

Experiments in Sensor Sharpening for Color Constancy

*Kobus Barnard and Brian Funt
Simon Fraser University
Burnaby B.C.*

Abstract

Sensor sharpening has been proposed as a method for improving color constancy algorithms but it has not been tested in the context of real color constancy algorithms. In this paper we test sensor sharpening as a method for improving color constancy algorithms in the case of three different cameras, the human cone sensitivity estimates, and the XYZ response curves. We find that when the sensors are already relatively sharp, sensor sharpening does not offer much improvement and can have a detrimental effect. However, when the sensors are less sharp, sharpening can have a substantive positive effect. The degree of improvement is heavily dependent on the particular color constancy algorithm. Thus we conclude that using sensor sharpening for improving color constancy can offer a significant benefit, but its use needs to be evaluated with respect to both the sensors and the algorithm.

Introduction

Sensor sharpening has been proposed as a method for improving color constancy algorithms [1], but it has not been tested in the context of real color constancy algorithms. Rather, the experimental results available are limited to the minimum error possible with and without sharpening. Since the error in current color constancy methods is often substantially larger than the minimum error possible, we felt it necessary to investigate further the utility of sensor sharpening for color constancy. In this paper we provide results of color constancy with and without sharpening for a Sony DXC-930 CCD video camera, a Kodak DCS-200 digital camera [2], a Kodak DCS-420 digital camera [2], the XYZ response curves [3], and the Vos and Walraven human cone sensitivity estimates [3]. The general conclusion is that when the sensors are already relatively sharp (e.g. the Sony camera), further sensor sharpening is not worth the trouble, and often has a small detrimental effect. However, when the sensors are not sharp (e.g. the DCS-200 and DCS-420), sensor sharpening can have a substantial positive effect, depending on the algorithm, thus validating the original work.

Sensor Sharpening

We begin with an explanation of sensor sharpening [1]. The motivation for sensor sharpening is the observation that most color constancy algorithms make use of a diagonal model of illumination change. To understand this model, consider a white patch under two different illuminants. Suppose that under the first illuminant the color is $[r, g, b]$ and under the second illuminant the color is $[r\hat{r}, g\hat{g}, b\hat{b}]$. It is possible to map the color of white under the first illuminant to the color under the second by post-multiplication by a diagonal matrix: $[r\hat{r}, g\hat{g}, b\hat{b}] = [r, g, b] \text{diag}(r\hat{r}/r, g\hat{g}/g, b\hat{b}/b)$. If the same diagonal matrix transforms the RGB of all surfaces (not just the white ones) to a good approximation, then we say that we have a diagonal model of illumination change. It turns out that the accuracy of the approximation is a function of the vision system's sensors.

The idea of sensor sharpening is to map the data by a linear transform T into a new space where the diagonal model holds more faithfully. Colour constancy algorithms which rely on the diagonal model can then proceed more effectively. The final result is then mapped back to the original RGB space with the inverse transformation. Working in the transform space is like having new sensors which are a linear transformations of the old ones. Further, the sensitivity functions of sensors which support the diagonal model tend to look sharper with narrower peaks than ones that do not. In the extreme case, if the sensors are delta-functions, the diagonal model holds exactly. From these two observations, we get the name: sensor sharpening.

The main technical result in sensor sharpening is finding the transformation T . In [1], Finlayson et al propose three methods for finding T : 'sensor based sharpening', 'database sharpening', and 'perfect sharpening'. For this work we chose database sharpening over sensor based sharpening due to the clean correspondence between the sharpening method and a color constancy error metric. Perfect sharpening did not work well for us because our test illuminant set did not meet the key requirement of being two-dimensional, partly due to the inclusion of fluorescent lights.

In database sharpening, RGB are generated using a database of reflectance spectra, together with an illuminant spectrum and the sensors. This is done for two separate illuminants. Let A be the matrix of RGB for the first illuminant and B be the matrix for the second, with the RGBs placed row-wise. In the sharpening paradigm we map from B to A with a sharpening transform, followed by a diagonal map, followed by the inverse transform. If we express each transform by post multiplication by a matrix we get: $A \approx BTD^{-1}$. In database sharpening the matrix T (and implicitly D) is found that minimizes the RMS error, $\|A - BTDT^{-1}\|_2$. The sharpening transform gives exactly the same error as the best linear transform M . In fact, T is found by diagonalizing M , where M minimizes $\|A - BM\|_2$.

One implementation issue should be noted. The result of the diagonalization is ambiguous up to scaling and swapping of the columns of T . As is standard, we use columns of norm 1. Furthermore, we put the element of T of largest absolute value on the diagonal by swapping columns, and ensure that it is positive by multiplying the column by -1 if necessary. Then in a similar way we attempt to make the other diagonal elements as large as possible. This procedure is used to reduce the number of negative components of sharpened data.

In this work we view color constancy as finding a transformation from the image of a scene taken under an unknown illuminant, to the image of the same scene as

maximum in each channel intuitively estimates white in RGB space, but its choice as an estimator in sharp space is less clear. Similar considerations are also relevant in analyzing methods for choosing a solution from the constraints sets found with gamut mapping algorithms.

Additional problems can occur with the gamut mapping algorithms due to negative RGB which can be introduced by sharpening. Depending on the variant of the algorithm, it may or may not be problematic to have negative components in the sharpened input data, canonical data, or illuminant database data. We found that the chromaticity version of Forsyth's method [10] is very sensitive to these problems, and therefore we do not include results for it. In the case of the RGB variants used below, a few of the illuminants cause trouble; in the context of an application, one could default back to non-sharp computation. However, for the purposes of testing we simply exclude that generated scene from the test, thus ensuring that all algorithms are run on the same data.

Results

We have investigated sensor sharpening in the case of a Sony DXC-930 CCD video camera, a Kodak DCS-200 digital camera [2], a Kodak DCS-420 digital camera [2], the XYZ response curves [3], and the Vos-Walraven human cone sensitivity estimates [3]. As our main interest here is machine vision and image reproduction, we are most interested in the results for the cameras. However, sharpening is also of interest in relation to human vision[1].

A sampling of our results is shown in Table 1. The data was used was generated from a test set of 100 measured illuminants (normalized to be the same magnitude), a database of roughly 2000 reflectances, and the 5 different sensors. The 100 measured illuminants include a variety of indoor and sources and outdoor illumination, as well as a complex combinations thereof obtained at random locations in and around our university campus. We note that the results of sharpening are a function of the spectra databases; our data sets were chosen to be as general as possible.

As discussed above, the result of each algorithm can be interpreted as supplying a mapping from unknown illuminant images to canonical ones. To obtain the errors presented here, we applied that mapping to a large set of RGB computed using the entire reflectance database, together with the test illuminant and the sensors. We then computed a similar set using the canonical illuminant, which was chosen to be a Tungsten illuminant, and tabulated the RMS difference of the two sets. This error metric was chosen to coincide with the error which database sharpening strives to reduce. Lack of space prevents us from providing errors using other metrics.

Each entry in the table is the average of 600 results. The first five rows in the two tables are target results. The first row is the best linear fit (which is also the best diagonal fit with optimal database sharpening), the second is the best diagonal fit, the third is the result of 'knowing' the RGB of the illuminant, the fourth is the same with optimal database sharpening, and the fifth is the same with sharpening using the global average illuminant. Following the target results are the results of several algorithms estimating the best diagonal map on the basis of 8 randomly chosen data points.

The 'sharpness' of the sensors can be taken as the relative magnitudes of the best linear fit and the best diagonal fit. The results below indicate that sharpening the already 'sharp' DXC-930 sensors is troublesome at best. However, in the case of the less sharp DCS-200 and DCS-420 sensors, sharpening can give a substantial improvement, as is the case with the E-CRULE-HA algorithm. For example, with the DCS-200, average illuminant based sharpening reduces the error of E-CRULE-HA from 81.3 to 33.0. E-CRULE-HA is better tuned than E-CRULE-MV for both the data set and the error metric, and thus the difference between the two algorithms in the case of sharp sensors is not surprising. However, it is interesting to note that in the case of the DCS-200 and DCS-420 sensors, sharpening was required to obtain the advantages of this algorithm in this situation. In summary, the results indicate that the benefit of sharpening is quite dependent on the algorithm, which is understandable based on the discussion above, but the magnitude of the effect is still surprisingly large.

In Table 1 we exclude results for the cone sensitivity estimates. Most algorithms did not work well in the sharp space for these sensors. Only the comparison algorithms and the gray world algorithm gave reasonable results. With these sensors, the sharpened RGB typically had negative R and G. This almost always made the gamut mapping approach untenable, and also made the Retinex algorithm perform very poorly.

Finally, we note that when sharpening is beneficial for the real color constancy algorithms, using the global average illumination for sharpening is not much worse than the optimal. On the other hand, in the case of the known illuminant 'algorithm', optimal sharpening gives substantially better results (relative to the already small error) than average illuminant sharpening. This suggests that as color constancy improves, the method of choosing the sharpening transform becomes more relevant. However, when we ran some of the experiments again with the number of surfaces increased to 32 in order to improve color constancy, the advantage of using the optimal sharpening was still slight, even though the RMS error for some of the algorithms was less than 20.

Table 1: Mapping error vs. algorithm

		DXC 930	DCS 200	DCS 420	XYZ
Linear Fit		2.88	1.32	1.67	2.04
Diagonal Fit		4.42	8.48	10.7	9.79
Known Illum.		4.53	8.91	11.2	10.6
Known Illum.	(opt)	2.99	1.33	1.68	2.07
Known Illum.	(ave)	3.90	1.63	2.02	3.00
Gray World		44.5	39.8	39.0	40.3
Gray World	(opt)	45.8	41.4	40.5	42.9
Gray World	(ave)	45.5	41.3	40.4	43.4
Retinex		146	112	120	123
Retinex	(op)	144	102	111	114
Retinex	(ave)	143	102	111	113
E-CRULE-MV		74.3	69.8	66.0	52.4