Image Color Constancy Using EM and Cached Statistics

Charles Rosenberg CHUCK+@CS.CMU.EDU
Department of Computer Science, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213 USA

Abstract

Cached statistics are a means of extending the reach of traditional statistical machine learning algorithms into application areas where computational complexity is a limiting factor. Recent work has shown that cached statistics greatly reduce the computational requirements of building a mixture model of a distribution using Expectation-Maximization for a small trade off in model error. This paper describes a method whereby a mixture model built using cached statistics is used as a means of improving the color normalization performance of two standard color constancy algorithms. Color constancy algorithms factor out illumination effects such that normalized pixel color values become an invariant representation of surface reflectance properties. This can improve the performance of machine vision and image database algorithms which use color as a feature. This processing is also important in digital camera and scanning applications where a preferred rendition of a scene is to be realized independently of the lighting conditions at the time of image capture. The details and experimental evaluation of two modified color constancy algorithms which utilize a parametric mixture model are described.

1. Introduction

Statistical machine learning algorithms are an extremely powerful class of techniques for manipulating and analyzing large sets of data. These techniques also have the advantage of being firmly based on probability theory and therefore provide certain guarantees about correctness and convergence properties. Unfortunately, many of these techniques are quite complex computationally and although many applications would benefit by their use, they have not been generally applied because the computational requirements are too great for these applications. Recently techniques, such as those described in Moore (1999), have been developed which allow these machine learning techniques to be applied to large data sets for a greatly reduced computational cost in exchange for a small tradeoff in model error. The goal of this work is to explore an application area which can benefit from the use of machine learning techniques which have typically been deemed too complex.

In machine vision and image database applications color is often an appealing feature to use as a simple means of segmenting or identifying a specific object, as described in Funt (1998). Although color in and of itself is often insufficient to perform such a task reliably it can be used robustly in conjunction with other features or as a means of identifying candidate regions of the image likely to contain the desired object. When used to identify likely candidate regions it can speed up the recognition process immensely as described in Rowley (1998).

This works well when all of the images are captured with a single camera and under uniform illumination conditions. Problems arise when the images to be compared are taken under different illumination conditions and with different cameras. The problem is that the measured red, green, and blue (RGB) pixel values are a function of the original object surface reflectance properties, the properties of the illuminant incident on the object surface, and the properties of the camera sensors. The full spectrum of the light source interacts with the surface properties of the object being imaged and is then filtered down to an RGB triplet by the camera sensors. Obviously methods which attempt to measure the distance between the RGB value of an unidentified color in an image and a target RGB value will not be robust if the image pixel in the unidentified image was captured under different conditions than the target image. What is needed is a method to measure the similarity of two image colors independent of the illumination and sensor conditions under which the image was captured.

Color constancy is a method for determining the surface reflectance properties of objects in a scene regardless of scene illumination and camera characteristics. It is this task which will be explored in this paper. The values generated by image capture devices do not represent the surface reflectance properties of the objects in the scene. The measured pixels values for channel k at location (x,y), denoted as $\rho_k(x,y)$, are the product of the incident illumination $E(\lambda)$, the surface reflectance properties $S(x,y,\lambda)$ and camera sensor channel spectral sensitivity $C_k(\lambda)$ as a function of the wavelength of the incident light λ, integrated over the visible spectrum, ω:

$$\rho_k(x,y) = \int_\omega E(\lambda)S(x,y,\lambda)C_k(\lambda)d\lambda$$

The goal of color constancy is to recover the original surface reflectance properties, $S(x,y,\lambda)$, regardless of the incident illumination $E(\lambda)$. However, since only three camera sensor channels are typically employed, it is not possible to recover the surface reflectance properties as a function of any incident wavelength of light. However, if $E(\lambda)$ can be estimated, then it is possible to determine the sensor response $\rho_k(x,y)$ under some canonical camera model. It has been shown in Finlayson (1993) that with a relatively mild set of assumptions, the previous equation can be collapsed to the following simple form:

$$R = \rho_r(x,y) = \alpha S_r(x,y);\ G = \rho_g(x,y) = \beta S_g(x,y);$$
$$B = \rho_b(x,y) = \gamma S_b(x,y)$$

In this form all that needs to be estimated are the three values α, β, and γ in order to recover surface reflectance information. The accurate estimation of these parameters allows a machine vision application to compute a color feature value which is *invariant* of incident illumination.

Color constancy is not only important for machine vision applications but is also important for digital cameras and scanners. Frequently images are captured under varying lighting conditions. This can often lead to images having an reddish tone if captured under tungsten illumination or a greenish tone if captured under fluorescent lighting. Color constancy can be used as a means of removing those color casts resulting in a more visually pleasing image.

The body of this paper describes and evaluates a pair of color constancy algorithms which use parametric models of the distribution of image pixel color values as a basis for estimating scene illumination.

2. Approach

2.1 Introduction

Many color constancy algorithms (Land, 1977; Maloney, 1986; Gershon, 1987; Finlayson, 1995; Funt, 1996; Finlayson, 1998; Funt, 1999) are discussed in the literature. All of these algorithms consist of two steps: an estimation step and a correction step. In the estimation step an assumption is made about the distribution of observed pixel color values which is in turn used to estimate the scene illuminant properties. In the correction step, the illumination estimate is used to calculate a correction factor which is then applied to the image to render it as it would be if captured under a canonical illuminant. In this work we construct a parametric model of the distribution of the observed sensor color channel responses in the image and use that model as the basis of an improved illuminant estimation for two basic color constancy algorithms.

The two basic color constancy algorithms we will be working with are often referred to as the "gray world" algorithm and "white patch" retinex. Even though these algorithms are exceedingly simple, a recent study by Funt (1998) empirically demonstrated that they often work as well as or better than other more complex algorithms when color is used as the basis for matching objects in a database retrieval task. Also, recent work by Finlayson (1998) and Funt (1999) utilizes these basic algorithms as a component of more complex systems. Therefore any method which improves the performance of these basic algorithms could be used to improve the performance of these more complex systems.

In the gray world algorithm, the assumption is made that the means of each of the three color channels over the entire image form a fixed ratio under the canonical illuminant:

$$\frac{1}{\alpha}\overline{\rho_r} = \overline{S_r},\ \frac{1}{\beta}\overline{\rho_g} = \overline{S_g},\ \frac{1}{\gamma}\overline{\rho_b} = \overline{S_b}$$

The values $\overline{S_r}$, $\overline{S_g}$, $\overline{S_b}$ are determined empirically or based on prior knowledge of the distribution of colors in the images to be corrected. The obvious weakness of this algorithm is that the distribution of colors in an individual image may vary from the desired ratio. For example, a group of images containing the same pair of differently colored objects will have a very different measured color distribution depending on which of the objects appears larger in the scene.

In the white patch retinex algorithm, a simplified version of the retinex algorithm described in Land (1977), the assumption is made that the peak values in each channel represent the maximum possible reflectance of that component of the illuminant, as might occur from a specular reflection in the image or an object originally colored white. Here too, a fixed ratio of colors is assumed, in this case over the maximum of the channel values:

$$\frac{1}{\alpha}\rho_r^{max}(x,y) = S_r^{max},\ \frac{1}{\beta}\rho_g^{max}(x,y) = S_g^{max},\\ \frac{1}{\gamma}\rho_b^{max}(x,y) = S_b^{max}$$

The values S_r^{max}, S_g^{max}, S_b^{max} are also determined empirically or based on prior knowledge. One weakness of this algorithm is that it may be hard to identify the maximum value in a particular color channel given noise and possible clipping of the channel values as would occur if the input reflectance exceeds the input channel dynamic range. The other weakness is that, given the distribution of colors in the scene, the maximum response from a particular channel may not be characteristic of the illuminant color.

2.2 Parametric Model

This work attempts to overcome some of the deficiencies of these basic algorithms through the construction of a parametric model of the distribution of color channel responses in an image and the use of that model as the basis for computation in lieu of the raw pixel values. The parametric model chosen was a mixture of Gaussians each with a diagonal covariance matrix. In this model every three dimensional pixel value (R,G,B) is modeled as being generated by one of the model Gaussians with a hidden indicator variable, z_{ij}, representing which Gaussian generated which pixel. To simplify the equations below we index the pixels in the image by the variable i, ranging from 1 to m where m is the number of pixels in the image, the other variables are defined as follows: the variable j indexes the n Gaussians in the mixture and the variable k indexes the color channels (dimensions), ranging from 1 to 3:

$$Pr(\rho_r(i), \rho_g(i), \rho_b(i) \mid z_{ij}, \mu, \sigma) \sim \\ \left(\prod_{k=1}^{3}(2\pi\sigma_{jk}^2)\right)^{-1/2} \exp\left(-\frac{1}{2}\sum_{k=1}^{3}\frac{(\rho_k(i)-\mu_{jk})^2}{\sigma_{jk}^2}\right)$$

To build the model, we must estimate the Gaussian parameters as well as the expected values of the indicator variables. As is typically done, as per Duda (1973) and Mitchell (1997), we use Expectation-Maximization (EM) to estimate these parameter values. We alternate between expectation and maximization steps as follows:

Expectation step:

$$E[z_{ij}] = \frac{\left(\prod_{k=1}^{3}\sigma_{jk}^2\right)^{-1/2}\exp\left(-\frac{1}{2}\sum_{k=1}^{3}\frac{(\rho_k(i)-\mu_{jk})^2}{\sigma_{jk}^2}\right)}{\sum_{c=1}^{n}\left(\prod_{k=1}^{3}\sigma_{ck}^2\right)^{-1/2}\exp\left(-\frac{1}{2}\sum_{k=1}^{3}\frac{(\rho_k(i)-\mu_{ck})^2}{\sigma_{ck}^2}\right)}$$

Maximization step:

$$\mu_{jk} \leftarrow \frac{\sum_{i=1}^{m}E[z_{ij}]\rho_k(i)}{\sum_{i=1}^{m}E[z_{ij}]}$$

$$\sigma_{jk} \leftarrow \frac{\sum_{i=1}^{m}E[z_{ij}]\rho_k^2(i) - \left(\sum_{i=1}^{m}E[z_{ij}]\rho_k(i)\right)^2}{\sum_{i=1}^{m}E[z_{ij}]}$$

2.3 Modified Algorithms

Once the parameters of the mixture model have been estimated, they are used to implement modified versions of the two basic color constancy algorithms described previously.

In the gray world algorithm the major deficiency was an unbalanced distribution of colors in the image. In the modified version of the algorithm, which we call gray world EM, we use the estimated "centers" of the Gaussians instead of the pixel values of the image as the values we are trying to balance to a predetermined ratio. The modified version of the gray world algorithm becomes:

$$\frac{1}{\alpha\cdot n}\sum_{j=1}^{n}\mu_{jr} = \overline{S_r},\ \frac{1}{\beta\cdot n}\sum_{j=1}^{n}\mu_{jg} = \overline{S_g},\\ \frac{1}{\gamma\cdot n}\sum_{j=1}^{n}\mu_{jb} = \overline{S_b}$$

Because the Gaussians represent clusters of pixels with similar values, using the clusters instead of the cluster means will make the algorithm less sensitive to the absolute number of pixels of a particular color in the image.

One of the major problems in white patch retinex is obtaining a reliable estimate of the peak value of each color channel. In the modified version we propose here, we look for the channel maximums in the Gaussian cluster means instead of the raw pixels. We believe this estimate to be more robust in the face of noise and clipping. The EM version of white patch retinex becomes:

$$\frac{1}{\alpha}\mu_{jr}^{max} = S_r^{max},\ \frac{1}{\beta}\mu_{jg}^{max} = S_g^{max},\ \frac{1}{\gamma}\mu_{jb}^{max} = S_b^{max}$$

Prior work by Gershon (1987) attempts to achieve similar results through the use of more traditional image segmentation techniques.

2.4 Approximate Methods

As is stated in Moore (1999) estimating the parameters of a mixture model for a large data set can be quite computationally complex; and images are quite large data sets, a typical 512 x 512 image contains 262,144 pixels. A 1280 x 1280 image, not atypical in today's digital cameras, contains approximately 1.6 million pixels. In image applications speed is often important as operations need to be performed in close to real time. To achieve performance goals it is often acceptable to trade off some accuracy in the final answer for drastically reduced computational complexity. Moore (1991, 1999) describes a technique for quickly constructing an efficient specialized data structure and the associated cached statistics. We have taken a much more basic and less powerful approach here. We uniformly divide up the input space into small cubes. Each cube represents a *bin* and statistics are cached and associated with each bin, based on the pixels which fall into a specific bin. As detailed in the previous section, we assume a diagonal covariance structure for the components of our mixture model to simplify computations. For each bin, bin_i, we cache seven values: N_i, the number of pixels falling into that bin, $\widehat{\rho}_k(i) = \sum \rho_k(x,y)/N_i$, the mean for each of the color component values, k, for the pixels which fall into that bin, and $\widehat{\rho}_k^2(i) = \sum \rho_k^2(x,y)/N_i$ the mean of the squared color component values for the pixels falling into that bin. This leads to modified expectation and maximization steps in our EM algorithm. In our approximate version, the expected value of the hidden variable, z_{ij}, is now based on the mean of the pixels stored in the bin as follows and where i now represents the bin index, N_i is the number of pixels falling in that bin:

Expectation step:

$$E[z_{ij}] = \frac{(\prod_{k=1}^{3} \sigma_{jk}^2)^{-1/2} \exp\left(-\frac{1}{2}\sum_{k=1}^{3} \frac{(\widehat{\rho_k}(i)-\mu_{jk})^2}{\sigma_{jk}^2}\right)}{\sum_{c=1}^{n} (\prod_{k=1}^{3} \sigma_{ck}^2)^{-1/2} \exp\left(-\frac{1}{2}\sum_{k=1}^{3} \frac{(\widehat{\rho_k}(i)-\mu_{ck})^2}{\sigma_{ck}^2}\right)}$$

Maximization step:

$$\mu_{jk} \leftarrow \frac{\sum_{i=1}^{m} E[z_{ij}] N_i \widehat{\rho}_k(i)}{\sum_{i=1}^{m} E[z_{ij}] N_i}$$

$$\sigma_{jk} \leftarrow \frac{\sum_{i=1}^{m} E[z_{ij}] N_i \widehat{\rho}_k^2(i) - \left(\sum_{i=1}^{m} E[z_{ij}] N_i \widehat{\rho}_k(i)\right)^2}{\sum_{i=1}^{m} E[z_{ij}] N_i}$$

Utilizing our prior knowledge of the application domain, we also decided to explore another method of dividing the inputs space into bins. We experimented with bins in which all of the pixels had equal sums, that is $\rho_r(x,y) + \rho_g(x,y) + \rho_b(x,y)$ is equal to a constant. This creates bins, actually "slabs", which are separated by planes which are perpendicular to the line $R = G = B$. The result is that all of the pixels in the slabs have roughly the same brightness.

3. Results

3.1 Evaluation Metrics

To evaluate the modified algorithms and approximate versions we compared the color constancy performance of the standard algorithms, the EM versions operating on the raw pixels, and the approximate EM versions using a number of different bins. The error in the illuminant estimated by a specific algorithm was measured as the Euclidean distance between the chromaticity of the "true" illuminant to the estimated illuminant. The chromaticity measure used in this work is a two dimensional quantity which can be calculated from the triplet α, β, γ of illuminant scaling values as follows:

$$C_1 = \frac{\alpha}{\alpha+\beta+\gamma},\ C_2 = \frac{\beta}{\alpha+\beta+\gamma}$$

Only two values are needed because $C_3 = 1 - C_1 - C_2$. The "true" illuminant was estimated as the C_1 and C_2 values of a known white region in the image. Algorithms were also timed to measure relative computational complexity.

3.2 Data Set

To perform the evaluation described in the previous section a set of images collected under controlled conditions is necessary. The data set used for the evaluation was collected by the Computational Vision Lab at Simon Fraser University as detailed in Funt (1998). They describe the conditions under which the dataset was collected as the following. The camera used to capture the images was a Sony DXC-930 3-CCD color video camera balanced for 3200K lighting with gamma correction turned off. The data was post processed to remove offsets and was linear in reflectance. Images of 11 different household objects on black backgrounds were collected: two different colored balls, a book, a disposable coffee cup, a cereal box, flowers, a bottle of bleach, a box of macaroni, some colored rope, a shampoo bottle and a detergent box. The images were captured in a Macbeth Judge II light booth under the following five illuminants: Sylvania Halogen, Sylvania Cool White Fluorescent, a Phillips Ultralume Fluorescent, the Macbeth Judge II 5000 Fluorescent, and the Macbeth Judge II 5000 Fluorescent together with a Roscolux 3202 blue filter.

Two images of each object under each illuminant were collected for a total of 110 images. In addition to the selected object, each image also included the Gretag / Macbeth ColorChecker color rendition chart. This chart includes a set of neutral gray patches of varying reflectance. The portion of the image which contained this chart was used to determine the true illuminant chromaticities. This portion of the image was cropped from each image before use since it would give an advantage to the white patch algorithms.

Because the images were exposed to be quite dark, image intensity values were scaled by a factor of 2.5 and values over 255 were clipped to more closely match the exposure settings under which images would typically be captured, as suggested in Funt (1998).

3.3 Algorithm Specifics

The following algorithms were evaluated: white patch retinex, EM based white patch retinex, gray world, EM based gray world. The white patch retinex algorithm simply used the maximum value of each of the color channels to estimate the illuminant. The gray world algorithm use the channel means to estimate the illuminant. The EM based white patch algorithm utilized the R,G,B values of the means of the Gaussians to estimate the illuminant. The EM based gray world algorithm took the average of the cluster means. A probability cutoff value was also used such that clusters which had a very small probability of generating any of the pixels in the image were not used in the calculation. A total of sixteen Gaussians were used in each mixture model.

The images described previously were split into two equal sized sets. One set was used to estimate the canonical color channel response ratios of the images when corrected. The second set was used as test data. Four variants of our EM based algorithms were evaluated. One variant used just the raw pixel values. A second version used 16 bins per dimension for a total of 4096 bins. A third version used 8 bins per dimension for a total of 512 bins. A fourth version used 1024 bins (or "slabs") which were oriented perpendicular to the line $R = G = B$.

3.4 Experimental Results

One simple, albeit imprecise, way to evaluate these algorithms is to visually examine the results of correcting an image with the given illumination estimate. Figure 1 contains a set of color images, the original "cereal box" image with two different illuminants, the result of using simple white patch, the result of using EM with uniform bins, and the best correction based on the illumination model used. As can be seen, the corrected images have a more neutral tone and the colors in these images more closely match one another than those in the original images. (If the images do not appear in color, they can be found online at: http://www.cs.cmu.edu/~chuck/cc1/)

Table 1 contains the distance results for the white patch algorithms, larger distances represent larger errors. The last column labeled "Significance" is the probability that the specific algorithm actually performs worse than the basic algorithm, even though the mean error appears to be smaller. This probability was estimated using a student t distribution. As this table shows, all of the EM variants of the white patch algorithm perform better than the basic algorithm with a very low probability of this being a chance result. Interestingly, the approximate binned versions of the algorithm seem to perform better than the version which utilizes the raw pixels.

Table 2 contains the distance results for the gray world algorithms. Here the EM algorithm which utilizes the raw pixels and the binned versions perform better than the standard version. However, in the raw pixel and projected bin cases, there is a reasonably high probability that this is a chance result.

Table 3 summarizes the timing performance of the algorithms. Since the computational complexity of the white patch and the gray world algorithms is nearly identical, only the white patch measurements are reported. The time reported is the program run time in seconds on a Intel Pentium II at 266 MHz. (Time required for file operations are not included.) As this table shows, at a typical image size of 512 x 512, the binned EM version is approximately 16.5 times slower than the basic algorithm, but still takes less than one second for a typically sized image. The full EM version is approximately 2900 times slower than the basic version, and takes nearly 3 minutes to process a typically sized image. As the image size increases the binned EM version gains a larger advantage because the time it spends iterating over a constant sized data structure (the bins) is amortized over a larger set of pixels. This relationship can be seen in Figure 2. This is a log-log plot of processing time on the y-axis versus image pixel count on the x-axis. The linear relationship between pixel count and processing time can be observed for all algorithms, the binned version reaching a slope asymptote at about 500,000 pixels.

4. Related Work

Many color constancy methods are described in the literature: Land (1977), Maloney (1986), Gershon (1987), Finlayson (1995), Funt (1996), Finlayson (1998), Funt (1999). The goal of these methods is to estimate the parameters of the scene illumination. The primary method used for this task is to make a set of prior assumptions about the overall distribution of color values in a typical image, or correspondingly typical illuminants and surfaces in the world. These assumptions are then used as a basis for illuminant parameter estimation. The most basic methods, as previously described, are the gray world algorithm and white patch retinex.

Unmodified Image | Simple White Patch | EM White Patch | Optimal Estimate

Figure 1. These color images provide an example of algorithm performance for visual inspection. Images in the first row were captured under halogen illumination. Images in the second row were captured under Macbeth 5000 illumination. The columns represent, respectively, from left to right: no correction, simple white patch correction, EM white patch correction with 4096 uniform bins, optimal correction. If the images do not appear in color, they can be found online at: http://www.cs.cmu.edu/~chuck/cc1/

Table 1. A comparison of the average error of the estimated chromaticities for variations of the white patch algorithm. The column labeled “standard deviation” is the standard deviation of the average error. The column labeled “significance” is the probability that the observed sign of the difference of that algorithm’s mean from the basic version of the algorithm was observed by chance based on a student t distribution.

Algorithm	Average Distance Error	Standard Deviation	Significance
White Patch Retinex	0.0992	0.0611	—
White Patch EM / Raw Pixels	0.0665	0.0469	4.99×10^{-7}
WP EM / 4096 Uniform Bins	0.0644	0.0445	1.83×10^{-7}
WP EM / 512 Uniform Bins	0.0639	0.0436	1.44×10^{-7}
WP EM / 1024 Projection Bins	0.0637	0.0475	1.94×10^{-7}

Table 2. A comparison of the average error of the estimated chromaticities for variations of the gray world algorithm. The column labeled “standard deviation” is the standard deviation of the average error. The column labeled “significance” is the probability that the observed sign of the difference of that algorithm’s mean from the basic version of the algorithm was observed by chance based on a student t distribution.

Algorithm	Average Distance Error	Standard Deviation	Significance
Gray World	0.0625	0.0420	—
Gray World EM / Raw Pixels	0.0611	0.0400	0.169
GW EM / 4096 Uniform Bins	0.0595	0.0380	0.053
GW EM / 512 Uniform Bins	0.0594	0.0391	0.050
GW EM / 1024 Projection Bins	0.0620	0.0444	0.322

Table 3. This table contains measured run times in seconds for three different variants of the white patch algorithm as image size is varied.

Image Size	White Patch	White Patch EM 4096 Bins	White Patch EM Raw Pixels
128 x 128	0.00	0.29	10.70
256 x 256	0.01	0.49	42.49
512 x 512	0.06	0.99	175.40
768 x 768	0.11	1.66	396.68
1024 x 1024	0.18	2.65	740.66
1280 x 1280	0.32	3.72	1089.98
1536 x 1536	0.40	5.20	1575.80

An algorithm with a different methodology, as discussed in Finlayson (1995), is one which utilizes constraints about the distribution of the extreme colors in an image. This method uses the gamut of the color in an image, which one can think of as the three dimensional convex hull which encloses the color histogram. The constraints are formed by knowledge of typical object surfaces and illuminants in the world. A search procedure is used to determine the transformation which best satisfies the given constraints.

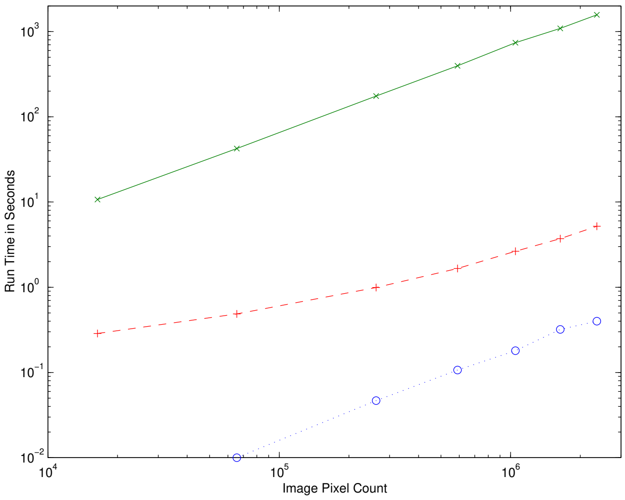

Figure 2. Plot of white patch algorithm run time in seconds versus image pixel count. The dotted line plots data from the basic version of the algorithm. The dashed line plots data from the EM version of the algorithm with 4096 bins. The solid line plots data from the version of the algorithm utilizing raw pixels.

A method introduced in Funt (1996) uses a neural network which takes as input the image color histogram and outputs the estimated chromaticities. One problem with this method is that it requires training data to train the neural network. Funt (1998) reports that the performance of this algorithm in a database matching application is quite good. More recent work by Funt (1999) has made use of bootstrapping techniques to eliminate the need for highly controlled training data.

The "retinex" family of algorithms has been around for quite some time and were originated by Land (1977). The basic concept is that the lightest regions in the image provide a measurement of the parameters of the original scene illumination. The simplest version of this algorithm is called the white patch algorithm.

Another algorithm which attempts to improve on the gray world algorithm is described in Gershon (1987). Here traditional image segmentation techniques are used to divide an image into regions. These regions are used to measure sensor surface responses and perform computations in a modified gray world algorithm.

5. Conclusions and Future Work

We have demonstrated that constructing and using a parametric model of the distribution of image color pixel values can improve the performance of color constancy algorithms. In the case of the white patch and gray world algorithms a significant increase in the accuracy of the illuminant estimate was measured. In both cases approximations using cached statistics did as well as or better than using the raw data and provided a major improvement in speed. A possible explanation for the improved accuracy in the approximate method is that the binned data introduces a prior toward smoothness which helps EM to avoid local maxima in likelihood space. A surprising result was the strong performance of the projected binning approach.

Our next line of investigation in this work will be to experiment with mixture models which use the full covariance matrix. Our performance results suggest that performance could be improved in that manner. Also tantalizing is the projected binning algorithm which seemed to improve both algorithm accuracy as well as computational performance. This suggests that for certain applications a carefully chosen set of bins might provide some of the benefit of using the full covariance matrix without the complexity.

Acknowledgements

I would like to thank my advisor Sebastian Thrun for his support throughout this work. I would especially like the members of the Computational Vision Lab at Simon Fraser University: Dr. Brian Funt, Vlad Cardei, and Kobus Barnard. I am deeply indebted to them for collecting the data set used in this work and making it publicly available. This research is sponsored in part by National Science Foundation LIS grant number REC-9720374, which is gratefully acknowledged.

References

Duda, R. O. & Hart, P. E. (1973) *Pattern classification and scene analysis*. New York: John Wiley & Sons.

Finlayson, G. & Drew, M. S. & Funt, B. (1993). Diagonal transforms suffice for color constancy. *IEEE Proceedings: International Conference on Computer Vision* (pp. 164-171). Berlin.

Finlayson, G. & Funt, B. & Barnard, J. (1995). Color constancy under a varying illumination. *Proceedings of the Fifth International Conference on Computer Vision.*

Finlayson, G. & Schiele, B. & Crowley, J. (1998). Comprehensive color image normalization. *Proceedings of the Fifth European Conference on Computer Vision*. Freiburg, Germany.

Funt, B. & Cardei, V. & Barnard, K. (1996). Learning color constancy. *Proceedings of Imaging Science and Technology / Society for Information Display Fourth Color Imaging Conference* (pp. 58-60). Scottsdale.

Funt, B. & Barnard, K. & Martin, L. (1998). Is colour constancy good enough? *Proceedings of the Fifth European Conference on Computer Vision* (pp. 445-459).

Funt, B. & Cardei, V. C. (1999). Bootstrapping color constancy. *Proceedings of SPIE: Electronic Imaging* IV, 3644.

Gershon, A., Jepson, A. D., Tsotsos, J. K. (1987). From [R,G,B] to surface reflectance: computing color constant descriptors in images. *Proceedings of the International Joint Conference on Artificial Intelligence* (pp. 755-758).

Land, E.H. (December 1977). The retinex theory of color vision. *Scientific American*, 108-128.

Maloney, L. T. and Wandell, B. A. (1986). Color constancy: a method for recovering surface spectral reflectance. *Journal of the Optical Society of America A*, 3:29-33.

Mitchell, T. M. (1997) *Machine learning*. New York: McGraw Hill.

Moore, A. W. (1991). *Doctoral dissertation: Efficient Memory-based Learning for Robot Control*. Computer Laboratory, University of Cambridge, Technical Report No. 209.

Moore, A. W. (1999). Very fast EM-based mixture model clustering using multiresolution kd-trees. *Advances in Neural Information Processing Systems 11.*

Rowley, H. A. & Baluja, S. & Kanade, T. (January 1998). Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, 1, 23-38.