Colour by Correlation in a Three-Dimensional Colour Space

Kobus Barnard[1], Lindsay Martin[2], and Brian Funt[2]

[1]Computer Science Division, U. C. Berkeley, Berkeley, CA, 94720
email: kobus@cs.berkeley.edu

[2]Department of Computing Science, Simon Fraser University,
8888 University Drive, Burnaby, BC, Canada, V5A 1S6
email: {funt, colour}@cs.sfu.ca

Abstract. We improve the promising Colour by Correlation method for computational colour constancy by modifying it to work in a three dimensional colour space. The previous version of the algorithm uses only the chromaticity of the input, and thus cannot make use of the information inherent in the pixel brightness which previous work suggests is useful. We develop the algorithm for the Mondrian world (matte surfaces), the Mondrian world with fluorescent surfaces, and the Mondrian world with specularities. We test the new algorithm on synthetic data, and on a data set of 321 carefully calibrated images. We find that on the synthetic data, the new algorithm significantly out-performs all other colour constancy algorithms. In the case of image data, the results are also promising. The new algorithm does significantly better than its chromaticity counter-part, and its performance approaches that of the best algorithms. Since the research into the method is still young, we are hopeful that the performance gap between the real and synthetic case can be narrowed.

1 Introduction

The image recorded by a camera depends on three factors: The physical content of the scene, the illumination incident on the scene, and the characteristics of the camera. It is the goal of computational colour constancy to identify, separate, or mitigate the effects of these factors. Doing so has applications in computer vision and image reproduction.

In this paper we improve the promising Colour by Correlation [1] method for computational colour constancy by casting it in a three-dimensional colour space. Colour by Correlation is promising because it can combine more sources of information than the related state-of-the-art gamut-mapping approach [2-6], and thus it is potentially even more effective. The extra source of information that becomes available is the statistical distribution of expected surfaces and illuminants, and how their interactions affect the expected distribution of the observed camera responses. However, the current version of Colour by Correlation uses only chromaticity

D. Vernon (Ed.): ECCV 2000, LNCS 1842, pp. 375–389, 2000.
Springer-Verlag Berlin Heidelberg 2000

information. Since the algorithm works in a chromaticity space, it has no way of using any information that might be available in the pixel brightness. However, previous work has shown that it is beneficial to use the pixel brightness information [4, 6, 7], even if only the illuminant chromaticity is being sought. Thus it is natural to modify Colour by Correlation so that it can also use this information.

In this paper we provide details of the changes required to have this algorithm work in a three-dimensional colour space. The modified algorithm naturally allows extensions for both metallic and non-metallic specularities, in analogy with previous work on gamut-mapping for these conditions [7, 8]. In addition, the algorithm can deal with fluorescent surfaces, much like the two-dimensional version, as described in [7, 9]. We begin with a brief discussion of the two-dimensional Colour by Correlation algorithm.

2 Colour by Correlation in Chromaticity Space

Finlayson et el. introduced Colour by Correlation [1] as an improvement on the Colour in Perspective method [3]. The basic idea of Colour by Correlation is to pre-compute a correlation matrix which describes how compatible proposed illuminants are with the occurrence of image chromaticities. Each row in the matrix corresponds to a different training illuminant. The matrix columns correspond to possible chromaticity ranges resulting from a discretization of (r,g) space, ordered in any convenient manner. Two versions of Colour by Correlation are described in [1]. In the first version, the elements of the correlation matrix corresponding to a given illuminant are computed as follows: First, the (r,g) chromaticities of the reflectances in the training set under that illuminant are computed using the camera sensors. Then the convex hull of these chromaticities is found, and all chromaticity bins within the hull are identified as being compatible with the given illuminant. Finally, all entries in the row for the given illuminant corresponding to compatible chromaticities are set to one, and all other elements in that row are set to zero.

To estimate the illuminant chromaticity, the correlation matrix is multiplied by a vector whose elements correspond to the ordering of (r,g) used in the correlation matrix. The elements of this vector are set to one if the corresponding chromaticity occurred in the image, and zero otherwise. The i'th element of the resulting vector is then the number of chromaticities which are consistent with the illuminant. Under ideal circumstances, all chromaticities in the image will be consistent with the actual illuminant, and that illuminant will therefore have maximal correlation. As is the case with gamut-mapping methods, it is possible to have more than one plausible illuminant, and in our implementation we use the average of all candidates close to the maximum. We label this algorithm C-by-C-01 in the results.

In the second version of Colour by Correlation, the correlation matrix is set up to compute the probability that the observed chromaticities are due to each of the training illuminants. The best illuminant can then be chosen using a maximum likelihood estimate, or using some other estimate as discussed below. To compute the correlation matrix, the set of (r,g) for each illuminant using our database of surface reflectances is again found. The frequency of occurrence of each discrete (r,g) is then recorded. If additional information about the frequency of occurrence of these reflectances is available, then the frequency counts are weighted accordingly. However, since such a distribution is not readily available for the real world, in our implementation we simply use uniform statistics. The same applies for the illuminant data set. The counts are proportional to the probability that a given (r,g) would be observed, given the specific illuminant. The logarithms of these probabilities for a given illuminant are stored in a corresponding row of the correlation matrix. The application of the correlation matrix, which is done exactly as described above, now computes the logarithm of the posterior distribution.

This computation of the posterior distribution is a simple application of Bayes's rule. Specifically, the probability that the scene illuminant is I, given a collection of observed chromaticities C, is given by:

$$P(I|C) = \frac{P(C|I)P(I)}{P(C)} \tag{1}$$

Since we are assuming uniform priors for I, and since P(C) is a normalization which is not of interest, this reduces to:

$$P(I|C) \quad P(C|I) \tag{2}$$

Assuming that the observed chromaticities are independent, P(C|I) itself is the product of the probabilities of observing the individual chromaticities c, given the illuminant I:

$$P(C|I) = \underset{c\quad C}{} P(c|I) \tag{3}$$

Taking logarithms gives:

$$log(P(C|I)) = \underset{c\quad C}{} log(P(c|I)) \tag{4}$$

This final quantity is exactly what is computed by the application of the correlation matrix to the vector of chromaticity occurrences. Specifically, the i'th element of the resulting vector is the logarithm of the posterior probability for the i'th illuminant.

The method described so far will work fine on synthetic data, provided that the test illuminant is among the training illuminants. However, once we apply the method to the real world, there are several potential problems. First, due to noise, and other sources of mismatches between the model and the real world, an observed set of chromaticities can yield zero probability for all illuminants, even if the illuminant, or a similar one, is in the training set. Second, the illumination may be a combination of two illuminants, such as an arbitrary mix of direct sunlight and blue sky, and ideally we would like the method to give an intermediate answer. We deal with these problems as follows. First, as described below, we ensure that our illuminant set covers (r,g) space, so that there is always a possible illuminant not too far from the actual. Second, as we build the correlation matrices, we smooth the frequency distribution of observed (r,g) with a Gaussian filter. This ensures that there are no holes in the distribution, and compensates for noise.

The final step is to choose an answer, given the posterior probability distribution, an example of which is shown in Figure 1. The original work [1] mentions three choices: The maximum likelihood, mean likelihood, or local area mean, introduced in [10]. That work discusses these methods in detail with respect to a related colour constancy algorithm, where they are referred to as the MAP, MMSE, and MLM estimators, respectively. We will adopt this notation here. The MAP estimate is simply the illuminant which has the maximum posterior probability. To compute the MMSE estimate of the chromaticity estimate we take the average (r,g) weighted by the posterior distribution. The MLM estimator is computed by convolving the posterior distribution with a Gaussian mask, and then finding the maximum. In general, one would like to choose the particular Gaussian mask which minimizes the error of some specific task. Unfortunately, the bulk of our results are not of much help here, as they are based on RMS error, and thus we already know that the MMSE method will work better. Thus we provide results only for the computationally cheaper MAP method, and least-squares optimal MMSE method.

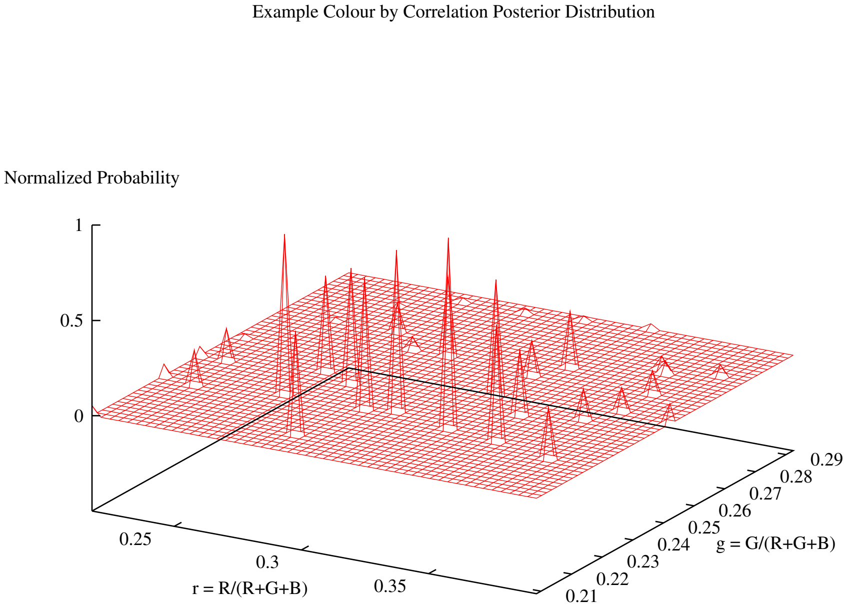

Fig. 1: An example posterior distribution, showing the probabilities that the illuminant in the training set with chromaticity (r,g) explains the observed data produced from a randomly selected illuminant and 8 randomly selected surface reflectances.

3 The Extension to Three Dimensions

We now consider what a three-dimensional analog to the two-dimensional algorithm entails. In the two-dimensional case, the observed two-dimensional descriptors (chromaticities) were tested against possible theories of the distributions of those descriptors, each theory corresponding to one of the illuminants in the training set. In the three-dimensional version, we wish to do the same with three-dimensional descriptors. However, we run into the problem that the brightness of the illuminant changes the observed values. In effect, not only does each illuminant produce a theory, but every brightness level of each illuminant produces a theory. Thus we must attempt to match over possible illuminant brightnesses, as well as over illuminants. This leads to several problems.

The first problem is that, a priori, the illuminant can have any non-negative brightness. This is different than chromaticity which is naturally constrained, and thus easily discretized. To solve this problem we propose making an initial estimate of the illuminant brightness using some other means. For this, we found a grey world type estimate to be adequate. Specifically, we compute the average of R+G+B over the image pixels, and multiply the result by a factor chosen to give the best estimate when the same procedure was applied to synthetic data. The value of the factor used for the experiments was 4.3. For some applications the median or other estimation method may very well be superior to the average, being more robust in the face of outliers. However, on our data, the mean worked better than the median.

Having determined an estimate of the illuminant brightness, we reason that it is unlikely to be wrong by more than a factor of k=3. Now, on the assumption that the illuminant brightness is between L/k and kL, we discretize this range on a logarithmic scale, giving us a finite number of possible illuminant brightness theories. We verified that the specific choice of k=3 gives the same results as providing the algorithm with the exact illuminant brightness in its place. Clearly, a larger or smaller value could be more appropriate, depending on circumstances.

The next problem that we faced is that the literal analogy of the two-dimensional method leads to unmanageably large correlation matrices. There are two contributions to the increase in size. First, the matrix row length increases because of the added descriptor—the rows now store linearized versions of three-dimensional arrays where two-dimensional arrays were previously stored. Second, the strategy of considering each illuminant at each brightness level implies, a priori that we would further need to increase the number of rows by a factor of the brightness resolution because now we would need a row for every brightness of every illuminant. The combined effect of these two factors lead to correlation matrices which are simply too large.

Fortunately, the second increase in size is not necessary. We instead loop over the possible brightnesses, and simply scale the input by an appropriate amount each time. Conceptually, this amounts to the same thing as having a correlation matrix row for each illuminant at each brightness. In practice, however, it leads to a subtle problem due to the discretization. If we consider the alternative of building a correlation matrix row for each possible brightness, we see that as the proposed illuminant brightness decreases, the bins become proportionally more populated. For example, if the illuminant brightness is halved, then the same data is put into half as many bins. Now the algorithm proceeds by summing terms for each observed response. The terms are the logarithms of quantities proportional to the probability that a proposed illuminant occurs with the observed response. If we consider each term to be negative, then we see that decreasing the number of terms increases the sum. Since we are trying to maximize this sum, the algorithm will favor low brightness values, because these tend to put the observations into as few bins as possible, leading to fewer terms. This is an artifact of the discretization, and clearly is not wanted.

We discuss two possible approaches to deal with this problem. First, we can allow duplicate entries into the bins. In order to minimize the effect of duplicates present in the data, the data could be pre-processed to remove all initial duplicates. This method gives excellent results when used in conjunction with generated data. However, we have not had equal success with image data.

A second approach to the above problem is to compensate for the discretization problem directly. We reason as follows: If we were to have constructed correlation matrices for each brightness level, then the frequency counts placed in the bins to compute the probabilities would have been roughly inversely proportional to the brightness. Thus the probabilities themselves would be inversely proportional to the brightness, and to obtain a fair estimate, we need to divide each probability in the product by a value proportional to the brightness. In the log representation, this means that we subtract the log of the brightness times the number of occupied bins. This method also yields excellent results when used in conjunction with generated data. More importantly, the results using image data are also promising. We feel that this algorithm can be substantially improved, and one of the key areas for further study is this discretization problem.

We now consider the choice of three-dimensional descriptors. One natural choice is RGB. However, given the asymmetry of the role of brightness and chromaticity in computational colour constancy, we feel that a better choice is to use (r,g) chromaticity, together with R+G+B. This has several advantages over using RGB. First, due to the above mentioned asymmetry, we may wish to use different resolutions for the chromaticity and the brightness. Second, this choice provides conceptual clarity, in that our method then subsumes the two-dimensional version as the sub-case where there is only one division for the R+G+B coordinate. Finally, we find it convenient to have only one coordinate which can be arbitrarily large.

The algorithm as described is easily extended to model complex physical scenes. For example, we can model fluorescent surfaces, as already done in the two-dimensional case in [9], and we can model specular surfaces, including metallic ones, as was done for gamut-mapping algorithms in [8]. The Colour by Correlation method has an advantage over the gamut-mapping methods in that the expected frequency of occurrence of these phenomena can be modeled. Unfortunately we currently do not know these statistics for the real world, and hence it is difficult to exploit this in the case of image data. Nevertheless, doing so holds promise for the future because if some estimate of the likelihood of occurrence of these classes of surfaces could be made, then three-dimensional Colour by Correlation would be more robust than the extended versions of three-dimensional gamut mapping. This is due to the fact that it can allow for the possibility of, for example, metallic surfaces, while compensating for the fact that there is only a low likelihood that such surfaces are present. Gamut-mapping, on the other hand, is forced to use uniform statistics.

4 Algorithm Summary

We now provide a summary of the method. The implementation of the algorithm consists of two parts. First the correlation matrices are built, and then these matrices are used to perform colour constancy. The first stage is a one time operation, and consequently, we are not concerned about resource usage. We begin with a data set of illuminant and reflectance spectra. Ideally, we would know the expected frequency of occurrence of these surfaces and illuminants, but since we do not, we assume that there are all equally likely. For surface reflectances we used a set of 1995 spectra compiled from several sources. These surfaces included the 24 Macbeth colour checker patches, 1269 Munsell chips, 120 Dupont paint chips, 170 natural objects, the 350 surfaces in Krinov data set [11], and 57 additional surfaces measured by ourselves.

The choice of illuminant spectra must be made with more care, as the algorithms are sensitive to the statistics of the occurrence of the illuminants in the training set. We feel that it is best to have the training and testing sets both at least roughly uniformly distributed in (r,g) space. To obtain the appropriate illuminant sets, we first selected 11 sources to be used for the image data. These were selected to span the range of chromaticities of common natural and man made illuminants as best as possible, while bearing in mind the other considerations of stability over time, spectral nearness to common illuminants, and physical suitability. To create the illuminant set used for training, we divided (r,g) space into cells 0.02 units wide, and placed the 11 illuminants described above into the appropriate cells. We then added illumination spectra from a second set of 97, provided that their chromaticity bins were not yet occupied. This second set consisted of additional sources, including a number of illumination spectra measured in and around our university campus. Then, to obtain the desired density of coverage, we used random linear combinations of spectra from the two sets. This is justified because illumination is often the blending of light from two or more sources. Finally, to produce the illuminant set for testing, we followed the same procedure, but filled the space 4 times more densely.

Given the illuminant and reflectance data sets, we generate the required sensor responses using estimates of our camera sensitivity functions, determined as described in [12]. Thus to apply the algorithms to image data, we must first map the data into an appropriate linear space (also described in [12]), and perform other adjustments to compensate for the nature of the imaging process as described more fully in [7].

We use the colour space (r,g,L) where L=R+G+B, r=R/L, and g=G/L. We divide the space into discrete bins. The resolution of the discretization of the three components do not need to be equal. There is no reason to make the first two different from each other, but, as discussed above, it can be advantageous to use a different value for the third. For all experiments we used 50 divisions for (r,g), which is consistent with the discretization resolution used in this work for two-dimensional Colour by Correlation. When specularities are added, as discussed shortly, the overall number of bins required for L increases. We express the resolution for L in terms of the number of bins devoted to matte reflection. For the experiments with generated data, we generally used a value for L which also leads to 50 divisions for matte reflection, but this is likely higher resolution than is necessary, and in fact, preliminary results indicate that a smaller number is likely better. Thus for the image data experiments, we used 25 divisions.

Given a discretization of colour space, we then map this space into a vector, using any convenient method. We note that since half of the values in (r,g) are impossible, a more compact representation can be used than the naive one. Since the three-dimensional correlation matrices are large, we make use of this observation to reduce storage requirements.

Thus we form a two-dimensional array, where each row is the above linearization of colour space, and the rows correspond to training illuminants. We then build up the matrix by computing, for each illuminant, the RGB of the reflectances in our database. We then compute the frequency of occurrence of the colours within each discrete cell in our colour space. These frequencies are proportional to the probabilities; they can be converted to probabilities by dividing by the total number of surfaces. Finally, for convenience, we store the logarithm of the probabilities.

To add fluorescent surfaces, we compute the responses which occur for each illuminant using the model described in [9]. The relative expected frequency of such surfaces is expressed by simply adjusting the frequency counts during the construction of the correlation matrix. In our experiments with fluorescent surfaces, we set the frequency of occurrence of any fluorescent surface to be about 20%. Since we only model 9 such surfaces, the frequency of occurrence of each was set to be 50 times that of each of the surfaces in the set of roughly 2000 reflectances.

We can also model specular reflection. This is a little more involved than handling fluorescent surfaces. First, we need to extend the number of bins in the L direction, as specular reflection is modeled as reflection which exceeds that of a perfect white. Then, we must model both the relative frequency of occurrence of specularities, as well as the frequency of each degree of specular reflection. It should be clear that the model can well be used with metallic specularities, an analogy with the work in [8], but we do not study those here.

The second part of the algorithm is the use of the above matrix for colour constancy. We wish to compute the likelihood of an illuminant-brightness combination. We loop over the possible illuminants, and then the possible brightnesses, to obtain an estimate for each combination. To compute a maximum likelihood estimate, we simple keep track of the maximum value reached and the corresponding illuminant and brightness. However, since we are also interested in studying the mean likelihood estimate, we store all values in order to make that estimate from them as a second step. We now provide additional details of the likelihood calculation.

Again, for each proposed illuminant, we loop over a discretization of possible brightnesses on a log scale. We remind the reader that the range is set by an initial rough estimate of the brightness. We generally use 101 brightness levels; This is likely excessive. For each proposed brightness level, we scale the input accordingly, using the brightness of the proposed illuminant. We then form a vector representing the observed scene assuming this brightness level. The components of this vector correspond to the linearized form of the discretized colour space.

To compute the entries of this vector we begin by initializing all components to zero. We then compute the corresponding bin for each colour observed in the scene. If we are using the first method to solve the discretization problem discussed in the previous section, then we store a count of the number of colours falling in each bin. Alternatively, if we are using the second method we simply note the presence of the colour with a count of one. All bins corresponding to colours not observed remain zero.

To obtain the likelihood of the proposed illuminant-brightness combination, we simply take the dot product of the computed vector with the row in the correlation matrix corresponding to the proposed illuminant. Since the values stored in the correlation matrix are the logarithms of probabilities, the dot product computes the logarithm of the product of the probability contributions for each observation (see Equation 5). If we are using the second method to compensate for the discretization problem discussed above, we then adjust the result by subtracting the logarithm of the proposed brightness times the count of the occupied bins.

5 Experiments

We tested the new algorithm on generated and image data. For the first two sets of results on generated data we used the first method of dealing with the discretization problem. For the third set of results with generated data, as well as for the image data results, we used the second method. For the experiments with generated data we used the set of test illuminants describe above. We remind the reader that both the training illuminant set and the test illuminant set were designed to systematically cover (r,g) space, but the test illuminant set covered that space four times more densely.

Figure 2 shows the chromaticity performance of the method using both maximum likelihood and mean likelihood estimation as a function of the number of surfaces in the generated scenes. We also provide the results for corresponding two-dimensional versions of the algorithms, as well as the results for two gamut mapping methods—the original method [2], labeled CRULE-MV in the results, and a new variant introduced in [7], and labeled as ND-ECRULE-SCWIA-12 to maintain consistency with the literature. This later algorithm has been shown to be comparable to the best computational colour constancy algorithms over a wide range of conditions, and thus provides a good counter-point to the three-dimensional version of Colour by Correlation.

The results clearly show that the new method excels when tested on data with similar statistics to that used for training. The error drops to the minimum possible given the discretization when only 16 surfaces are used, clearly out-performing the other algorithms.

For the second experiment we looked at the performance of the method under a variety of simulated conditions. We developed three-dimensional Colour by Correlation algorithms for fluorescent surfaces and specular reflection, and tested these algorithms, along with the version for matte surfaces, under the conditions of matte surfaces, the matte and fluorescent surfaces, and matte surfaces with specularities. The test conditions were similar to the training conditions, especially in the fluorescent case. In the specular case, the rough discretization of specular reflection used for creating the correlation matrices only approximates what the algorithms were tested against.

Again the results, shown in Table 2, are very promising. As expected, the algorithms do very well when tested under the conditions they were designed for. More promising is that the algorithms seem quite robust to the absence of these conditions. For example, adding fluorescent capability reduced the error from 0.060 to 0.022 when fluorescent surfaces were present, but using the algorithm with fluorescent capability in the case of standard surfaces incurred minimal penalty (0.026 instead of 0.025). (These figures are using the MMSE estimator). In general, it is clear that for generated data, these algorithms perform better than any of the others which are listed in Table 2.

For the third experiment, we tested the second method of dealing with the discretization problem discussed above. The results are shown in Table 3. We also include additional comparison algorithms in this table. Again, the new methods do significantly better than the next best strategy, which is the ND-ECRULE-SCWIA-12 algorithm developed in [7]. Using the MMSE estimator, the three-dimensional Colour by Correlation error is 0.24; using the MAP estimator it is 0.29; and using ND-ECRULE-SCWIA-12 it is 0.39. The results also indicate that the second method of dealing with our discretization problem may be better than the first, as the errors are lower, but we note that the difference can also easily be explained by random fluctuations within our error estimates.

We also tested the method on a data set of 321 images. These images were of 30 scenes under 11 different illuminants (9 were culled due to problems). The images are described in more detail in [7]. As mentioned above, we have not yet been able to significantly improve upon the two-dimensional method using the first method of dealing with our discretization problem. Using the second method, however, the results, shown in Table 4, are very promising. We see that the error of the new method (0.46) is approaching that of the best performer listed, namely ECRULE-SCWIA-12 (0.37) and CRULE-MV (0.045). This error is significantly less than that for the two-dimensional counter-part (0.077).

6 Conclusion

We have shown how to modify the Colour by Correlation algorithm to work in a three-dimensional colour space. This was motivated by the observations that the correlation method is more powerful than the chromaticity gamut-mapping method due to the use of statistical information, and that three-dimensional gamut mapping is also more effective than its chromaticity counterpart due to the use of information inherent in the pixel brightness. We wished to combine these two features into one algorithm The resulting algorithm is also suitable for modification to deal with complex physical surfaces such as fluorescence, and standard and metallic specularities. In fact, if the frequency of occurrence of these surfaces could be estimated, then this algorithm could also exploit these statistics. In summary, this algorithm is able to use more sources of information than any other, and thus is potentially the most powerful colour constancy method.

We tested a number of versions of the algorithm on synthetic and image data. The results with synthetic data are excellent, and it seems that these algorithms are in fact the best performers in this situation. The results with image data are also promising. In this case the new methods perform significantly better than their two dimensional counterparts. Currently, however, the performance still lags a little behind the best algorithms for image data. It is quite possible that the performance gap between real and image data can be reduced, as we have only recently begun to study the algorithm in this context. However, previous work [7] has shown that statistical algorithms do tend to shine during synthetic testing, and therefore, we must be cautious not to over-sell the method until the image data performance exceeds that of the current best methods.

Finally we note that the algorithm as described is computationally quite expensive, both in terms of memory use, and CPU time. Since our initial intention was to push the limits of the error performance, we have not addressed ways to speed up the algorithm. If the performance on image data can be made comparable to that for generated data, then an important next step is to consider what can be done in this regard.

7 References

1. G. D. Finlayson, P. H. Hubel, and S. Hordley, Color by Correlation, Proc. *IS&T/SID Fifth Color Imaging Conference: Color Science, Systems and Applications*, pp. 6-11 (1997).
2. D. Forsyth, A novel algorithm for color constancy, *International Journal of Computer Vision*, **5**, pp. 5-36 (1990).
3. G. D. Finlayson, Color in perspective, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**, pp. 1034-1038 (1996).
4. K. Barnard, Computational colour constancy: taking theory into practice, MSc Thesis: Simon Fraser University, School of Computing (1995).
5. G. Finlayson and S. Hordley, A theory of selection for gamut mapping colour constancy, *Proc. IEEE Conference on Computer Vision and Pattern Recognition* (1998).
6. B. Funt, K. Barnard, and L. Martin, Is Colour Constancy Good Enough?, *Proc. 5th European Conference on Computer Vision*, pp. I:445-459 (1998).
7. K. Barnard, Practical colour constancy, Ph.D. Thesis: Simon Fraser University, School of Computing Science (1999), ftp://fas.sfu.ca/pub/cs/theses/1999/KobusBarnardPhD.ps.gz
8. K. Barnard and B. Funt, Color constancy for specular and non-specular surfaces, *Proc. IS&T/SID Seventh Color Imaging Conference: Color Science, Systems and Applications* (1999).
9. K. Barnard, Color constancy with fluorescent surfaces, *Proc. IS&T/SID Seventh Color Imaging Conference: Color Science, Systems and Applications* (1999).
10. D. H. Brainard and W. T. Freeman, Bayesian color constancy, *Journal of the Optical Society of America A*, **14**, pp. 1393-1411 (1997).
11. E. L. Krinov, *Spectral Reflectance Properties of Natural Formations*: National Research Council of Canada, 1947.
12. K. Barnard and B. Funt, Camera calibration for color research, *Proc. Human Vision and Electronic Imaging IV*, pp. 576-585 (1999).

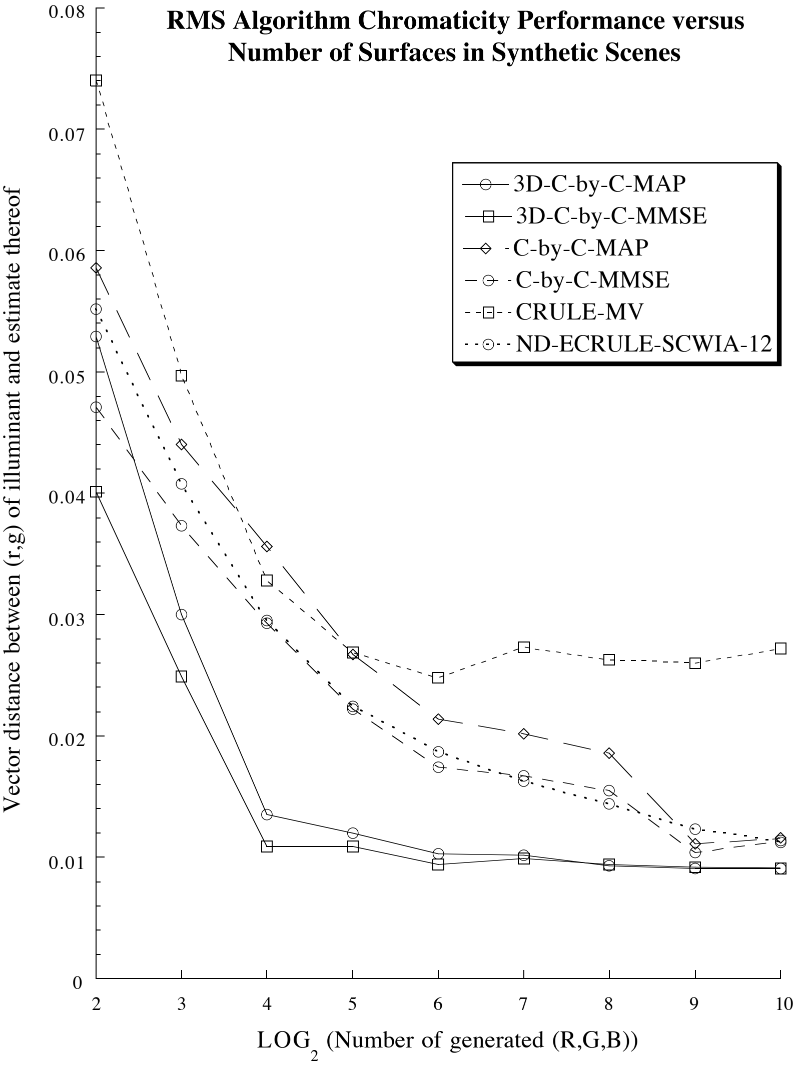

Fig. 2: The chromaticity performance of the new method as compared to the two-dimensional version of the algorithm and two gamut mapping methods. For both Colour by Correlation methods we provide results using both maximum likelihood (MAP) and mean likelihood (MMSE) estimation.

NOTHING	The result of doing no colour constancy processing
AVE	The illuminant is assumed to be the average of the illuminant data base (normalized), regardless of the input.
MAX	Estimate illuminant by the max RGB in each channel.
GW	Estimate illuminant colour by assuming that image average is the colour of a 50% reflectance
DB-GW	Estimate illuminant colour by assuming that image average is the colour of the average of a reflectance database.
CRULE	Original gamut constraint method described in
ECRULE	CRULE with illumination constraint
MV	Solutions are chosen from the feasible set delivered by the gamut mapping method using the max volume heuristic
ICA	Solutions are chosen from the feasible set delivered by the gamut mapping method by averaging.
SCWIA	Solutions are the average over feasible illuminant chromaticities, weighted by a function chosen to emphasize illuminants with chromaticities around the maximum volume solution, as described in [7].
ND	Gamut mapping algorithm is extended to reduce diagonal model failure as described in [7, 9]
C-by-C-MAP	Colour by Correlation [1], with a Gaussian mask to smooth the correlation matrix and maximum likelihood estimate.
C-by-C-MMSE	Colour by Correlation [1], with a Gaussian mask to smooth the correlation matrix and mean likelihood estimate.
3D-C-by-C-MAP	The Colour by Correlation method for a three-dimensional colour space as developed in this paper, and using the maximum likelihood estimate.
3D-C-by-C-MMSE	The Colour by Correlation method for a three-dimensional colour space as developed in this paper, and using the mean likelihood estimate.
FL	Algorithm is extended for fluorescence. For gamut mapping and two-dimensional Colour by Correlation, this is described in [9]. For 3D-C-by-C, the algorithm is developed in this work.
SPEC	Algorithm is extended for fluorescence. For gamut mapping this is described in [8]. For 3D-C-by-C, the algorithm is developed in this work.

Table 1: Key to the algorithms referred to in the results.

	Synthetic scenes with 8 matte surfaces	Synthetic scenes with 8 matte and fluorescent surfaces	Synthetic scenes with 8 matte and specular surfaces
NOTHING	0.116	0.114	0.110
AVE-ILLUM	0.088	0.086	0.084
GW	0.057	0.116	0.034
DB-GW	0.047	0.092	0.032
MAX	0.066	0.104	0.033
C-by-C-01	0.078	0.071	0.079
C-by-C-MAP	0.044	0.059	0.040
C-by-C-MMSE	0.037	0.048	0.033
3D-C-by-C-MAP	0.030	0.066	0.043
3D-C-by-C-MMSE	0.025	0.060	0.037
FL-3D-C-by-C-MAP	0.033	0.023	*
FL-3D-C-by-C-MMSE	0.026	0.022	*
SPEC-3D-C-by-C-MAP	0.038	*	0.023
SPEC-3D-C-by-C-MMSE	0.032	*	0.017
CRULE-MV	0.050	0.103	0.027
CRULE-AVE	0.061	0.088	0.052
ECRULE-MV	0.045	0.078	0.026
ECRULE-ICA	0.051	0.065	0.045
FL-ECRULE-MV	0.049	0.061	0.027
FL-ECRULE-ICA	0.058	0.051	0.056
SP-ND-ECRULE-MV	0.053	0.085	0.029
SP-ND-ECRULE-ICA	0.047	0.062	0.026

Table 2: Algorithm chromaticity performance under three different conditions of variants of the new methods designed for the various conditions, as well as that for a number of comparison algorithms. For these results, the first method of dealing with our discretization problem was used.

Algorithm	Performance estimating (r,g) chromaticity of the illuminant. (4%)
NOTHING	0.111
AVE-ILLUM	0.083
GW	0.055
DB-GW	0.047
MAX	0.061
C-by-C-01	0.076
C-by-C-MAP	0.043
C-by-C-MMSE	0.035
3D-C-by-C-MAP	0.029
3D-C-by-C-MMSE	0.024
CRULE-MV	0.048
ND-ECRULE-SCWIA-12	0.039

Table 3: Algorithm chromaticity performance in the Mondrian world of the new method (MAP and MMSE), as well as that for a number of comparison algorithms. For these results, the second method of dealing with our discretization problem was used.

Algorithm	Performance estimating (r,g) chromaticity of the illuminant. (4%)
NOTHING	0.125
AVE-ILLUM	0.094
GW	0.106
DB-GW	0.088
MAX	0.062
C-by-C-01	0.075
C-by-C-MAP	0.084
C-by-C-MMSE	0.077
3D-C-by-C-MAP	0.047
3D-C-by-C-MMSE	0.046
CRULE-MV	0.045
ECRULE-SCWIA-12	0.037

Table 4: Algorithm chromaticity performance on 321 images of the new method with two estimators (MAP and MMSE), as well as that for a number of comparison algorithms. For these results, the second method of dealing with our discretization problem was used.