A Colour Constancy Algorithm Based on the Histogram of Feasible Colour Mappings*

Jaume Vergés-Llahí and Alberto Sanfeliu

Institut de Robòtica i Informàtica Industrial
Technological Park of Barcelona, U Building
Llorens i Artigas 4-6, 08028 Barcelona, Catalonia.
{jverges, asanfeliu}@iri.upc.es

Abstract. Colour is an important cue in many applications in machine vision and image processing. Nevertheless, colour greatly depends upon illumination changes. Colour constancy goal is to keep colour images stable. This paper's contribution to colour constancy lies in estimating both the set and the likelihood of feasible colour mappings. Then, the most likely mapping is selected and the image is rendered as it would be seen under a canonical illuminant. This approach is helpful in tasks where light can be neither controlled nor easily measured since it only makes use of image data, avoiding a common drawback in other colour constancy algorithms. Finally, we check its performance using several sets of images of objects under quite different illuminants and the results are compared to those obtained if the true illuminant colour were known.

Keywords: Colour, colour mappings, colour change, colour constancy, colour histograms.

1 Introduction

In a number of applications from machine vision tasks such as object recognition, image indexing and retrieval, to digital photography or new multimedia applications, it is important that the recorded colours remain constant under changes in the scene illumination. Hence, a preliminary step when using colour must be to remove the distracting effect of the illumination change. This problem is usually referred to in the literature as *colour constancy*, i.e., the stability of surface colour appearance under varying illumination conditions. Part of the difficulty is that this problem is entangled with other confounding phenomena such as the shape of objects, viewing and illumination geometry, besides the changes in the illuminant spectral power distribution and the reflectance properties of the imaged objects.

A general approach to colour constancy is to recover a *descriptor* for each different surface in a scene as it would be seen by a camera under a canonical illuminant. This is similar to pose the problem as that of recovering an estimate

* Partially funded by a grant of the Gov. of Catalonia and the CICyT DPI2001-2223.

A. Sanfeliu and J. Ruiz-Shulcloper (Eds.): CIARP 2003, LNCS 2905, pp. 171–179, 2003.
© Springer-Verlag Berlin Heidelberg 2003

of the colour of the scene illumination from an image taken under an unknown illumination, since it is relatively straightforward to map image colours back to illuminant independent descriptors [1].

Therefore, *finding a mapping between colours or the colour of the scene illuminant are equivalent problems*. This path has been traced by a great amount of algorithms, being those related to the *gamut–mapping* the most successful [2, 3,4,5].

Lately, the trend has slightly changed to make a guess on the illumination, as in *colour–by–correlation* [1] or *colour–voting* [6], rather than attempting to recover only one single estimate of the illuminant. A measure of the *likelihood* that each of a set of feasible illuminants was the scene illuminant is set out instead, which is afterwards used to select the corresponding mapping to render the image back into the canonical illuminant.

2 Discussion

These approaches have two common drawbacks. First, as a rule, all of them rely on the fact that the set of all possible colours seen under a canonical illuminant is, somehow, known and available. That is, we must know *a priori* how any possible surface will appear in an image.

The collection of gamut–mapping algorithms uses them to constrain the set of feasible mappings, while the colour–by–correlation algorithm builds the correlation matrix up with them, which in addition implies that this set of colours must be known for each single illumination taken into account.

Secondly, in gamut–mapping algorithms the set of realizable illuminants also needs to be known *a priori* to restrict the feasible transformations. Besides, while this set is a *convex hull* in the gamut–mapping family, it is a finite set in the colour–by–correlation algorithm not covering any intermediate illuminant.

In short, before any of the previous colour constancy algorithms can even be set to work, a pretty big chunk of *a priori* knowledge about reflectances and lights is needed, which reduces the scope of those methods. We point out this lack in two basic tasks where a mechanism of colour constancy is required [7], namely, *colour indexing* and *colour-based object recognition*. In both cases, it may be very difficult or simply impossible to have an *a priori* realistic database of surface and illuminant colours. Image indexing may be using images of unknown origin such as Internet while recognition may be part of a higher task where light conditions are uncontrollable or unknown.

Thus, this paper suggests a less information-dependent colour constancy algorithm which just relies on pixels and is capable of rendering images from an unknown illumination back into a task-dependent canonic illuminant. We only require that the set of images to transform shows similar scenes without caring about the number of imaged objects since no segmentation is carried out.

3 Diagonal Model and Chromaticity Coordinates

First of all, the problem of modelling the colour change must be considered. If referred to the literature, from Forsyth [2] to Finlayson et al. [3,1], the algorithms with best performance are based on a *diagonal model*, i.e., colours recorded under one illuminant can be mapped onto those under a different illuminant by applying *individual scaling factors* to each coordinate. Forsyth's gamut–mapping algorithm used $3D$ diagonal matrices to transform RGB sensor responses:

$$(R', G', B')^t = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & \gamma \end{bmatrix} \cdot (R, G, B)^t \tag{1}$$

That algorithm worked well only on a restricted set of images which included flat, matte, uniformly illuminated scenes. To alleviate problems found in images with *specularities* or *shape* information and to reduce the computational burden, Finlayson [3] discarded intensity information by just working in a $2D$ chromaticity space usually referred to as *perspective colour coordinates*:

$$(r, g) = \left(\frac{R}{B}, \frac{G}{B}\right) \tag{2}$$

Therefore, the diagonal matrix of Eq. (1) expressed in perspective coordinates changes into the following relation:

$$(r', g')^t = \begin{bmatrix} \frac{\alpha}{\gamma} & 0 \\ 0 & \frac{\beta}{\gamma} \end{bmatrix} \cdot (r, g)^t = \begin{bmatrix} \tilde{\alpha} & 0 \\ 0 & \tilde{\beta} \end{bmatrix} \cdot (r, g)^t \tag{3}$$

Later, Finlayson and Hordley proved in [8] that there is no further advantage in using $3D$ algorithms because the set of feasible mappings after being projected into $2D$ is the same as the set computed by $2D$ algorithms. Hence, both chromaticity coordinates in Eq. (2) and $2D$ diagonal mappings in Eq. (3) will be used throughout this paper.

4 Measuring the Performance of Colour Constancy

The performance of a colour constancy algorithm is usually measured as the error of the illuminant estimates or the RMS error between the transformed and canonic images, which is useless if the point of view changes or objects move.

As reported in [9,7], colour histograms are an alternative way to globally represent and compare images. Thus, the *Swain&Ballard* intersection–measurement in [9] computes the *resemblance*[1] between two histograms $\mathcal{H}$ and $\mathcal{T}$:

$$\cap(\mathcal{H}, \mathcal{T}) = \sum_k \min\{H_k, T_k\} \in [0, 1] \tag{4}$$

[1] A distance measure can be similarly defined as $Dist(\mathcal{H}, \mathcal{T}) = 1 - \cap(\mathcal{H}, \mathcal{T}) \in [0, 1]$.

The advantages of this measure are that it is very fast to compute if compared to other matching functions [10], and more importantly, if histograms are sparse and colours equally probable, this is a robust way of comparing images [9,10].

Since both colour indexing and colour-based object recognition using the *Swain&Ballard* measure fail miserably when scene light differs from that used in creating the database of model images [7], we suggest Eq. (4) as a mean of both computing the *performance* of a colour constancy algorithm and also that of *measuring* the suitability of a particular colour mapping if one histogram corresponds to a transformed image and the other to the canonic one.

5 Colour Constancy Algorithm

We suggest an algorithm to estimate the *set* and *likelihood* of feasible colour mappings from image pixels. This set is analogous to the set of possible mappings in [2,3,8], but here the likelihood of each mapping is computed, as in [1].

The algorithm supposes we have images of similar scenes under different illuminants and that we want to render them as seen under a canonic illuminant[2]. The number of objects in the scene does not matter since we do not segment the image and only the pixels are used to find a colour mapping as those of Eq. (3).

More precisely, let I^a and I^b be two colour images of nearly the same scene taken under *different* and *unknown* illuminants. We take I^b as the *canonic* and our goal is to find a colour transformation $\mathcal{T} \in \mathbf{T}$ which maps the colour of the pixels of image I^a *as close as possible* onto those of image I^b. $\mathbf{T}$ is the set of *feasible colour mappings*. We note the *transformed* image as $\mathcal{T}(I)$, which is formed by applying $\mathcal{T}$ to every pixel in I.

The main idea of this algorithm is to estimate the *likelihood* $\mathcal{L}(\mathcal{T} \mid I^a, I^b)$ of every feasible mapping $\mathcal{T} \in \mathbf{T}$ just from pixel data of images I^a and I^b. Afterwards, we will select the most likely transformation $\mathcal{T}_0$:

$$\text{find } \mathcal{T}_0 = \underset{\mathcal{T} \in \mathbf{T}}{\text{argmax}} \; \{\mathcal{L}(\mathcal{T} \mid I^a, I^b)\} \tag{5}$$

According to Eq. (3), $\mathcal{T} = diag(\tilde{\alpha}, \tilde{\beta})$, where $\tilde{\alpha}, \tilde{\beta} \in \left[\frac{1}{255}, 255\right]$. Therefore, for every pair of chromaticities $(r^a, g^a) \in I^a$ and $(r^b, g^b) \in I^b$ it is possible to compute the transformation relating them as the quotient:

$$(\tilde{\alpha}, \tilde{\beta}) = (\frac{r^b}{r^a}, \frac{g^b}{g^a}) \tag{6}$$

Extending these quotients to all the pixels in I^a and I^b, the set of all the feasible transformations can be computed as $\mathbf{T} = \{\left(r_i^b/r_j^a, g_i^b/g_j^a\right) \mid (r_j^a, g_j^a) \in I^a \text{ and } (r_i^b, g_i^b) \in I^b\}$, where $j, i = 1, \ldots, N$ correspond to the jth and i^{th} pixels of images I^a and I^b, respectively. N is the total number of pixels of an image. $\mathbf{T}$

[2] What is *canonic* is a convenience, so any illuminant could be the canonic one.

could be further constrained if any extra knowledge about surface or illuminant colour were available.

Whether this is the case or not, the key idea is that *the more proper a mapping is, the more occurrences must exist in* $\mathbf{T}$. Once the set $\mathbf{T}$ and its histogram $\mathcal{H}(\mathbf{T})$ are obtained, the probability of a certain $\mathcal{T} \in \mathbf{T}$, $Pr(\mathcal{T} \mid I^a, I^b)$, can be estimated as the relative frequency of the bin corresponding to $\mathcal{T}$ from the histogram $\mathcal{H}(\mathbf{T})$. This way, a likelihood function depending of $\mathcal{T}$ could be defined as:

$$\mathcal{L}(\mathcal{T} \mid I^a, I^b) = log(Pr(\mathcal{T} \mid I^a, I^b)),\ \mathcal{T} \in \mathbf{T} \tag{7}$$

where, according to Eq. (5) and (7), the most likely mapping $\mathcal{T}_0$ would correspond to the bin of highest relative frequency in $\mathcal{H}(\mathbf{T})$, fulfilling the idea that *the most appropriate mapping must have the most occurrences.*

Unfortunately, the previous approach needs a large number of computations –$O(N^2)$– to build the set $\mathbf{T}$ and resources to store it. To alleviate those computations, a far better approach is the use of image histograms rather than pixels.

It is possible to construct the histogram of mappings $\mathcal{H}(\mathbf{T})$ and to estimate the probability $Pr(\mathcal{T} \mid I^a, I^b)$ by means of the chromaticity histograms $H^a = \mathcal{H}(I^a)$ and $H^b = \mathcal{H}(I^b)$ of images I^a and I^b, respectively. The relative frequency of each bin in $\mathcal{H}(\mathbf{T})$ is the summation of the frequencies of each pair of chromaticities giving rise to a certain mapping $\mathcal{T}$ by means of Eq. (6):

$$Pr(\mathcal{T} \mid I^a, I^b) = \sum_{\mathcal{T} \cap \mathbf{T}} Pr((\frac{r^b}{r^a}, \frac{g^b}{g^a}) \mid I^a, I^b),\ \mathcal{T} \in \mathbf{T} \tag{8}$$

where $\mathcal{T} \cap \mathbf{T} = \{(r^a, g^a) \in I^a \text{ and } (r^b, g^b) \in I^b \mid \mathcal{T} = (\frac{r^b}{r^a}, \frac{g^b}{g^a})\}$. The probability of every element of $\mathcal{T} \cap \mathbf{T}$ is:

$$Pr((\frac{r^b}{r^a}, \frac{g^b}{g^a}) \mid I^a, I^b) = Pr((r^a, g^a) \mid I^a) \cdot Pr((r^b, g^b) \mid I^b) \tag{9}$$

where $Pr((r^a, g^a) \mid I^a)$ and $Pr((r^b, g^b) \mid I^b)$ are the relative frequencies of chromaticities $(r^a, g^a) \in H^a$ and $(r^b, g^b) \in H^b$, respectively.

This procedure greatly reduces the number of computations to less than $O(M^2)$, where M is the number of bins in a histogram, since only non-zero bins are taken into account and $M << N$, considering that $M \sim O(10^3)$ and $N \sim O(10^5)$. We average the set of mappings falling into a bin to get a better estimate of the mapping corresponding to that bin. The number of histogram bins affects the precision of the mapping estimate only if it is too low.

In practice, some spurious peaks may appear due to the accumulation of noisy or to redundant mappings which might mislead the algorithm. Hence, the intersection-measure of Eq. (4) is used to evaluate the performance of each particular mapping since it globally measures the colour resemblance between two images. The better a mapping is, the higher the histogram intersection is.

Therefore, to improve the chances of obtaining a more precise estimate, we newly define the likelihood function combining both Eq. (4) and Eq. (8) as:

$$\mathcal{L}(\mathcal{T} \mid I^a, I^b) = log(\cap(\mathcal{T}(H^a), H^b) \cdot Pr(\mathcal{T} \mid I^a, I^b)),\ \mathcal{T} \in \mathbf{T} \tag{10}$$

where $\mathcal{T}(H) = \mathcal{T}(\mathcal{H}(I))$ is the transformation of a histogram $\mathcal{H}(I)$ by $\mathcal{T}$, which is not as straightforward as mapping an image I since the discrete nature of histograms and the absence of *one–to–one* correspondence among histogram bins generally produce gaps and bin overlays.

To avoid gaps, the procedure begins from the bins in $\mathcal{T}(H)$ and computes their corresponding bin in H using the inverse $\mathcal{T}^{-1}$. Bin overlays mean that some bins may have been repeatedly counted. Hence, $\mathcal{T}(H)$ must be normalised.

Furthermore, the previous likelihood function is only computed on a limited set of mappings to reduce the computational burden. Only those of higher probability $Pr(\mathcal{T} \mid I^a, I^b)$ are checked by Eq. (4) to be a good mapping. Finally, the most likely transformation $\mathcal{T}_0$ is selected, as stated in Eq. (5).

6 Results

In this section, we perform the previous algorithm in a set of 220 images coming from 20 different colourful objects taken under 11 different illuminants[3]. We show the set of objects in Fig. 1. We have chosen this image database to benchmark the algorithm since it presents a wide range of both real objects and lights.

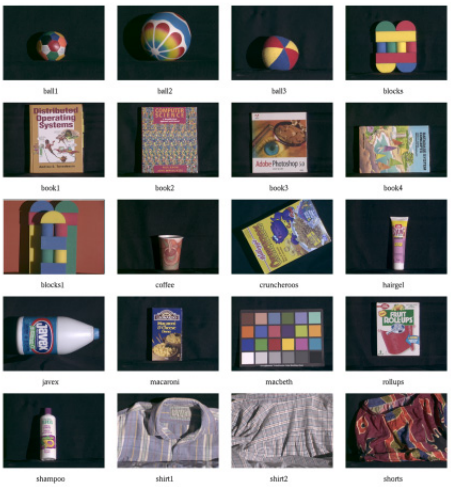

Fig. 1. Set of objects.

The experiment consists, for each object, in taking in turn each illuminant as the canonic while computing colour mappings from the rest of illuminants onto the canonic. We measure the performance of each computed mapping by

[3] These sets belong to the public database of the *Computational Vision Lab* at the *Simon Fraser University* located at URL: `http://www.cs.sfu.ca/~colour/`.

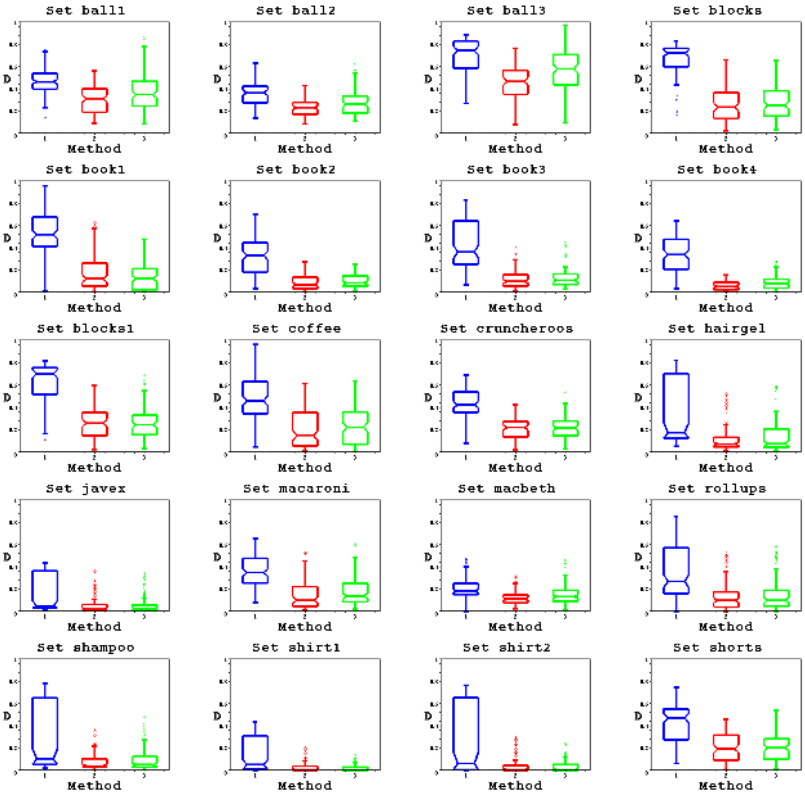

Fig. 2. Boxplots of the results per object set and method.

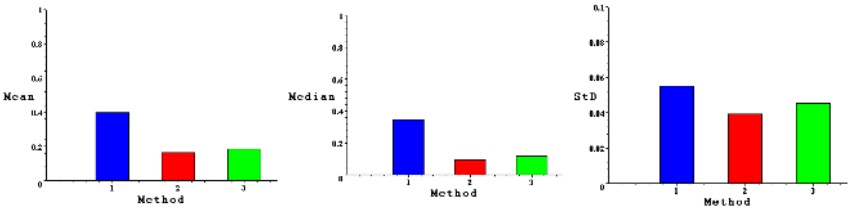

Fig. 3. Mean, median and standard deviation of the results per method.

comparing the chromaticity histogram of the transformed image with that of the canonic by means of the *distance* between histograms defined using the Eq. (4).

To compare the results with a *ground truth*, we directly calculate the colour transformation out of the *real illuminant colour*. That is, if two illuminants E^a and E^b have colours (R^a, G^a, B^a) and (R^b, G^b, B^b), respectively, then the change from E^a onto E^b is $(\alpha_0, \beta_0) = \left(r^b/r^a, g^b/g^a\right)$, where (r^a, g^a) and (r^b, g^b) are the illuminant chromaticities, according to Eq. (2). This information was measured at the same time as the image database using a diffuse white surface at the scene

[4,5]. The mappings computed in this way are the limit of performance of any colour constancy algorithm using Eq. (3) to model the colour change.

In Fig. 2, for each object, we plot the histogram distances into three sets. In *blue* –(1)–, when no colour correction is carried out. These of using the real illuminant colour in *red* –(2)– and those of our algorithm in *green* –(3)–. Each set is depicted as a *boxplot*, where the three quartiles form the box with a notch at the median, and the maximum and the minimum are the extrema of the bars. It can be appreciated in all the sets that there has been a reduction in the colour difference with regard to not doing any colour correction –blue sets–. And more importantly, the performance of the algorithm is close to that of the mappings computed from the real illuminant colour –red sets–.

Table 1. Global results per method.

Method	**Blue** (1)	**Red** (2)	**Green** (3)
Mean	0.398	0.166	0.186
Median	0.346	0.097	0.118
St. Dev.	0.055	0.040	0.046

To globally describe the performance, we put together the former results and compute the mean, the median and the standard deviation for each category, as can be seen in Table 1 and Fig. 3. Thus, we can state that globally the colour difference has decreased from 0.394 to 0.186, a percentage reduction of 56.6%. Secondly, these values are close to those obtained when using the true illuminant colour, i.e., a distance of 0.166 and a percentage reduction of 60.6%.

7 Conclusions

The present paper shows a procedure based on image raw data that, in a framework where the colour change is modelled as a $2D$ diagonal matrix, finds a colour mapping so that the image colours can be rendered as seen under a canonic illumination reducing their dependence on the light conditions. The performance of the algorithm was checked with a wide range of real images of objects under different illuminants. The results show its performance is comparable to the case of knowing the real illuminant colour. Finally, we can state our algorithm improves colour images since stabilises pixel colours in front of illuminant changes.

References

1. Finlayson, G., Hordley, S., Hubel, P.: Colour by correlation: A simple, unifying framework for colour constancy. IEEE Trans. on Pattern Analysis and Machine Intelligence **23** (2001) 1209–1221

2. Forsyth, D.: A novel algorithm for color constancy. Int. Journal of Computer Vision **5** (1990) 5–36
3. Finlayson, G.: Color in perspective. IEEE Trans. on Pattern Analysis and Machine Intelligence **18** (1996) 1034–1038
4. Barnard, K., Cardei, V., Funt, B.: A comparison of computational colour constancy algorithms: Part one: Methodology and experiments with synthesized data. IEEE Trans. on Image Processing **11** (2002) 972–983
5. Barnard, K., Martin, L., Coath, A., Funt, B.: A comparison of computational colour constancy algorithms: Part two: Experiments with image data. IEEE Trans. on Image Processing **11** (2002) 985–996
6. Sapiro, G.: Color and illuminant voting. IEEE Trans. on Pattern Analysis and Machine Intelligence **21** (1999) 1210–1215
7. Funt, B., Barnard, K., Martin, L.: Is colour constancy good enough? In: Proc. 5th European Conference Computer Vision. (1998) 445–459
8. Finlayson, G., Hordley, S.: Improving gamut mapping color constancy. IEEE Trans. on Image Processing **9** (2000) 1774–1783
9. Swain, M., Ballard, D.: Indexing via color histograms. In: Proc. Int. Conf. on Computer Vision. (1990) 390–393
10. Schiele, B., Crowley, J.: Object recognition using multidimensional receptive field histograms. In: Proc. Int. European Conf. on Computer Vision. (1996) 610–619