonstration, we selected a blog page as the target layout (http://purecss.io/layouts/blog/). The webpage has seven

fixed regions of color groups. Thus, we retrieved two color schemes from Kuler, plus an extra neutral gray. An 11-color palette is created (see Fig. 8)

There is a total of $11!/4! = 1,663,200$ possible alternatives. We use a brute force search to evaluate all 1,663,200 alternatives. A depth first searching tree in terms of the states of the seven color groups is constructed. The color groups containing texts are placed closer to the root node so that we can efficiently prune the branches of illegible solutions (violating the WCAG 2.0 luminance standard) at the very beginning. Those that have a text luminance below 2.2 are killed without further evaluations. Approximately 200,000 solutions remained after the pruning. The solutions are then further evaluated and sorted with a sequence of objective models.

To evaluate the solutions with the contrast model introduced in Section 5.1, basic components of the page must be parsed at the beginning. There is a total of 15 text/block components on the page, with their weights and predictive features being measured. The overall score is the weighted sum of the evaluations on the components (see Eq. (3)). Fig. 9 shows results with the first and third scores of the overall contrast evaluations.

The top 10 results have apparent diversity (see Fig. 10). This verifies that the contrast qualities alone may be too general to converge the results, even with a small palette. More objectives, such as semantic preferences, may be necessary.

If we apply the semantic models at first and sort out a portion of the total 200,000 for further contrast evaluation (or vice versa), the final results can be significantly different. Fig. 11 shows the best contrast results with two different prerequisites of keywords. Fig. 12 shows that with the prerequisites, the top contrast results converge much more than those in Fig. 10.

In every lexicographic stage, a sample of sorted high scored results must be saved. The users can browse through it to obtain the sense of how diverse the results are so far and then decide on the portion reserved for the next stage of screening.

In another case, the layout is designed with a template (http://demo.mekshq.com/?theme=seashell). It has 7 color groups. We create a palette by acquiring colors from the theme image on the target page. This is a popular practice for designers to achieve harmony and semantic consistency on webpages [36]. Fig. 13 shows a palette that has 10 colors from van Gogh's Wheatfield with Crows.

The total number of solutions is 604,800, in which only 1442 meet the WCAG legibility standard because most colors on the palette are neutral and dim. We sort the 1442 solutions with a semantic objective before applying the contrast models to them. Fig. 14 shows the results of two different semantic objectives: the left is the best contrast setting among the top 200 "warm soft". The right is obtained from the top 200 "cool business".

An essential detail of lexicographic optimization is to set a proper tolerance δ. This refers to how to determine the surviving portion of the last screening to enter the next stage. There is no mature guideline to follow to date. Generally, we must keep sufficient variety in the retained set to leave room for the next stage of screening. In this paper, we interactively expand the retained set to alter the best ones until we obtain sufficient variety.

In the experiments, we found that a harmonious color scheme generally ensures harmonious page coloring effects. However, a larger palette may result in potential incompatible combinations. In the next trial, we use a 20-color palette (see Fig. 15). The target page is still the same as the last. There is a total of 390,700,800 possible candidate solutions. 706,800 remained after legibility checking. Then, the further screenings have three layers. Harmonization is inserted in between semantic and contrast
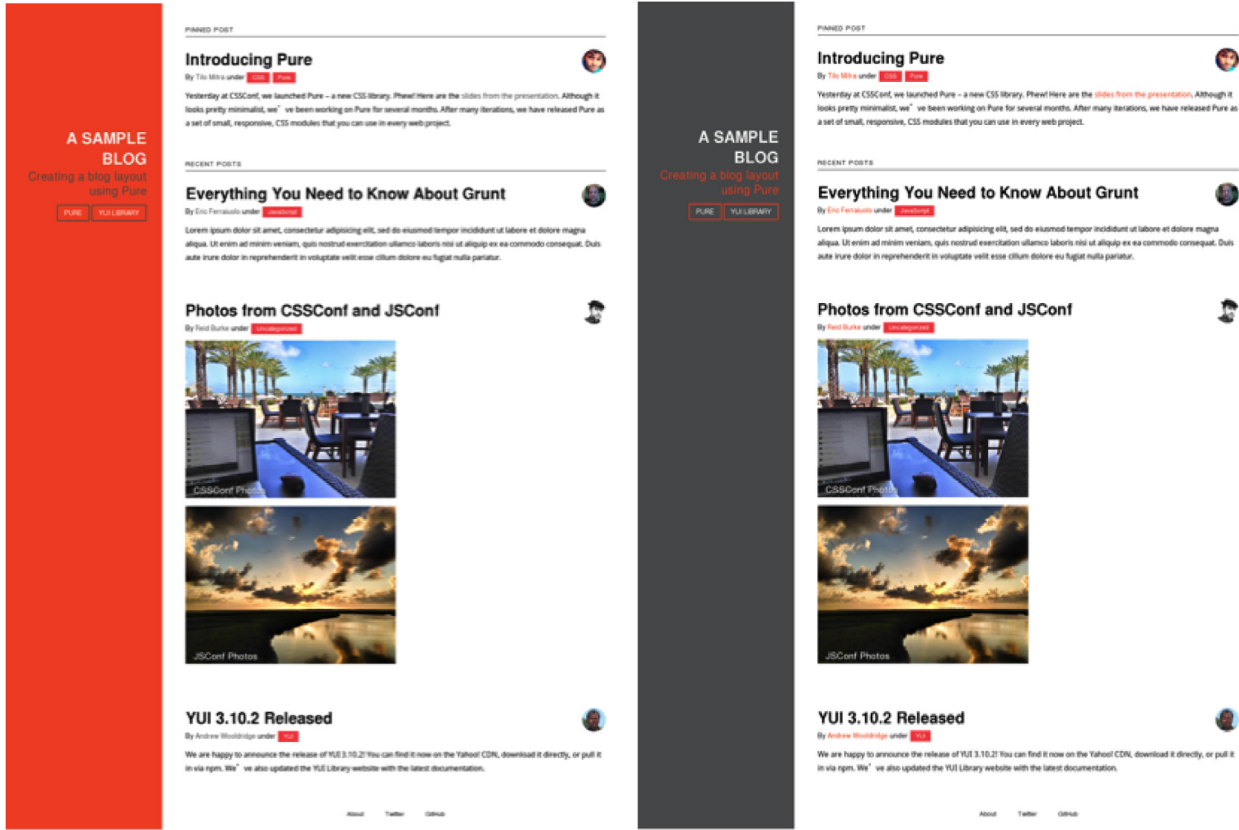
Fig. 9. The 1st and 3rd score coloring effects after the contrast evaluation.
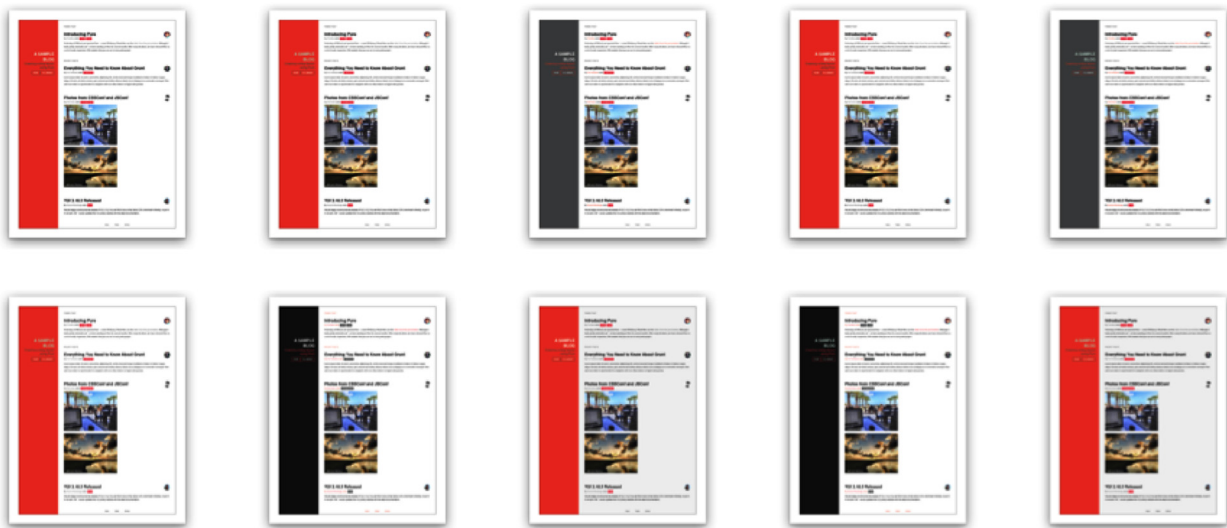

Fig. 10. The top 10 in 200,000 after the contrast evaluation. The results shown are somewhat diverse. This means that the contrast model alone may not be able to converge the final solutions. Other objectives, such as semantic preferences, may be necessary.

evaluations, considering contrast evaluations are computationally heaviest among the three. Fig. 16 demonstrates the best ones at each of the three consecutive evaluations from left to right with notable improvements.

In the demos, the contrast model generally performs well. However, it is not perfect, as some types of components, such as the div of the headline, text input box and navigation bars, seem to be mixed up in the predictions. This may be because they share too many similar features. It would be better to add more discriminative tags of their natures.

7.2. Fine-tuning a color design

Some webpage makers like to start from a roughly specified color tone on the page following tiresome adjusting processes. The data-driven models may assist them in improving their initial colorings with slight modifications. That is, given a current page coloring X, the system automatically finds better-looking alternatives X' nearby in the space of solutions.

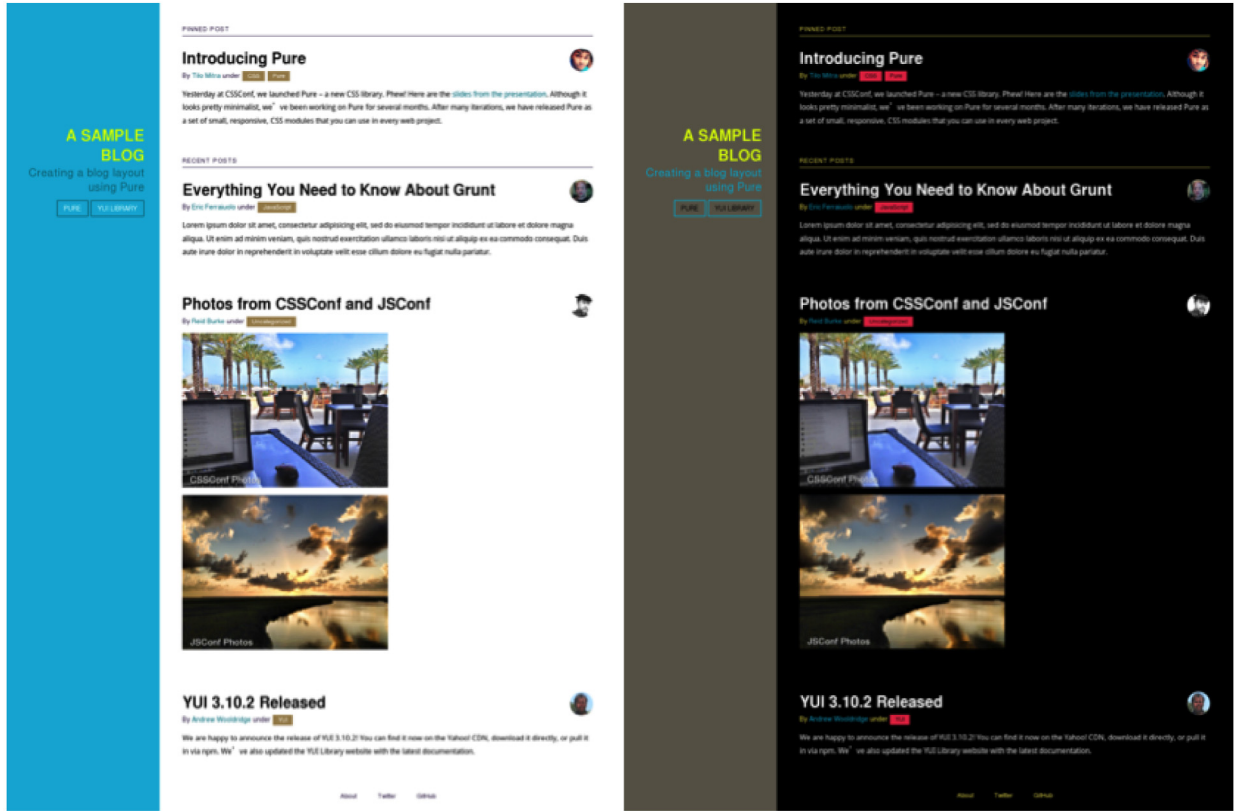$$Max f(X'), \quad \text{s.t. } \|X' - X\| \leq D. \tag{10}$$

Fig. 11. We firstly use keywords to specify two different semantic objectives and sort out the top 500 for each. Then, within them, we search for the best contrast settings. The left is the best contrast result among the top 500 of "fresh", while the right is that for "business".
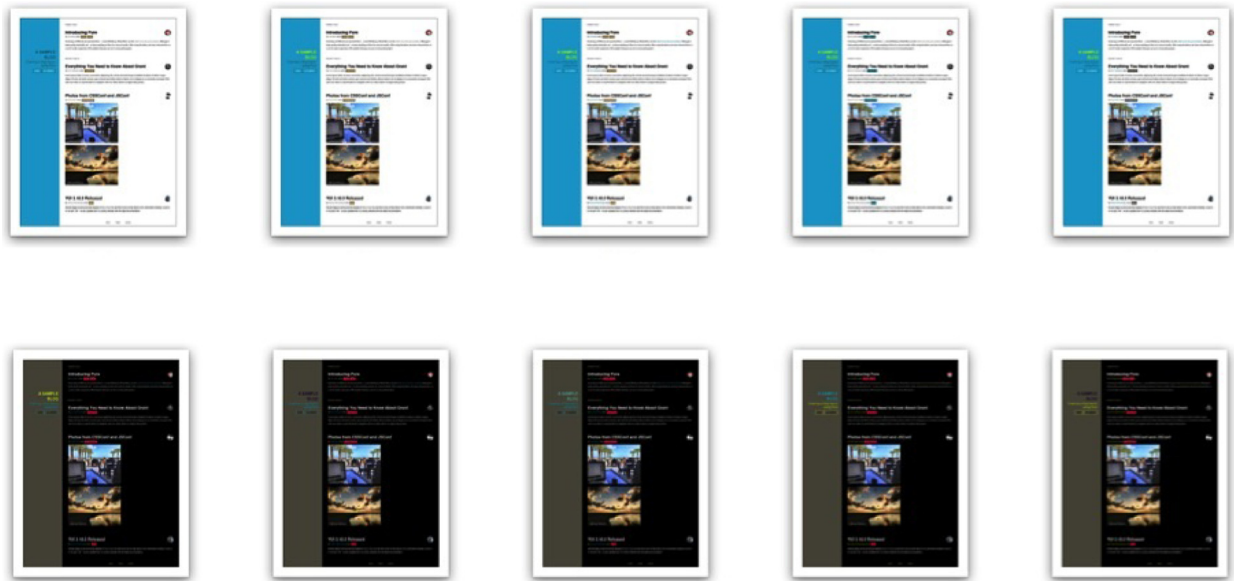

Fig. 12. The top row shows the top 5 contrasts among the top 500 "fresh". The bottom row shows those for "business". Compared to Fig. 10, the results here are much more converged.

The L_2 distance D is computed in HSV space, constraining the variations of X. D can be set individually for each color region and each color channel. Here, we set D on the HSV dimensions separately with $D_h = 0.01$, D_s and $D_v = 0.1$, because people are more sensitive to the change of hue than the other two channels.

Firstly, we demonstrate "finding a better direction of the next move". We enumerate all possible slight moves away from the original. Four foreground color groups on the page are set as

Fig. 13. A color palette created by quantizing a theme image.

variables. Suppose each channel of a color has three states: move upward, move downward and no move; then, there is a total of $3^{12} - 1 = 531{,}441$ combinations (directions) around the original
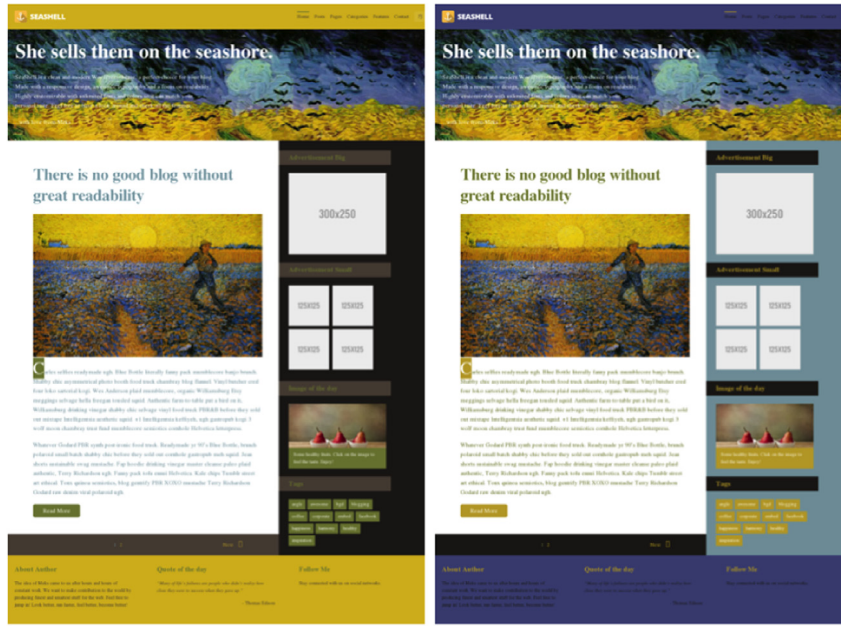
Fig. 14. The best contrast settings in prerequisites of two semantic objectives separately. The left is "warm soft" and the right is "cool business".

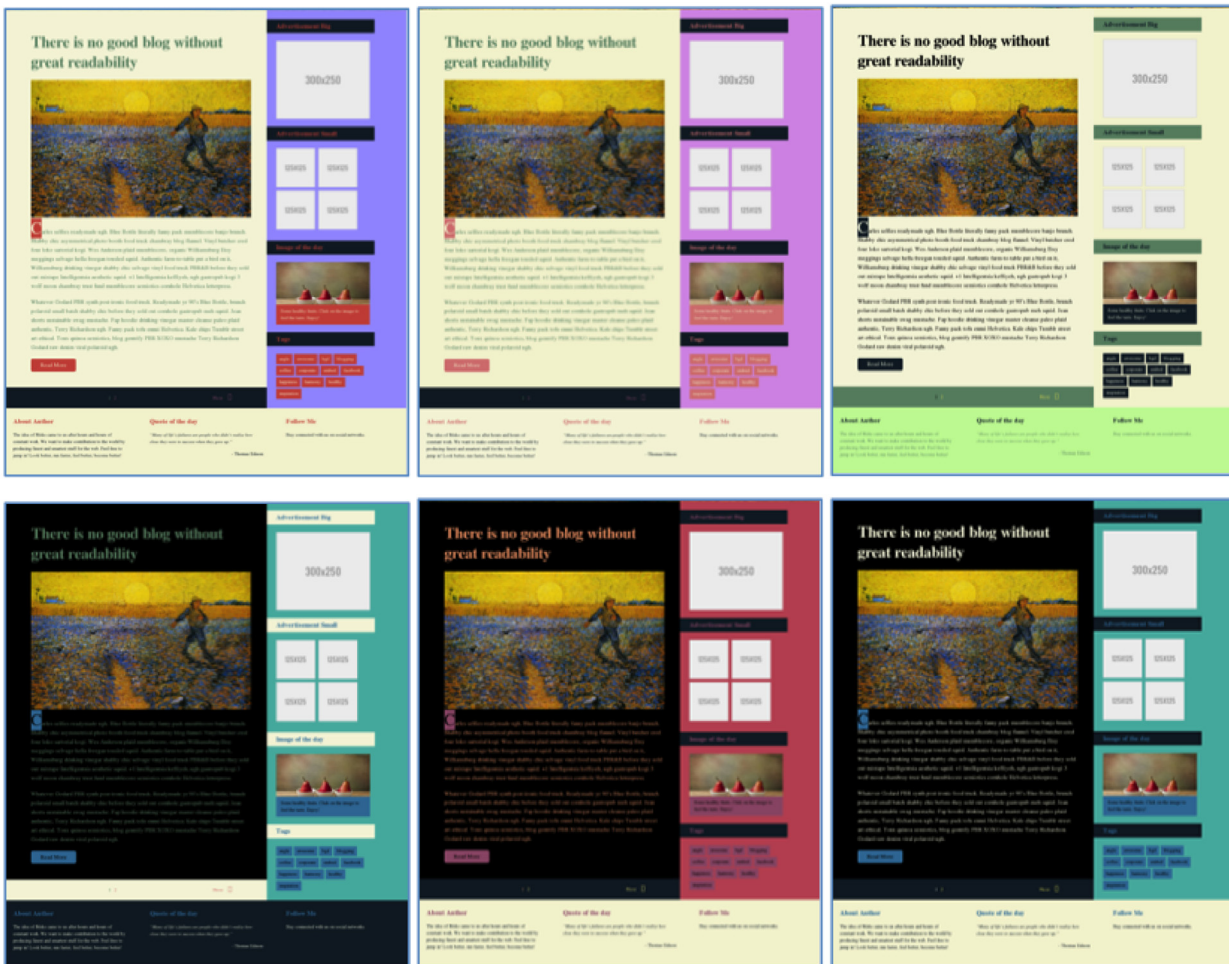**Fig. 15.** Four schemes retrieved from Kuler with keywords "fresh", "soft" and "business".

Fig. 16. The optimizations under two semantic preferences. The top row is "fresh" and the bottom row is "business". From left to right are the stage results produced by the three consecutive evaluations. The left column shows the best of the semantics in 706 800 solutions. The middle shows the most harmonious ones in the top 18 000 of semantics. The right shows the best contrast in the top 2000 of harmony. Improvement is notable as more evaluations added.
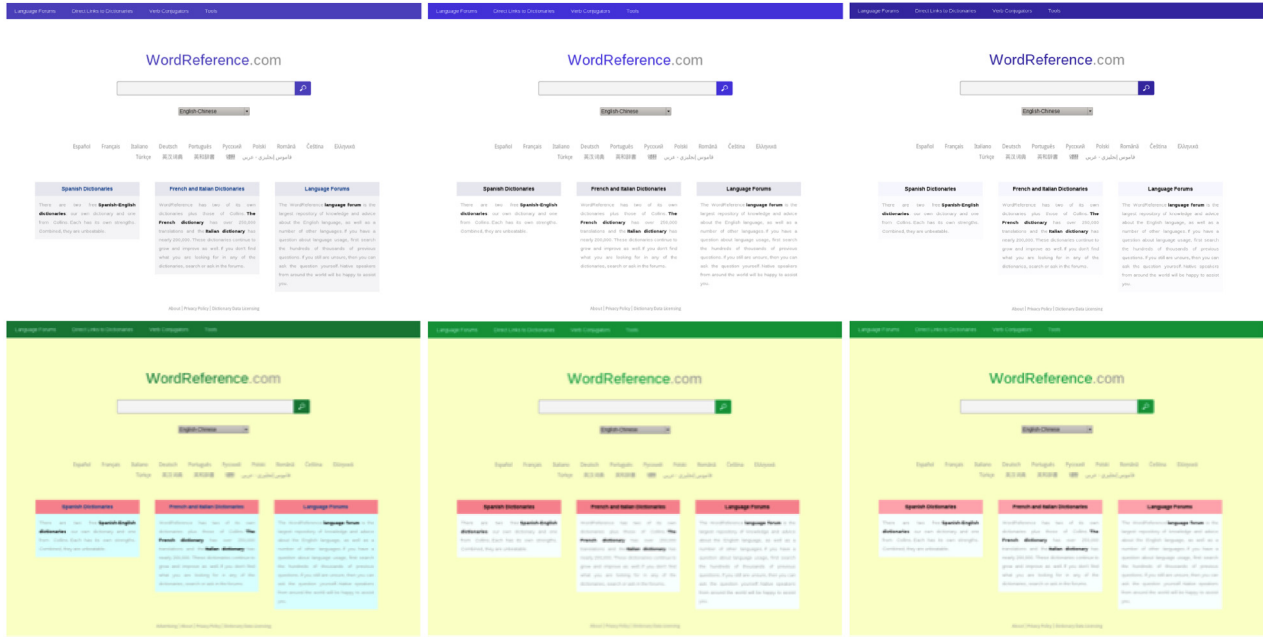
Fig. 17. Top left: the original colored page of http://www.wordreference.com. Top middle: the best contrast adaptation. There are 3042 adaptations with contrast evaluations better than the original. The top right is evaluated as the best harmonization in the set of 3042 with a contrast rank at 1423, while the original's harmony in the set ranks at 1896. The color of the text boxes becomes paler due to the texts on them being a fixed gray. Bottom row: the left is the original, designed by a student. The middle is the best contrast adaptation. There are 157 045 with contrast scores better than the original. The set is further sorted with the harmony model to obtain the final result. The bottom right is the most harmonious one in the set, with a contrast at 4537, while the harmony rank of the original in the set is 13 212.

coloring state. We try to find a direction better than the original from both contrast and harmony points of views (see Fig. 17).

Another demo uses Covariance Matrix Adaptation (CMA) [42], an evolutionary approach, to fine tune the page, with the constraints being enforced by assigning large penalties to generated alternatives that violate the constraints. CMA is run for 50 iterations to converge with a sample size $N = 30$.

The CMA is run twice. In the first run, the original page is adapted to an optimum contrast state. In the second run, the optimum contrast ratios on the page are fixed as constraints. The harmony model leads the adaptation. Fig. 18 shows two original pages and their subsequent consecutive adaptations.

8. User perception studies

We evaluated the effectiveness of the approach through user perception studies. Twenty undergraduates from various departments (15 males and 5 females) were recruited to participate in the study. All participants had normal color vision.

In the first test, 30 coloring solutions of the same input layout using the same color palette were prepared as the test material, in which 10 were made by nonprofessional university students, 10 generated by the system randomly, and 10 generated by the system with optimization to two objectives (contrast and harmony).

The participants were shown each of the 30 effects for 5 s on an Apple 27 in cinema display in randomized order with a 5 s black screen interval and were asked to indicate their attitude to each coloring effect: like, dislike or no opinion. 8 effects were presented once again to check the consistency of the participants' attitudes.

Fig. 19 shows that human design and our model received more "like"s and less "dislike"s than the random base-line. Pairwise T test shows that the difference between the generated colorings with the model and without the model is significant (like: $t = 8.986$, dislike: $t = -15.449$, with $p < 0.001$). The model-generated colorings received more "like"s than those generated by non-professional students ($t = 2.236$, $p < 0.05$), while the "dislike"s they received have no difference (-0.237, $P = 0.815$).

We also conducted an A–B test on the effectiveness of the fine-tuning. 10 pairs of test material were prepared. 20 participants were asked to compare each original webpage coloring with its tuned version on a screen at the same time in a five second interval and to select either their preferable one, both or none. The analysis of the results shows that the tuned versions received more preferences (see Fig. 20). We also found that those that originally had less "like"s were improved more significantly.

To evaluate the predictability of the semantic model, a user test presented participants with a randomized grid of 16 colorings generated for the same webpage. Participants were asked to rate the colorings (-1, 0, $+1$) in terms of given keywords. The test uses 3 unary keywords (fresh, soft, business) and 3 compound keywords (fresh + business, soft + business, soft + fresh). We compared the human ratings on the 16 colorings with the model evaluations. The Pearson correlation between the human ratings and model predictions is 0.826 for the unary and 0.653 for the compound keywords with $p < 0.05$.

9. Conclusions and future research

In this paper, we present a data-driven design framework for automatically coloring webpages. Three fundamental objectives are addressed in the optimization. The contrast and semantic models are newly introduced in webpage design and formulated with probabilistic density estimations in the established feature spaces with the design samples we prepared. The framework supports an interactive lexicographic strategy to coordinate the multi-objectives. The demonstrations of the initial implementation of the framework in two typical design scenarios are encouraging. The user perception experiments suggest the effectiveness of the models. The system-generated designs are generally preferable to those designs by nonprofessionals.

The internal mechanism of color aesthetics perception is still not very clear. For instance, the colors on a page interact with each other. The areas, shapes, distances and so on all mutually affect the effects of colors in a composition. Thus, the evaluation
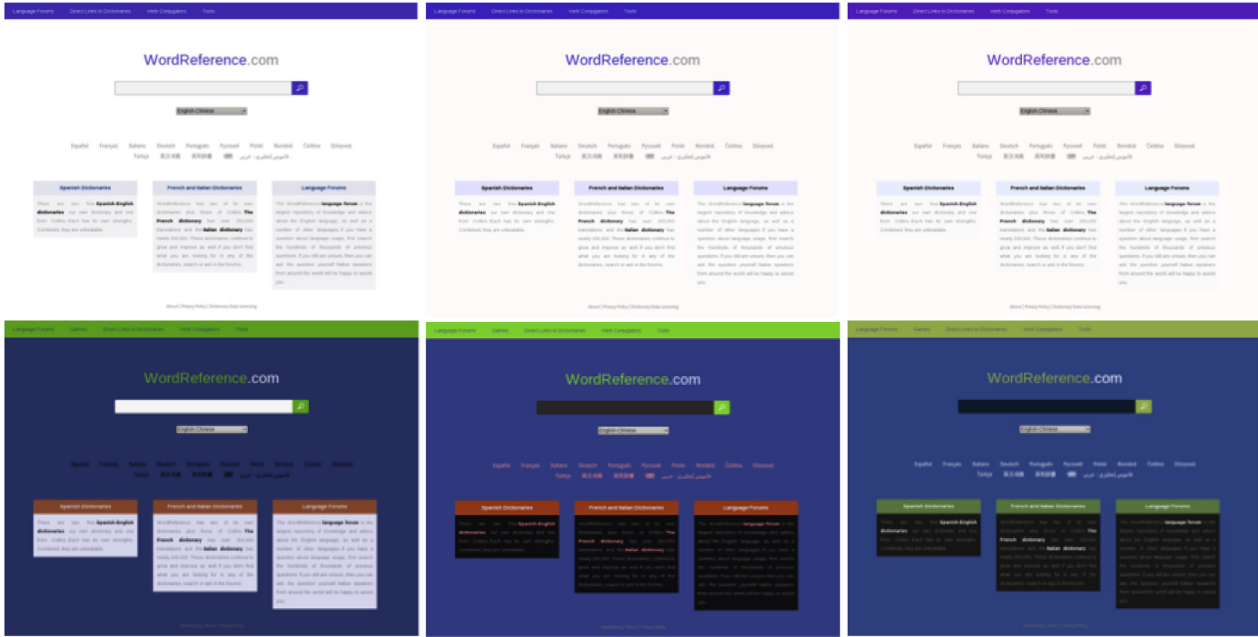
Fig. 18. Two-stage CMA fine-tuning. Top left is the original colored page of http://www.wordreference.com, the same as in Fig. 17. Bottom left is a randomly colored page. In CMA, smaller color blocks are set with larger D values, and the background is also variable. The middle two are the optimized contrasts. The right two are results of further harmonization, while the contrast ratios are fixed as constraints.
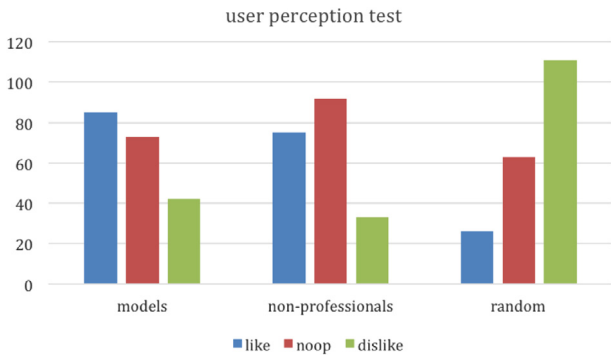

Fig. 19. Percentage of "like" and "dislike" on three sets of coloring effects. The left refers to those generated and optimized with our model. The middle refers to those generated by human design. The right refers to those generated without optimization.
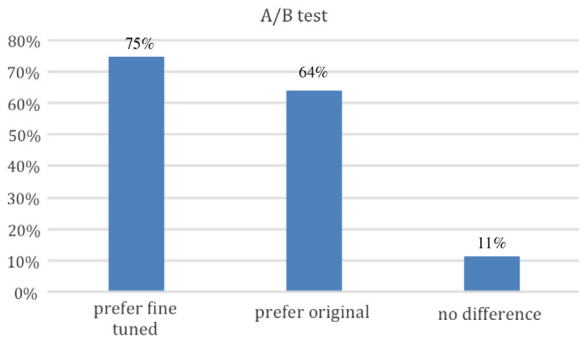

Fig. 20. The refined versions received more preferences than the originals.

of page harmony using five linearly arranged color swatches might be oversimplified.

There remains considerable work to be performed to identify which color properties matter most to the semantic expression needs of a systematic features selection process. For a better contrast prediction, the contrast model needs to explore more discriminative features of the webpage components, try different granularities in the contrast space, and enlarge the design repository with more examples.

For the sake of the simplification of computing, presently, in design generation, the number of color groups and their zoning are not adjustable. If so, the total number of alternatives may increase exponentially, making brute force searching certainly not a reasonable option. More efficient searching, such as Stochastic, iterative and evolutionary techniques, genetic algorithms and the ant colony algorithm may be more suitable.

To date, the objective model has been implemented using MATLAB and the generative engine using Python and JavaScript. The demonstrations of color assignments in this paper normally require 30 min ~6 h on a 1.7 GHz Intel core i5, 8 GB DDR to complete these optimizations (it generally depends on the sizes of solution sets) and the color fine tunings with CMA require about 2 h to converge. The performance of the system can still be greatly improved.

This research presents a new vision for future adaptive webpage coloring. A webpage may quietly harmonize itself, adjust its visual saliencies, and change its color mood based on its contents, different audiences or different seasons.

Acknowledgments

References

[1] Peters G. Aesthetic primitives of images for visualization. In: Information visualization, 11th international conference. IEEE; 2007. p. 316–25.

[2] Lindgaard G, Fernandes G, Dudek C, Brown J. Attention web designers: You have 50 milliseconds to make a good first impression! Behav Inf Technol 2006; 25(2):115–26.

[3] Lin S, Ritchie D, Fisher M, Hanrahan P. Probabilistic color-by-numbers: Suggesting pattern colorizations using factor graphs. ACM Trans Graph (TOG) 2013;32(4):37.

[4] Hearn D, Baker MP. Computer graphics, C version, Vol. 2. Upper Saddle River: Prentice Hall; 1997.

[5] Powell TA. Web design: the complete reference. McGraw-Hill, Inc.; 2001.

[6] Itti L, Koch C, Niebur E. A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Pattern Anal Mach Intell 1998;20(11):1254–9.

[7] Itten J. The elements of color. Vol. 4. John Wiley &; Sons; 1970.

[8] O'Donovan P, Agarwala A, Hertzmann A. Color compatibility from large datasets. ACM Trans Graph (TOG) 2011;30:63.

[9] Rentmeesters MJ, Tsai WK, Lin K-J. A theory of lexicographic multi-criteria optimization. In: Second IEEE international conference on engineering of complex computer systems, Proceedings. IEEE; 1996. p. 76–9.

[10] Holtzschue L. Understanding color: an introduction for designers. John Wiley & Sons; 2011.

[11] Munsell AH, Birren F. A grammar of color. Van Nostrand Reinhold; 1969.

[12] Ostwald W. The color primer: a basic treatise on the color system of Wilhelm Ostwald. Van Nostrand Reinhold Co.; 1969.

[13] Matsuda Y. Color design. Tokyo, Japan: Asakura Shoten; 1995.

[14] Wang L, Giesen J, McDonnell KT, Zolliker P, Mueller K. Color design for illustrative visualization. IEEE Trans Vis Comput Graphics 2008;14(6):1739–54.

[15] Jahanian A, Liu J, Lin Q, Tretter DR, O'Brien-Strain E, Lee S, Lyons N, Allebach JP. Automatic design of colors for magazine covers. In: Proc. IS & T/SPIE electronic imaging. International Society for Optics and Photonics; 2013.

[16] Tokumaru M, Muranaka N, Imanishi S. Color design support system considering color harmony. In: FUZZ-IEEE'02. Proceedings of the 2002 IEEE international conference on fuzzy systems, Vol. 1. IEEE; 2002. p. 378–83.

[17] Cohen-Or D, Sorkine O, Gal R, Leyvand T, Xu Y-Q. Color harmonization. ACM Trans Graph (TOG) 2006;25(3):624–30.

[18] Meier BJ, Spalter AM, Karelitz DB. Interactive color palette tools. IEEE Comput Graph Appl 2004;24(3):64–72.

[19] Sauvaget C, Boyer V. Harmonic colorization using proportion contrast. In: Proceedings of the 7th international conference on computer graphics, virtual reality, visualisation and interaction in Africa. ACM; 2010. p. 63–9.

[20] Ou L-C, Luo MR. A colour harmony model for two-colour combinations. Color Res Appl 2006;31(3):191–204.

[21] Yu L-F, Yeung SK, Terzopoulos D, Chan TF. Dressup!: outfit synthesis through automatic optimization. ACM Trans Graph (TOG) 2012;31(6):134.

[22] Zhang Q, Zhao C, Gu Z, Ming X. A color harmony measure model with shape information. In: International joint conference on computational sciences and optimization. CSO, Vol. 1. IEEE; 2009. p. 627–30.

[23] Wang B, Yu Y, Wong T-T, Chen C, Xu Y-Q. Data-driven image color theme enhancement. ACM Trans Graph (TOG) 2010;29(6):146.

[24] Consortium WWW. et al. Web content accessibility guidelines (wcag) 2.0.

[25] Cheng M-M, Zhang G-X, Mitra NJ, Huang X, Hu S-M. Global contrast based salient region detection. In: 2011 IEEE conference on computer vision and pattern recognition (CVPR). IEEE; 2011. p. 409–16.

[26] Jänicke H, Chen M. A salience-based quality metric for visualization. In: Computer graphics forum. Vol. 29. Wiley Online Library; 2010. p. 1183–92.

[27] Brown RO, MacLeod DI. Color appearance depends on the variance of surround colors. Curr Biol 1997;7(11):844–9.

[28] Kumar R, Satyanarayan A, Torres C, Lim M, Ahmad S, Klemmer SR, Talton JO. Webzeitgeist: Design mining the web. In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM; 2013. p. 3083–92.

[29] Lin S, Fortuna J, Kulkarni C, Stone M, Heer J. Selecting semantically-resonant colors for data visualization. In: Computer graphics forum. Vol. 32. Wiley Online Library; 2013. p. 401–10.

[30] Hsu SH, Chuang MC, Chang CC. A semantic differential study of designers' and users' product form perception. Int J Ind Ergon 2000;25(4):375–91.

[31] Ou L-C, Luo MR, Woodcock A, Wright A. A study of colour emotion and colour preference. Part I: Colour emotions for single colours. Color Res Appl 2004;29(3):232–40.

[32] Ou L-C, Luo MR, Woodcock A, Wright A. A study of colour emotion and colour preference. Part II: Colour emotions for two-colour combinations. Color Res Appl 2004;29(4):292–8.

[33] Kobayashi S, Matsunaga L. Color image scale. Kodansha International Tokyo; 1991.

[34] Eisemann L. Pantone's guide to communicating with color. Grafix Press, Ltd.; 2000.

[35] Morse BS, Thornton D, Xia Q, Uibel J. Image-based color schemes. In: IEEE international conference on image processing. ICIP, Vol. III. IEEE; 2007. p. 497–505.

[36] Gu Z, Wu Z, Yu J, Lou J. A color schemer for webpage design using interactive mood board. In: Human–computer interaction. Human-centred design approaches, methods, tools, and environments. Springer; 2013. p. 555–64.

[37] Csurka G, Skaff S, Marchesotti L, Saunders C. Learning moods and emotions from color combinations. In: Proceedings of the seventh Indian conference on computer vision, graphics and image processing. ACM; 2010. p. 298–305.

[38] Marler RT, Arora JS. Survey of multi-objective optimization methods for engineering. Struct Multidiscip Optim 2004;26(6):369–95.

[39] Bribiesca E. Measuring 2-d shape compactness using the contact perimeter. Comput Math Appl 1997;33(11):1–9.

[40] Charpiat G, Hofmann M, Schölkopf B. Automatic image colorization via multi-modal predictions. In: Computer Vision—ECCV. Springer; 2008. p. 126–39.

[41] Peng H, Long F, Ding C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell 2005;27(8):1226–38.

[42] Hansen N. The cma evolution strategy: a comparing review. In: Towards a new evolutionary computation. Springer; 2006. p. 75–102.