

Enhancing Resolution Along Multiple Imaging Dimensions Using Assorted Pixels

Srinivasa G. Narasimhan, *Member, IEEE*, and Shree K. Nayar

Abstract—Multisampled imaging is a general framework for using pixels on an image detector to simultaneously sample multiple dimensions of imaging (space, time, spectrum, brightness, polarization, etc.). The mosaic of red, green, and blue spectral filters found in most solid-state color cameras is one example of multisampled imaging. We briefly describe how multisampling can be used to explore other dimensions of imaging. Once such an image is captured, smooth reconstructions along the individual dimensions can be obtained using standard interpolation algorithms. Typically, this results in a substantial reduction of resolution (and, hence, image quality). One can extract significantly greater resolution in each dimension by noting that the light fields associated with real scenes have enormous redundancies within them, causing different dimensions to be highly correlated. Hence, multisampled images can be better interpolated using local structural models that are learned offline from a diverse set of training images. The specific type of structural models we use are based on polynomial functions of measured image intensities. They are very effective as well as computationally efficient. We demonstrate the benefits of structural interpolation using three specific applications. These are 1) traditional color imaging with a mosaic of color filters, 2) high dynamic range monochrome imaging using a mosaic of exposure filters, and 3) high dynamic range color imaging using a mosaic of overlapping color and exposure filters.

Index Terms—Image formation, multisampling, dynamic range, color, resolution, interpolation, structural models, learning, Bayer pattern.

1 MULTISAMPLED IMAGING USING ASSORTED PIXELS

CURRENTLY, vision algorithms rely on images with 8 bits of brightness or color at each pixel. Images of such quality are simply inadequate for many real-world applications. Significant advances in imaging can be made by exploring the fundamental trade offs that exist between various dimensions of imaging (see Fig. 1). The relative importance of these dimensions clearly depends on the application at hand. In any practical scenario, however, we are given a finite number of pixels (residing on one or more detectors) to sample the imaging dimensions. Therefore, it is beneficial to view imaging as the judicious assignment of resources (pixels) to the dimensions of imaging that are relevant to the application.

Different pixel assignments can be viewed as different types of samplings of the imaging dimensions. In all cases, however, more than one dimension is simultaneously sampled. In the simplest case of a gray-scale image, image brightness and image space are sampled simultaneously. More interesting examples result from using image detectors made of an assortment of pixels, as shown in Fig. 2. Fig. 2a shows the popular Bayer mosaic [1] of red, green, and blue spectral filters placed adjacent to pixels on a detector. Since multiple color measurements cannot be captured simultaneously at a pixel, the pixels are assigned to specific colors to trade off spatial resolution for spectral resolution. Over the last three decades, various color mosaics have been

suggested, each one resulting in a different trade off (see [7], [8], [17], [20], [14]).

Historically, multisampled imaging has only been used in the form of color mosaics. Only recently has the approach been used to explore other imaging dimensions. Fig. 2b shows the mosaic of neutral density filters with different transmittances used in [19] to enhance an image detector's dynamic range. In this case, spatial resolution is traded-off for brightness resolution (dynamic range). In [22], spatially varying transmittance and spectral filters were used with regular wide FOV mosaicing to yield high dynamic range and multispectral mosaics. Fig. 2c shows how space, dynamic range, and color can be sampled simultaneously by using a mosaic of filters with different spectral responses and transmittances. This type of multisampling is novel and, as we shall show, results in high dynamic range color images. Another example of assorted pixels is proposed in [2], where a mosaic of polarization filters with different orientations is used to estimate the polarization parameters of light reflected by scene points. This idea can be used in conjunction with a spectral mosaic, as shown in Fig. 2d, to achieve simultaneous capture of polarization and color.

Multisampled imaging can be exploited in many other ways. Fig. 2e shows how temporal sampling can be used with exposure sampling. This example is related to the idea of sequential exposure change proposed in [18], [9], [16] to enhance dynamic range. However, it is different in that the exposure is varied as a periodic function of time, enabling the generation of high dynamic range, high frame rate video. The closest implementation appears to be the one described in [12], where the electronic gain of the camera is varied periodically to achieve the same effect. A more sophisticated implementation may sample space, time, exposure, and spectrum simultaneously, as shown in Fig. 2f.

The above examples illustrate that multisampling provides a general framework for designing imaging systems that extract information that is most pertinent to the

- S.G. Narasimhan is with the Robotics Institute (NSH 3211), Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213.
E-mail: srinivas@cs.cmu.edu.
- S.K. Nayar is with the Computer Science Department (Room 450), Columbia University, 500 West 120th Street, New York, NY 10027.
E-mail: nayar@cs.columbia.edu.

Manuscript received 1 May 2003; revised 2 May 2004; accepted 3 Aug. 2004; published online 10 Feb. 2005.

Recommended for acceptance by D. Fleet.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0061-0503.

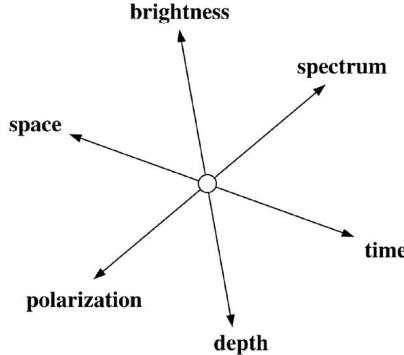


Fig. 1. A few dimensions of imaging. Pixels on an image detector may be assigned to multiple dimensions in a variety of ways depending on the needs of the application. Such pixels are called “assorted pixels.”

application. Though our focus is on the visible light spectrum, multisampling is, in principle, applicable to any form of electromagnetic radiation. Therefore, the pixel assortments and reconstruction methods we describe in this paper are also relevant to other imaging modalities, such as X-ray, magnetic resonance (MR), and infrared (IR). Furthermore, the examples we discuss are two-dimensional, but the methods we propose are directly applicable to higher-dimensional imaging problems, such as ones found in tomography and microscopy.

2 LEARNED STRUCTURAL MODELS FOR RECONSTRUCTION

How do we reconstruct the desired image from a captured multisampled one? Nyquist’s theory [4] tells us that, for a continuous signal to be perfectly reconstructed from its discrete samples, the sampling frequency must be at least twice the largest frequency in the signal. In the case of an image of a scene, the optical image is sampled at a frequency determined by the size of the detector and the number of pixels on it. In general, there is no guarantee that this sampling frequency satisfies Nyquist’s criterion. Therefore, when a traditional interpolation technique is used to enhance spatial resolution, it is bound to introduce errors in the form of blurring and/or aliasing. In the case of multisampled images (see Fig. 2), the assignment of pixels to multiple dimensions causes further undersampling of scene radiance along at least some dimensions. As a result, conventional interpolation methods are even less effective.

Our objective is to overcome the limits imposed by Nyquist’s theory by using prior models that capture redundancies inherent in images. The physical structures of real-world objects, their reflectances and illuminations, impose strong constraints on the light fields of scenes. This causes different imaging dimensions to be highly correlated with each other. Therefore, a local mapping function can be learned from a set of multisampled images and their corresponding correct (high-quality) images. As we shall see, it is often beneficial to use multiple mapping functions, as shown in Fig. 5a. Then, given a novel multisampled image, these mapping functions can be used to reconstruct an image that has enhanced resolution in each of the dimensions of interest (see Fig. 5b). We refer to these learned mapping functions as *local structural models*.

We are not claiming that any learned model (mapping function) can guarantee perfect reconstruction of an image

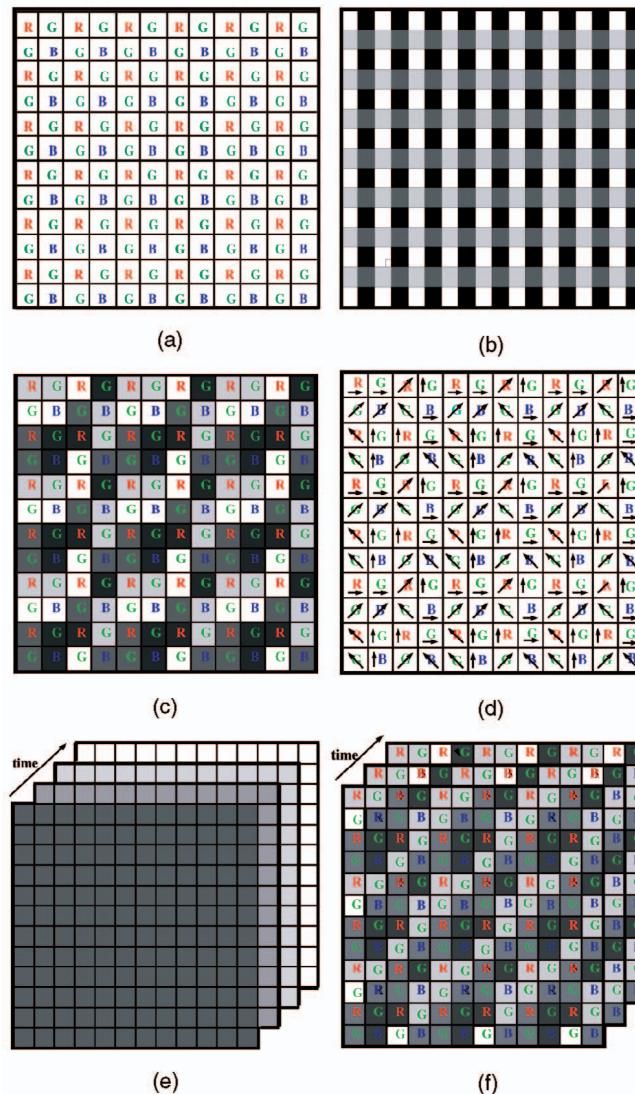


Fig. 2. A few examples of multisampled imaging using assorted pixels. (a) A color mosaic. Such mosaics are widely used in solid-state color cameras. (b) An exposure mosaic. (c) A mosaic that includes different colors and exposures. (d) A mosaic using color and polarization filters. (e) and (f) Multisampling can also involve varying exposure and/or color over space and time.

irrespective of its sampling. Instead, we posit that the models can provide the most likely (frequently occurring in real scenes) value for an image point given neighboring measurements. This permits us to develop reconstruction algorithms that compute high-quality images from low-resolution multisampled images.

The general idea of learning interpolation functions is not new. In [10], a probabilistic Markov network is trained to learn the relationship between sharp and blurred images and then used to increase spatial resolution of an image. In [3], a linear system of equations is solved to estimate a high-resolution image from a sequence of low-resolution images wherein the object of interest is in motion. Note that both of these algorithms are developed to improve spatial resolution, while our interest is in resolution enhancement along multiple imaging dimensions.

Learning-based algorithms have also been applied to the problem of interpolating images captured using color



Fig. 3. Some of the 50 high-quality images ($2,000 \times 2,000$ pixels, 12 bits per color channel) used to train the local structural models described in Sections 4, 5, and 6.

mosaics. The most relevant among these are the works of [24], [14] that estimate an interpolation kernel from training data (high-quality color images of test patterns and their corresponding color mosaic images). The same problem was addressed in [5] using a Bayesian method.

We are interested in a general method that can interpolate not just color mosaic images, but any type of multisampled data. For this, we propose the use of a structural model where each reconstructed value is a polynomial function of the image brightnesses measured within a local neighborhood. The size of the neighborhood and the degree of the polynomial vary with the type of multisampled data being processed. It turns out that the model proposed in [24], [14] is a special instance of our model as it is a first-order polynomial applied to the specific case of color mosaic images. As we shall see, our polynomial model produces excellent results for a variety of multisampled images. Since it uses polynomials, our method is very efficient and can be easily implemented in hardware. In short, it is simple enough to be incorporated into any imaging device (digital still or video camera, for instance).

3 TRAINING USING HIGH QUALITY IMAGES

Since we wish to learn our model parameters, we need a set of high-quality training images. These could be real images of scenes, synthetic images generated by rendering, or some combination of the two. Real images can be acquired using professional grade cameras whose performance we wish to emulate using lower quality multisampling systems. Since we want our model to be general, the set of training images must adequately represent a wide range of scene features. For instance, images of urban settings, landscapes, and indoor spaces may be included. Rotated and magnified versions of the images can be used to capture the effects of scale and orientation. In addition, the images may span the gamut of illumination conditions encountered in practice, varying from indoor lighting to overcast and sunny conditions outdoors. Synthetic images are useful as one can easily include in them specific features that are relevant to the application. For instance, edges, lines, curves, or more complex features are easily rendered at various orientations and scales. In addition, specific types of textures can be synthesized.

Some of the 50 high-quality images we have used in our experiments are shown in Fig. 3. Each of these is a $2,000 \times 2,000$ color (red, green, blue) image with 12 bits of information in each color channel. These images were captured using a 35mm Nikon N90 SLR film camera and scanned using a 12-bit Nikon LS2000 slide scanner. Though

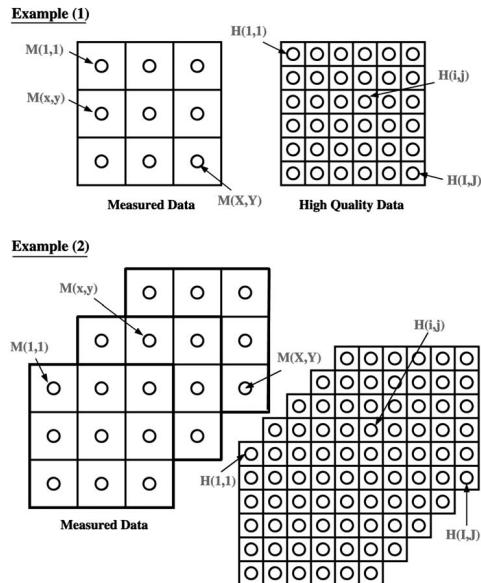


Fig. 4. Relationship between measured (low-quality) and desired (high-quality) data.

the total number of training images is small, they include a sufficiently large number of local (say, 7×7 pixels) appearances for training our structural models.

Given such high-quality images, it is easy to generate a corresponding set of low-quality multisampled images. For instance, given a $2,000 \times 2,000$ RGB image with 12 bits per pixel per color channel, simple downsampling in space, color, and brightness results in a $1,000 \times 1,000$, 8 bits per pixel multisampled image with the sampling pattern shown in Fig. 2c. We refer to this process of generating multisampled images from high-quality images as *downgrading*.

With the high-quality images and their corresponding (downgraded) multisampled images in place, we can learn the parameters of our structural model. A structural model is a function f that relates measured data $M(x, y)$ in a multisampled image to a desired value $H(i, j)$ in the high-quality training image (see Fig. 4):

$$(X, Y) \text{ is a neighborhood of } (i, j), \text{ for each } (i, j) \text{ in high quality images}$$

$$H(i, j) = f(M(1, 1), \dots, M(x, y), \dots, M(X, Y)), \quad (1)$$

where X and Y define some neighborhood of measured data around or close to the high-quality value $H(i, j)$. Note that a large variety of linear and nonlinear functions can be used for f . Depending on the nature of f , a suitable estimation algorithm can be used to find its parameters from the training data. In the experiments reported here, we have chosen a simple function; the desired value $H(i, j)$ is expressed as a low-order polynomial in the measured values $M(x, y)$. The coefficients of the polynomial define the structural model. The polynomial structural model is linear in its coefficients. Therefore, the coefficients can be efficiently computed from training data using linear regression. As we will see, this simple structural model is very effective in predicting local appearances.

Note that a single structural model may be inadequate. This is particularly the case when we are sampling the scene simultaneously along several dimensions. Such an example is shown in Fig. 2. If we set aside the measured data and focus on

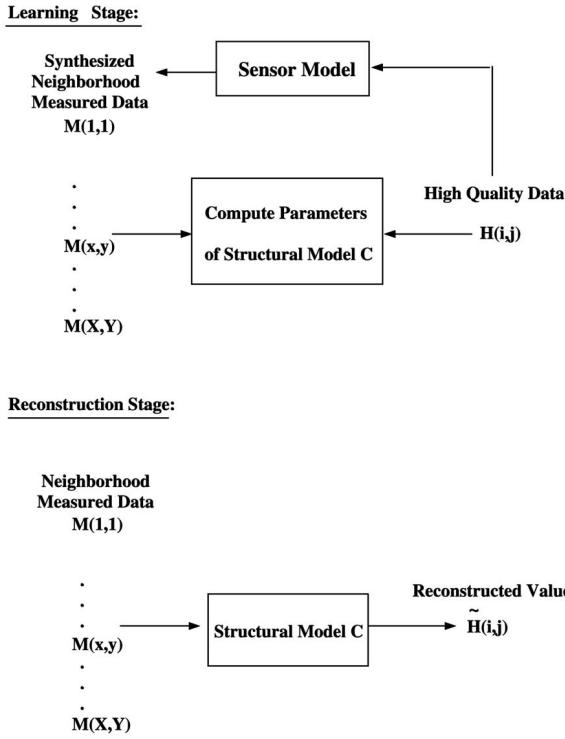


Fig. 5. Estimating structural models and using them to generate high-quality data.

the type of multisampling used (see Fig. 2), we see that pixels can have different types of neighborhood sampling patterns. With respect to Fig. 2, we may be interested in high-quality *R*, *G*, and *B* values at each of the pixels shown. However, the pattern of exposures and spectral filters around a pixel is different from the patterns around its neighbors. Hence, using a single structural model to predict the brightness of the image in any one color channel may be asking for too much from the model itself. That is, if we want our models to be compact (small number of coefficients) and effective, we cannot expect them to capture variations in scenes as well as changes in the sampling pattern. Hence, we use a single structural model for each type of local sampling pattern. Since our imaging dimensions are sampled in a uniform manner, in all cases, we have a small number of local sampling patterns. Therefore, only a small number of structural models are needed. During reconstruction, given a pixel of interest, the appropriate structural model is invoked based on the pixel's known neighborhood sampling pattern.

4 SPATIALLY VARYING COLOR (SVC)

Most color cameras have a single image detector with a mosaic of red, green, and blue spectral filters on it. The resulting image is, hence, a widely used type of multi-sampled image. We refer to it as a spatially varying color (SVC) image. When one uses an NTSC color camera, the output of the camera is nothing but an interpolated SVC image. Color cameras are notorious for producing inadequate spatial resolution and this is exactly the problem we seek to overcome using structural models.

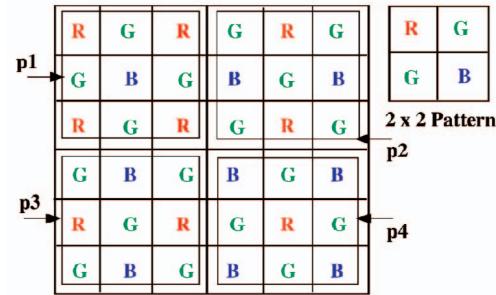


Fig. 6. Spatially varying color (SVC) pattern on a Bayer mosaic. Given a neighborhood size, all possible sampling patterns in the mosaic must be one of four types. In the case of a 3×3 neighborhood, these patterns are p_1 , p_2 , p_3 , and p_4 .

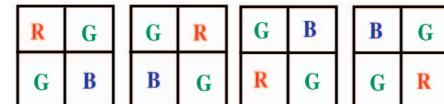


Fig. 7. Cyclic shifts of columns and rows of the 2×2 Bayer-mosaic neighborhood correspond to the four unique types of local patterns possible. For examples, see $p_1 \dots p_4$ in Fig. 6.

Since this is our first example, we will use it to describe some of the general aspects of our approach.

4.1 Bayer Color Mosaic

Several types of color mosaics have been implemented in the past [1], [7], [8], [17], [20], [14]. However, the most popular of these is the **Bayer pattern** [1], shown in Fig. 6. Since the human eye is more sensitive to the green channel, the Bayer pattern uses more green filters than it does red and blue ones. Specifically, the spatial resolutions of green, red, and blue are 50 percent, 25 percent, and 25 percent, respectively. Note that the entire mosaic is obtained by repeating the 2×2 pattern shown on the right in Fig. 6. Therefore, given a neighborhood size, all neighborhoods in a Bayer mosaic must have one of four possible sampling patterns. The different types of sampling patterns exactly correspond to the distinct cyclic shifts of the rows and columns of the 2×2 pattern (see Fig. 7). So, if the neighborhood is of size 3×3 , the resulting patterns are p_1 , p_2 , p_3 , and p_4 , shown in Fig. 6.

4.2 SVC Structural Model

From the measured SVC image, we wish to compute three color values (red, green, and blue) at each pixel, even though each pixel in the SVC image provides a single color measurement. Let the measured SVC image be denoted by \mathbf{M} and the desired high-quality color image by \mathbf{H} . A structural model relates each color value in \mathbf{H} to the measured data within a small neighborhood in \mathbf{M} . This neighborhood includes measurements of different colors and, hence, the model implicitly accounts for correlations between different color channels.

As shown in Fig. 8, let M_p be the measured data in a neighborhood with sampling pattern p , and $H_p(i + 0.5, j + 0.5, \lambda)$ be the high-quality color value at the center of the neighborhood. The center is off-grid because the neighborhood is an even number of pixels in width and height. Then, a polynomial structural model can be written as:

each row: IS_pl * N_p entries

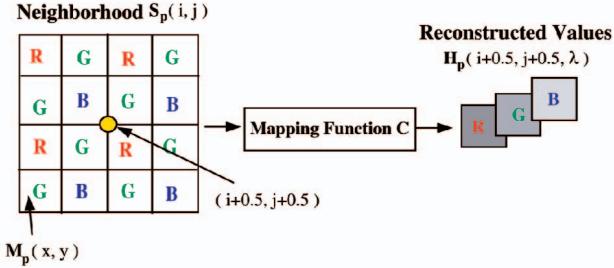


Fig. 8. The measured data M_p in the neighborhood $S_p(i, j)$ around pixel (i, j) are related to the high-quality color values $H_p(i + 0.5, j + 0.5, \lambda)$ via a polynomial with coefficients C_p .

$$H_p(i + 0.5, j + 0.5, \lambda) = \sum_{(x,y) \in S_p(i,j)} \sum_{(k \neq x, l \neq y) \in S_p(i,j)} C_p(a, b, c, d, \lambda, n) M_p^n(x, y) M_p^q(k, l). \quad (2)$$

1 C_p for each pattern p

$S_p(i, j)$ is the neighborhood of pixel (i, j) , N_p is the order of the polynomial, and C_p are the polynomial coefficients for the pattern p . The coefficient indices (a, b, c, d) are equal to $(x - i, y - j, k - i, l - j)$.

The product $M_p(x, y) M_p(k, l)$ explicitly represents the correlation between different pixels in the neighborhood. For efficiency, we have not used these cross-terms in our implementations. We found that very good results are obtained even when each desired value is expressed as just a sum of polynomial functions of the individual pixel measurements:

$$H_p(i + 0.5, j + 0.5, \lambda) = \sum_{(x,y) \in S_p(i,j)} \sum_{n=0}^{N_p} C_p(a, b, \lambda, n) M_p^n(x, y). \quad (3)$$

polynomial at each pixel

In short, a polynomial is computed for each on-grid pixel of a neighborhood in the low-quality SVC data, and the aggregation of polynomials of all pixels in that neighborhood gives the high-quality color values at the corresponding off-grid center pixel. It should be noted that, in general, the method holds for any neighborhood size and shape and for off-grid as well as on-grid computation.

The mapping function (3) for each color λ and each local pattern type p can be conveniently rewritten using matrices and vectors, as shown in Fig. 9:

$$H_p(\lambda) = A_p C_p(\lambda). \quad (4)$$

For a given pattern type p and color λ , A_p is the measurement matrix. The rows of A_p correspond to the different neighborhoods in the image that have the pattern p . Each row includes all the relevant powers (up to N_p) of the measured data M_p within the neighborhood. The vector $C_p(\lambda)$ includes the coefficients of the polynomial mapping function and the vector $H_p(\lambda)$ includes the desired high-quality values at the off-grid neighborhood centers. The estimation of the model parameters C_p can then be posed as a least squares problem:

$$C_p(\lambda) = (A_p^T A_p)^{-1} A_p^T H_p(\lambda). \quad (5)$$

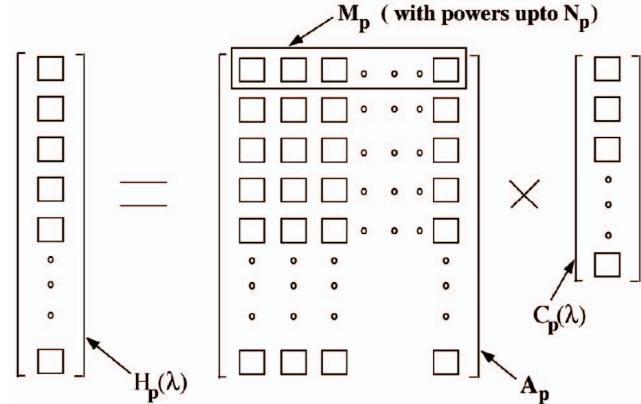


Fig. 9. The mapping function in (3) can be expressed as a linear system using matrices and vectors. For a given pattern p and color λ , A_p is the measurement matrix, $C_p(\lambda)$ is the coefficient vector, and $H_p(\lambda)$ is the reconstruction vector.

When the signal-to-noise characteristics of the image detector are known, (5) can be estimated using weighted least squares to achieve greater accuracy by solving:

$$A_p^T W_p^2 A_p C_p(\lambda) = A_p^T W_p^2 H_p(\lambda), \quad (6)$$

where W_p is a diagonal matrix that can be used to assign weights to different neighborhoods based on their signal-to-noise characteristics. $A_p^T W_p^2 A_p$ is called the weighted normal matrix and $A_p^T W_p^2 H_p(\lambda)$ is called the weighted regression vector.

4.3 Total Number of Coefficients

The number of coefficients in the model (3) can be calculated as follows: Let the neighborhood size be $u \times v$ and the polynomial order corresponding to each pattern p be N_p . Let the number of distinct local patterns in the SVC image be P and the number of color channels be Λ . Then, the total number of coefficients needed for structural interpolation is:

$$|C| = \left(P + u * v * \sum_{p=1}^P N_p \right) * \Lambda. \quad (7)$$

why + P, not * P

For the Bayer mosaic, $P = 4$ and $\Lambda = 3$ (R, G, B). If we use $N_p = 2$ and $u = v = 6$, the total number of coefficients is 876. Since these coefficients are learned from real data, they yield greater precision during interpolation than standard interpolation kernels. In addition, they are very efficient to apply. Since there are $P = 4$ distinct patterns, only 219 (a quarter) of the coefficients are used for computing the three color values at a pixel. Note that the polynomial model is linear in the coefficients. Hence, structural interpolation can be implemented in real-time using a set of linear filters that act on the captured image and its powers (up to N_p).

4.4 Experiments

A total of 30 high-quality training images (see Fig. 3) were used to compute the structural model for SVC image interpolation. Each image is downgraded to obtain a corresponding Bayer-type SVC image. For each of the four sampling patterns in the Bayer mosaic and for each of the three colors, the appropriate image neighborhoods were used to compute the measurement matrix A_p and the reconstruction vector $H_p(\lambda)$. While computing these, one additional step was taken; each measurement is normalized.

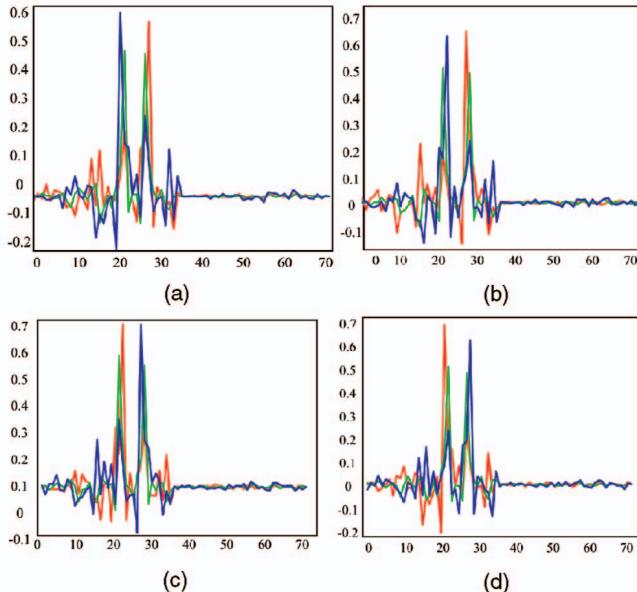


Fig. 10. Coefficients of the SVC structural model for the four different patterns. A second order polynomial is used and the neighborhood size is 6×6 . The first 36 coefficients denote the coefficients corresponding to the first order terms and the remaining correspond to the second order terms in the SVC structural model. The first order terms capture most of the energy. (a) RGB coefficient for SVC pattern 1. (b) RGB coefficient for SVC pattern 2. (c) RGB coefficient for SVC pattern 3. (d) RGB coefficient for SVC pattern 4.

by the energy within its neighborhood to make the structural model insensitive to changes in illumination intensity and camera gain. The resulting A_p and $H_p(\lambda)$ are used to find the coefficient vector $C_p(\lambda)$ using linear regression (see (5)). In our implementation, we used the parameter values $P = 4$ (Bayer), $N_p = 2$, $u = v = 6$, and $\Lambda = 3(R, G, B)$ to get a total of 876 coefficients. The coefficients computed using all the training images are shown in Fig. 10.

The above structural model was used to interpolate 20 test SVC images that are different from the ones used for training. In Fig. 11a, a high-quality (8 bits per color channel) image is shown. Fig. 11b shows the corresponding (downgraded) SVC image. This is really a single channel 8-bit image and its pixels are shown in color only to illustrate the Bayer pattern. Fig. 11c shows a color image computed from the SVC image using bicubic interpolation. As is usually done, the three channels are interpolated separately using their respective data in the SVC image. The magnified image region clearly shows that bicubic interpolation results in a loss of high frequencies; the edges of the tree branches and the squirrels are severely blurred. Fig. 11d shows the result of applying structural interpolation. Note that the result is of high-quality with minimal loss of details. Three more results are shown in Figs. 12, 13, and 14.

We also compared the structural interpolation against other commonly used techniques for demosaicing, such as [6], [11], [15], [13]. A comparison between the reconstruction results obtained is shown in Fig. 15. The structural model produces results that are visually comparable to the best demosaicing method while not introducing spurious high frequencies.

4.5 Quantitative Evaluation

We have also quantitatively verified our results. Figs. 16 and 17 show histograms of the luminance error for bicubic

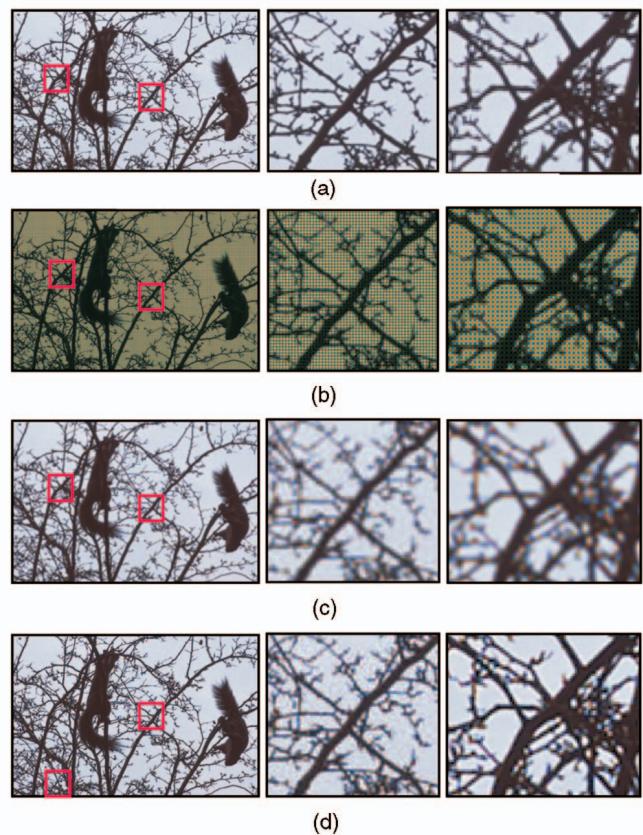


Fig. 11. **(SVC Reconstruction)** (a) Original (high-quality) color image with 8 bits per color channel. (b) SVC image obtained by downgrading the original image. The pixels in this image are shown in color only to illustrate the Bayer mosaic. Color image computed from the SVC image using (c) bicubic interpolation and (d) structural interpolation.

and structural interpolation as well as four other commonly used demosaicing techniques. These histograms are computed using all 20 test images (not just the ones in Figs. 11, 12, 13, and 14). The difference between the two histograms may appear to be small but is significant because a large fraction of the pixels in the 20 images belong to “flat” image regions that are easy to interpolate for all methods. The RMS errors (computed over all 20 images) are 6.12 and 3.27 gray levels for bicubic and structural interpolation, respectively. The other demosaicing methods have RMS error values in between. These quantitative results indicate that structural models clearly outperform the other commonly used demosaicing techniques.

We also performed experiments to study the effectiveness and stability of the learning technique. First, we performed testing on the images used for training. The average RMS error in luminance was very low, as expected. Second, we performed the experiments where the testing and training images were not identical but similar (for example, different types of grass, trees, etc). Even in this case, the RMS error was low, as shown in Fig. 18. Then, we performed experiments by randomly selecting a set of images from the training set. The training set of images was different from the testing set and contained scenes of similar as well as dissimilar content (for example, urban versus natural scenes). For very small training sets, the average luminance RMS error is high. However, the average RMS error is more or less stable for larger training sets, indicating the stability in the learning technique. It is conceivable that more complex learning

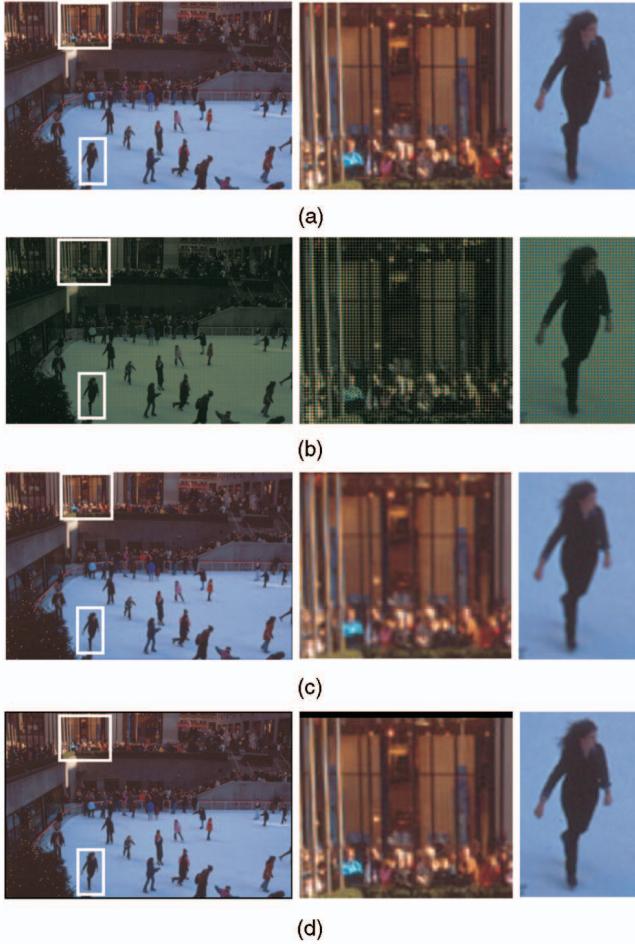


Fig. 12. **(SVC Reconstruction)** (a) Original (high-quality) color image with 8 bits per color channel. (b) SVC image obtained by downgrading the original image. The pixels in this image are shown in color only to illustrate the Bayer mosaic. Color image computed from the SVC image using (c) bicubic interpolation and (d) structural interpolation.

techniques (carefully chosen, for example, from the machine learning literature) could be used to obtain better results. However, our focus here is to present a simple and efficient technique that may also be implemented in hardware.

5 SPATIALLY VARYING EXPOSURES (SVE)

In [19], it was shown that the dynamic range of a gray-scale image detector can be significantly enhanced by assigning different exposures (neutral density filters) to pixels, as shown in Fig. 19. This is yet another example of a multi-sampled image and is referred to as a spatially varying exposure (SVE) image. In [19], standard bicubic interpolation was used to reconstruct a high dynamic range gray-scale image from the captured SVE image; first, saturated and dark pixels are eliminated, then all remaining measurements are normalized by their exposure values, and, finally, bicubic interpolation is used to find the brightness values at the saturated and dark pixels. As expected, the resulting image has enhanced dynamic range but lower spatial resolution. In this section, we apply structural interpolation to SVE images and show how it outperforms bicubic interpolation.

Consider the array of pixels shown in Fig. 19. The brightness level associated with each pixel represents its sensitivity such that the brighter pixels have greater

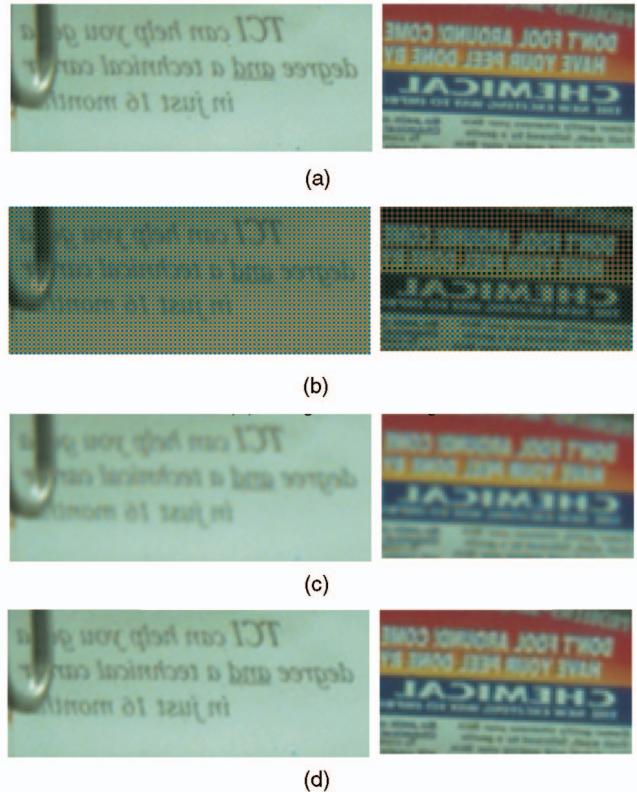


Fig. 13. **(SVC Reconstruction)** (a) Zoomed in regions of an original color image with 8 bits per color channel. (b) SVC image obtained by downgrading the original image. The pixels in this image are shown in color only to illustrate the Bayer mosaic. Color image computed from the SVC image using (c) bicubic interpolation and (d) structural interpolation.

exposure to image irradiance and the darker ones have lower exposure. In the example shown, four neighboring pixels have different exposures ($e_1 < e_2 < e_3 < e_4$) and this 2×2 neighborhood is repeated over the detector array.

The key feature here is that we are simultaneously sampling the spatial dimensions as well as the exposure dimension of image irradiance. Note that, when a pixel is saturated in the acquired image, it is likely to have a neighbor that is not and, when a pixel produces zero brightness, it is likely to have a neighbor that produces nonzero brightness. Our goal is to exploit this spatio-exposure sampling and compute a high dynamic range image of the scene.

It is worth noting that we are by no means restricted to the mosaic shown in Fig. 19. The number of discrete exposures can differ (hence, the number of local pattern types) and the pattern does not have to be periodic. There may be instances where a random exposure mosaic may be useful. The mosaic can be implemented in many ways as discussed in [19]. One approach is to place a mask with cells of different optical transparencies adjacent to the detector array. As discussed in [19], other implementations also result in the same effect, namely, a detector array with spatially varying exposures.

5.1 SVE Structural Model

As in the SVC case, let the measured SVE data be M and the corresponding high dynamic range data be H . Again, we use simple polynomial structural models to relate H and M . If the SVE detector uses only four discrete exposures (see Fig. 19), it is easy to see that a neighborhood of any given

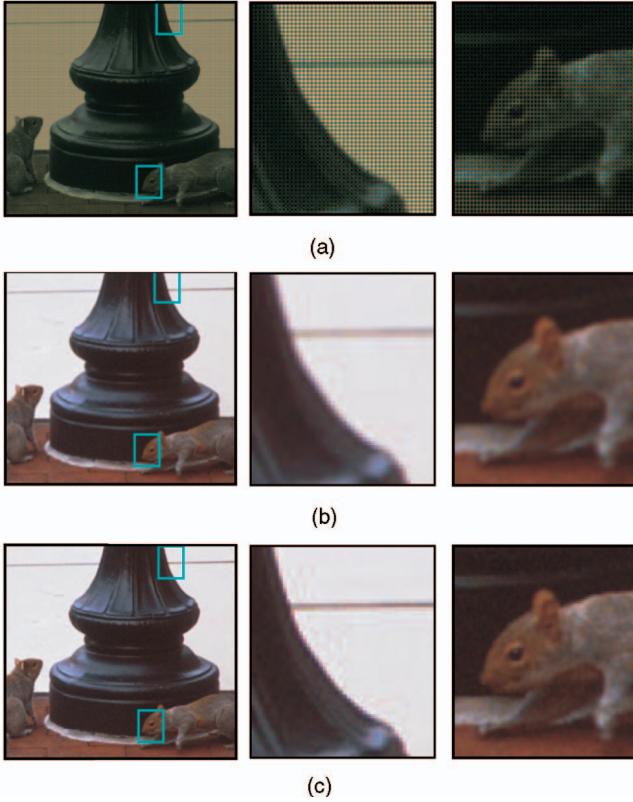


Fig. 14. **(SVC Reconstruction)** (a) SVC image obtained by downgrading the original image. The pixels in this image are shown in color only to illustrate the Bayer mosaic. Color image computed from the SVC image using (b) bicubic interpolation and (c) structural interpolation.

size can have only one of four different sampling patterns ($P = 4$). Therefore, for each sampling pattern p , a polynomial structural model is used that relates the captured data M_p within the neighborhood to the high dynamic range value H_p at the center of the neighborhood:

$$H_p(i + 0.5, j + 0.5) = \sum_{(x,y) \in S_p(i,j)} \sum_{n=0}^{N_p} C_p(a, b, n) M_p^n(x, y), \quad (8)$$

where, as before, $(a, b) = (x - i, y - j)$, $S_p(i, j)$ is the neighborhood of pixel (i, j) , N_p is the order of the polynomial mapping, and C_p are the polynomial coefficients for the pattern p . Note that there is only one channel in this case (gray-scale) and, hence, the parameter λ is omitted. To avoid asymmetries in the number of exposures in a neighborhood, we use square neighborhoods of even length and width. The high dynamic range value is computed at the off-grid neighborhood center $(i + 0.5, j + 0.5)$. It should be noted that, in general, the method holds for any neighborhood size and shape and off-grid as well as on-grid computation. Once again, we could have used a more general mapping function like the one in (2), but have chosen a simpler model as it is does very well and is more efficient to learn and apply.

The above model is rewritten in terms of a measurement matrix A_p and a reconstruction vector H_p , and the coefficients C_p are found using (5). The number of coefficients in the SVE structural model is determined as:

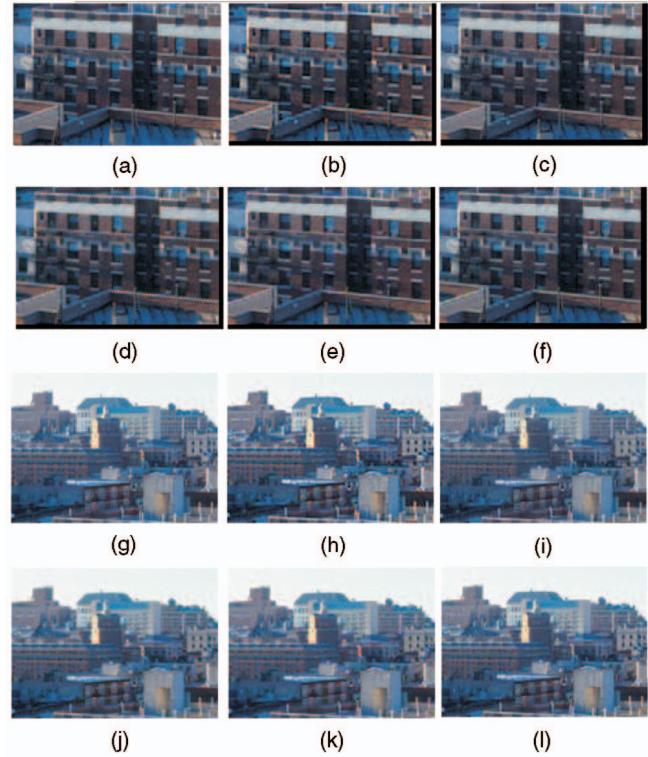


Fig. 15. **(SVC Reconstruction)** Comparison of the reconstruction technique with bicubic interpolation and four other commonly used demosaicing techniques. Two regions of an urban scene are shown. As can be seen, the visual quality of our reconstruction is at least as good as the best demosaicing technique with the added advantage of avoiding spurious high frequencies. See Fig. 16 for a quantitative comparison. The methods used for demosaicing were obtained from four different papers [6], [11], [15], [13]. The implementations of these demosaicing methods (excluding bicubic interpolation) were obtained from Dr. Rajeev Ramanath [21]. (a) Original. (b) Structural Model. (c) Freeman. (d) Cok. (e) Laroche. (f) Hamilton. (g) Original. (h) Structural Model. (i) Freeman. (j) Cok. (k) Laroche. (l) Hamilton.

$$|C| = P + u * v * \sum_{p=1}^P N_p. \quad (9)$$

In our implementation, we have used $P = 4$, $N_p = 2$, and $u = v = 6$, which gives a total of 292 coefficients. Since $P = 4$, only 73 coefficients are needed for reconstructing each pixel in the image.

5.2 Experiments

The SVE structural model was trained using 12-bit gray-scale versions of six of the images shown in Fig. 3 and their corresponding downgraded 8-bit SVE images. Each SVE image was obtained by applying the exposure pattern shown in Fig. 19 (with $e_4 = 4e_3 = 16e_2 = 64e_1$) to the original image, followed by a downgrade from 12 bits to 8 bits. The structural model was tested using six test images, one of which is shown in Fig. 20. Fig. 20a shows the original 12-bit image, Fig. 20b shows the downgraded 8-bit SVE image, Fig. 20c shows a 12-bit image obtained by bicubic interpolation of the SVE image, and Fig. 20d shows the 12-bit image obtained by structural interpolation. The magnified images shown on the right are histogram equalized to bring out the details (in the clouds and walls) that are lost during bicubic interpolation but extracted by structural interpolation. Fig. 21 compares

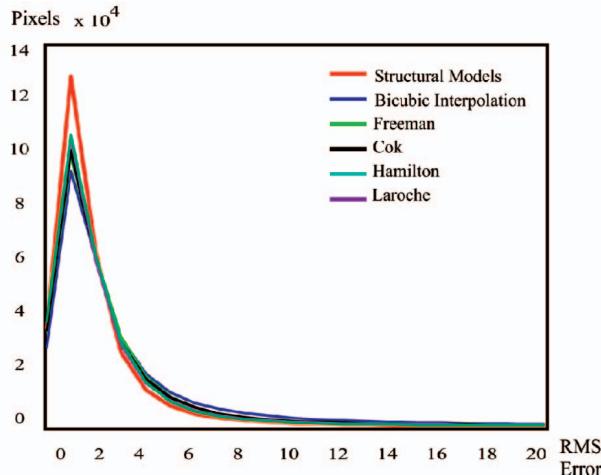


Fig. 16. Comparison of the histograms of luminance error (averaged over 20 test images) obtained by using five different methods. The average RMS error in luminance is 6.12 gray levels for bicubic interpolation and **3.27 gray levels for structural interpolation**. The RMS values for the methods—Freeman [11], Hamilton and Adams [13], Laroche and Prescott [15], and Cok [6]—are in between. Note that our method outperforms the other five methods. Structural: 3.7, Freeman 4.21, Hamilton 4.63, Laroche 4.78, Cok 4.96, and bicubic 6.12.

the error histograms (computed using all six test images) for the two cases. The RMS errors were found to be 33.4 and 25.5 gray levels (in a 12-bit range) for bicubic and structural interpolations, respectively. Note that, even though a very small number (six) of images were used for training, our method outperforms bicubic interpolation.

6 SPATIALLY VARYING EXPOSURE AND COLOR (SVEC)

Since we are able to extract high spatial and spectral resolution from SVC images and high spatial and brightness resolution from SVE images, it is natural to explore how these two types of multisamplings can be combined into one. The result is the simultaneous sampling of space, color, and exposure (see Fig. 22). We refer to an image obtained in this manner as a **spatially varying exposure and color (SVEC) image**. If the SVEC image has 8 bits of a **single color channel at each pixel**, we would like to compute, at each pixel, three color values, each with 12 bits of precision. Since the same number of pixels on a detector are now being used to **sample three different dimensions**, it should be obvious that this is a truly challenging interpolation problem. We

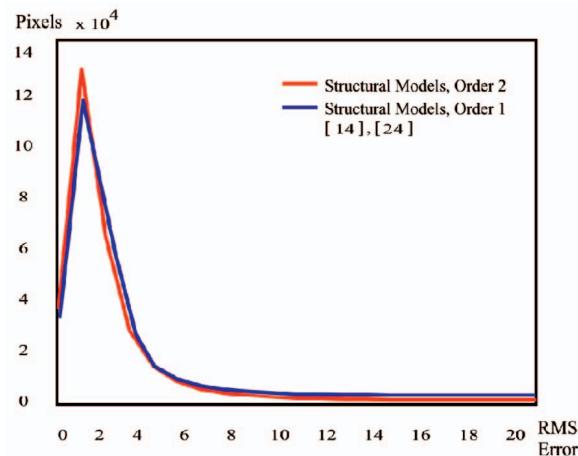


Fig. 17. Comparison of the histograms of luminance error (averaged over 20 test images) obtained by using structural models of orders 1 and 2, respectively. Note that the structural model with order 1 reduces to the works proposed in [14], [24]. Second order polynomials perform slightly better than the first order polynomials. Later, we will show that higher orders can have more significance in other multisampled images.

will see that structural interpolation does remarkably well at extracting the desired information.

6.1 SVEC Patterns and Neighborhoods

Color and exposure filters can be used to construct an SVEC sensor in many ways. If the number of pixels is p , the number of exposures used is e , and the number of color channels is c , then the cardinality $(ce)^p$ of the entire space of possible SVEC sensors is enormous. A simple way to decrease the number of possibilities is to assign colors and exposures within a small neighborhood and then to repeat it over the entire detector array. We call this repeated pattern the **base pattern**. In Fig. 22, the 4×4 neighborhood $p1$ (base pattern) is repeated over the entire detector array.

The advantage of repeating a base pattern throughout the detector array is that the number of distinct types of sampling patterns within a local neighborhood corresponds exactly to the distinct cyclic shifts of columns and rows of the base pattern. As an example, in Fig. 22, $p16$ is obtained from the base pattern $p1$ by cyclically shifting the columns of $p1$ by four steps and then cyclically shifting the rows of $p1$ by four steps. For a base pattern size of $u \times v$, the number of distinct cyclic shifts and, hence, the number of unique patterns is at most uv . For the 4×4 base pattern $p1$ (see Fig. 22), the unique patterns correspond to the neighborhoods $p1 \dots p16$.

Even with the repetitive pattern, we are left with $(ce)^n$ SVEC mosaic possibilities, where n is the number of pixels

Images	Same Images	Similar Images
RMS Error	1.5	2.7

(a)

Images	5	15	20	25	30	40
RMS Error	6.5	4.0	3.4	3.6	3.3	4.3

(b)

Fig. 18. **(Effectiveness and Stability of the Learning Technique**) (a) When the training and testing image sets are identical or when the images used in training are very similar to the test images (say, different types of grass), the average RMS errors in the luminance channel are very small. (b) If the training set consists of scenes that are different as well as similar to the testing set, then the results vary with the number of training images. The training images are selected randomly from a set of 40 images. The luminance RMS errors are computed over all 20 test images. The RMS errors shown are averaged over five trials of random selection. Note that the RMS errors are more or less stable (apart from the five image case).

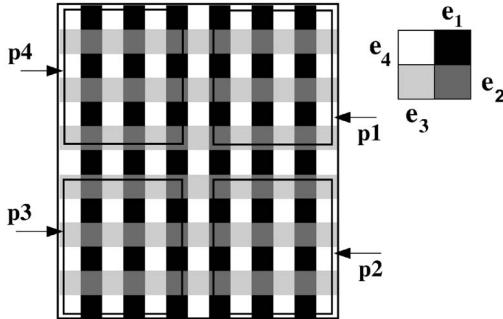


Fig. 19. The dynamic range of an image detector can be improved by assigning different exposures to pixels. In this special case of four exposures, any 6×6 neighborhood in the image must belong to one of four possible sampling patterns shown as p1...p4.

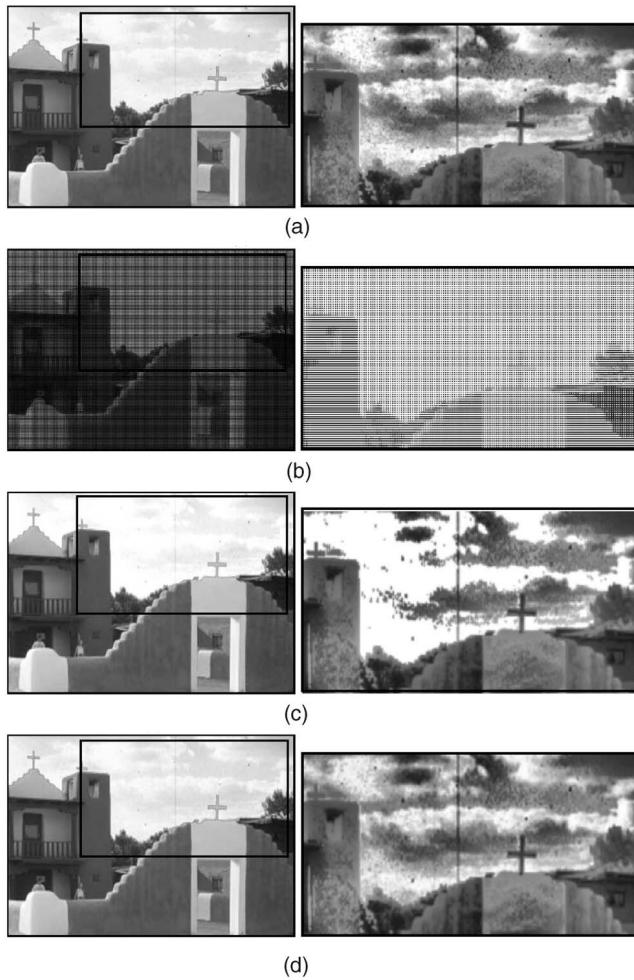


Fig. 20. (SVE Reconstruction) (a) Original 12-bit gray scale image. (b) 8-bit SVE image. (c) 12-bit (high dynamic range) image computed using bicubic interpolation. (d) 12-bit (high dynamic range) image computed using structural models. The magnified image regions on the right are histogram equalized.

in the neighborhood. Two questions need to be addressed at this point: 1) What is the size of base neighborhood pattern that is repeated throughout the detector array? 2) How do we assign the colors and exposures within a neighborhood?

We now describe some guidelines to assign the various exposures and colors to pixels in a neighborhood. These guidelines are decided based on the following parameters:

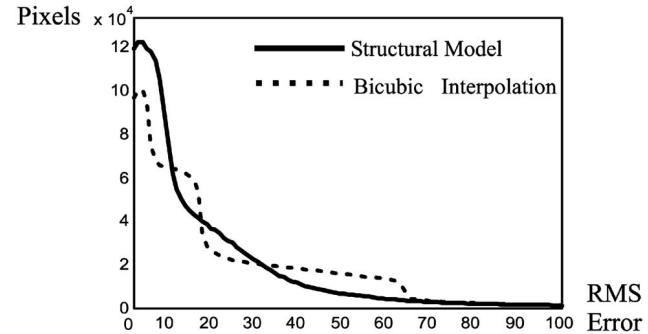


Fig. 21. Error histograms for the two cases (averaged over six test images). The RMS error for the six images is 33.4 and 25.5 gray levels (on a 12-bit scale) for bicubic and structural interpolation, respectively.

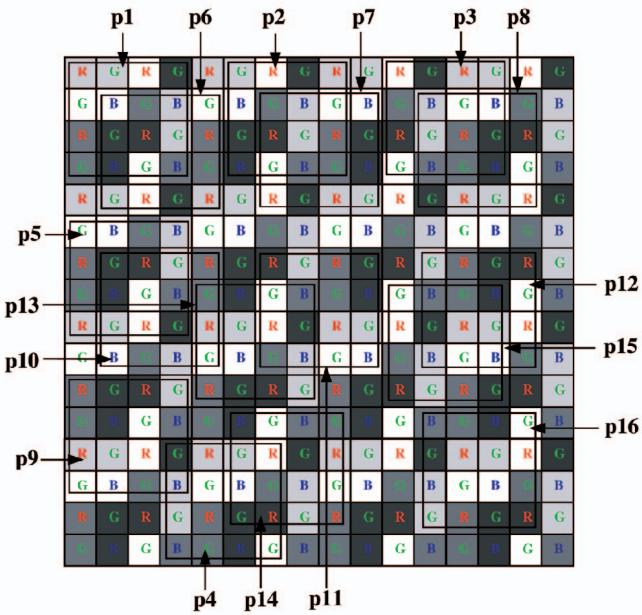


Fig. 22. A combined mosaic of three spectral and four neutral density filters used to simultaneously sample space, color, and exposures using a single image detector. The captured 8-bit SVEC image can be used to compute a 12-bit (per channel) color image by structural interpolation. For this mosaic, for any neighborhood size, there are only 16 distinct sampling patterns. For a 4×4 neighborhood size, the patterns are p1...p16.

1. Neighborhood size and shape, parameterized by length u , width v . The neighborhood aspect ratio is then $a = \frac{v}{u}$.
2. Number of R, G, or Bs in any neighborhood and their relative positions.
3. Number of discrete exposures e , their transmittances, and relative positions in the neighborhoods.
4. Number of R, G, or Bs assigned to each discrete exposure value.

In this paper, we choose the spatial distribution of the colors on the detector array to follow the Bayer mosaic (see Fig. 6). Now, we need to find the size of the minimum repeatable SVEC neighborhood (base neighborhood) and also find assignments to the exposures.

Within each local neighborhood of size $u \times v$, we constrain the spatial resolution of R, G, and B to be 25 percent, 50 percent, and 25 percent, respectively (see Fig. 6),

$$|R| = |B| = 0.25 u v; |G| = 0.5 u v. \quad (10)$$

From the above equation, we see that the neighborhood length u and width v must be even. Also, it is desirable to have each color channel combined with all the different exposures, within each neighborhood. Therefore,

$$\begin{aligned} |R| = |B| &= k e; \quad |G| = 2 k e \\ 4 k e &= u v, \end{aligned} \quad (11)$$

where R and B are combined with each exposure k times and G is combined with each exposure $2k$ times. Furthermore, in terms of the aspect ratio a of the neighborhood

$$4 k e = a u^2. \quad (12)$$

Note that k, e , and u are positive integers. Our goal is to find the *smallest local SVEC pattern* that satisfies the above constraints. For a fixed number of exposures e and aspect ratio a , the minimum size u of the local pattern corresponds to the smallest integer value for k that satisfies (12). For example, if $e = 2$ and $a = 1$ (for square neighborhoods), then the smallest neighborhood that needs to be repeated over the entire detector array is of size $u = 4$ (corresponding to $k = 2$). Similarly, for $e = 4$ and $a = 1$, we have $u = 4$ (and $k = 1$). In Fig. 22, we have used $e = 4$ (shown using different gray levels), the Bayer color mosaic, and, hence, the 4×4 square neighborhood p_1 , is repeated over the entire array.

Equation (12) gives the relationship between the number of exposures and the local neighborhood pattern size for a Bayer mosaic. Note that, without fixing the positions of the colors on the detector array, we are still left with a large number of possibilities for the SVEC mosaic. However, since the Bayer mosaic is very commonly used in CCDs, we assume the Bayer pattern for the colors and, hence, assign the exposures to the various pixels in the neighborhood.

6.2 SVEC Structural Model

The polynomial structural model used in the SVEC case is the same as the one used for SVC and is given by (3). The only caveat is that, in the SVEC case, we need to ensure that the neighborhood size used is large enough to adequately sample all the colors and exposures. That is, the neighborhood size is chosen such that it includes all colors and all exposures of each color.

The total number of polynomial coefficients needed is computed in the same way as in the SVC case and is given by (7). In our experiments, we have used the mosaic shown in Fig. 22. Therefore, $P = 16$, $\Lambda = 3(R, G, \text{and } B)$, $N_p = 2$ for each of the 16 patterns, and $u = v = 6$ to give a total of 3,504 coefficients. Once again, at each pixel for each color, only $3,504/48 = 73$ coefficients are used. Therefore, even for this complex type of multisampling, our structural models can be applied to images in real-time using a set of linear filters.

6.3 Experiments

The SVEC structural model was trained using six of the images in Fig. 3. In this case, the 12-bit color images in the training set were downgraded to 8-bit SVEC images. The original and SVEC images were used to compute the 3,504 coefficients of the model. The model was then used to map 10 different test SVEC images to 12-bit color images. One of these results is shown in Fig. 23. The original 12-bit image shown in Fig. 23a was downgraded to obtain the 8-bit SVEC image shown in Fig. 23b. This image has a single channel and is shown here in color only to illustrate the effect of simultaneous color and exposure

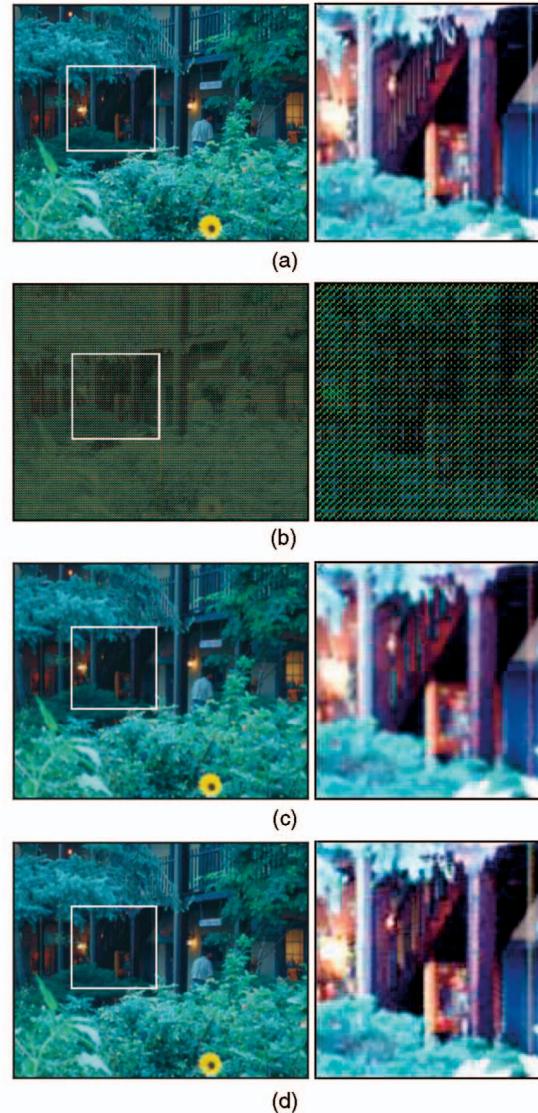


Fig. 23. **(SVEC Reconstruction)** (a) Original 12-bit color image. (b) 8-bit SVEC Image. 12-bit color images reconstructed using (c) bicubic interpolation and (d) structural interpolation. The magnified images on the right are histogram equalized.

sampling. Fig. 23c and Fig. 23d show the results of applying bicubic and structural interpolation, respectively. It is evident from the magnified images on the right that structural interpolation yields greater spectral and spatial resolution. The two interpolation techniques are compared in Fig. 24, which shows error histograms computed using all 10 test images. The RMS luminance errors were found to be 118 gray-levels and 80 gray-levels (on a 12 bit scale) for bicubic and structural interpolations, respectively.

Note that learned polynomial models for SVEC lie in a high-dimensional space. So, naturally, the following question arises: Is the learning and, hence, the reconstruction algorithm, sensitive to the size of the neighborhood and/or the order of the polynomial chosen for the structural model? We analyze the reconstruction error by performing several experiments with different polynomial orders and neighborhood sizes. To see a general trend, we just show reconstruction errors corresponding to three different neighborhood sizes and three polynomial orders. To avoid

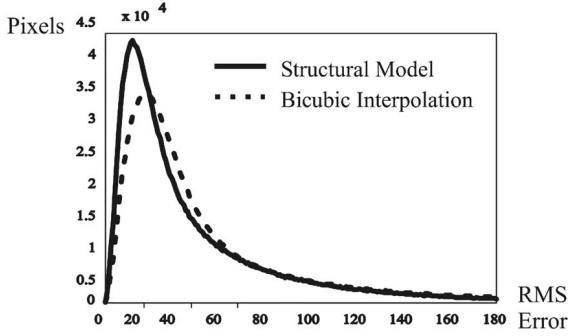


Fig. 24. Luminance error histogram computed using 10 test images. RMS luminance errors were found to be 118 and 80 (on a 12-bit scale) for bicubic and structural interpolation, respectively.

asymmetries in the local sampling pattern, we consider only neighborhoods that have even length and width.

Fig. 25 shows the average error histograms plotted for reconstruction using polynomial structural models of orders 1, 2, and 3 and neighborhood sizes 6×6 , 8×8 , and 10×10 . When the neighborhood size is small, we see that the reconstruction does better with higher order polynomials. Note also the RMS values of error obtained by using polynomials of orders 1, 2, and 3 (Fig. 26). As we increase the neighborhood size, the second and third order polynomials do not perform any better than the first order polynomials. This trend does not continue further as we increase neighborhood size since the correlation between the center location in the neighborhood and the outer edge of the neighborhood is minimal. Note, however, that, in all cases, the structural models perform much better than pure bicubic interpolation.

7 FUTURE WORK

We presented a general framework for using pixels on an image detector to simultaneously sample multiple dimensions of imaging (space, time, spectrum, brightness, polarization, etc.). Such pixels are termed *Assorted Pixels*. High-quality reconstructions along different imaging dimensions are done based on learning structural models from high-quality images of real scenes.

In the entire paper, we used polynomials of image intensities to represent structural models. We chose polynomials because of their simplicity, efficiency, and potential hardware implementation. It is conceivable that more complex learning techniques could lead to better reconstructions. For instance, Bayesian nets and belief propagation techniques or neural nets could be carefully chosen to represent the structural models. These techniques have been used in the context of increasing spatial resolution of single channel images, as well as in the context of demosaicing the Bayer pattern [10], [23]. However, it still remains to be seen how these methods can be extended to other multisampled images. Moreover, convergence issues, the requirement of priors, and the computational complexity of these methods make them less favorable when compared to the simple polynomial models. Another way to increase flexibility and accuracy is to use multiple structural models per multisampled pattern. This was done for the SVC case by Tappen et al. [23], where it was shown to be better than the simple polynomial-based learning technique. However, again, it will be interesting to explore

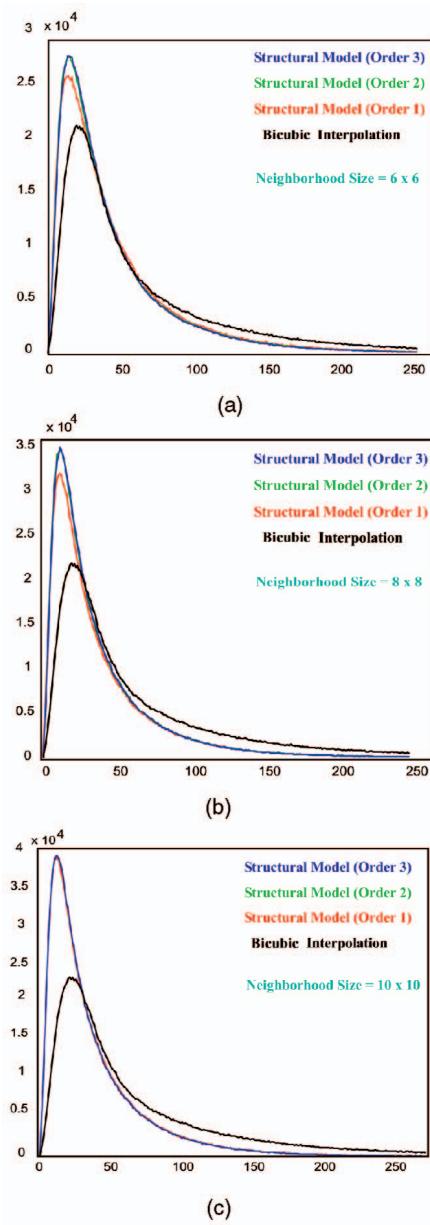


Fig. 25. Comparing error histograms computed using all the test images with structural models of different polynomial orders and different neighborhood sizes and with bicubic interpolation. In all the cases, the reconstruction using structural models is better than bicubic interpolation. (a) The neighborhood size is 6×6 and it can be seen that higher order polynomials (order 2 or 3) do better than the first order polynomial. (b) The neighborhood size is 8×8 and now the performance of the structural model is better than the 6×6 case. (c) The neighborhood size is 10×10 and the results are only marginally improved. However, notice that the polynomial orders do not effect the reconstruction errors. This shows we don't need very high order polynomials or very large neighborhoods to capture the inherent structure of the scene.

how these methods could be extended to other types of multisampled images. Also, it is important to note that our focus in this work is to derive a general framework for enhancing the resolution of arbitrary multisampled images and is not limited to the popular Bayer mosaic (SVC) case.

The multisampled framework can also include the temporal dimension. Thus, it is possible to apply learning techniques to increase temporal and spatial resolution of video as well. Note that this is fundamentally different

Neighborhood	Order 1	Order 2	Order 3
6 x 6	80.3	69.7	68.3
8 x 8	64.1	58.0	57.5
10 x 10	57.1	56.8	54.9

Fig. 26. RMS errors for different orders and neighborhood sizes.

from other traditional superresolution techniques that mostly correspond to the enhancement of spatial resolution using images captured using a moving camera. In future work, we wish to study different learning techniques that can be used to obtain better temporal resolution in addition to the dimensions of imaging considered in this work (exposure and color).

ACKNOWLEDGMENTS

This research was supported in part by an ONR/ARPA MURI Grant (N00014-95-1-0601) and a US National Science Foundation ITR Award (IIS-00-85864). Some of the results presented in this manuscript were published in the *Proceedings of the European Conference on Computer Vision* in 2002. The authors thank Dr. Rajeev Ramanath for providing the code for the state of the art demosaicing methods used for comparison in this work.

REFERENCES

- [1] B.E. Bayer, "Color Imaging Array," US Patent 3,971,065, July 1976.
- [2] M. Ben-Ezra, "Segmentation with Invisible Keying Signal," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2000.
- [3] S. Baker and T. Kanade, "Limits on Super-Resolution and How to Break Them," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pp. II:372-379, 2000.
- [4] R.N. Bracewell, *The Fourier Transform and Its Applications*. McGraw-Hill, 1965.
- [5] D. Brainard, "Bayesian Method for Reconstructing Color Images from Trichromatic Samples," *Proc. Soc. for Imaging Science and Technology 47th Ann. Meeting*, pp. 375-380, 1994.
- [6] D.R. Cok, "Signal Processing Method and Apparatus for Producing Interpolated Chrominance Values in a Sampled Color Image Signal," US Patent 4,642,678, 1987.
- [7] P.L.P. Dillon, "Color Imaging Array," US Patent 4,047,203, Sept. 1977.
- [8] P.L.P. Dillon, "Fabrication and Performance of Color Filter Arrays for Solid-State Imagers," *IEEE Trans. Electronic Devices*, vol. 25, pp. 97-101, 1978.
- [9] P. Debevec and J. Malik, "Recovering High Dynamic Range Radiance Maps from Photographs," *Proc. ACM SIGGRAPH '97*, pp. 369-378, 1997.
- [10] W.T. Freeman, E.C. Pasztor, and O.T. Carmichael, "Learning Low-Level Vision," *Intl. J. Computer Vision*, vol. 40, no. 1, pp. 25-47, 2000.
- [11] W.T. Freeman, "Median Filter for Reconstructing Missing Color Samples," US Patent 4,724,395, 1988.
- [12] R. Ginosar, O. Hilsenrath, and Y. Zeevi, "Wide Dynamic Range Camera," US Patent 5,144,442, Sept. 1992.
- [13] J.F. Hamilton and J.E. Adams, "Adaptive Color Interpolation in Single Sensor Color Electronic Camera," US Patent 5,629,734, 1997.
- [14] K. Knop and R. Morf, "A New Class of Mosaic Color Encoding Patterns for Single-Chip Cameras," *IEEE Trans. Electronic Devices*, vol. 32, 1985.
- [15] C.A. Laroche and M.A. Prescott, "Apparatus and Method for Adaptively Interpolating a Full Color Image Utilizing Chrominance Gradients," US Patent 5,373,322, 1994.
- [16] T. Mitsunaga and S.K. Nayar, "Radiometric Self Calibration," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pp. I:374-380, 1999.
- [17] D. Manabe, T. Ohta, and Y. Shimidzu, "Color Filter Array for IC Image Sensor," *Proc. IEEE Custom Integrated Circuits Conf.*, pp. 451-455, 1983.
- [18] S. Mann and R. Picard, "Being 'Undigital' with Digital Cameras: Extending Dynamic Range by Combining Differently Exposed Pictures," *Proc. Soc. for Imaging Science and Technology 48th Ann. Conf.*, pp. 442-448, May 1995.
- [19] S.K. Nayar and T. Mitsunaga, "High Dynamic Range Imaging: Spatially Varying Pixel Exposures," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pp. I:472-479, 2000.
- [20] K.A. Parulski, "Color Filters and Processing Alternatives for One-Chip Cameras," *IEEE Trans. Electronic Devices*, vol. 32, 1985.
- [21] R. Ramanath, W.E. Snyder, G.L. Bilbro, and W.A. Sander, "Demosaicing Methods for Bayer Color Arrays," *J. Electronic Imaging*, vol. 11, no. 3, July 2002.
- [22] Y.Y. Schechner and S.K. Nayar, "Generalized Mosaicing," *Proc. Int'l Conf. Computer Vision*, 2001.
- [23] M.F. Tappen, B.C. Russell, and W.T. Freeman, "Exploiting the Sparse Derivative Prior for Super-Resolution and Image Demosaicing," *Proc. Third Int'l Workshop Statistical and Computational Theories of Vision*, 2003.
- [24] M.A. Wober and R. Soini, "Method and Apparatus for Recovering Image Data through the Use of a Color Test Pattern," US Patent 5,475,769, Dec. 1995.



Srinivasa G. Narasimhan received the MS and PhD degrees in computer science from Columbia University in December 2000 and 2003, respectively. From January 2004 to July 2004, he was a postdoctoral research scientist at Columbia University. Since August 2004, he has been an assistant professor in the Robotics Institute at Carnegie Mellon University. His research interests are in the fields of computer vision and computer graphics. In particular, he is

interested in the development of physics-based models and algorithms for understanding and rendering complex scene appearance. His paper on modeling and removing atmospheric effects from images received a Best Paper Honorable Mention award at the 2000 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). He is a member of the IEEE.



Shree K. Nayar received the PhD degree in electrical and computer engineering from the Robotics Institute at Carnegie Mellon University in 1990. He is the T.C. Chang Professor of Computer Science at Columbia University. He currently heads the Columbia Automated Vision Environment (CAVE), which is dedicated to the development of advanced computer vision systems. His research is focused on three areas, namely, novel vision sensors, physics-based models for vision, and algorithms for scene understanding. His work is motivated by applications in the fields of computer vision, computer graphics, human-machine interfaces, and robotics. Dr. Nayar has authored or coauthored papers that have received the Best Paper Award at the 2004 IEEE CVPR Conference, the Best Paper Honorable Mention Award at the 2000 IEEE CVPR Conference, the David Marr Prize at the 1995 ICCV, the Siemens Outstanding Paper Award at the 1994 IEEE CVPR Conference, the 1994 Annual Pattern Recognition Award from the Pattern Recognition Society, the Best Industry Related Paper Award at the 1994 ICPR, and the David Marr Prize at the 1990 ICCV. He holds several US and international patents for inventions related to computer vision and robotics. Dr. Nayar was the recipient of the David and Lucile Packard Fellowship for Science and Engineering in 1992, the US National Young Investigator Award from the National Science Foundation in 1993, and the Excellence in Engineering Teaching Award from the Keck Foundation in 1995.

▷ For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.