

TOWARDS AUTOMATIC CONCEPT TRANSFER

Naila Murray*
Computer Vision Center
Universitat Autònoma de Barcelona
Barcelona, Spain

Sandra Skaff†
Xerox Research Centre Webster
Webster, NY, USA

Luca Marchesotti‡
Florent Perronnin§
Xerox Research Centre Europe
Meylan, France

Abstract

This paper introduces a novel approach to automatic concept transfer; examples of concepts are “romantic”, “earthy”, and “luscious”. The approach modifies the color content of an input image given only a concept specified by a user in natural language, thereby requiring minimal user input. This approach is particularly useful for users who are aware of the message they wish to convey in the transferred image while being unsure of the color combination needed to achieve the corresponding transfer. The user may adjust the intensity level of the concept transfer to his/her liking with a single parameter. The proposed approach uses a convex clustering algorithm, with a novel pruning mechanism, to automatically set the complexity of models of chromatic content. It also uses the Earth-Mover’s Distance to compute a mapping between the models of the input image and the target chromatic concept. Results show that our approach yields transferred images which effectively represent concepts, as confirmed by a user study.

CR Categories: I.4.3 [Image Processing and Computer Vision]: Enhancement; I.4.10 [Image Processing and Computer Vision]: Image Representation—Statistical

Keywords: color concepts, color transfer, concept transfer, chromatic modeling

1 Introduction

The manipulation of the chromatic content of images has applications ranging from color correction in movie post-production, to document or user profile personalization and graphic design. In particular, color transfer modifies image color content by transferring the chromatic characteristics of a target image to an input image. There has been limited work in transferring a *concept* to an input image, where a concept is represented by a collection of colors referred to as a *color scheme*. We use the term concept to refer to emotions, moods, or aesthetics which are amenable to a chromatic representation. Examples of concepts include “romantic,” “serene,” and “cool.” Therefore, concept-based color transfer modifies the chromatic appearance of an image or document such that the transferred version assumes a specific conceptual message.

There has been very little work in the literature on concept transfer of images. In [Hou and Zhang 2007] Hou and Zhang proposed a concept transfer method which required manual color editing and

in which the concepts include semantic ones such as “beach”. Yang and Peng [Yang and Peng 2008] described an automatic method of transferring moods to an input image by extracting the color information from a suitable image chosen from a database, where each image is associated with a particular mood. Because each mood is represented by a single color, the concept has a rather narrow interpretation. In addition, it requires each of the target and input images to contain a dominant color, in order to successfully convey the concept. The work described in this paper is closest to the method proposed by Wang *et al.* [Wang et al. 2010] in which the colors of an input image are modified according to a color combination representing a specific theme such as “graceful”. However, their approach requires user scribbles in order to enhance theme transfer.

Unlike the state of the art methods, our proposed method is automatic and it uses abstract concepts specified in natural language. The method is particularly appropriate for the unskilled user, who is often aware of the message he or she wants to convey, but uncertain about the colors to use for his/her document or image. This is the main contribution of our method with respect to other transfer methods such as [Wang et al. 2010]. The second contribution of the paper concerns the way we describe the images during the transfer operation: it leverages a Gaussian Mixture Model (GMM) learned through Convex Clustering. Finally, we also propose a method based on the Earth Mover’s Distance (EMD) to map input chromatic content to target colors. The flow between input and target colors is computed using EMD and it is used as a mapping strategy. Finally, color transfer based on affine transformations is performed, after which a technique to maintain the brightness of the input image in the concept-transferred image is applied. This final stage includes a mechanism which allows the user to tune the level of concept transfer to his or her liking. Figure 1 summarizes our proposed concept transfer framework.

The remainder of the paper is organized as follows. In Section 2 the state-of-the-art in image-based and concept-based color transfer is described. Section 3 describes the stages of our automatic concept transfer framework. Section 4 shows and discusses the results obtained using this framework and compares our results to those of a state-of-the-art approach. In addition, this section describes a test performed on non-expert subjects who were asked to identify, between a pair of images, which image is more consistent with a concept. To the best of our knowledge, this is the first time a user study of this nature is performed. Conclusions are given in Section 5.

2 Previous Work

There is considerable work in the literature on color transfer. With the recent trend of labeling color combinations, or color schemes, with abstract concepts [Ou et al. 2004b; Csurka et al. 2010], the problem of concept transfer is being increasingly addressed. We first start with an overview of color transfer and previous work. We then give an overview of the work on concept transfer.

Color Transfer. Color transfer methods typically consist of three main steps which are described below along with the related work.

Color model representation: The colors present in input and target

*e-mail: nmurray@cvc.uab.es

†e-mail: sandra.skaff@xerox.com

‡e-mail: luca.marchesotti@xrce.xerox.com

§e-mail: florent.perronnin@xrce.xerox.com

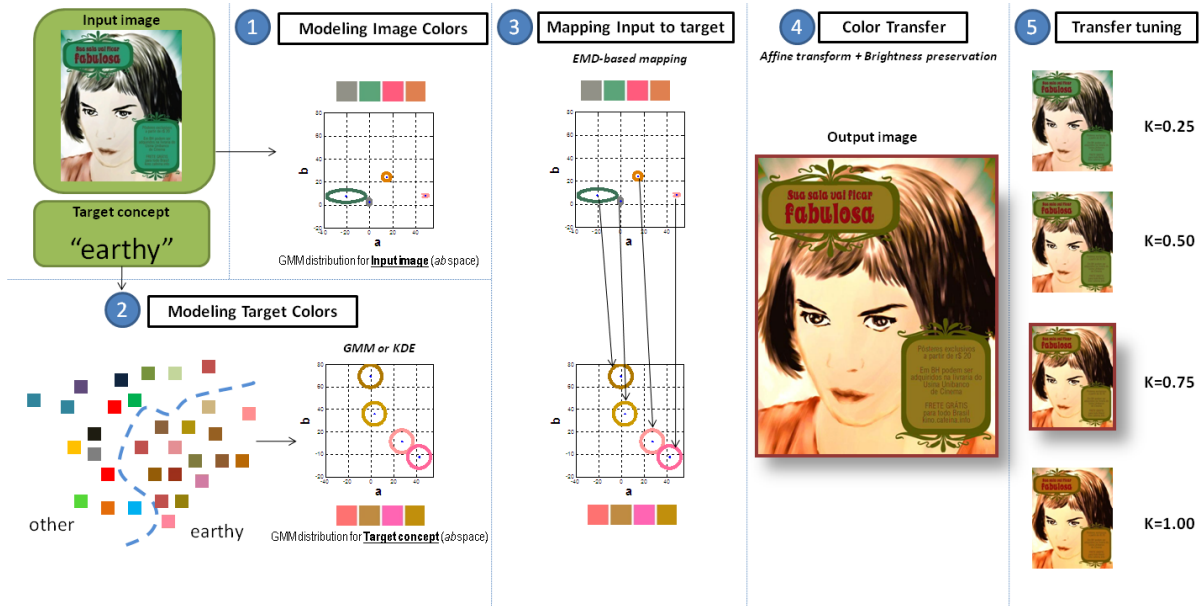


Figure 1: Our concept transfer framework. K is the parameter used to adjust the transfer level. See Section 1 for details.

images are modeled using representations such as GMMs, three-dimensional color histograms, or probability distribution functions. GMMs and color histograms contain components which are representative of the modeled color data, specifically the means and covariances of Gaussian components and the centers of (non-empty) histogram bins, respectively. We refer to these representative color components as *swatches*. Reinhard *et al.*'s seminal work on color transfer models the color profile of an image by the mean and covariance matrices of the pixel representation values in the color space channels [Reinhard *et al.* 2001]. To avoid computing a full covariance matrix, the color representation of an image is usually transformed to the CIELAB (hereafter referred to as the *Lab*) color space to decorrelate the channels. Other methods use full covariance matrices and operate in RGB space [Xiao and Ma 2006]. For a more representative profile, color histograms [Pitié *et al.* 2005; Yang and Peng 2008; Freedman and Kisilev 2010; Pouli and Reinhard 2010] and GMMs [Tai *et al.* 2005; Shapira *et al.* 2009] have been employed. A short, coarse color histogram for an image, termed a color palette, has also been used [Greenfield and House 2003]. A target color model is usually obtained from a reference image supplied either by the user [Reinhard *et al.* 2001; Greenfield and House 2003; Freedman and Kisilev 2010], or via image retrieval from a database [Lalonde and Efros 2007; Yang and Peng 2008]. The composition of this reference, or target image, must be similar to that of the input image, in order for color transfer to occur on a region-to-region basis. For complicated scenes, more than one reference image may be required [Charpiat *et al.* 2008] as finding a reference image with similar content to the input image is difficult.

Input-to-target mapping: Input model colors are mapped to suitable target model swatches using various criteria, most notably distance metrics and EMD-based flows. When input and target color models are created using a single Gaussian, mapping is straightforward [Reinhard *et al.* 2001; Xiao and Ma 2006]. For histogram-based color models, the EMD is a popular choice for which the transportation problem between the input and target histograms is solved. When the mapping is one-to-many, the final color used for transfer is a combination of target model swatches [Lalonde and Efros 2007; Huang and Chen 2009; Freedman and Kisilev 2010; Dong *et al.* 2010]. Colorization may be seen as a special case of

color transfer from a colored target image to a greyscale input image. In this case, input-to-target mapping is often performed based on texture similarity [Qu *et al.* 2006; Charpiat *et al.* 2008]. Artificial color boundaries tend to appear when histogram-based models are used, as spatially close input pixels with similar colors may be placed in different bins and consequently mapped to very different target colors [Yang and Peng 2008]. This phenomenon is less common with GMMs but may be noticeable if the occupancy probability for a pixel is significantly different from that of a neighboring pixel. To improve spatial coherence, constraints on the color differences which are allowed between neighboring pixels were introduced [Charpiat *et al.* 2008; Freedman and Kisilev 2010]. In object-to-object or background-to-foreground color transfer applications, segmentation is required, which puts the final result at the mercy of the quality of the segmentation [Lalonde and Efros 2007; Freedman and Kisilev 2010].

Color Transfer: Transformations between mapped input and target model swatches are first determined. The affine transformation technique introduced by Reinhard *et al.* is the most common one for transferring color to an input pixel x^{in} to produce an output pixel x^{out} [Reinhard *et al.* 2001; Tai *et al.* 2005; Freedman and Kisilev 2010]. First, the pixel colors are represented in a decorrelated color space. This allows the use of diagonal covariance matrices for the input and target color representations, Σ^{in} and Σ^t respectively, which are computed along with the means μ_{in} and μ_t . Using these statistics, the transform for which the statistics of x^{out} match the target statistics is: $x^{out} = \mu^t + (\Sigma^t)^{1/2}(\Sigma^{in})^{-1/2}(x^{in} - \mu^{in})$. For input and target representations with multiple model swatches, a linear combination of such affine transformations are used to transfer color to x^{in} . Pitié and Kokaram introduced a linear transformation derived from the Monge-Kantorovich theory of mass transportation, which minimizes the amount of changes in the image colors due to the transfer [Pitié and Kokaram 2007]. Another common color transfer technique involves transforming the input image such that the resultant output image has the same color distribution as that of the target image. Pitié *et al.* [Pitié *et al.* 2005] transformed the probability density function of the input image into that of the target image. As opposed to automatic color transfer, the approach in [An and Pellacini 2010], following [Levin *et al.* 2004], requires

the user to specify regions of the input and target images whose colors should be matched by drawing strokes across these regions. The cumulative density functions of the input regions are then modified to match that of their target regions using a model whose parameters are determined by solving a constrained optimization problem. For intricate images, many region pairs may need to be defined, making this type of user interaction laborious.

Concept Transfer. One of the earlier approaches is that of Hou and Zhang on color conceptualization [Hou and Zhang 2007]. Their approach considered semantic concepts, such as *beach*, in addition to abstract ones. Images obtained from a database are manually separated into semantic categories, after which the images in each category are algorithmically clustered. Each resulting cluster is represented by an average of the hue wheels of its associated images and is manually labeled with a concept such as “warm”. Histogram matching is performed to alter the colors of an image in order to match those of a desired concept. Since this method does not transfer the saturation and intensity components of the concept colors, the corresponding components of the input image must be similar for the concept to be evident in the output image.

Another method which performs concept transfer is that of Yang and Peng [Yang and Peng 2008]. Their method is automatic as it does not require user intervention. However, in their method the concept needs to be specified through a target image which contains a dominant color, thus making the identification of a suitable target image tedious for a user. In addition, a concept is represented by one color in this method, unlike in our case where it is a combination of colors. More details on Yang and Peng’s approach are provided in Section 4.

The work described in this paper is closest to that of Wang *et al.* who propose a technique for color transfer based on concepts [Wang et al. 2010]. Their approach uses a database of natural images to learn relationships between color and texture. For example, “grass” is typically associated with “green” or “yellow”. These relationships serve as constraints during color transfer, limiting the appeal of this approach for non-photo realistic color-modification applications. In addition, prior knowledge is important for forming these relationships, and user input is necessary to impose realism on content which lacks texture. Finally, in the user test, the authors of [Wang et al. 2010] do not ask the subjects to rate images according to their consistency with a *concept* but rather according to their consistency with colors associated with a concept.

3 Automatic Concept Transfer

Our method for automatic concept transfer aims to enable a user to modify any input image such that the modified version conveys a desired concept. The concept is chosen by the user from a selection of concepts described in natural language, such as “earthy”. As illustrated in Figure 1, our method consists of 4 main stages:

1. Modeling the input image;
2. Modeling the target concept;
3. Mapping input colors to target colors;
4. Color Transfer.

As a fifth optional stage, the user may adjust the level of concept transfer. We describe these stages next.

3.1 Modeling the Input Image

In order to model the colors of a natural image, it is traditionally assumed that the pixel values represented in a given color space

are generated by a probabilistic mixture model. This model can be either discrete or continuous. In the discrete case, an image is modeled through a color histogram. However, such a representation is sensitive to quantization errors and setting the number of bins is not trivial. Therefore we use a mixture model, namely a GMM, which is the most common continuous model used in this context.

The parameters of the GMM are denoted by $\lambda^{in} = \{w_i^{in}, \mu_i^{in}, \Sigma_i^{in}, i = 1, \dots, N\}$ where w_i^{in} , μ_i^{in} , and Σ_i^{in} are respectively the weight, mean vector and covariance matrix of Gaussian i .

Let $X = x_t, t = 1, \dots, T$ denote the set of observations which are all the pixels of the input image. If q denotes which Gaussian emitted x_t , the likelihood that x_t was generated by the GMM is:

$$p(x_t|\lambda^{in}) = \sum_{i=1}^N w_i^{in} p_i(x_t|\lambda^{in}), \quad (1)$$

where $p_i(x_t|\lambda^{in}) = p(x_t|q = i, \lambda^{in})$. The weights are subject to the constraint: $\sum_{i=1}^N w_i^{in} = 1$. The components p_i are given by:

$$p_i(x_t|\lambda^{in}) = \frac{e^{-1/2(x_t - \mu_i^{in})'(\Sigma_i^{in})^{-1}(x_t - \mu_i^{in})}}{(2\pi)^{D/2} |\Sigma_i^{in}|^{1/2}}, \quad (2)$$

where $|\cdot|$ denotes the determinant operator and D denotes the dimensionality of the feature space. The parameters of the GMM can be obtained by maximizing the log-likelihood function $\log p(X|\lambda^{in})$, which is referred to as Maximum Likelihood Estimation (MLE). The standard approach to MLE uses the Expectation-Maximization (EM) algorithm. The EM algorithm is only guaranteed to converge to a local optimum, not to a global one. The location of convergence is dependent on the initialization parameters. In other words, different initialization conditions will, in general, lead to different schemes.

We choose to represent the pixel values of an input image in the *Lab* color space where the Euclidean distance correlates with the human perception of color. Since our transfer framework employs only the chromatic content of images, we use only the *ab* components.

Setting the complexity of the GMM model (i.e. the number of Gaussian components) that represents the input image is a challenging task. In general, there is no one number of components which fits all images. Moreover, in the specific case of color transfer, determining a suitable number is fundamental. If the number of components in the input image GMM is too small, perceptually different colors may be associated with the same component. On the other hand, if the number of components is too large, perceptually similar colors will be associated with different components. The convex clustering algorithm proposed next is an efficient solution for determining the model complexity.

3.1.1 The Convex Clustering Algorithm

This algorithm involves the optimization of an objective function using EM, and includes a novel pruning method that dramatically increases efficiency.

The Objective Function. Let $x_t, t = 1 \dots T$ be a set of T pixels to be clustered. Each of the x_t ’s is two-dimensional, represented in the *ab* space. In convex clustering, a kernel is centered on each point, with an assigned weight w_t , and all points are potential cluster centers. The objective function to be optimized is [Lashkari and

Golland 2008]:

$$L = \sum_{t=1}^T \log \left(\sum_{s=1}^T w_s k_s(x_t) \right), \quad (3)$$

with respect to the w_t 's under the following set of constraints:

$$\begin{aligned} 0 &\leq w_s \leq 1; \\ \sum_{s=1}^T w_s &= 1. \end{aligned} \quad (4)$$

The values $k_s(x_t)$ are non-negative and express a similarity between points x_s and x_t . Since we use the Euclidean distance, we compute the similarity measure using the Gaussian kernel:

$$k_s(x_t) = \frac{1}{2\pi^{D/2}\sigma^D} \exp\left(-\frac{\|x_t - x_s\|^2}{2\sigma^2}\right), \quad (5)$$

where σ is the standard deviation of the kernel and D is the dimensionality of the vectors (2 in this case). Using a normalized kernel $k(\int_x k(x)dx = 1)$ provides a probabilistic interpretation of convex clustering: $\sum_{s=1}^T w_s k_s$ is a probability density function which is a GMM in the case of the Gaussian kernel and the function of Equation 3 is a log-likelihood objective function.

The Solution. We solve the convex clustering problem using EM [Lashkari and Golland 2008]. This algorithm consists of an E-step (Expectation) and an M-step (Maximization). The E-step computes the T^2 assignments γ_{st} of point x_s to the Gaussian centered on x_t as such: $\gamma_{st} = \frac{w_t k_t(x_s)}{\sum_{j=1}^T w_j k_j(x_s)}$. The M-step re-estimates the w_t

values as such: $w_t = \frac{1}{T} \sum_{s=1}^T \gamma_{st}$. To ensure convergence to the optimal solution, all w_t 's should be initialized to non-zero values. In our experiments, we initialize all w_t to $1/T$. As the number of iterations increases, most of the w_t 's converge to zero and the number of clusters, or equivalently Gaussians, is therefore reduced.

Pruning Clusters. If we knew beforehand which w_t values would converge to zero, then we could speed-up the EM algorithm significantly by initializing these values to zero. As this is not the case, we attempt to predict which values will converge to zero and prune them at the early stage of the algorithm. Such pruning implies removing unlikely cluster candidates by adding a third step to the EM algorithm. The authors in [Lashkari and Golland 2008] proposed setting all the w_t 's below a certain threshold to zero after the M-step. However, such a modification would have a limited impact on the speed of convergence as only a very small fraction of w_t 's are zeroed at each pruning step. We propose a novel pruning scheme which significantly improves the convergence rate by zeroing a large percentage of w_t 's without compromising the optimality of the solution. We note that if $w_t \neq 0$, then x_t is a cluster center and we expect x_t to be assigned with higher probability to its own cluster than to any other one. This can be translated as follows: for all $s \neq t$, $\gamma_{tt} \geq \gamma_{ts}$ or $w_t = 0$. Therefore, we propose the following pruning step:

$$\begin{aligned} \text{if there exists an index } s \text{ such that } \gamma_{tt} &\leq \gamma_{ts}, \\ \text{then set } w_t &= 0. \end{aligned} \quad (6)$$

3.1.2 Comparative Results

On Synthetic Data. When a pruning step is introduced into the algorithm, convergence is not guaranteed. However, experiments show that the log-likelihood of the solutions arrived at are on par with those obtained when running the EM algorithm without the

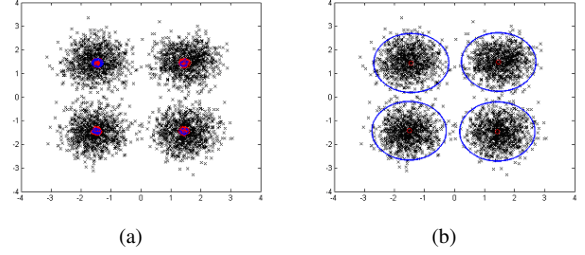


Figure 2: Solutions found by the convex clustering algorithm using (a) the pruning mechanism of [Lashkari and Golland 2008] after 2 min; (b) the proposed pruning mechanism after 3 sec. The small red circles show the cluster centers, i.e. those points x_t with a non-zero w_t . The blue circles centered on the cluster centers have a radius which is proportional to the w_t value.

pruning step. In addition, convergence is sped up with pruning as it sets a large number of w_t 's to zero. We illustrate the performance of the convex clustering approach on synthetic data, which consists of 4,000 points drawn at random from four isotropic Gaussians. Each Gaussian produced 1000 points and was isotropic with $\sigma = 1$ in a 2D space. Results are shown in Figure 2. The left plot of the figure shows that the solution using the EM algorithm of [Lashkari and Golland 2008] contains 150 clusters, and the solution is arrived at after 2 minutes with a log-likelihood value of $-6.2847e+03$. The right plot of the figure shows that the solution using the EM algorithm with our proposed pruning mechanism contains four clusters, and the solution is arrived at after 3 seconds with a log-likelihood value of $-6.2814e+03$. Therefore, the EM algorithm using our proposed pruning method as compared with that of [Lashkari and Golland 2008] results in faster convergence with a similar log-likelihood value.

On Real Data. We illustrate the performance of our approach on real data in the context of concept transfer. We compare concept-transferred images when an input image GMM is initialized using: (a) agglomerative clustering where the number of Gaussians N is set to an ad-hoc value of 16, which is a good general value; (b) agglomerative clustering where N is set to a reasonable value of 1, determined by visual inspection of the points distribution; and (c) convex clustering where the number of Gaussians is determined automatically using our proposed pruning mechanism. In each case, the resulting GMM is used to initialize a MLE estimation of the GMM, which re-estimates the means of the Gaussians and estimates their covariances matrices. Figure 3 shows a comparison of concept transfer results for all three cases. The first row shows the input image and its GMM for (a), (b) and (c). The second row shows the target concept, represented by a sample of associated colors, and its GMM for (a), (b) and (c). The last row shows the concept transfer results in each case. Notice that when an input image GMM of $N = 16$ is used, artifacts occur in the transfer due to the over-segmentation of the image regions resulting from too many Gaussian components. The convex clustering approach automatically sets $N = 1$ and consequently no artifacts in the transfer results can be seen. Therefore, this method can be seen as reducing redundancy by retaining only those sample points which are sufficient for explaining the data.

3.2 Modeling the Target Concept

Recent works have attempted to associate color combinations to abstract concepts [Ou et al. 2004b; Csurka et al. 2010]. We con-

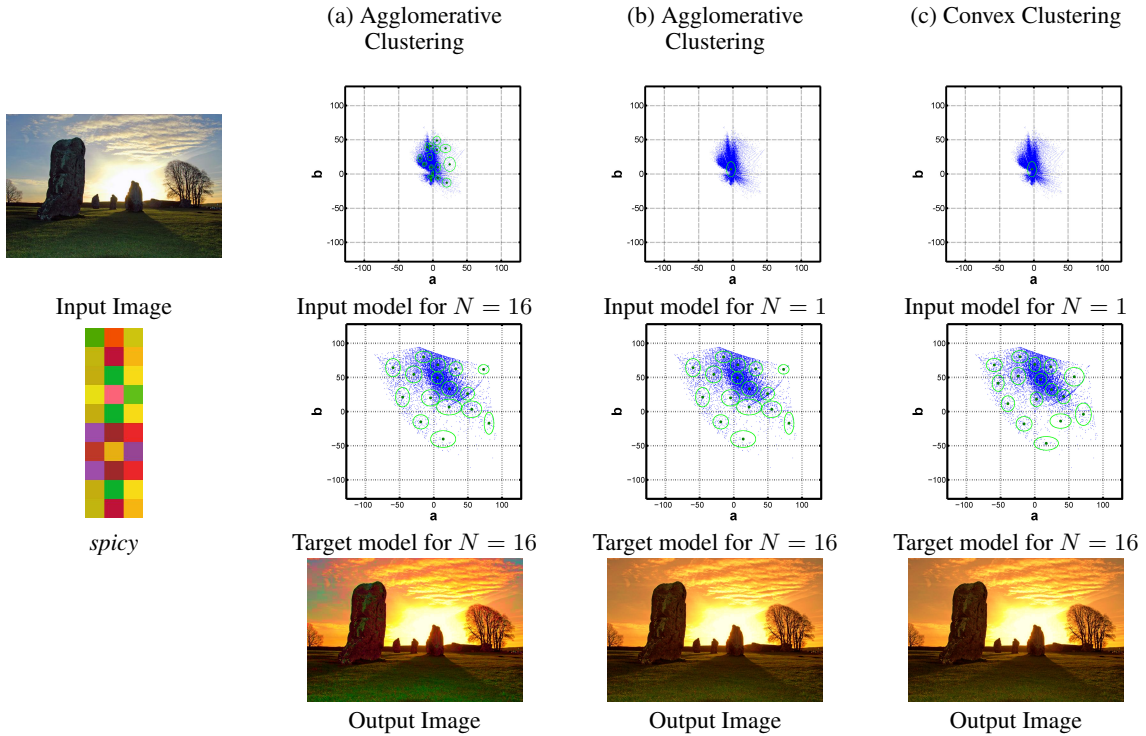


Figure 3: Comparison between agglomerative and convex clustering: (a) shows agglomerative clustering for an ad-hoc number of $N = 16$ clusters. The output image in this case contains significant artifacts due to an overly granulated clustering of the input image chromatic data. The distribution of the data must be known beforehand in order to set $N = 1$ for the input image data (shown in (b)). As shown in (c), convex clustering automatically generates $N = 1$ clusters for this data.

consider a dataset of color schemes labeled with textual tags [Eiseman 2000], that we call the Communicating with Color database. The textual tags are the 15 following concepts: “capricious”, “classic”, “cool”, “delicate”, “earthy”, “elegant”, “luscious”, “playful”, “robust”, “romantic”, “sensual”, “serene”, “spicy”, “spiritual”, and “warm”. Each of these concepts has 24 associated color schemes with three swatches each, some of which are shown in Figure 4. Additional colors were associated to these concepts using a method similar to that described in [Csurka et al. 2010]. This new database is referred to as the Colour Lovers database. Note that additional concepts can also be used. The most straightforward method to model these concepts is to fit a statistical model such as a GMM using the swatches as observations. This ensures that the concept is well represented by a combination of colors, as compared with one- or two-color representations of concepts as in [Ou et al. 2004a; Ou et al. 2004b; Ou et al. 2004c]. In a few cases, the number of colors associated with a concept in the Colour Lovers database is very small and therefore a GMM cannot be fit to them. In this case a good alternative is to use a non parametric technique such as the Parzen Window or Kernel Density Estimation (KDE). We detail these two techniques which are developed and tested in our framework.

Kernel-based Density Estimation. We interpret the colors associated with the target concept as a sequence of i.i.d. random samples $x_1, \dots, x_i, \dots, x_n$ drawn according to some unknown probability law $p(x)$. To estimate the density distribution of these samples we perform convex clustering on colors x_i using the approach described in Section 3.1. The number of cluster centroids N is thus automatically determined. The covariance matrix of a Gaussian component, Σ , is diagonal with σ^2 set to 100 for both a and b channels.

Gaussian Mixture Model Estimation. For concepts with a sufficient number of distinct color schemes, we learn a probabilistic model, namely a GMM. The procedure for creating a GMM to represent concept colors is identical to that for representing input image colors. In this case however, the observations of the GMM are all the swatches of the schemes associated with the target concept [Csurka et al. 2010].

3.3 Mapping Input Colors to Target Colors

We relate the target and the input image model through a mapping which finds soft correspondences between their Gaussians. In Figure 1, we show two GMMs representing a sample input image and the concept “earthy”. We also display the corresponding color schemes using the means of each of the Gaussians in the two models. To perform the association we use the EMD [Rubner et al. 2000], which attempts to solve the following optimization problem:

$$\begin{aligned} \min_{\{f_{i,j}\}} \quad & \sum_{i=1}^M \sum_{j=1}^N f_{i,j} \text{Dist}(s_i^{in}, s_j^c); \\ \text{subject to} \quad & \sum_{j=1}^N f_{i,j} = w_i^{in}, \quad i = 1, \dots, M; \\ & \sum_{i=1}^M f_{i,j} = w_j^c, \quad j = 1, \dots, N, \end{aligned} \quad (C1) \quad (C2)$$

where M is the number of swatches in the input model and Dist is the cost matrix containing the Euclidean distance between each input swatch s_i^{in} and concept swatch s_j^c . The quantities w_i^{in} and w_j^c

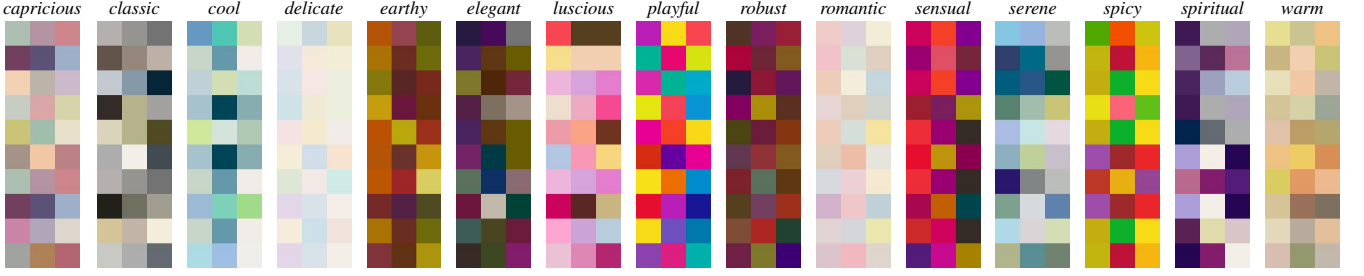


Figure 4: Examples of schemes from the Communicating with Color database for each of the 15 concepts considered. Each scheme comprises 3 swatches. Each concept has 24 associated schemes, 10 of which are shown.

are the weights for the i -th input and concept model swatch respectively. The flow $f_{i,j}$ can be considered to be the part of swatch s_j^{in} which is mapped to s_j^c . The target swatch s_i^t for the input swatch s_i^{in} is computed as a weighted average of the concept swatches, where the weight for s_j^c is $f_{i,j}$. Freedman and Kisilev [Freedman and Kisilev 2010] introduced the idea of mapping input swatches to concept swatches using EMD-derived flows. However, in their case, input and target models were represented by color histograms. Constraint (C1) requires that each input model swatch has flows that sum to its weight. Similarly, constraint (C2) requires that each concept model swatch has flows that sum to its weight. Therefore each concept swatch is guaranteed to be associated with at least one input swatch and vice versa. This ensures that each input color is transferred to a concept color and, in addition, that all concept colors are used.

3.4 Color Transfer

The color-transferred image is obtained in the *Lab* color space by applying an affine transformation to the *ab* channel values of the input image pixels. The lightness channel values of the output image pixels are set to be the same as those of the input image pixels. The transformation is calculated given the *ab* values of the swatch colors of the input and target schemes. We match the statistics of the i -th input swatch to those of the i -th target swatch using a linear transform of the form:

$$A_i = (\Sigma_i^t)^{1/2} (\Sigma_i^{in})^{-1/2}; B_i = \mu_i^t - \mu_i^{in} A_i, \quad (8)$$

where Σ_i^t and Σ_i^{in} denote the covariance matrices of the i -th components of the models representing the target and input schemes respectively, and μ_i^t and μ_i^{in} denote their means. Note that since we choose the covariances to be diagonal, the square-root is uniquely defined. The color transfer functions used in the affine transformation are:

$$A(x) = \sum_{i=1}^N \gamma_i(x) A_i; B(x) = \sum_{i=1}^N \gamma_i(x) B_i. \quad (9)$$

Finally, the transformation is applied to the *ab* channel values of the input image pixel x^{in} to obtain the corresponding values in the output image pixel, denoted by x^{out} :

$$x^{out} = A(x^{in}) x^{in} + B(x^{in}). \quad (10)$$

Different levels of transfer can be performed by introducing a parameter K into the affine transformation using the following formulations of A_i and B_i :

$$\begin{aligned} A_i &= K(\Sigma_i^t)^{1/2} (\Sigma_i^{in})^{-1/2} + (1 - K)I; \\ B_i &= K\mu_i^t + (1 - K)\mu_i^{in} - \mu_i^{in} A_i, \end{aligned} \quad (11)$$

where I is the identity matrix. Using these formulations, for $K = 1$, A_i and B_i revert to the original case (Eq. 8), while for $K = 0$, A_i and B_i revert to the identity matrix and the null matrix respectively so that the input image pixels are unchanged. Therefore, by varying K from 0 to 1, the level of concept transfer may be controlled in a linear fashion. Figure 5 illustrates the effect of K on color transfer. Note that $K = 1$ for all the other results shown in this paper.

Since the image pixel representation values in the channels of the *Lab* color space are not completely decorrelated [Reinhard et al. 2008], our transfer method may still modify the brightness of pixels in the output image. These modifications might involve important features or design elements of the image thus altering its semantic content (see Figure 6) or impact the feel and meaning of the input image. In addition, the converse problem exists wherein features not present in the input image, such as sharp color gradients, are observed in the output image as artifacts. We alleviate these issues by preserving the brightness of the input image within the color transferred image as follows. Firstly we scale the transferred image pixel representation values in the R, G, and B channels to attain the brightness of the input image. However, a few pixels might be assigned to values >255 for one or more channels. We cap such out-of-gamut pixels to 255. We then increase the remaining in-gamut pixel values so as to compensate for the capping. This cap-and-increase procedure is repeated until all pixels are in-gamut. Figure 6 shows the improved results obtained by preserving the image brightness.

4 Results

We show concept transfer results using graphic design, photo-book template, and natural images. We choose images with a variety of scene content and with both smooth and intricate textures. Figure 7 shows results for our approach using GMM-based concept models. More results with GMM-based as well as KDE-based concept models are provided in the supplementary material. Note that constraint (C2) ensures that all concept colors are present in the output image in proportion to their weights in the concept model. For a MATLAB implementation running on an Intel Core 2 Duo CPU at 3.00 GHz with 2GB RAM, a typical run time for an image of size 768x1024 pixels is 12 seconds.

Comparison to State-of-the-Art. In order to compare to previous work, we choose Yang and Peng’s mood-transferring method [Yang and Peng 2008] as a baseline because it is automatic and does not require user scribbling as in [Wang et al. 2010]. Figure 8 shows results for a comparison between our method and Yang and Peng’s mood-transferring method [Yang and Peng 2008]. For more results the reader is referred to the supplementary material. The method of [Yang and Peng 2008] defines a mood or concept with one associated color. The mood of an image is taken to be the one whose

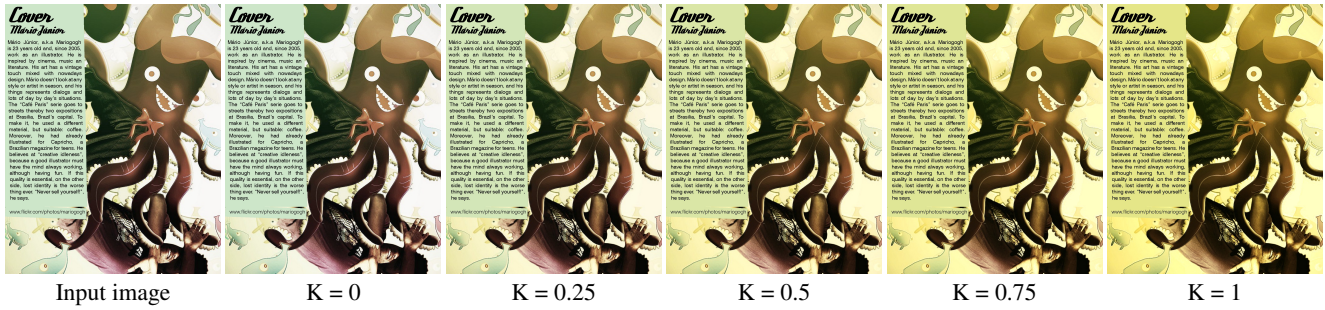


Figure 5: The effect of varying K on concept transfer for “warm”. For $K = 0$ the image is unchanged while for $K = 1$ full transfer has been applied, producing the “warmest” version of the output image. Transfer was performed using GMM-based concept models.

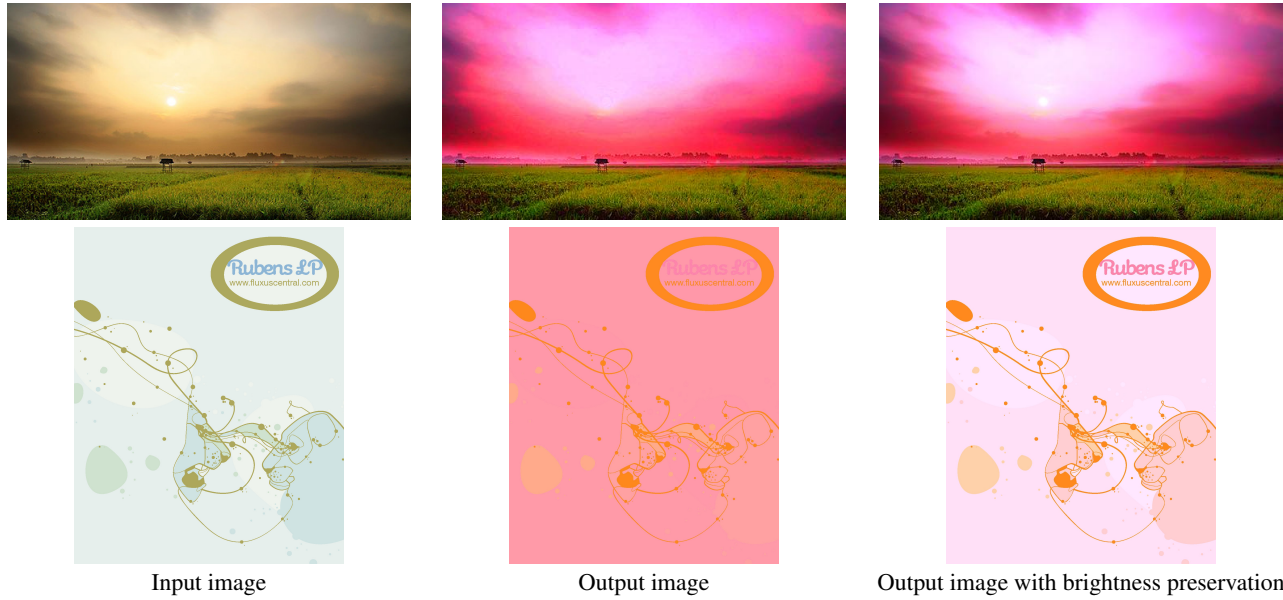


Figure 6: Row 1: After concept transfer with “playful”, the sun in the original image has been recovered from the output image and fewer color gradient artifacts are visible. Row 2: After concept transfer with “sensual”, the contrast between various design elements is lost in the output image. It is then recovered after the brightness preservation step. Transfer was performed using KDE-based concept models.

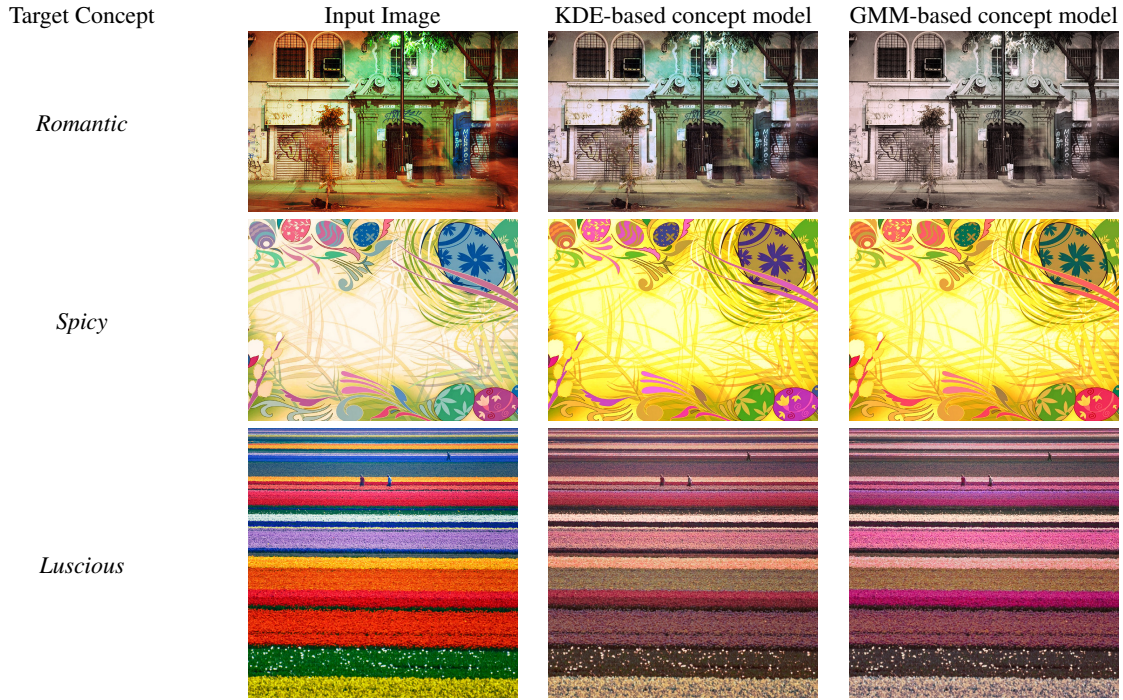


Figure 7: Comparing our method in the cases of KDE- and GMM-based models for the concepts “romantic”, “spicy”, and “luscious”.

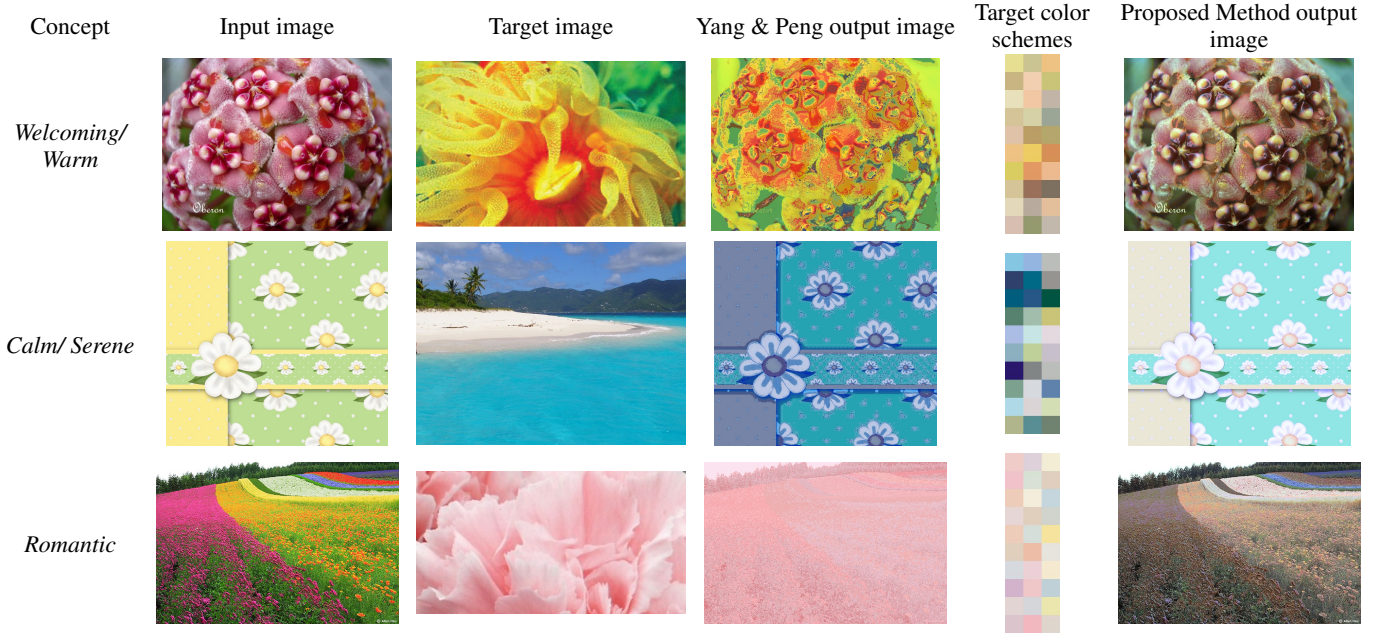


Figure 8: Comparing to [Yang and Peng 2008] when our method uses KDE-based concept models. A few schemes from each concept are shown in the fourth column. For the welcoming/warm and earthy examples, both input images satisfy the dominance constraint, while the constraint is not satisfied for the calm/serene and romantic ones. All target images satisfy the dominance constraint.

predefined associated color is most similar to the most frequent color of the image’s mood color histogram. The latter comprises 24 bins corresponding to the 24 color moods defined. To transfer the mood of a target image to an input image, the colors of the input image in RGB space are rotated such that the principal color axis is aligned to that of the target image. A matching technique between the modified input image and target image histograms is

then applied. This technique iteratively switches colors in input bins that are over-represented with respect to the target bin with colors in bins that are under-represented, by adjusting the colors of the pixels assigned to these bins appropriately. To select a target image, a database of images with pre-computed mood definitions are used. The number of pixels in the second highest frequency bin must be less than 75% of the pixels in the highest frequency

bin of the histogram of each image. This is referred to as a dominance constraint and is imposed on both the input and the target images used. It ensures that the mood colors dominate in the images, thus leading to an unambiguous mood transfer. However, this constraint greatly restricts the input and target images available to a user of this method. The target images used in Figure 8 satisfy this constraint. For the “welcoming”/“warm” example, the input image satisfies the dominance constraint, while the “calm”/“serene” and “romantic” examples illustrate the more general case in which the input images do not satisfy the constraint. As the “romantic” example shows, the output image has many artifacts due to the histogram matching and it contains a smaller variety of colors as compared to the input image. In addition, the non-probabilistic concept model of Yang and Peng produces results which, though conforming with the target image, are often particularly unsuccessful with color gradients. The “welcoming/warm” example is one such case. Our concept model however, is multi-modal probabilistic, and therefore allows for a richer and more extensive variety of concept colors. For this reason the approach is flexible enough for transfers involving input images with varied color content.

User Study. To evaluate our method’s ability to automatically transfer concepts to images, we performed a user study. We first collected 2 test images for each of the 15 concepts listed in Section 3.2. These images were chosen such that their semantic content was broadly consistent with their concept. Concept transfer was performed on these 30 images, resulting in 30 pairs of images, each pair containing an original and a color-transferred image. These image pairs constituted the stimuli of the user study. The pairs were displayed side-by-side on a 14-inch computer screen with 1440x900 resolution. The positions of the images in each pair, either left or right, were randomly assigned.

We invited 20 participants, all with normal or corrected-to-normal color vision, to participate in the user study, which was conducted in the same room and on the same computer for each participant. None of the participants were design professionals. For each of the 30 image pairs, these users were instructed to select the image they considered to be more consistent with the concept. Note that the users were provided only with the name of the concept, and were not shown the color schemes associated with that concept. The users were not informed or aware of the manner in which the images were obtained, that is, that one of the images in each pair was a version of the other image, in which the color content had been modified.

Each image pair was therefore compared 20 times, resulting in 40 pair-wise comparisons per concept. We used these comparisons to fit a probabilistic choice model, namely the Bradley-Terry-Luce (BTL) model, that allows for a quantitative evaluation of our method’s performance for each concept. The BTL model for a concept gives a measure of the degree of preference for the concept-transferred image. This measure is set relative to the original image, such that values above 1 indicate that the transferred image is more consistent with the concept than the original image and values below 1 indicate the contrary. The value of this measure, denoted by u , and the probability, P , of a user preferring the concept-transferred image are reported in Table 1, for each concept. For 10 of the 15 concepts, the concept-transferred image was preferable. The preference for the transferred image was especially high for concepts which are strongly associated with certain colors. For instance, our algorithm performed best for the “romantic”, “sensual” and “serene” concepts, which are associated with pink, red and blue respectively. As shades of these colors were dominant in our learned color schemes for these 3 concepts, the users consistently chose the transferred image. The 5 concepts for which the original image was preferred were “elegant”, “robust”, “luscious”, “delicate” and “spiritual”. None of these concepts are strongly as-

Concept	u	P
romantic	9.0000	0.9000
sensual	7.0000	0.8750
serene	5.6667	0.8500
capricious	4.0000	0.8000
cool	3.4444	0.7750
earthy	3.0000	0.7500
classic	3.0000	0.7500
spicy	3.0000	0.7500
playful	2.0769	0.6750
warm	1.5000	0.6000
spiritual	0.7391	0.4250
delicate	0.6667	0.4000
luscious	0.4815	0.3250
robust	0.3333	0.2500
elegant	0.2903	0.2250

Table 1: BTL model parameters for each concept: u is a measure of the ability of the algorithm to transfer the concept; P is the probability of the user choosing the transferred image as more consistent with the concept.

sociated with specific colors, such that it is difficult to transfer these concepts to an image solely by manipulating its color content.

Limitations. Because the lightness of pixels in the input image are maintained in their corresponding output pixels, concepts whose colors are particularly light or dark, such as “delicate”, or “robust” are not well transferred. In addition, while the EMD algorithm encourages input swatches to be mapped to concept swatches with similar colors due to the minimization expressed in Equation 7, our approach does not include color constraints related to the typical colors of objects or materials. For this reason, grass may be colored *pink*, in addition to *green* and/or *yellow*. Therefore, our method is suited to applications where a fanciful concept transfer is acceptable or required.

5 Conclusions

We presented a novel framework for transferring concepts, specified by natural language, to images. The framework used a convex clustering algorithm to automatically set the complexity of models of the chromatic content of input images and target concepts. A pruning mechanism was introduced to ensure fast convergence of the clustering algorithm. The framework used the EMD algorithm to map input image chromatic content to target concept chromatic content. A user study, which is the first of its nature in the literature, showed the efficacy of the transfer of concepts to images using our framework.

An unresolved challenge in color and concept transfer is ensuring that contiguous regions of an image, which may have the same underlying color but appear differently due to illumination conditions, are recolored to the same color. This is especially difficult when shadows or highlights are present. In the future, we would like to address this challenge by introducing color constancy techniques and spatial constraints into our framework.

Acknowledgments

The authors thank those who served as subjects in the user studies and Tommaso Colombino for his invaluable advice regarding the user study.

References

- AN, X., AND PELLACINI, F. 2010. User-controllable color transfer. *Computer Graphics Forum (Eurographics)* 29, 2, 263–271(9).
- CHARPIAT, G., HOFMANN, M., AND SCHÖLKOPF, B. 2008. Automatic image colorization via multimodal predictions. In *European Conference on Computer Vision*, 126–139.
- CSURKA, G., SKAFF, S., MARCHESOTTI, L., AND SAUNDERS, C. 2010. Learning moods and emotions from color combinations. In *Indian Conference on Computer Vision, Graphics, and Image Processing*.
- DONG, W., BAO, G., ZHANG, X., AND PAUL, J.-C. 2010. Fast local color transfer via dominant colors mapping. In *ACM SIGGRAPH ASIA 2010 Sketches*, ACM, New York, NY, USA, SA '10, 46:1–46:2.
- EISEMAN, L. 2000. *Pantone Guide to Communicating with Color*. Grafix Press, Ltd.
- FREEDMAN, D., AND KISILEV, P. 2010. Object-to-object color transfer: optimal flows and smp transformations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 287–294.
- GREENFIELD, G., AND HOUSE, D. 2003. Image recoloring induced by palette color associations. In *International Conference on Computer Graphics, Visualization and Computer Vision*, vol. 11, 189–196.
- HOU, X., AND ZHANG, L. 2007. Color conceptualization. In *ACM International conference on Multimedia*, 265–268.
- HUANG, T.-W., AND CHEN, H.-T. 2009. Landmark-based sparse color representations for color transfer. In *IEEE International Conference on Computer Vision*, 199–204.
- LALONDE, J.-F., AND EFROS, A. 2007. Using color compatibility for assessing image realism. In *IEEE International Conference on Computer Vision*, 1–8.
- LASHKARI, D., AND GOLLAND, P. 2008. Convex clustering with exemplar-based models. In *Advances in Neural Information Processing Systems*, vol. 20, 825–832.
- LEVIN, A., LISCHINSKI, D., AND WEISS, Y. 2004. Colorization using optimization. *ACM Transactions on Graphics* 23, 3, 689–694.
- OU, L.-C., LUO, M. R., WOODCOCK, A., AND WRIGHT, A. 2004. A study of colour emotion and colour preference. Part I: Colour emotions for single colours. *Color Research And Application* 29, 3, 232–240.
- OU, L.-C., LUO, M. R., WOODCOCK, A., AND WRIGHT, A. 2004. A study of colour emotion and colour preference. Part II: Colour emotions for two-colour combinations. *Color Research And Application* 29 (May), 292–298.
- OU, L.-C., LUO, M. R., WOODCOCK, A., AND WRIGHT, A. 2004. A study of colour emotion and colour preference. Part III: Colour preference modeling. *Color Research And Application* 29 (October), 381–389.
- PITIÉ, F., AND KOKARAM, A. 2007. The linear monge-kantorovitch linear colour mapping for example-based colour transfer. In *European Conference on Visual Media Production*, 1–9.
- PITIÉ, F., KOKARAM, A. C., AND DAHYOT, R. 2005. N-dimensional probability density function transfer and its application to colour transfer. In *IEEE International Conference on Computer Vision*, vol. 2, 1434–1439.
- POULI, T., AND REINHARD, E. 2010. Progressive histogram reshaping for creative color transfer and tone reproduction. In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '10, 81–90.
- QU, Y., WONG, T.-T., AND HENG, P.-A. 2006. Manga colorization. *ACM Transactions on Graphics* 25, 3, 1214–1220.
- REINHARD, E., ASHIKHMIN, M., GOOCH, B., AND SHIRLEY, P. 2001. Color transfer between images. *IEEE Computer Graphics Applications* 21, 5, 34–41.
- REINHARD, E., KHAN, E., AKYZ, A., AND JOHNSON, G. 2008. *Color Imaging: Fundamentals and Applications*. A. K. Peters, Ltd.
- RUBNER, Y., TOMASI, C., AND GUIBAS, L. J. 2000. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision* 40, 2, 99–121.
- SHAPIRA, L., SHAMIR, A., AND COHEN-OR, D. 2009. Image appearance exploration by model-based navigation. In *Eurographics*, vol. 28, 629638.
- TAI, Y., JIA, J., AND TANG, C. 2005. Local color transfer via probabilistic segmentation by expectation-maximization. In *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 1, 747–754.
- WANG, B., YU, Y., WONG, T. T., CHEN, C., AND XU, Y.-Q. 2010. Data-driven image color theme enhancement. *ACM Transactions on Graphics (SIGGRAPH Asia 2010 issue)* 29, 6, 146:1–146:10.
- XIAO, X., AND MA, L. 2006. Color transfer in correlated color space. In *ACM International Conference on Virtual Reality Continuum and its Applications*, 305–309.
- YANG, C.-K., AND PENG, L.-K. 2008. Automatic mood-transferring between color images. *IEEE Computer Graphics and Applications* 28, 2, 52–61.