Selection of High Contrast Color Sets

P. Campadelli and R. Schettini*
**Dipartimento di Scienze dell'Informazione*
Universitá degli Studi di Milano, via Comelico, Milano, Italy
ITIM, CNR, Via Ampere, Milano, Italy

Abstract

We present an algorithm for the selection of high contrast color sets which performs very well and does not require that the function adopted to code the similarity between two colors is a distance function.

Introduction

The visual identification of different information items in graphical environments can be achieved with the help of colors. However, the number of items to be coded in a graphic environment often exceeds the number of easily discriminable colors. In these cases color coding may become an inefficient, time-consuming activity.[1,2] Several heuristic procedures have been proposed to define high contrast sets of colors feasible for nominal color coding. Kelly[3] has proposed a list of 22 maximally contrasting surface colors, such that each color of the list is maximally different from the one immediately preceding it. In 1982 Carter and Carter[4] proposed the first algorithm to compute easily discriminable sets of colors. Several authors[5,6,7] have devised algorithms which can also fulfil a number of ergonomical requirements. We address the algorithmic aspect of the selection of high contrast color sets, and present an algorithm which has a novel, interesting feature: it does not require that the function adopted to code the similarity between two colors (or graphical codes obtained by combining color with other features, such as shape or texture) is a distance function. This should facilitate the definition of perceptual measures of color differences, useful in particular applications. The algorithm, which works whatever the color space selected, performs very well. We report some experimental data regarding the selection of high contrast color sets within both the whole Munsell gamut, and within a reduced palette composed of the Munsell coordinates of the centroid of the 267 color blocks of the ISCC-NBS color naming system.

Problem Definition

According to Carter and Carter[4], selecting a subset of highly contrasting colors from a given gamut of colors means choosing a subset such that in it the minimal distance among all possible couples of colors is maximal. More formally, we let $N = \{c_1, \ldots, c_n\}$ be the given gamut of n colors and K be any subset of N of cardinality $k < n$. Denoting by C the set of all possible subsets of N with exactly k elements, by d_{ij} the distance between color c_i and color c_j, and by $d_{\min}$ the minimal distance among couples of colors belonging to an arbitrary subset K, the problem can be formulated as follows:

$$\max_{C}\left\{d_{min}\right\} = \max_{C}\left\{ \min_{\{c_i, c_j\} \subset K} \left\{d_{ij}\right\}\right\}$$

This problem can be transformed into a combinatorial optimization problem on graphs;[9] we do this because we want to be free to consider any color space and arbitrary dissimilarity functions among colors (not necessarily a distance in a given color space), and thus take into account some ergonomical requirements. Given this abstract formulation of the problem, we can precisely state its computational complexity.

Let $G = (V, E, W)$ be a complete, undirected weighted graph; V is the set of nodes, E is the set of edges, and W: $E \rightarrow R^+$ is a function which assigns to each edge a positive weight. We interpret each element of V as representing a color of the given gamut N, and assign a weight to each edge $\{i,j\} \in E$ throughout the function $W(\{i,j\}) = d_{ij}$, where d_{ij} is the dissimilarity measure between the colors whose corresponding nodes in G are connected by the edge $\{i,j\}$. Of course $W(\{i,i\}) = 0$ for all i and $W(\{i,j\}) = W(\{j,i\})$ for all i,j $(1 \leq i, j \leq n)$. For the sake of simplicity, from now on we shall write Wij instead of $W(\{j,i\})$.

The task of selecting a subset K of k highly contrasting colors from the set N of n colors can be considered equivalent to that of choosing, among all subgraphs of G having k nodes, the subgraph $\hat{G}$ such that the minimal weight is maximal. More precisely:

Problem 1 (Maximal Dissimilarity among colors):
Instance: a complete, undirected, weighted graph $G = (V, E, W)$, a positive integer $k < |V|$.
Solution: $\hat{G}$ = (V, E, W), a subgraph of G having exactly k nodes; $\hat{W}$ is the restriction of the function W on the set $\hat{E} \subset E$.
Cost function: $H(\hat{G}) = \min_{\{i,j\} \in \hat{E}} \{Wij\}$
Goal: Max

It can easily be shown that the decision version of this combinatorial optimization problem is **NP**—complete.[8] Because of this the corresponding optimization version can not be solved exactly by polynomial time algorithms, unless {**P = NP**}.

Following an idea presented,[5] instead of addressing this problem directly we take a different, but related one for which we have designed an efficient algorithm which gives very good approximate solutions. We call this different problem Problem 2. We are able to show the precise relation among the two problems, in particular, we estimate the error that we may make solving Problem 2, instead of Problem 1.

Problem 2 Minimum of the sum of the inverses of the dissimilarity raised to α:

Instance: a complete, undirected, weighted graph $G = (V, E, W)$, a positive integer $k < |V|$.

Solution: $\hat{G} = (\hat{V}, \hat{E}, \hat{W})$, a subgraph of G having exactly k nodes; W is the restriction of the function W on the set $\hat{E} \subset E$.

Cost function: $F_\alpha(G) =$

$$\sum_{\{i,j\}\in\hat{E}} \frac{1}{\left(\hat{W}_{i,j}\right)^a}$$

Goal: Min

It can easily be shown that also the decision version of this problem is also **NP**—complete.

Solving Problem 2 exactly means finding the minimum of the sum of the inverses of the dissimilarities raised to α; this may produce couples of colors which are not dissimilar enough. However, the next theorem, which we give without proof, shows how to set α in order to make the error that can be incurred by solving Problem 2, instead of Problem 1 negligible.

Given two arbitrary complete, undirected, weighted graphs $G_1 = (V_1, E_1, W_1)$ and $G_2 = (V_2, E_2, W_2)$ such that $|V_1| = |V_2| = k$ we state that:

Fact 2.1 If $H(G_1) < H(G_2)$ than there exists α such that $F_\alpha(G_1) > F_\alpha(G_2)$

If we set $\in$, the relative error that we can accept for an approximate solution of Problem 1 (of course $0 < \in < 1$), we can prove that taking

$$a > \frac{2 \lg k}{\in}$$

we make the maximization of the minimal dissimilarity among couples of colors equivalent to the minimization of the sum of the inverses, raised to α. This result is useful since the design of efficient algorithms for minimizing the sum is easier. Indeed we have already used Hopfield networks[9] to do so, since this problem formulation can be very simply transformed into a neural network algorithm. In the following section we present a local search algorithm that performs much better than the Hopfield network.

Algorithm

Before describing the algorithm itself we observe that, given a complete undirected weighted graph $G = (V, E, W)$ with n nodes, the problem we are interested in can be expressed in this way:

$$\begin{cases} \min \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \frac{2}{W_{ij}^a} x_i x_j \\ \sum_{i=1}^{n} x_i = k \end{cases}$$

where x_i, x_j denote nodes of G. We also recall the bijective correspondence among subsets with k elements of a given set with n elements and the vectors in $\{0, 1\}^n$ with exactly k elements equal to 1. In particular, denoting with V_k an arbitrary subset with k elements of the set V with n elements, the corresponding vector $x(V_k) = (x_1, \ldots, x_n) \in \{0, 1\}^n$ is so defined: $x_i = 1 \; iff \; \mathrm{i} \in V_k$.

Given $x(V_k)$ we say that the vector x' is reachable from x in a step if it is obtainable selecting two elements $x_i = 1$ and $x_j = 0$ of x and exchanging their values, that is setting $x_i = 0$ and $x_j = 1$. A neighborhood of x is given by all the vertices of the hypercube $\{0, 1\}^n$ that are obtainable from the vertex x, setting at 1 one of the $n - k$ elements which have value 0, and setting at 0 one of the k elements which have value 1.

Without loss of generality let us consider a vector $x = (x_1, \ldots, x_n)$ with k elements with value 1 and the vector x' obtained from x by interchanging x_1 and x_2. We have:

$$\begin{aligned} F_a(x) &= \sum_{i=1}^{n} \sum_{j=1, j\neq 1}^{n} \frac{1}{W_{ij}^a} x_i x_j \\ &= \sum_{i\neq 1}^{n} \sum_{j\neq 2}^{n} \frac{1}{W_{ij}^a} x_i x_j + \frac{1}{W_{12}^a} x_1 x_2 + \\ &\quad \sum_{j\neq 1, j\neq 2} \frac{1}{W_{1j}^a} x_1 x_j + \sum_{i\neq 1, i\neq 2} \frac{1}{W_{i2}^a} x_1 x_2 \end{aligned}$$

Now let us study how F_α changes selecting x' after x. If $F_\alpha(x) = F_\alpha(x_1, x_2, \ldots, x_n)$ then $F_\alpha(x') = F_\alpha(1 - x_1, 1 - x_2, \ldots, x_n)$, by simple manipulations we obtain:

$$\begin{aligned} \Delta F_a &= F_a(x_1, x_2, \ldots, x_n) - F_a(1 - x_1, 1 - x_2, \ldots, x_n) \\ &= (x_1 - 1 + x_1) \sum_{j\neq 1, j\neq 2} \frac{1}{W_{1j}^a} x_j + (x_2 - 1 + x_2) \sum_{j\neq 1, j\neq 2} \frac{1}{W_{2j}^a} x_j \end{aligned}$$

Denoting $(x_1 - 1 + x_1)$ with t, it is easily seen that $(x_2 - 1 + x_2) = -t$, and:

$$\Delta F_a = t \left(\sum_{j\neq 1, j\neq 2} \frac{1}{W_{1j}^a} x_j - \sum_{j\neq 1, j\neq 2} \frac{1}{W_{1j}^a} x_j \right)$$

Since the goal is the minimization of F_α, we must find the conditions which make $\Delta F_\alpha > 0$, that is:

$$t > 0 \text{ and } \sum_{j\neq 1, j\neq 2} \frac{1}{W_{1j}^a} x_j > \sum_{j\neq 1, j\neq 2} \frac{1}{W_{2j}^a} x_j$$

or

$$t > 0 \text{ and } \sum_{j \neq 1, j \neq 2} \frac{1}{W_{1j}^a} x_j < \sum_{j \neq 1, j \neq 2} \frac{1}{W_{2j}^a} x_j$$

Let us suppose without loss of generality that $x_1 = 1$ (then $t = 1$), then $\Delta F_\alpha > 0$ only if:

$$\sum_{j \neq 1, j \neq 2} \frac{1}{W_{1j}^a} x_j > \sum_{j \neq 1, j \neq 2} \frac{1}{W_{1j}^a} x_j$$

If this condition is satisfied, then the interchange between the values of x_1 and x_2 makes F_α decrease. Having made these preliminary observations we can sketch the algorithm.

Algorithm

Input: A complete undirected weighted graph $G = (V, E, W)$

Step 1: Choose randomly a subset $V_k \subset V$ with k elements, and build the vector $x(V_k)$

Step 2: Choose two elements x_i and x_j in $x(V_k)$ such that $x_i = 1$ and $x_j = 0$

Step 3: Estimate

$$A = \sum_{l \neq i, l \neq j} \frac{1}{W_{il}^a} x_l \quad \text{and} \quad B = \sum_{l \neq i, l \neq j} \frac{1}{W_{jl}^a} x_l$$

Step 4: If $A > B$ then $x_i := 0$ and $x_j := 1$

Step 5: If $x(V_k)$ has been scanned completely and during the last scan no changes have occurred, STOP, else GO TO **Step 2**

Output: The chosen subset $\hat{V} \subset kV$

The algorithm works whatever the color space selected and it can be easily modified if some fixed colors are desired in the high contrast color set to be selected.

Experimental Results

In our experiments to evaluate the algorithm's performance in selecting high contrast color sets within the whole Munsell gamut and within the reduced palette, of the Munsell coordinates of the centroid of the 267 color blocks of the ISCC-NBS color naming system, the dissimilarity function between two colors was simply their euclidean distance in the CIELUV space. We compared the results with those obtained euristically by Kelly[3] and those obtained by Van Laar and Flavell.[10] The 22 colors defined by Kelly have an average color distance of 103 cdu (color difference units), with minimum and maximum values of 19 and 219 cdu respectively. Van Laar and Flavell's algorithm selected a set of 13 colors within a NEC CRT gamut, with an avarage color difference of 111.7 cdu, the minimum value of 49 cdu, and the maximum value of 240 cdu.

When 13 colors are selected within the ISCC-NBS color palette our algorithm, with α set at 65, gives average, minimum, and maximum color difference values of 119.2 cdu, 64.2 cdu and 222 cdu respectively. The corresponding values for 22 colors are 110.3 cdu, 45.2 cdu and 219 cdu. If the algorithm is applied to the whole Munsell set, the average, minimum and maximum color difference values are 167 cdu, 84 cdu, and 311.6 cdu respectively when 13 colors are selected and 147.3 cdu, 67 cdu and 306 cdu, for 22 colors.

We are currently performing experiments to test the algorithm in selecting colors in a typical CRT gamut. Another problem addressed is the definition of dissimilarity functions that take into account perceptual constraints on luminance and chromaticity.

Acknowledgments

We thank Prof. Alberto Bertoni for many fruitful discussions and helpful suggestions.

References

1. A. Della Ventura and R. Schettini, (1993) Computer Aided Color Coding. Communication with Virtual Word (N.M. Thalmann, D. Thalmann eds), Springer-Verlag, Tokyo, 62-75.
2. L. W. MacDonald, (1990) Using Colour Effectively In Displays for Computer-Human Interface, *Displays*, **11**, 129-141.
3. K. L. Kelly, (1976) Twenty-Two Colors of Maximum Contrast, *Color Engineering*, **3**, 26-27.
4. R. C. Carter and E. C. Carter, (1982) High-Contrast Sets of Colors, *Applied Optics,* 2936-2939.
5. W. DeCorte, (1988) Ergonomically Optimal CRT Colours for Non Fixed Ambient Illumination Conditions, *Color Research and Application*} **13**, 5, 327-331.
6. W. De Corte, (1990) Recent Developments in the Computation of Ergonomically Optimal Contrast Sets of CRT Colours, *Displays*, **11**, 123-128.
7. L. D. Silverstein, J. S. Lepkowski, R. C. Carter and E. C. Carter, (1986) Modeling of Display Color Parameters and Algorithm Color Selection.
8. M. R. Garey and D. S. Johnson, (1979) Computers and Intractability, Freeman and Company.
9. P. Campadelli, P. Mora, and R. Schettini, (1995) Color Set Selection for Nominal Coding by Hopfield Networks, *The Visual Computer*, **11**, 150-155.
10. D. Van Laar, A. Flavell, (1994) A Name Based Algorithm for Generating Maximally Discriminable Color Sets, *J. of Photo. Science*, **42**, 33-34.