Automatic Color Correction based on Generic Content Based Image Analysis

Michael Schröder and Stefan Moser
Gretag Imaging AG
Regensdorf, Switzerland

Abstract

In this paper, we present a novel concept of color correction for consumer digital still camera (DSC) images. This concept is based on a hierarchical Bayesian image content analysis consisting of feature extraction and unsupervised clustering and on a set of color correction algorithms that have been optimized on the obtained characteristic image classes. Since the concept uses Bayesian inference to combine several color correction results, further available information (e.g., obtained from camera metadata) can be easily integrated into the color correction process.

Introduction

Many images generated by today's consumer DSC still suffer from color casts. Photofinishing labs report that up to 50% of all images need to be *manually* color corrected before printing in order to achieve a satisfying quality, resulting in very high labor costs. Therefore, *automatic* algorithms for color correction are highly desirable.

However, many automatic color correction algorithms assume calibrated systems and do not work for images of consumer DSC that already make automatic white balance and that provide image data in the sRGB color space. The sRGB color space is an output color space and is obtained from the colorimetric color space in the camera (e.g., ISO RGB) by a rendering step. Therefore, algorithms that estimate the illumination from the distribution of colors in some color space (e.g., "gamut mapping color constancy" or "color by correlation" [1]) are of very limited use for the output of consumer DSCs.

A frequently made observation for all simple color correction algorithms (such as "white patch", "grey world", or "Retinex") is that each algorithm performs well for some images, but very badly for others. Therefore, if a particular algorithm is adjusted in such a way that on the average the color correction provides some benefit, then the algorithm will be very weak and it will provide only marginal improvements. For some algorithms it might even turn out, that it is better to abstain from applying a particular color correction algorithm at all (a similiar observation has been made when using color constancy to make classification more robust [2]).

The solution that we propose in this contribution is to do a classification of the image into certain characteristic classes first and then to apply the mentioned simple color correction algorithms, which have been optimized for the individual image classes. These image classes can be either *semantic* classes (certain scene types, e.g., "indoor scene", "vegetation scene", or "mountain scene") or signal-oriented, *generic* classes (e.g., "scene of high color complexity"). After the class-specific application of a set of simple color correction algorithms, we combine the results of these algorithms in such a way as to take into account the class-specific reliabilities of each algorithm. We do this in a step of Bayesian inference that allows us to take all available information into consideration: from values of certain features, via characteristic image classes up to image meta data provided by the DSC image file [3], such as the camera model or details of the image exposure. Altogether, this results in a hierarchical system of automatic color correction that significantly reduces the number of consumer DSC images that require manual color correction.

This paper is structured as follows. In the next section, we sketch the hierarchical image analysis that provides us with a signal-oriented description of the content in the image. Then we use this description to perform several class-specific color corrections, which we combine using Bayesian inference and the set of class probabilities obtained by the initial image content description. We present quantitative experimental results using a test database of typical DSC images before we conclude with a short summary.

Generic content-based image analysis

In the recent years, research on generic image content description has been mainly driven by the application for content-based image retrieval (CBIR; for a recent overview of the research area see [4]). In this application, a large database of images can be accessed *by content*, in most cases by providing an example image and feedback on the

Level	Description	Explanation/examples
5	semantic image classes A_ν	scene types (e.g., 'outdoor') or problem types (e.g., 'underexposed')
4	generic image classes w_i	classes derived from level 3 via unsupervised clustering
3	image features	moments of pixel features (level 1) or statistics of pixel classes (level 2)
2	pixel classes	classes derived from level 1 via unsupervised clustering
1	pixel features	luminance, color, local contrast, texture, ...
0	image data I_k	digital images (from DSCs, scanners, etc.) including meta information

Table 1: *Hierarchic scheme of generic content description: Starting from the actual image data (level 0), we calculate local features (level 1), which are classified into characteristic pixel classes (level 2). From the distributions of these pixel classes, we can calculated image features (level 3) that lead (in a second step of unsupervised clustering) to characteristic image classes w_i (level 4). These classes can be used to classify the image into semantic image classes A_ν (level 5) such as scene or problem types.*

subsequent search results.

In order to answer content-based "query-by-example" queries, a generic signal-oriented content index has to be set up and compared with the user examples. Early approaches [5–7] simply described the image using low-level features such as color, texture, and shape resulting in large low-level feature vectors, which can be used to calculate similarities. However, these straightforward approaches have not been able to demonstrate the ability to express content on a higher abstraction level (that is, user semantics such as 'indoor scene' or 'sunset image'). Therefore, more advanced approaches [8, 9] apply a hierarchic scheme and first determine characteristic classes (that is, a "codebook" of image content) based on low-level features (e.g., using self-organizing maps or vector quantization) and then link these classes to semantically meaningful labels.

On the first glance, the application of automatic color correction has not much in common with CBIR. However, most of the simple color correction algorithms (white patch/greyworld/retinex) perform well on some image and badly on others. Therefore, if we are able to automatically detect these images then we will be able to apply the simple algorithms in an optimum way: for each characteristic image class we can select the optimum parameters of each algorithm and decide which algorithm or which combination of algorithm performs best. In addition to *characteristic* image classes, which correspond to classes inhererent in the kind of data used (e.g., a characteristic class of DSC images are B/W images), methods of CBIR can also be used to automatically detect *semantic* image classes, that is, classes that have a certain meaning to humans, such as scene types (e.g., 'outdor' or 'sunset') or problem types (e.g., 'underexposed' or 'strong color cast').

In order to extract the image content information from the image data (a large collection of typical image data, possibly accompanied by information from photographic experts), we apply an hierarchical scheme of Bayesian stochastic modeling [10], which attempts to automatically detect existing, "typical" image content and to describe it in a robust way. This hierarchic scheme consists of five levels of different semantic abstraction that are linked via steps of Bayesian inference. We depict and shortly explain it in Tab. 1.

We abstain from providing more details on the Bayesian inference process on levels 1 to 3 and just note that at the end of this inference process we obtain the posterior class probabilities

$$p(w_i|I_k) \quad \forall i, \tag{1}$$

of characteristic image classes w_i given a particular image I_k. Similiarly, in a supervised scenario, we obtain the posterior probabilities $p(A_\nu|I_k)$ of the semantic classes A_ν given I_k. Note that, in the following, we restrict our discussion of the application to color correction to the case of the *signal* classes w_i. However, for certain applications, one might prefer to use the *semantic* classes A_ν instead.

In order to demonstrate the nature of our generic image content description, we present two examples: characteristic pixel classes (level 2) in Fig. 1 and characteristic image classes in Fig. 2. Both examples will be explained in more detail in the "Experimental Results" section later in this paper.

Class-specific color correction

For a given image class w_i (or in the supervised case A_ν), we apply a set of automatic color correction algorithms $\{V_j\}$, each of which has been optimized for that particular image class and each of which provides us with a certain color correction transformation $\theta_{V_j,w_i}(I_k)$. Thereby, the color correction θ denotes the transformation that is applied to the image data to remove the color cast. It can be given as, e.g., the color of the illumination or the three factors to be applied to the RGB channels.

We can model the relationship between the (unknown) actual color transformation θ and the resulting transformation of each automatic color correction algorithm V_j as likelihood

$$p(\theta_{V_j,w_i}(I_k)|\theta), \tag{2}$$

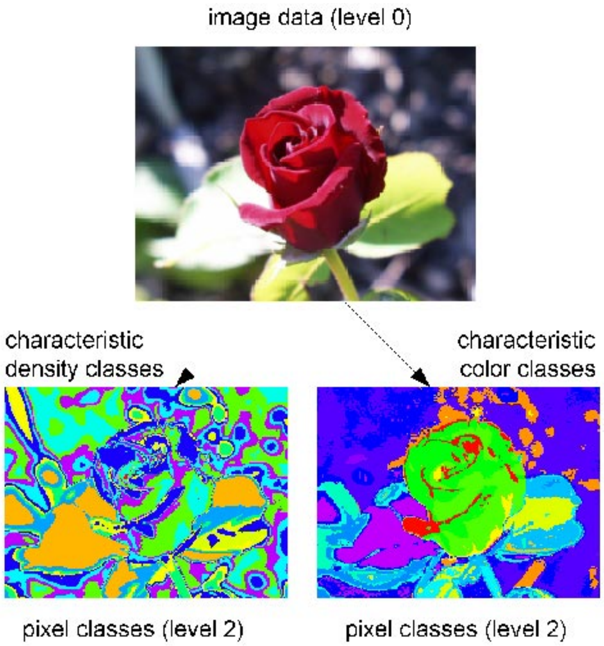

Figure 1: *Examples of characteristic* pixel *classes (level 2) using* density *(bottom left) and* color *(bottom right) as local features. Each class is depicted as a particular color (or if printed in B/W as a particular grey value). The characteristic classes have been obtained via unsupervised clustering over a large dataset of consumer DSC images.*

which specifies the probability of the algorithm's result in dependence of the true value of θ. A very precise algorithm is modelled by a very sharp distribution, whereas a very unprecise algorithm (that is, a very bad one) by a very broad distribution. The detailed shape of this distribution depends on both the algorithm and the signal class to which it is applied. It can be determined from a training dataset together with manually corrections by photographic experts. It's mathematical model can be, e.g., a multivariate Gaussian.

In order to combine the result of several algorithms, we can use the total likelihood of all results, which is given as product of the individual likelihoods if we assume independance of the individual algorithms. Together with the prior distribution

$$p(\theta|w_i, \mathrm{meta\ data}) \tag{3}$$

of the color transformation, we arrive at the posterior distribution of the color transformation as

$$p(\theta|w_i, I_k) \ \propto \ p(\theta|w_i, \mathrm{meta\ data}) \cdot \prod_{\mathrm{j}} \mathrm{p}(\theta_{\mathrm{V_j, w_i}}(\mathrm{I_k})|\theta), \tag{4}$$

where the product over j extends over all algorithms V_j. By maximizing Eq. (4) we obtain the final color transformation for this image, given it is in the specified class w_i.

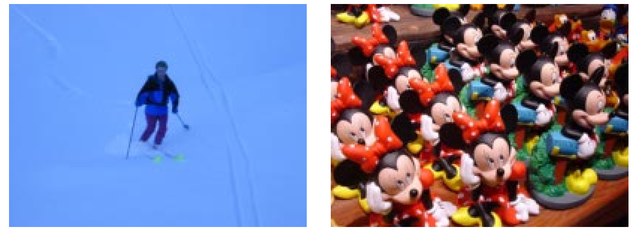

Figure 2: *Examples of characteristic* image *classes (level 4) resulting from features based on characteristic pixel classes: images of low and high* color complexity *(left and right, respectively). We obtain this complexity measure from the distribution of classes using information theoretical methods. It is not equivalent to conventional features, such as size of the color gamut, since it takes into account the statistics of* characteristic *colors in a large dataset of typical DSC images.*

The prior $p(\theta|w_i, \mathrm{meta\ data})$ is independent of the currently analyzed image I_k and models the expected color corrections for the current image class w_i and available meta data. Later, in Fig. 3, we show an example in which we demonstrate the dependence of the width of this prior on a particular image class.

Since the stochastic content description provides us with the probability distribution $p(\theta|I_k)$, we can use $p(\theta|w_i, I_k)$ to calculate the posterior probability distribution of the color transformation as

$$p(\theta|I_k) = \sum_i p(\theta|w_i, I_k) \cdot p(w_i|I_k), \tag{5}$$

where the sum over i extends over all image classes w_i. From this posterior probability we can calculate the maximum a posteriori (MAP) color correction by simply maximizing it.

Since the computationally intensive task of the whole procedure is to determine the characteristic classes and to find image features that unravel useful information on possible color casts—or on their absence—the actual application is very fast. The overall speed is determined by the number of simple color correction algorithms used and the number of image classes taken into consideration. If one or more simple color correction algorithms are very slow, approximations of Eqns. (4) and (5) can be easily derived by taking into account only those image classes and algorithms that have a significant contribution.

Experimental Results

We have applied the presented concept to a test database of about 7300 images taken with 60 standard consumer DSCs ranging from simple VGA models up to recent three-megapixel-models.

As image features, we simply use luminance $L*$ (of CIELAB) as measure of pixel brightness and $a*$ and $b*$

(of CIELAB) as two-dimensional measure of pixel colour. After collecting these features from all images in this dataset, we searched each feature space for characteristic pixel classes (level 2) using unsupervised methods of exploratory data analysis. This provided us with sets of characteristic pixel classes for the corresponding features, such as shown in the example in Fig. 1. We used the statistical description of the pixel classes in a particular image to derive image features (level 3) that—in a second step of exploratory data analysis—provided us with characteristic image classes (level 4), such as shown in the example in Fig. 2.

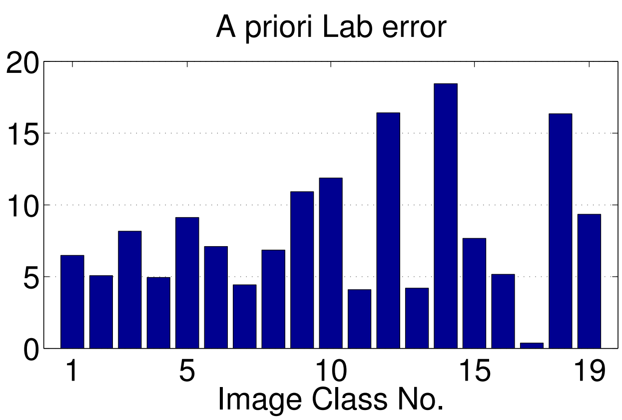

Figure 3: *Average color cast of the images in the training data set in dependence on the signal classes resulting from unsupervised segmentation of color complexity. We can clearly identify image classes that have a high or low* a priori *probability for a color cast (classes 9/10/12/14/18 and 7/11/13/17, respectively). The information on the class-dependent color cast expectation enters our inference process via* $p(\theta|w_i, \mathrm{meta\ data})$ *in Eq. (4).*

In order to demonstrate that the obtained characteristic image classes (level 4) indeed contain useful information for color correction, we calculate the average strength of the observed color casts for each color complexity class and display it in Fig. 3. There is a significant dependence of the strength of the color cast on the image class. Note in particular class 17 which corresponds to B/W images—a class of images that arouse in a completely un-supervised way.

The information on the color casts to be expected for a particular image (that is, the a priori distribution of the color casts) is a very valuable information and enters our inference process via $p(\theta|w_i, \mathrm{meta\ data})$ in Eq. (4). In particular, a prior distribution that is sharply localized around "no correction" prevents us from applying an aggressive color correction algorithm to an image that is very likely to get harmed by it. On the contrary, if the image is assigned to a class with a strong prior color cast, the flat prior will result in comparatively strong changes in color.

In order to demonstrate the overall performance of the proposed scheme on our dataset, we use two simple color correction algorithms: white patch (WP) and a variant of grey world (GW). When performing the classification and calculating the color correction using our method (that is, using Eq. 5), the overall result shows a slight improvement, as we show in Tab. 2.

	A priori error	white patch	grey world	generic classification
mean error	7.1	6.3	6.8	5.5

Table 2: *Mean Lab error for the dataset using white patch, grey world, and our classification-based method in comparison to the a priori error (equivalently, "do nothing" algorithm).*

This difference between the algorithms might seem to be marginal. However, if we categorize the image into color cast classes (on a subjective basis by defining Lab error intervals for 'no'/'weak'/'strong'/'very strong'), more practically useful information is obtained as we show in Tab. 3. The detailed example tables reveil that the globally optimized algorithms provide only very limited improvements: using the white patch (grey world) algorithm, about 32–41% (13–17%) of the images improve, but 7–15% (2%) get worse. The detailed statistics of the combination using generic classification shows a significant higher ratio of improved images (47–58%), with only a slightly higher ratio of deteriorated images (6–17%).

Summary and Conclusion

We have shown how generic content-based image analysis can be utilized to support automatic color correction. As main result of the generic content description we obtain characteristic image classes. Their contribution is two-fold: (1) they can unravel important information on whether a color cast has to be expected or not for a particular image and (2) we can make use of them to apply the individual simple color correction algorithms in an optimal way. The first information can be incorporated as prior into the inference process and the latter via class-specific optimizations and models of the algorithm's performance.

Several extensions are possible. More information might be integrated into the inference process: information on the camera model and manufacturer, camera settings (such as white balance mode, exposure time, and f-stop). Furthermore, due to its class-dependent nature, interactive refinements of corrections can be implemented on a per class basis.

However, we note that the actual "intelligence for color correction" still originates from the used simple algorithms and that the presented scheme is only an assistance for

White Patch

↓before / after→	no	weak	strong	v.str.
no	**0.85**	0.13	0.02	0.00
weak	0.32	**0.61**	0.07	0.00
strong	0.05	0.36	**0.57**	0.02
very strong	0.00	0.02	0.32	**0.66**

Grey World

↓before / after→	no	weak	strong	v.str.
no	**0.98**	0.02	0.00	0.00
weak	0.17	**0.81**	0.02	0.00
strong	0.00	0.17	**0.82**	0.01
very strong	0.00	0.00	0.13	**0.86**

Using Generic Classification

↓before / after→	no	weak	strong	v.str.
no	**0.83**	0.13	0.04	0.00
weak	0.47	**0.47**	0.06	0.00
strong	0.16	0.42	**0.40**	0.02
very strong	0.08	0.09	0.41	**0.42**

Table 3: *Distribution of color casts* after *the application (columns) of various color correction algorithms given the strength of color casts* before *applying the algorithms (rows). Here we compare the results of white patch (top), grey world (middle), and the combined algorithm using the method described in this paper (bottom). Entries on the diagonals specify the proportion of images that remain unchanged (printed in bold) and entries below or above the diagonal specify images that have a weaker or stronger color cast after applying the algorithm, respectively.*

their application (using the best algorithm combination with optimum parameters for each image). In addition, since the overall algorithm is more complex than simple approaches, significantly more training data is needed. Furthermore, it is critical that this training data represents an average cross section of future image data to which the algorithm will be applied. Of course, the latter two drawbacks have to be taken into account for all color correction algorithms with many internal parameters (such as approaches based on neural networks).

Altogether, the described hierarchical system can be applied for any device or tool in which automatic color correction of consumer DSCs is needed, such as digital photofinishing systems, Photo CD systems, and standalone image enhancement tools. It might help to further improve the quality of fully automatic digital imaging systems. In addition, further applications of generic image content characterization to other kinds of image enhancement tasks can be thought of.

References

1. G. Finlayson, S. Hordley, and P. Hubel. Colour by correlation: a simple, unifying approach to colour constancy. In ICCV'99, volume 2, pages 835–842, 1999.
2. B. Funt, K. Barnard, and L. Martin. Is machine colour constancy good enough? In ECCV'98, pages 445–459, 1998.
3. J. Milch and R. Reisch. Eastman Kodak Company picture metadata guidelines. Technical report, Eastman Kodak Company, 2000. White paper available on the Internet.
4. A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. IEEE Tr. on Pattern Analysis and Machine Intelligence, 22(12):1349–1380, December 2000.
5. C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz. Efficient and effective querying by image content. Journal of Intelligent Information Systems, 3:231–262, 1994.
6. R. W. Picard. Toward a visual thesaurus. Technical Report 358, MIT, 1995.
7. J. Smith and S. Chang. VisualSEEk: A fully automated content-based image query system. In ACM Multimedia 96, Boston, MA, 1996.
8. T. P. Minka and R. W. Picard. Interactive learning with a 'society of models'. Pattern Recognition, 30(4):565–581, 1997.
9. A. Vailaya, M. Figueiredo, A. Jain, and H. Zhang. Image classification for content-based indexing. IEEE Tr. on Pattern Analysis and Machine Intelligence, 10(1):117–130, January 2001.
10. M. Schröder. Stochastic Modeling of Image Content in Remote Sensing Image Archives, volume 2 of Selected Readings in Vision and Graphics. Hartung-Gorre, Konstanz, 2000. ISBN 3-89649-539-9.

Biography

Michael Schröder received the Dipl. Phys. degree in physics from the University of Ulm, Germany in 1996. Afterwards, he joined the Computer Vision Group of the Communication Technology Lab at ETH Zurich, Switzerland. He received his Ph.D. focussing on stochastic modeling of image content in remote sensing image archives in 2000. Since then he has been with the Imaging Information Technology group of Gretag Imaging, Switzerland. His current research interests are digital photography, color correction, and image content characterization.

Stefan Moser received the Masters degree in Electrical Engineering with a focus on image processing, process control and automation systems from the ETH in Zurich, in 1999. Afterwards he completed the "Young Graduate Trainee Program" at the European Space Agency (ESA), where he worked on on-board real time image processing. In 2000 he joined the Imaging Information Technology group of Gretag Imaging, Switzerland, and is now working on statistical image processing and image enhancement.