Journal of Imaging Science and Technology® 50(4): 341–348, 2006.
© Society for Imaging Science and Technology 2006

Estimating Illumination Chromaticity Via Support Vector Regression[1]

Weihua Xiong and Brian Funt
School of Computing Science, Simon Fraser University, Vancouver, Canada
E-mail: weihuax@sfu.ca

Abstract. *Support vector regression is applied to the problem of estimating the chromaticity of the light illuminating a scene from a color histogram of an image of the scene. Illumination estimation is fundamental to white balancing digital color images and to understanding human color constancy. Under controlled experimental conditions, the support vector method is shown to perform well. Its performance is compared to other published methods including neural network color constancy, color by correlation, and shades of gray.* © *2006 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.(2006)50:4(341)]

INTRODUCTION

Accurate estimation of the spectral properties of the light illuminating an imaged scene by automatic means is an important problem. It could help explain human color constancy and it would be useful for automatic white balancing in digital cameras. Here we will focus on machine based color constancy. A color imaging system will be considered to be color constant to the degree to which it is able to account for changes in the color of the scene illumination and thereby maintain a stable representation of object colors.

More precisely we can formulate color constancy as: Given a digital image acquired under unknown illumination conditions, predict what the image would have been if the same scene had been illuminated instead by some chosen known "canonical" illuminant. For example, the canonical illuminant might be specified as equal energy white. Color constancy can be divided into two subproblems: (1) estimate the color of the illumination and (2) adjust the image colors based on the difference between the estimated and canonical illuminants. The second problem is often addressed by the von Kries coefficient rule or an equivalent diagonal transformation model.[1] Because it is very under constrained, the first problem, illumination estimation, is the more difficult of the two. We propose a new method based on support vector regression to solve it.

Many papers have been published on illumination estimation. Some aim to recover the full spectrum of the illumination,[2,3] while others aim to recover either a two-parameter (e.g., *xy* or *rg*) estimate of its chromaticity[4,5] or a three-parameter description of its color (e.g., *XYZ* or *RGB*).[6,7] The method we propose here is similar to previous work by Funt et al.[4,8] and Finlayson et al.[5] in that it aims to recover the chromaticity of the scene illumination based on the statistical properties of binarized color or chromaticity histograms; however, the proposed method replaces the neural networks and Bayesian statistics of these previous methods with support vector machine regression.

Vapnik's[9,10] support vector machine theory has been applied successfully to a wide variety of classification problems.[11–14] Support vector machines have been extended as well to regression problems including financial market forecasts, travel time prediction, power consumption estimation, and highway traffic flow prediction.[15–17]

Depending on the problem domain, support vector machine based regression (SVR) can be superior to traditional statistical methods in many ways. SVR enables inclusion of a minimization criterion into the regression, training can be easier, and it achieves a global rather than local optimum. It also facilitates explicit control of the tradeoff between regression complexity and error. We show how the illumination estimation problem can be formulated in SVR terms and find that, overall, SVR performs well.

SUPPORT VECTOR REGRESSION

SVR estimates a continuous valued function that encodes the fundamental interrelation between a given input and its corresponding output in the training data. This function then can be used to predict outputs for given inputs that were not included in the training set. This is similar to a neural network. However, a neural network's solution is based on empirical risk minimization. In contrast, SVR introduces structural risk minimization into the regression and thereby achieves a global optimization, while a neural network achieves only a local minimum.[18]

Most classical regression algorithms require knowledge of the expected probability distribution of the data. Unfortunately, in many cases, this distribution is not known accurately. Furthermore, many problems involve uncertainties such that it is insufficient to base a decision on the event probability alone. Consequently, it is important to take into account the potential cost of errors in the approximation. SVR minimizes the risk without prior knowledge of the probabilities.

Smola and Schölkopf[9] provide an introduction to SVR.

[1]Presented in part at the IS&T/SID 12th Color Imaging Conference, Scottsdale, AZ, November 2004.

Received Feb. 1, 2005; accepted for publication Dec. 20, 2005.
1062-3701/2006/50(4)/341/8/$20.00.

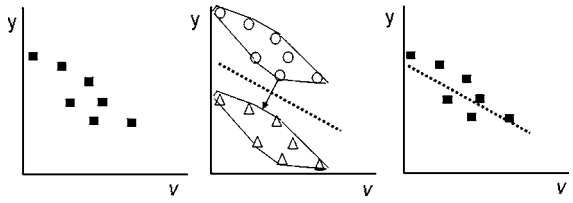


Figure 1. Geometrical interpretation of SVR [after Figs. 1 and 2 of Bi and Bennett (see Ref. 20)]. The left panel shows the input data (squares) as a function of the multidimensional feature vector *v*. The regression line is found by making two copies of the data and shifting them equal amounts up and down relative to the original data. The regression (dotted) line is found as the bisector of the line (arrow) between the two closest points on the convex hulls of the shifted data sets. The right panel shows the regression line from the middle panel superimposed on the original data.

Some simple intuition about it can be gained by comparison to least-squares regression in fitting a line in two dimensions. Least-squares regression minimizes the sum of squares distance between the data points and the line. SVR maximizes the space containing the data points subject to minimization of the distance of the points to the resulting line. The width of the space is called the "margin." Points within an "insensitivity" region are ignored. The technique represents the region defined by the margin by a subset of the initial data points. These data points are called the support vectors. SVR is extended to the fitting of a nonlinear function by employing the kernel trick,[9] which allows the original nonlinear problem to be reformulated in terms of a kernel function. The reformulated problem is linear and can be solved using linear SVR. We used the Chang and Lin[19] SVR implementation.

An intuitive geometric interpretation of SVR in terms of distances between the convex hulls of the training sets is provided by Bi and Bennett.[20] Figure 1 shows the basic idea for the simplest case of a linear fit with hard margins. Copies of the original data are made and shifted vertically, one up, one down, by equal amounts. The two sets of data are then considered to be two groups to be classified. The regression line is determined as the line that best separates the two groups into two classes. The best separation is found by considering the convex hulls of the two sets and the locations where the hulls come closest to one another. The perpendicular bisector of the line between the two closest points provides the optimum separation between the classes, and also is the regression line to the original data.

SVR FOR ILLUMINATION CHROMATICITY ESTIMATION

In this section, we discuss how the SVR technique can be applied to analyze the relationship between the image of a scene and the chromaticity of the illumination chromaticity incident upon it. As introduced in the neural network method,[8] we will first use binarized two-dimensional (2D) chromaticity space histograms to represent the input image data. Later, we extend these histograms to three dimensional (3D) to include intensity as well as chromaticity. Chromaticity histograms have the potential advantage that they discard intensity shading, which varies with the surface geometry and viewing direction, but is most likely unrelated to the illumination's spectral properties.

The training set consists of histograms of many images along with the measured *rg* chromaticities [$r=/(R+G+B)$ and $g=G/(R+G+B)$] of the corresponding scene illuminants. Each image's binarized chromaticity histogram forms a SVR binary input vector in which each component corresponds to a histogram bin. A "1" or "0" indicates that the presence or absence of the corresponding chromaticity in the input image. Partitioning the chromaticity space equally along each component into N equal parts yields $N\times N$ bins. The resulting SVR binary input vector is of size N^2. We experimented with various alternative choices for N and eventually settled on $N=50$. Generally speaking, for $N<50$, the bins are too large so the color space is quantized too coarsely, with the result that the illumination estimation error increases. For $N>50$, the training time increases, but without a corresponding improvement in overall performance. All the results reported below are based on $N=50$, so the chromaticity step size is 0.02. With $0\leq r, g\leq 1$ only half these bins can ever be filled, so a sparse matrix representation was used. Support vector regression then finds the function mapping from image histograms to illuminant chromaticities.

Since some other illumination estimation methods[7,21] (gamut mapping and color by correlation) benefit from the inclusion of intensity data, it is natural to consider it in the SVR case as well. The neural network method has thus far not been applied to 3D data (chromaticity plus intensity) because the number of input nodes becomes too large and the space too sparse for successful training, given the relatively small size of the available training sets. Support vector regression handles sparse data reasonably well, so we experimented with 3D binarized histograms in the training set. Intensity, defined as $L=R+G+B$, becomes the third histogram dimension along with the *r* and *g* chromaticity. We quantized *L* into 25 equal steps, so the 3D histograms consist of 62 500 $(25\times 50\times 50)$ bins.

HISTOGRAM CONSTRUCTION

To increase the reliability of the histograms, the images are preprocessed to reduce the effects of noise and pixels straddling color boundaries. We have chosen to follow the region-growing segmentation approach described by Barnard et al.[21] This also facilitates comparison of the SVR method to the other color constancy methods Barnard et al. tested. The region-growing method is good because the borders it finds are perfectly thin and connected. Membership in a region is based on chromaticity and intensity. A region is only considered to be meaningful if it has a significant area. For the sake of easy comparison, we used the same thresholds as in Ref. 21, namely, to be in the same region, the *r* and *g* chromaticities at a pixel must not differ from their respective averages for the region containing the pixel by more than 0.5%, or its intensity by 10%. Also, regions that result in an area of fewer than five pixels are discarded. The RGBs of all pixels within each separate region are then averaged, converted to *L*, *r*, *g* and histogrammed.

Table I. Admissible kernel functions.

Name	Definition	Parameters
Linear	$K(x_i, x_j) = (x_i)^T x_j$	-
Polynomial	$K(x_i, x_j) = [(x_i)^T x_j + 1]^d$	d
Radial basis function (RBF)	$K(x_i, x_j) = e^{-\gamma\|x_i - x_j\|^2}$	γ
Sigmoid[a]	$K(x_i, x_j) = \tanh[(x_i)^T x_j + r]$	r

[a]For some τ values, the kernel function is invalid.

k-FOLD CROSS VALIDATION FOR SVR PARAMETERS

The performance of SVR is known to depend on its insensitivity parameter ϵ, regularization parameter C, the choice of kernel function and associated parameters. Different kernel functions work better on some problem domains than others. Four of the commonly used kernel functions are listed in Table I. From a practical and empirical standpoint, the bigger the insensitivity parameter ε, the fewer the support vectors, and the higher the error in estimating the illumination. After much experimentation with different ϵ values, we fixed it to be 0.0001.

In the case of SVR for illumination estimation, the best choice of kernel function and its parameters may depend on the training set. We eliminated the Sigmoid kernel function from further consideration since it is known to be invalid for some values of the parameter r (Ref. 10) and focus instead on the RBF and polynomial kernel functions. This leaves the choice of either the RBF or polynomial kernel functions and for each of these kernels their parameters: penalty C and width γ for the RBF kernel function; or penalty C and exponential degree d for polynomial kernel function. The parameters γ and d control the corresponding kernel function's shape, while C determines the penalty cost of estimation errors. The kernel choice and parameter settings are made during the training phase by k-fold cross validation, which involves running the training using several different parameter choices and then selecting the choice that works best for that particular training set. This is described in more detail below.

For the RBF kernel function, we allow the penalty parameter to be chosen from four different values $C \in \{0.01, 0.1, 1, 10\}$ and the width value from $\gamma \in \{0.025, 0.05, 0.1, 0.2\}$. For the polynomial kernel function, we used the same four penalty candidates and selected the best degree d from the set {2 3 4 5}. Thus for each training data set, 32 test cases (two kernel choices with 16 pairs of parameter settings each) are tested to find the best choice.

Among the algorithms generally used to find the best parameters for support vector regression, we chose k-fold cross validation because it does not depend on a priori knowledge or user expertise and it handles the possibility of outliers in the training data. The disadvantage of the k-fold method is that it is computationally intensive.

In k-fold cross validation, the whole training set is divided evenly into k distinct subsets. Every kernel function and each of its related parameters forms a candidate parameter setting. For any candidate parameter setting, we conduct the same process k times during which $(k-1)$ of the subsets are used to form a training set and the remaining subset is taken as the test set. The root-mean-square (rms) chromaticity distance errors from k trials are averaged to represent the error for that candidate parameter setting. The parameter setting leading to the minimum error is then chosen and the final SVR training is done using the entire training set based on the chosen parameter setting.

EXPERIMENTS

We tested the proposed SVR-based illumination estimation method on both synthetic and real images. The implementation is based on the SVR implementation by Chang and Lin.[19] To this we added a MATLAB interface that reads data files representing the image histograms and associated illumination chromaticities. Each row in the training data file represents one training image and consists of two parts: the true illumination chromaticity followed by the bin number for each nonzero histogram bin.

Barnard et al.[21,22] reported tests of several illumination estimation methods, including neural-network based and color by correlation. We have tried to follow their experimental procedure as closely as possible and used the same image data so that SVR can be compared fairly to these other methods. In addition, we compare SVR to the new "shades of gray" (SoG) technique[23] based on the Minkowski family of norms, Max RGB, and Grayworld.

ERROR MEASURES

Several different error measures are used to evaluate performance. The first is the distance between the actual (r_a, g_a) and estimated chromaticity of the illuminant (r_e, g_e) as[21–23]

$$E_{i-dist} = \sqrt{(r_a - r_e)^2 + (g_a - g_e)^2}. \tag{1}$$

For the distance error, we also compute the rms, mean, and median errors over a set of N test images. It has been argued that the median is the most appropriate metric for evaluating color constancy.[24] The standard rms is defined as

$$\mathrm{rms}_{dist} = \frac{1}{N}\sqrt{\sum_{i=1}^{N} E_{i-dist}^2}. \tag{2}$$

The second error measure is the angular error between the chromaticity three vectors when the b-chromaticity component is included. Given r and g, $b = 1 - r - g$. Thus, we can view the real illumination and estimated illumination as two $\langle r, g, b\rangle$ vectors in 3D chromaticity space and calculate the angle between them. The angular error represented in degrees is

$$E_{i-angular} = \mathrm{COS}^{-1}\left[\frac{(r_a, g_a, b_a)(r_e, g_e, b_e)}{\sqrt{r_a^2 + g_a^2 + b_a^2} \times \sqrt{r_e^2 + g_e^2 + b_e^2}}\right] \times \frac{2\pi}{360}. \tag{3}$$

We also compute the rms, mean, and median angular error

over a set of images.

Even if the median angular error for one method is less than for another, the difference may not be statistically significant. To evaluate whether a difference is significant, we use the Wilcoxon signed-rank test.[24] In the following experiments, the error rate for accepting or rejecting null hypothesis is always set to 0.01.

SYNTHETIC DATA TRAINING, REAL-DATA TESTING

The first tests are based on training with synthesized image data constructed using the 102 illuminant spectra and 1995 reflectances described by Barnard[22] along with the sensor sensitivity functions of the calibrated SONY DXC-930 CCD.[25] Testing is based on Barnard's[21] 321 real images taken with the SONY DXC-930 of 30 scenes under 11 different light sources. These images are linear (a gamma of 1.0) with respect to scene intensity. These data are available on-line from the Simon Fraser University Computational Vision Laboratory color image database.[26]

The number of distinct synthesized training "scenes" was varied from 8 to 1024 in order to study the effect of training size on performance. Each synthetic scene was "lit" by each of the 102 illuminants in turn to create 102 images of each scene. The synthesized camera RGB values, their corresponding chromaticities, and the illuminant chromaticity are mapped to 2D and 3D binary vectors for input to SVR. Table II shows that the parameters vary with the training set as expected. Although the basis function type was allowed to vary during the cross validation, the RBF was eventually selected in all cases.

To test on real data, Barnard's calibrated 321 SONY images were first segmented and histogrammed according to the "generic preprocessing" strategy.[21] Illumination estimation by SVR compares favorably to the methods Barnard tested[21] and Finlayson reported[24] as shown in Table III. The rms and median errors for Color By Correlation with Binary Histogram (CC01), Color By Correlation with Maximum Likelihood (CCMAP), Color By Correlation with Mean Likelihood (CCMMSE), Color By Correlation (CCMLM), the Neural Network (NN), Database Grayworld (DB), Gamut Mapping (GM) are from Table II (p. 992 of Ref. 21) and Table 2 of Ref. 24, respectively. Figure 2 shows how the SVR performance initially improves as the size of the synthetic training set increases.

REAL IMAGE DATA TRAINING, REAL-DATA TESTING

Training on synthetic image data is convenient because large training sets can be calculated from existing databases of illuminant and reflectance spectra. The disadvantage of synthetic data is that it requires an accurate model of the camera and imaging process. On the other hand, creating a training set of real images is difficult because for each image the scene illumination must be measured.

Our testing with real data is based on three image data sets. To begin, we train and test on Barnard's[21] set of 321 SONY images and find that training with real data is in fact better than training with synthetic data. We continue with tests on Cardei's[4] set of 900 images from assorted cameras.

Table II. Results of *k*-fold kernel and parameter selection as a function of the histogram type and the number of training set images.

Training set size/102	Histogram dimension	Kernel selected	C	γ
8	2D	RBF	0.01	0.2
	3D	RBF	0.01	0.2
16	2D	RBF	1	0.1
	3D	RBF	1	0.05
32	2D	RBF	0.1	0.05
	3D	RBF	0.1	0.025
64	2D	RBF	1	0.05
	3D	RBF	0.1	0.1
128	2D	RBF	0.01	0.025
	3D	RBF	1	0.2
256	2D	RBF	0.01	0.1
	3D	RBF	0.1	0.05
512	2D	RBF	0.01	0.1
	3D	RBF	10	0.025
1024	2D	RBF	0.01	0.05
	3D	RBF	1	0.2

Finally, we train using the 11 346 image set that Ciurea et al.[27] built using a digital video camera. This very large, real-data training set improves overall performance.

EXPERIMENTS WITH 321 SONY REAL IMAGES

The training images are preprocessed, segmented and histogrammed in the same way as described above for the test images. The SVR kernel and parameters were selected based on the "1024" row of Table II; namely, for 3D, the radial basis function kernel with shape parameter $\gamma=0.2$ and penalty value $C=1$, while in 2D, these two parameters are set to 0.05 and 0.01, respectively.

Since it would be biased to train and test on the same set of images, we evaluate the illumination error using leave-one-out cross-validation procedure.[28] In the leave-one-out procedure, one image is selected for testing and the remaining 320 images are used for training. This is repeated 321 times, leaving a different image out of the training set each time, and the rms and median of the 321 resulting illumination estimation errors are calculated. The errors are significantly lower than those obtained with synthetic training data. The results and their comparison to the SoG,[23] Max RGB,[29] and Grayworld (GW)[30] are listed in Table IV. Table V tells us that 3D SVR has the best performance.

Table III. Comparison of competing illumination estimation methods. All methods are trained on synthetic images constructed from the same reflectance and illuminant spectra and then tested on the same SONY DXC930 (Ref. 21) camera images with identical preprocessing. Data marked by "*" are extracted from Ref. 21 (Table II, p. 992) while the data marked by "**" are extracted from Ref. 24 (Table 2, p. 79).

Method	RMS distance	RMS angle	Median angle
2D SVR	0.080	10.1	4.86
3D SVR	0.067	8.1	3.17
CC01	0.081	10.9*	-
CCMAP	0.071	9.9*	2.93**
CCMMSE	0.072	9.9*	-
CCMLM	0.072	9.9*	-
Neural network	0.070	9.5*	-
DB	-	12.25**	6.58**
GM	-	5.46**	2.92**

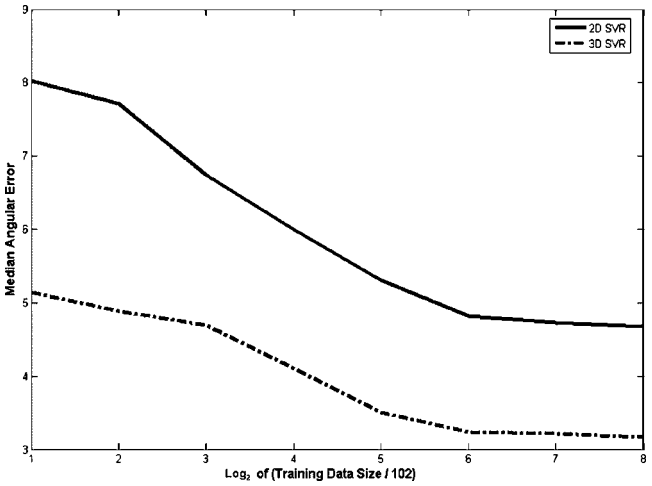


Figure 2. Median angular error in illumination chromaticity as a function of increasing training set size.

Although the median angular errors 2D SVR and SoG differ slightly, the difference is not statistically significant.

EXPERIMENTS WITH UNCALIBRATED 900 REAL IMAGES

We next consider Cardei's[4] set of 900 uncalibrated images taken using a variety of different digital cameras from Kodak, Olympus, HP, Fuji Polaroid, PDC, Canon, Ricoh, and Toshiba. A gray card was placed in each scene and its RGB value is used as the measure of the scene illumination. The SVR was based on a polynomial kernel function of degree 3 and 0.1 penalty. Its performance is also compared to the performance reported by Cardei[4] for Color by Correlation, the Neural Network, Shades of Grey, Max RGB, and Grayworld.

Table IV. Comparison of 2D and 3D SVR performance to SoG, Max RGB, Grayworld performance. The results involve real-data training and testing on the 321 SONY images. Errors are based on leave-one-out cross-validation evaluation and are reported in terms of both the rms angular chromaticity and distance error measures.

Method	SVR dimension/ norm power	Maximum angle	rms angle	Median angle	Maximum Distance ($\times 10^2$)	rms distance ($\times 10^2$)	Median distance ($\times 10^2$)
SVR	2D	22.99	10.06	4.65	16.41	7.5	3.41
	3D	24.66	8.069	2.17	16.03	6.3	3.07
SoG	6	28.70	9.027	3.97	19.77	6.21	2.83
Max RGB		36.24	12.28	6.44	25.01	8.25	4.46
GW		37.31	13.58	7.04	35.38	11.12	5.68

Table V. Comparison of the different algorithms via the Wilcoxon signed-rank test. A "+" means the algorithm listed in the corresponding the row is better than the one in corresponding column; "–" indicates the opposite; an "=" indicates that the performance of the respective algorithms is statistically equivalent.

	2D SVR	3D SVR	SoG (norm power = 6)	Max RGB	GW
2D SVR		–	=	+	+
3D SVR	+		+	+	+
SoG (norm power = 6)	=	–		+	+
Max RGB	–	–	–		–
GW	–	–	–	+	

Since a training set of 900 histograms is not very large, we used the histogram resampling strategy proposed by Cardei[4] in the context of neural network training to increase the training set size. Cardei et al. observed that each a histogram in the original training set could be used to generate many new training histograms by random sampling of its nonzero bins. Each sampling yields a new histogram of an "image" with the same illuminant chromaticity as the original. The number of possible sub-samplings is large, thereby making it possible to build a large training set based on real data extracted from a smaller number of images.

As before, we conduct leave-one-out tests. Hence, when we select an image for testing, we create a training set of 10 788 histograms from the remaining 899 real images and then measure the error in the SVR illuminant estimate for that one image. This process is repeated 900 times. The rms and median of the 900 errors is tabulated in Table VI. Table VII summarizes the Wilcoxon test among several of the algorithms. It also shows that on this training and test set, resampling the training set does not significantly change 3D SVRs performance; however, on the dataset discussed below, resampling does make a difference.

Table VI. Comparison of the performance of SVR to that of Color by Correlation, the Neural Network, SoG, Max RGB, Grayworld. The tests are based on leave-one-out cross-validation on a database of 900 uncalibrated images. The entries for C-by-C and the NN are from Ref. 4 (Table 7, p. 2385).

Method	Dimension/norm power	Maximum angle	rms angle	Median angle	Maximum distance ($\times 10^2$)	Mean distance ($\times 10^2$)	rms distance ($\times 10^2$)	Median distance ($\times 10^2$)
SVR (no resampling)	2D	20.43	4.47	2.40	18.40	2.40	3.27	1.74
	3D	17.46	3.94	2.02	15.42	2.09	2.94	1.40
SVR (with resampling)	3D	10.57	3.91	2.07	6.42	2.03	2.72	1.55
C-by-C	2D	-	-	-	-	2.92	3.89	-
NN	2D	-	-	-	-	2.26	2.76	-
SoG	6	19.71	4.99	3.02	15.96	2.96	3.80	2.19
Max RGB		27.16	6.39	2.96	22.79	3.36	4.75	2.17
GW		31.44	6.65	4.34	29.99	4.12	5.26	3.17

EXPERIMENTS WITH LARGE REAL IMAGE SET

Our final test with real data is based on the 11 346 real images extracted from over 2 h of digital video acquired with a SONY VX-2000. Ciurea et al.[27] built the database by partially automating the measurement of the illumination's RGB. Their setup consisted of a matte gray ball connected by a rod attached to the camera. In this way, the gray ball was made to appear at a fixed location at the edge of each video frame. The ball's pixels were thus easy to locate in each frame, and hence the chromaticity of the dominant illumination hitting the ball was easily measured as the average chromaticity of the pixels located in the ball's brightest region. The images include a wide variety of indoor and outdoor scenes including many with people in them.

Based on some initial experimentation, the RBF kernel function was chosen with 0.1 as the penalty parameter and 0.025 as the width parameter. All subsequent tests of SVR on the Ciurea database are based on these settings.

The size of the database means that leave-one-out validation is not feasible, although leave-*N*-out for a reasonable choice of *N* would be possible. In any case, it would not necessarily be a fair test because of the inherent regularities in the database. Since the database was constructed from a three-frame-per-second sampling of video clips, neighboring images in the database tend to be related. Hence, to ensure that SVR that the training and testing sets would be truly distinct, we partitioned the database into two sets in two different ways.

The first partitioning is based on geographical location. We take as the test set the 541 indoor and outdoor images taken exclusively in Scottsdale, Arizona. The training set becomes the 10 805 images in the remainder of the database, none of which is from Scottsdale. We call these datasets "Scottsdale" and "All-but-Scottsdale." The estimation errors are listed in Table VIII. The Wilcoxon signed-rank results are given in Table IX.

Table VII. Comparison of the performance based on the Wilcoxon signed-rank test. Labeling +, −, = as in Table V.

	2D SVR	3D SVR	3D SVR (with resampling)	SoG (norm power = 6)	MAX RGB	GW
2D SVR		−	−	+	+	+
3D SVR	+		=	+	+	+
3D SVR (with resampling)	+	=		+	+	+
SoG (norm power = 6)	−	−	−		=	+
MAX RGB	−	−	−	=		+
Wilcoxon signed-rank GW	−	−	−	−	−	

Table VIII. SVR (3D) illumination estimation errors for different training and test sets with comparisons to the SoG with norm power 6, Max RGB, and Grayworld methods.

Method	Training and test sets	Angular degrees			Distance ($\times 10^2$)		
		Maximum	rms	Median	Maximum	rms	Median
3D SVR	Train: all-but-Scottsdale	11.6	3.43	2.49	7.05	2.26	1.67
3D SVR with resampling		4.8	1.74	1.32	3.36	1.24	0.85
SoG	Test: Scottsdale	18.31	4.25	2.75	13.08	3.04	1.92
Max RGB		11.77	4.89	3.35	8.06	2.98	2.09
GW		24.96	6.78	4.47	25.05	5.10	3.26
3D SVR	Train: subset B	14.9	3.7	1.32	12.24	2.63	0.97
3D SVR with resampling		16.01	2.40	0.59	10.70	1.65	0.45
SoG	Test: subset A	35.87	7.82	4.89	27.99	5.70	3.44
Max RGB		27.42	10.63	6.41	21.72	7.65	4.48
GW		43.84	8.77	5.08	39.69	6.89	3.81
3D SVR	Train: subset A	16.8	3.6	2.87	15.00	2.61	2.06
3D SVR with resampling		11.7	1.41	1.22	8.44	1.03	0.08
SoG	Test: subset B	26.97	8.33	6.71	27.68	6.14	4.85
Max RGB		26.76	9.47	5.36	21.55	7.01	3.90
GW		33.12	9.24	7.62	32.41	7.28	5.68

Table IX. Comparison of the algorithms based on the Wilcoxon signed-rank test on angular error. SVR training set is All-but-Scottsdale. Test set for all methods is Scottsdale dataset. Labeling +, −, = as in Table V.

Method	3D SVR	3D SVR resampling	SoG (norm power = 6)	Max RGB	GW
3D SVR		−	=	+	+
3D SVR resampling	+		+	+	+
SoG (norm power = 6)	=	−		+	+
Max RGB	−	−	−		+
GW	−	−	−	−	

The second partitioning divides the entire database into two parts of similar size. Subset A includes 5343 images, and subset B includes 6003. Subset A contains images from Apache Trail, Burnaby Mountain, Camelback Mountain, CIC 2002 and Deer Lake. Subset B contains images from different locations: False Creek, Granville Island Market, Marine, Metrotown shopping center, Scottsdale, Simon Fraser University, and Whiteclyff Park. We then used A for training and B testing and vice versa. The results are again listed in Table VIII. Tables X gives the Wilcoxon sign results for this case.

Table X. Comparison of the algorithms based on the Wilcoxon signed-rank test on angular error. SVR training set is subset A. Test set for all methods is subset B. Labeling +, −, = as in Table V.

Test set = Subset B	3D SVR	3D SVR resampling	SoG (norm power = 6)	Max RGB	GW
3D SVR		−	=	+	+
3D SVR resampling	+		+	+	+
SoG (norm power = 6)	−	−		+	=
Max RGB	−	−	−		−
GW	−	−	=	+	

Table XI. Comparison of the algorithms based on the Wilcoxon signed-rank test on angular error. SVR training set is subset B. Test set for all methods is subset A. Labeling +, −, = as in Table V.

Method	3D SVR	3D SVR resampling	SoG (norm power = 6)	Max RGB	GW
3D SVR		−	+	+	+
3D SVR resampling	+		+	+	+
SoG (norm power = 6)	−	−		−	+
Max RGB	−	−	+		+
GW	−	−	−	−	

The histogram resampling strategy is also use here to expand the training dataset. The dataset excluding the Scottsdale images was increased to 162 075 histograms, dataset A to 154 462 and dataset B to 162 081. The corresponding test results are listed in Table VIII. Table XI includes the assessment through the Wilcoxon test. All of these Wilcoxon tests show that resampling strategy helps to improve the overall performance on this dataset.

CONCLUSION

Many previous methods of estimating the chromaticity of the scene illumination have been based in one way or another on statistics of the RGB colors arising in an image, independent of their spatial location or frequency of occurrence in the image. Support vector regression is a relatively new tool developed primarily for machine learning that can be applied in a similar way. We have tried it here, with good results, to the problem of learning the association between

color histograms and illumination chromaticity. Under almost the same experimentation conditions as those used by Barnard,[21,22] tests of the Shades-of-Gray, Neural-Network, Color-By-Correlation, Max RGB, and Grayworld methods, show that SVR performance generally is comparable to or better than these other methods. Using Ciurea's[27] large image database, SVR performance is shown, furthermore, to improve as the training set size is increased.

ACKNOWLEDGMENT

Funding was provided by the Natural Sciences and Research Council of Canada.

REFERENCES

[1] G. Finlayson, M. Drew, and B. Funt, "Color constancy: Generalized diagonal transforms suffice", J. Opt. Soc. Am. A **11**, 3011–3020 (1994).
[2] L. T. Maloney and B. A. Wandell, "Color constancy: A method for recovering surface spectral reflectance", J. Opt. Soc. Am. A **3**, 29–33 (1986).
[3] R. Buchsbaum, A. D. Jepson, and J. K. Tsotsos, "From [R,G,B] to surface reflectance: Computing color constancy descriptors in images", Perception **17**, 755–758 (1988).
[4] V. Cardei, B. Funt, and K. Barnard, "Estimating the scene illumination chromaticity using a neural network", J. Opt. Soc. Am. A **19**, 2374–2386 (2002).
[5] G. D. Finlayson, S. D. Hordley, and P. M. Hubel, "Color by correction: A simple, unifying framework for color constancy", IEEE Trans. Pattern Anal. Mach. Intell. **23**, 1209–1221 (2001).
[6] D. Forsyth, "A novel algorithm for color constancy", Int. J. Comput. Vis. **5**, 5–36 (1990).
[7] K. Barnard, L. Martin, and B. Funt, "Colour by correlation in a three dimensional colour space", *6th European Conference on Computer Vision* (Springer, Berlin, 2000), pp. 375–389.
[8] B. Funt, V. Cardei, and K. Barnard, "Learning color constancy", *Proc. IS&T/SID Fourth Color Imaging Conference* (IS&T, Springfield, VA, 1996), pp. 58–60.
[9] A. Smola and B. Schölkopf, "A tutorial on support vector regression", Stat. Comput. **14**, 199–222 (2003).
[10] V. Kecman, *Learning and Soft Computing* (MIT, Cambridge, MA, 2001), pp. 121–193.
[11] A. Chodorowski, T. Gustavsson, and U. Mattson, "Support vector machine for oral lesion classification", *Proc. 2002 IEEE International Symposium on Biomedical Imaging* (IEEE Press, Piscataway, NJ, 2000), pp. 173–176.
[12] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machine", IEEE Trans. Neural Netw. **13**, 415–425 (2002).
[13] Y. Lee and C. K. Lee, "Classification of multiple cancer types by multicategory support vector machines using gene expression data bioinformatics", Bioinformatics **19**, 1132–1139 (2003).
[14] Y. Lee, Y. Lin, and G. Wahba, "Multicategory support vector machine", *Proc. 33rd Symposium on the Interface* (Interface Foundation of No. America, Fairfax Station, VA, 2001).
[15] H. Yang, L. Chan, and I. King, "Support vector machine regression for volatile stock market prediction", *Intelligent Data Engineering and Automated Learning 2002* (Springer, Berlin, 2002), pp. 391–396.
[16] C.-H. Wu, C. C. Wei, M. H. Chang, D. C. Su, and J. M. Ho, "Travel time prediction with support vector regression", *Proc. IEEE Intelligent Transportation Conference* (IEEE, Piscataway, NJ, 2003) pp. 1438–1442.
[17] D. X. Zhao and L. Jiao, "Traffic flow time series prediction based on statistics learning theory", *Proc. IEEE 5th International Conference on Intelligent Transportation Systems* (IEEE, Piscataway, NJ, 2002) pp. 727–730.
[18] H. Van Khuu, H. K. Lee, and J. L. Tsai, "Machine learning with neural networks and support vector machines", Online Technical Report, available at: http://www.cs.wisc.edu/~hiep/Sources/Articles.
[19] C. C. Chang and C. J. Lin, *LIBSVM: A library for support vector machines* (2002); software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
[20] N. J. Bi and K. P. Bennett, "A geometric approach to support vector regression", Neurocomputing **55**, 79–108 (2003).
[21] K. Barnard, L. Martin, A. Coath, and B. Funt, "A comparison of computational colour constancy algorithms. Part two: Experiments on image data", IEEE Trans. Image Process. **11**, 985–996 (2002).
[22] K. Barnard, V. Cardei, and B. Funt, "A comparison of computational colour constancy algorithms. Part one: methodology and experiments with synthesized data", IEEE Trans. Image Process. **11**, 972–984 (2002).
[23] G. D. Finlayson and E. Trezzi, "Shades of gray and colour constancy", *Proc. IS&T/SID Twelfth Color Imaging Conference* (IS&T, Springfield, VA, 2004), pp. 37–41.
[24] S. D. Hordley and G. D. Finlayson, "Re-evaluating colour constancy algorithms", *Proc. 17th ICPR* (IEEE, Piscataway, NJ, 2004) Vol. 1, pp. 76–79.
[25] K. Barnard and B. Funt, "Camera characterization for color research", Color Res. Appl. **27**, 153–164 (2002).
[26] Simon Fraser University, Computational Vision Lab Data: www.cs.sfu.ca/~colour/data.
[27] F. Ciurea and B. Funt, "A large image database for color constancy research", *Proc. IS&T.SID Eleventh Color Imaging Conference* (IS&T, Springfield, VA, 2003) pp. 160–163.
[28] R. L. Eubank, *Spline Smoothing and Nonparametric Regression* (Marcel Dekker, New York, 1988).
[29] G. D. Finlayson, "Retinex viewed as a gamut mapping theory of color constancy", *Proc. AIC International Color Association Color 97* (Central Bureau of the CIE, Wien, Austria, 1997) pp. 527–530.
[30] B. Buchsbaum, "A spatial processor model for object color perception", J. Franklin Inst. **31**, 1–26 (1980).