Dynamic Demosaicing and Color Super-Resolution of Video Sequences

Sina Farsiu[a*], Dirk Robinson[a] Michael Elad[b], Peyman Milanfar[a]

[a] Electrical Engineering Department, University of California, Santa Cruz CA. 95064 USA.
[b] Computer Science Department, The Technion, Israel Institute of Technology, Israel.

ABSTRACT

In the last two decades a variety of super-resolution (SR) methods have been proposed. These methods usually address the problem of fusing a set of **monochromatic** images to produce a **single** monochromatic image with higher spatial resolution. In this paper we address the dynamic and color SR problems of reconstructing a high-quality **set** of **colored** super-resolved images from low-quality mosaiced frames. Our approach includes a hybrid method for simultaneous SR and demosaicing, this way taking into account practical color measurements encountered in video sequences. For the case of translational motion and common space-invariant blur, the proposed method is based on a very fast and memory efficient approximation of the Kalman filter. Experimental results on both simulated and real data are supplied, demonstrating the presented algorithm, and its strength.

1. INTRODUCTION

Theoretical and practical limitations usually constrain the achievable resolution of any imaging device. While higher quality images may result from more expensive imaging systems, often we wish to increase the resolution of images previously captured under non-ideal situations. For instance, enhancing the quality of a video sequence captured by surveillance cameras in a crime scene is an example of these situations.

The basic idea behind SR is the fusion of a sequence of low-resolution (LR) noisy blurred images to produce a higher resolution image. Early works on SR showed that it is the aliasing effects in the LR images that enable the recovery of the high-resolution (HR) fused image, provided that a relative sub-pixel motion exists between the under-sampled input images [1]. However, in contrast to the clean and practically naive frequency domain description of SR in that early work, in general SR is a computationally complex and even numerically ill-posed problem in many instances [2]. In recent years more sophisticated SR methods were developed (See [2–9] as representative works).

In this paper we focus on two common resolution-enhancement problems in digital video/photography that are typically addressed separately, namely, SR and demosaicing. While SR is naturally described for monochrome images, aiming to increase resolution by the fusion of several frames, the demosaicing is meant to recover missing color values, decimated deliberately by the sensor. In this work we propose a hybrid method of dealing with these two problems jointly.

Furthermore, in contrast to most of the existing works so far, we adopt a dynamic point of view, as introduced in [10], in developing the new SR solution. Dynamic SR refers to the situation in which a *sequence* of HR images are estimated from a sequence of LR frames. While it may appear that this problem is a simple extension of the static SR situation, simply solving static SR again and again, the memory and computational requirements for the dynamic case are so taxing as to preclude its application without highly efficient algorithms. It is natural to expect that if the SR problem is solved for time (t-1), our task for time (t) could be using this as a stepping stone towards a faster and more reliable SR solution. This is the essence of how dynamic SR is to gain its speed and better results, compared to a sequence of detached static SR routines.

Our goal in this paper is to develop a dynamic SR algorithm for color (filtered - see next section) sequences. For this algorithm, we sight both visual quality (resolution enhancement and color artifact reduction) and computational-memory efficiency. This paper has the following contributions. First, we develop a very fast and memory efficient dynamic SR method. Second, we introduce the multi-frame demosaicing problem as the

* (Corresponding Author) Email: farsiu@ee.ucsc.edu, Phone:(831)-459-4141, Fax: (831)-459-4829

Image Reconstruction from Incomplete Data III, edited by
Philip J. Bones, Michael A. Fiddy, Rick P. Millane Proc. of SPIE Vol. 5562
(SPIE, Bellingham, WA, 2004) · 0277-786X/04/$15 · doi: 10.1117/12.558936

generalization of the color SR problem. We propose a maximum a posteriori (MAP) estimation method to incorporate the proposed dynamic SR method in a general algorithm for addressing the multi-frame demosaicing problem.

This paper is organized as follows: In Section 2 we discuss the relation between color and SR. Section 3 addresses a fast dynamic image fusion method for the translational motion model. This method is then extended in Section 4 to consider the color-SR and multi-frame demosaicing cases. Simulations on both real and synthetic data sequences are presented in Section 5, and Section 6 concludes this paper.

Before departing to the details, we should like to note that this paper (with all color pictures and a MATLAB based software package for resolution enhancement) is available at *http://www.ee.ucsc.edu/∼milanfar* .

2. COLOR AND SR

A color image is represented by combining three separate monochromatic images. Ideally, each pixel should correspond to three scalar values; one for each of the color bands (red, green, or blue). In practice, to reduce production cost, many digital cameras have only one color measurement per pixel. The detector array is a grid of CCDs, each made sensitive to one color by placing a color filter array (CFA) in front of the CCD. The Bayer pattern shown in Figure 1 (left) is a very common example of such a color filter. The values of missing color bands at every pixel are then synthesized using some form of interpolation from neighboring pixel values. This process is known as color demosaicing.

While numerous single-frame demosaicing methods have been proposed (see [11–18] as representative works), the reconstructed images are almost always contaminated with different amounts of color artifacts. This results from the ill-posed nature of the demosaicing problem. However, if multiple, spatially offset, color filtered images of the same scene are available, one can combine them to increase both spatial resolution, and to produce a more effective overall demosaicing with significantly reduced artifacts. Such an approach may be termed multi-frame demosaicing. What makes multiframe demosaicing challenging is that almost none of the single-frame demosaicing methods (but the method in [19]) are directly applicable to it.

A related problem, color SR, addresses fusing a set of previously demosaiced color LR frames to enhance their spatial resolution. To date, there is very little work addressing the problem of color SR. The typical solution involves applying monochromatic SR algorithms to each of the color channels independently [20, 21], while using the color information to improve the accuracy of motion estimation. Another approach is transforming the problem to a different color space, where chrominance layers are separated from luminance, and SR is applied only to the luminance channel [3]. Both of these methods are sub-optimal as they do not fully exploit the correlation across the color bands.

3. DYNAMIC DATA FUSION

In this paper, we use a general linear model as in [3, 4]. A dynamic scene with intensity distribution X is seen to be warped at the camera lens because of the relative motion between the scene and camera, and blurred by camera lens and sensor integration. Then, it will be discretized at the CCD resulting in a digitized noisy frame Y. Discretization in many commercial digital cameras is a combination of color filtering and down-sampling processes.

We represent this forward model by the following equations:

$$\underline{X}(t) = F(t)\underline{X}(t-1) + \underline{U}(t), \quad \text{and} \tag{1}$$

$$\underline{Y}(t) = D(t)H(t)\underline{X}(t) + \underline{W}(t). \tag{2}$$

The equation (1) describes how the ideal super-resolved images relate to each other through time, stating that up to some innovation (depicted in $\underline{U}(t)$), the current image $\underline{X}(t)$ is a geometrically warped version of the previous one, where $F(t)$ represents this warp operator. The so-called system noise $\underline{U}$ is assumed to be additive white, zero mean Gaussian with $C_u(t)$ as its covariance matrix, reflecting the accuracy of the motion estimation process. We use the underscore notation such as $\underline{X}$ to indicate a vector arranged in lexicographic order.

As to equation (2), it describes how the measured image $\underline{Y}(t)$ is related to the ideal one $\underline{X}(t)$. The camera's point spread function (PSF) is modelled by the blur matrix $H(t)$. $D(t)$ represents the decimation operation at the CCD (decimation by the factor r). In mosaiced cameras this matrix also represents the effects of the color filter array, which further down-samples the color image. The noise $\underline{W}$ is assumed to be zero mean additive white, Gaussian (thus, its covariance matrix is $C_y(t) = I$). We further assume that $\underline{U}(t)$ and $\underline{W}(t)$ are independent of each other.

The equations given above describe a system in its *state-space* form, where the state is the desired ideal image. Thus, Kalman filter is required to estimate $(\underline{X}(t), t \in \{1, ..., N\})$ from the measurements $(\underline{Y}(t), t \in \{1, ..., N\})$ causally, assuming that $D(t), H(t), F(t),$ and $C_u(t)$ are all known [10, 22].

While each of the operators $D(t), H(t),$ and $F(t)$ could vary in time, for most situations the decimation, color filtering, and camera blurring operations remain constant over time. In this paper we further assume that the camera PSF is space-invariant, and the motion is composed of pure transalations. Thus, both H and $F(t)$ are block-circulant matrices, and as such, they commute.

We limit our model to the case of translational motion for several reasons. First, as we describe later, such a motion model allows for an extremely fast and memory efficient dynamic SR algorithm. Second, while simple, the model fairly well approximates the motion contained in image sequences, where the scene is stationary and only the camera moves. Third, for sufficiently high frame rates most motion models can be (at least locally) approximated by the translational model. Finally, we believe that an in-depth study of this simple case yields much insight into the more general cases of motion in dynamic SR. In Section 4 we show that the proposed dynamic SR method is easily applicable to multi-frame demosaicing and color SR problems.

By substituting $\underline{Z}(t) = H\underline{X}(t)$, we obtain from (1) and (2) an alternative model, where the state vector is $\underline{Z}(t)$,

$$\underline{Z}(t) = F(t)\underline{Z}(t-1) + \underline{V}(t), \quad \text{and} \tag{3}$$

$$\underline{Y}(t) = D\underline{Z}(t) + \underline{W}(t). \tag{4}$$

Note that the first of the two equations is obtained by multiplication (1) by H and using the fact that it commutes with $F(t)$. Thus, the vector $\underline{V}(t)$ is a colored version of $\underline{U}(t)$, leading to $C_v(t)$ as the covariance matrix.

This way, the solution of the inverse problem at hand decomposes without loss of optimality into the much simpler sub-tasks of fusing the available images to estimate the blurry image $\underline{Z}(t)$ followed by a de-blurring/interpolation step[†] estimating $\underline{X}(t)$ from $\hat{\underline{Z}}(t)$.

Kalman filtering should be used to estimate $\underline{Z}(t)$. In general, the application of Kalman filter requires the update of the state-vector's covariance matrix per each temporal point, and this update requires an inversion of this matrix. If the super-resolved image has N pixels, this matrix is of size $N \times N$, implying a prohibitive amount of computations and memory.

Fast and memory efficient alternative ways are to be found, and such methods has been proposed in [10]. Here we show that significant speedups are achieved for the case of translational motion and common space-invariant blur. However, unlike [10], where the information-pair approach to Kalman filtering was employed, our approach is based on the more traditional, direct mean-covariance formulation of the Kalman filter. The following defines the Kalman propagation and update equations [22]. We start with the covariance Propagation Matrix:

$$\tilde{M}(t) = F(t)M(t-1)F^T(t) + C_v(t), \tag{5}$$

[†]In this section, we treat the three color bands separately. For instance, only the red band values in the input frames, $\underline{Y}(t)$, contribute to reconstruct the red band values in $\underline{Z}(t)$. The correlation of the different color bands are discussed and exploited in the next Section.

the Kalman Gain Matrix is given by:

$$K(t) = \tilde{M}(t)D^T[C_y(t) + D\tilde{M}(t)D^T]^{-1}. \tag{6}$$

The updated state vector mean is computed by

$$\underline{\widehat{Z}}(t) = F(t)\underline{\widehat{Z}}(t-1) + K(t)[\underline{Y}(t) - DF(t)\underline{\widehat{Z}}(t-1)] \tag{7}$$

The final stage requires the update of the covariance matrix:

$$M(t) = (I - K(t)D)\tilde{M}(t) \tag{8}$$

While in general the above equations require the propagation of intolerable sized matrices in time, if we refer to $C_v(t)$ as a diagonal matrix, disregarding its off-diagonal entries, all the matrices involved here become diagonal. This is a key assumption in transferring the general Kalman filter into a simple and fast procedure, and as we shall see, the approximated version emerging is quite faithful.

The matrices F and D and their transposes can be exactly interpreted as direct image operators such as shift and decimation [5, 6]. Applying these matrices as a sequence of image operations spares us from explicitly constructing them as matrices. This property helps our method to be implemented in an extremely fast and memory efficient way

Note that the incorporation of each newly measured LR image effectively only updates the value of a few corresponding pixels, due to the diagonality imposed . This property enables us to simplify the equations (5), (6), (7), and (8) to the following rather intuitive procedure:

1. For a pixel at spatial position q with a current estimated value $[\underline{\widehat{Z}}(t-1)]_q$, and covariance $[M(t-1)]_q$, we need to update those two by motion compensation, applying the shift given by $F(t)$, and resulting with $[\underline{\widehat{Z}}_0(t)]_q$, and covariance $[M_0(t)]_q$.

2. For a pixel at spatial position q which is to be updated, we have to update the estimation by

$$[\underline{\widehat{Z}}(t)]_q = \frac{[C_y(t)]_q}{[C_y(t)]_q + [M_0(t)]_q + [C_v(t)]_q}[\underline{\widehat{Z}}_0(t)]_q + \frac{[M_0(t)]_q + [C_v(t)]_q}{[C_y(t)]_q + [M_0(t)]_q + [C_v(t)]_q}[\underline{Y}(t)]_q \tag{9}$$

   Similarly, the covariance update becomes:

$$[M(t)]_q = \frac{[C_y(t)]_q\,([M_0(t)]_q + [C_v(t)]_q)}{[C_y(t)]_q + [M_0(t)]_q + [C_v(t)]_q} \tag{10}$$

3. For pixel q which is not updated by a LR frame at time (t) (the effect of zero filling in up-sampling operation), we get that the estimate is simply

$$[\underline{\widehat{Z}}(t)]_q = [\underline{\widehat{Z}}_0(t)]_q. \tag{11}$$

   The covariance is updated by

$$[M(t)]_q = [M_0(t)]_q + [C_v(t)]_q \tag{12}$$

Such a recursive solution shows that there is no need to keep previous LR frames (except the most recent one) in memory. Only the HR image estimate $\underline{\widehat{Z}}(t-1)$ at any given time and a (same size) weighting image containing the updated diagonal matrix $M(t)$ need to be stored in memory, leading to a very efficient algorithm. Furthermore, the update operation is simply shifting the previous estimates $\underline{\widehat{Z}}(t-1)$, and $M(t-1)$ and updating the proper pixels using (9)-(10) or (11)-(12). Throughout this paper, we refer to this process as the Shift-And-Add process and call $\underline{\widehat{Z}}(t)$ the Shift-And-Add image.

At this point, we have an efficient recursive estimation algorithm producing estimates of the blurry HR image sequence $\underline{\widehat{Z}}(t)$. From these frames, the sequence $\underline{\widehat{X}}(t)$ must be estimated. Note that some (if not all) frames will not have estimates for every pixel in $\underline{\widehat{Z}}(t)$, necessitating a further joint interpolation and deblurring step. In any case, the above process is to be applied separately on the R, G, and B layers, producing the arrays we will start from in the next stage.

4. DEMOSAICING AND DEBLURRING

To perform robust deblurring and interpolation, we use the MAP cost function

$$\epsilon\left(\underline{X}(t)\right) = \|A(t)(H\underline{X}(t) - \underline{\widehat{Z}}(t))\|_2^2 + \lambda\Gamma(\underline{X}(t)), \tag{13}$$

and define our desired solution as

$$\underline{\widehat{X}}(t) = \underset{\underline{X}(t)}{\text{ArgMin}}\, \epsilon\left(\underline{X}(t)\right). \tag{14}$$

Here, the matrix $A(t)$ is a diagonal matrix whose values are chosen in relation to our confidence in the measurements that contributed to make each element of $\underline{\widehat{Z}}(t)$. These values have inverse relation to the corresponding elements in the matrix $M(t)$. The regularization parameter, λ, is a scalar for properly weighting the first term (data fidelity cost) against the second term (regularization cost), and $\Gamma(\underline{X})$ is the regularization cost function. The regularization term stabilizes the solution of this ill-posed problem, improves the rate of convergence, and helps remove artifacts. For the case of mono-chromatic SR, many regularization terms have been proposed (e.g. see [6] for a comparison of Tikhonov, Total Variation, and Bilateral regularization cost functions). In this section, we propose regularization terms that yield good results for the case of color SR and multi-frame demosaicing problems.

Referring to the mosaic effects, the geometry of the single-frame and multi-frame demosaicing problems are fundamentally different, making it impossible to simply cross apply traditional demosaicing algorithms to the multi-frame situation. To better understand the multi-frame demosaicing problem, we offer an example for the case of translational motion. Suppose that a set of color filtered LR images are available (images on the left in Figure 1). We use the two step process explained in Section 3 to fuse these images. The Shift-And-Add image in the right side of Figure 1 illustrates the pattern of sensor measurements in the HR image grid. In such situations, the sampling pattern is quite arbitrary depending on the relative motion of the LR images. This necessitates a different demosaicing algorithm than those designed for the original Bayer pattern.

Figure 1 shows that treating the green channel differently than the red or blue channels, as is done in many single-frame demosaicing methods before, is not particulary useful for the multi-frame case. While globally there are more green pixels than blue or red pixels, locally any pixel may be surrounded by only red or blue colors. So, there is no general preference for one color band over the others.

Another assumption, the availability of one and only one color band value for each pixel, is also not correct in the multi-frame case. In the under-determined cases [‡], there are not enough measurements to fill the HR grid. The symbol "?" in Figure 1 represents such pixels. On the other hand, in the over-determined case[§], for some pixels, there may in fact be more than one color value available.

We propose an algorithm for producing a high quality color image from a collection of LR color (filtered) images. Our computationally efficient MAP estimation method is motivated by the color image perception properties of human visual system. This method is directly applicable to both color SR (given full RGB LR frames), and the more general multi-frame demosaicing problems.

As described in Section 3, our method is a two step process of image fusion and simultaneous deblurring and interpolation. For the case of color SR, the first step involves nothing more than the application of the recursive image fusion algorithm separately on three different color bands. Image fusion of color filtered images is done quite similarly, where each single channel color filtered frame is treated as a sparsely sampled three-channel color image. The second step, deblurring and interpolating the missing values of different color channels, is based on minimizing a MAP cost function with several terms composing the overall function $\epsilon\left(\underline{X}(t)\right)$.

Data Fidelity Penalty Term: This term penalizes the dissimilarity between the raw data and the HR estimate, and is defined as:

$$J_0(\underline{X}(t)) = \sum_{i=R,G,B} \|A_i(t)\left(H\underline{\widehat{X}}_i(t) - \underline{\widehat{Z}}_i(t)\right)\|_2^2, \tag{15}$$

[‡] where the number of non-redundant LR frames is smaller than the square of resolution enhancement factor.

[§] where the number of non-redundant LR frames is larger than the square of resolution enhancement factor.

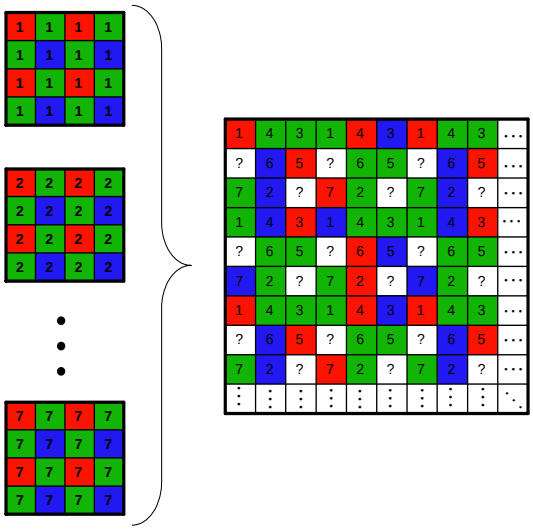

Figure 1. Fusion of 7 Bayer pattern LR images with relative translational motion (the figures in the left side of the accolade) results in a HR image ($\widehat{Z}$) that does not follow Bayer pattern (the figure in the right side of the accolade). The symbol "?" represents the High-resolution pixel values that were undetermined after the Shift-And-Add step (result of insufficient LR frames).

where $\underline{\widehat{Z}}_R$, $\underline{\widehat{Z}}_G$, and $\underline{\widehat{Z}}_B$ are the three color channels of the color Shift-And-Add image, $\underline{\widehat{Z}}$. A_R, A_G, and A_B keep track of the known red, green, and blue sensor values in $\underline{\widehat{Z}}$. The $A_{i\in\{R,G,B\}}$ matrices for the multi-frame demosaicing problem are sparser than the corresponding matrices in the color SR case. $\underline{\widehat{X}}_R$, $\underline{\widehat{X}}_G$, and $\underline{\widehat{X}}_B$ are the three color components of the reconstructed HR image $\widehat{X}$.

Luminance Penalty Term: The human eye is more sensitive to the details in the luminance component of an image than the details in the chrominance components [14]. Therefore, it is important that the edges in the luminance component of the reconstructed HR image look sharp. Applying Bilateral-TV regularization to the luminance component will result in this desired property [6], where L_1 norm is used to force spatial smoothness while creating sharp edges. The luminance image can be calculated as the weighted sum $\underline{X}_L = 0.299\underline{X}_R + 0.597\underline{X}_G + 0.114\underline{X}_B$ as explained in [23]. The luminance regularization term is defined as:

$$J_1(\underline{X}) = \underbrace{\sum_{l=-P}^{P}\sum_{m=0}^{P}}_{l+m\geq 0} \alpha^{|m|+|l|}\|\underline{X}_L(t) - S_x^l S_y^m \underline{X}_L(t)\|_1 \ , \tag{16}$$

S_x^l and S_y^m are the operators corresponding to shifting the image represented by $\underline{X}$ by l pixels in horizontal direction and m pixels in vertical direction, respectively. This cost function in effect computes derivatives across multiple scales. The scalar weight α, $0<\alpha<1$, is applied to give a spatially decaying effect to the summation of the regularization term. Note that image shifting and differencing operations are very cheap to implement.

Chrominance Penalty Term: The human eye is more sensitive to chromatic change in the low spatial frequency region than the luminance change [18]. As the human eye is less sensitive to the chrominance channel resolution, it can be smoothed more aggressively. Therefore, L_2 regularization is an appropriate method for smoothing the Chrominance term:

$$J_2(\underline{X}(t)) = \|\Lambda\underline{X}_{C1}(t)\|_2^2 + \|\Lambda\underline{X}_{C2}(t)\|_2^2 \ , \tag{17}$$

where Λ is the matrix realization of a high-pass operator such as the Laplacian filter. The images $\underline{X}_{C1}(t)$ and $\underline{X}_{C2}(t)$ are the I and Q layers in the YIQ color representation.

Orientation Penalty Term: This term penalizes the non-homogeneity of the edge orientation across the color channels. Although different bands may have larger or smaller gradient magnitudes at a particular edge, the statistics of natural images shows that it is reasonable to assume a same edge orientation for all color channels. That is, for instance, if an edge appears in the red band at a particular location, then an edge with the same orientation should appear in the other color bands at the same location as well. Following [13], minimizing the vector product norm of any two adjacent color pixels forces different bands to have similar edge orientation. With some modifications to what was proposed in [13], our orientation penalty term is a differentiable cost function:

$$\begin{aligned} J_3(\underline{X}) &= \underbrace{\sum_{l=-1}^{1}\sum_{m=0}^{1}}_{l+m\geq 0} \left[\|\underline{X}_G(t)\odot S_x^l S_y^m \underline{X}_B(t) - \underline{X}_B(t)\odot S_x^l S_y^m \underline{X}_G(t)\|_2^2 + \right. \\ &\quad \left. \|\underline{X}_B(t)\odot S_x^l S_y^m \underline{X}_R(t) - \underline{X}_R(t)\odot S_x^l S_y^m \underline{X}_B(t)\|_2^2 + \|\underline{X}_R(t)\odot S_x^l S_y^m \underline{X}_G(t) - \underline{X}_G(t)\odot S_x^l S_y^m \underline{X}_R(t)\|_2^2\right] \end{aligned}$$

where $\odot$ is the element by element multiplication operator.

The overall cost function $\epsilon\left(\underline{X}(t)\right)$ is the summation of these cost functions:

$$\widehat{\underline{X}}(t) = \underset{\underline{X}(t)}{\text{ArgMin}}\left[J_0(\underline{X}(t)) + \lambda' J_1(\underline{X}(t)) + \lambda'' J_2(\underline{X}(t)) + \lambda''' J_3(\underline{X}(t))\right] . \tag{18}$$

Steepest descent optimization may be applied to minimize this cost function. In the first step, the derivative of (18) with respect to one of the color bands is calculated, assuming the other two color bands are fixed. In the next steps, the derivative is computed with respect to the other color channels.

For example, the derivative with respect to the green band ($\underline{X}_G$) is calculated as:

$$\begin{aligned} \nabla\underline{X}_G(t) &= H^T A_G^T (A_G H \underline{X}_G(t) - A_G \widehat{\underline{Z}}_G(t)) + \\ &\quad \lambda' \underbrace{\sum_{l=-P}^{P}\sum_{m=0}^{P}}_{l+m\geq 0} 0.587\alpha^{m+l}[I - S_y^{-m}S_x^{-l}] sign\left(\underline{X}_L(t) - S_x^l S_y^m \underline{X}_L(t)\right) + \\ &\quad \lambda'' \underbrace{\sum_{l=-1}^{1}\sum_{m=0}^{1}}_{l+m\geq 0} \Big[2(\mathbf{X_B^{l,m}}(\mathbf{t}) - S_x^{-l}S_y^{-m}\mathbf{X_B}(\mathbf{t}))(\mathbf{X_B^{l,m}}(\mathbf{t})\underline{X}_G(t) - \mathbf{X_B}(\mathbf{t})S_x^l S_y^m \underline{X}_G(t)) + \\ &\quad 2(\mathbf{X_R^{l,m}}(\mathbf{t}) - S_x^{-l}S_y^{-m}\mathbf{X_R}(\mathbf{t}))(\mathbf{X_R^{l,m}}(\mathbf{t})\underline{X}_G(t) - \mathbf{X_R}(\mathbf{t})S_x^l S_y^m \underline{X}_G(t))\Big] + \\ &\quad \lambda''' \Lambda^T \Lambda \left[-0.1536 \times \underline{X}_R(t) + 0.2851 \times \underline{X}_G(t) - 0.1316 \times \underline{X}_B(t)\right] , \end{aligned} \tag{19}$$

where S_x^{-l} and S_y^{-m} define the transposes of matrices S_x^l and S_y^m, respectively and have a shifting effect in the opposite directions as S_x^l and S_y^m. $\mathbf{X_R}$, and $\mathbf{X_B}$ are the diagonal matrix representations of the red and blue bands and $\mathbf{X_R^{l,m}}$ and $\mathbf{X_B^{l,m}}$ are the diagonal representations of these matrices shifted by l and m pixels in the horizontal and vertical directions, respectively. Similar to the matrices in Section 3 all incorporated matrices such as H, Λ, S_x^l, and S_y^m and their transposes can be exactly interpreted as direct image operators such as blur, high-pass filtering, and shift.

The gradient of the other channels are computed in the same way, and the following steepest descent iterations are set up to calculate the HR image estimate iteratively,

$$\widehat{\underline{X}}_i^{n+1}(t) = \widehat{\underline{X}}_i^{n}(t) - \beta\nabla\widehat{\underline{X}}_i^{n}(t), \tag{20}$$

where $i \in \{R, G, B\}$ and the scalar β is the step size.

5. EXPERIMENTS

Experiments on synthetic and real world data sets are presented in this section. In the first experiment, we synthesized a sequence of low-resolution color filtered images from a single color image of size 1200×1600 captured with a 1CCD OLYMPUS C-4000 digital camera. A 128×128 section of this image was blurred with a symmetric Gaussian low-pass filter of size 4×4 with standard deviation equal to one. The resulting images were subsampled by the factor of four in each direction and further color filtered with Bayer pattern creating a 32×32 image. We consecutively shifted the 128×128 window on the original high-resolution image by one pixel in right, down, or up directions, and repeated the same image degradation process. In this fashion, we created a sequence of 250 frames.

Figures 2(a) & 2(b) show two sections of the HR image[¶]. Figures 2(c) & 2(d) show frames #50 and #250 of the LR sequence (for the sake of presentation each frame has been demosaiced following the method of [11]). We created a sequence of HR fused images using the method described in Section 3 (factor of four resolution enhancement). Figures 2(e) & 2(f) show frames #50 and #250 of this sequence, where the missing values were filled in using bilinear interpolation [‖]. Later each frame was deblurred-demosaiced using the method described in Section 4. Figures 2(g) & 2(h) show frames #50 and #250 of this reconstructed sequence, where the color artifacts have been almost completely removed. The corresponding parameters for this experiment (tuned by trial-and-error) were as follows: $\alpha = 0.9$, $\beta = 0.06$, $\lambda' = \lambda'' = 0.001$, and $\lambda''' = 10$. Fifteen iterations of steepest descent were used for this experiment.

Figure 2. A sequence of 250 LR color filtered images where recursively fused (Section 3), increasing their resolution by the factor of 4 in each direction. They were further deblurred and demosaiced (Section 4), resulting in images with much higher-quality than the input LR frames. In (a) & (b) we see the ground-truth for frames #50 and #250, and (c) & (d) are the corresponding synthesized LR frames. In (e) & (f) we see the recursively fused HR frames and (g) & (h) shows the deblurred-demosaiced frames.

Our next experiment was preformed on a real-world (already demosaiced) compressed image sequence courtesy of Adyoron Intelligent Systems Ltd., Tel Aviv, Israel. Two frames of this sequence (frames # 20 and #40) are shown in Figures 3(a) & 3(d). We created a sequence of HR fused images (factor of four resolution enhancement) using the method described in Section 3(Figures 3(b) & 3(e)). Later each frame in this sequence was deblurred using the method described in Section 4 (Figures 2(c) & 3(f)). The corresponding parameters for this experiment are as follows: $\alpha = 0.9$, $\beta = 0.1$, $\lambda' = \lambda'' = 0.005$, and $\lambda''' = 50$. Fifteen iterations of steepest descent were used for this experiment. The (unknown) camera PSF was assumed to be a 4×4 Gaussian kernel with standard deviation equal to one. As the relative motion between these images followed the translational model, we only

[¶]We cut out some border pixels in these images, and the corresponding LR and reconstructed images.

[‖]For this under-determined experiment, it is easy to show that only $\frac{1}{3}$ of the pixel values in $\underline{\widehat{Z}}(t)$ are determined by the Shift-And-Add process.

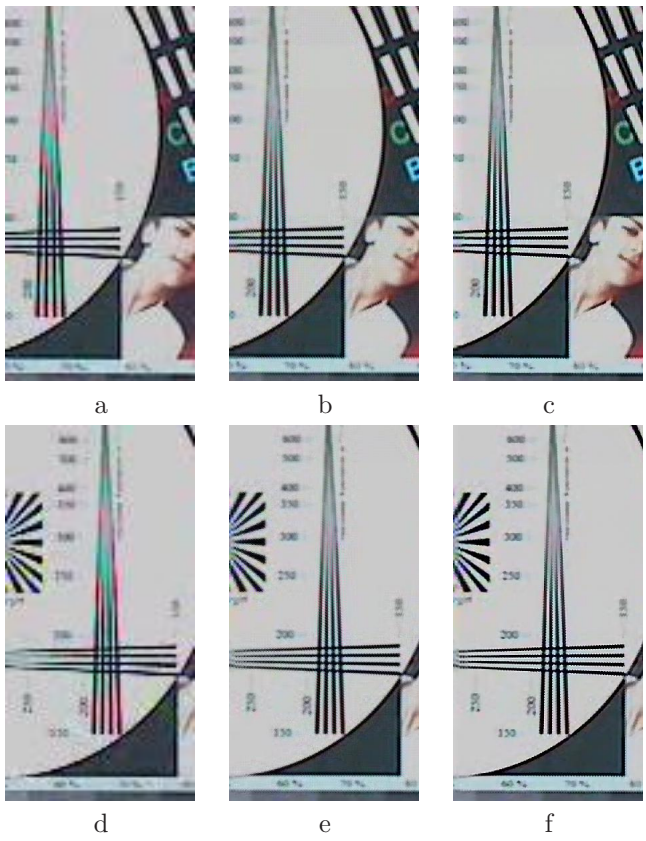

Figure 3. A sequence of 60 real-world LR compressed color frames (a & d) are recursively fused (Section 3), increasing their resolution by the factor of four in each direction (b & e). They were further deblurred (Section 4), resulting in images with much higher-quality than the input LR frames (c & f).

needed to estimate the motion between the luminance components of these images [24]. We used the method described in [25] to computed the motion vectors.

6. SUMMARY AND FUTURE WORK

In this paper, we presented algorithms to enhance the quality of a set of noisy, blurred, and possibly color filtered images to produce a set of color HR images with less noise and blur effects. We used MAP estimation technique to derive a hybrid method of dynamic SR and multi-frame demosaicing. Our method is also applicable to the case of color SR.

For the case of translational motion and common space-invariant motion we justified a two-step algorithm. In the first step, we used the Kalman filtering framework for fusing LR images recursively in a fast and memory efficient way. In the second step, while deblurring and interpolating the missing values, we reduced luminance and color artifacts by using appropriate penalty terms. These terms were based on our prior knowledge of the statistics of natural images and the properties of the human visual system. All matrix-vector operations in the proposed method are implemented as simple image operators.

While the proposed demosaicing method is applicable to a very wide range of data and motion models, our dynamic SR method is developed for the case of translational motion and common space-invariant blur. A fast and robust recursive data fusion algorithm based on using L_1 norm minimization applicable to general motion models is part of our ongoing work.

REFERENCES

1. T. S. Huang and R. Y. Tsai, "Multi-frame image restoration and registration," *Advances in computer vision and Image Processing*, vol. 1, pp. 317–339, 1984.
2. N. Nguyen, P. Milanfar, and G. H. Golub, "A computationally efficient image superresolution algorithm," *IEEE Trans. Image Processing*, vol. 10, no. 4, pp. 573–583, Apr. 2001.
3. M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP:Graph. Models Image Process*, vol. 53, pp. 231–239, 1991.
4. M. Elad and A. Feuer, "Restoration of single super-resolution image from several blurred, noisy and down-sampled measured images," *IEEE Trans. Image Processing*, vol. 6, no. 12, pp. 1646–1658, Dec. 1997.
5. A. Zomet and S. Peleg, "Efficient super-resolution and applications to mosaics," in *Proc. of the Int. Conf. on Pattern Recognition (ICPR)*, Sept. 2000, pp. 579–583.
6. S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multi-frame super-resolution," *To appear in IEEE Trans. Image Processing*, Oct. 2004.
7. S. Borman and R. L. Stevenson, "Super-resolution from image sequences - a review," in *In Proc. of the 1998 Midwest Symposium on Circuits and Systems*, Apr. 1998, vol. 5.
8. S.C. Park, M.K. Park, and M. G. Kang, "Super-resolution image reconstruction, a technical overview," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 21–36, May 2003.
9. S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution," *Invited paper: to appear in the International Journal of Imaging Systems and Technology*, Aug. 2004.
10. M. Elad and A. Feuer, "Super-resolution reconstruction of image sequences," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 9, pp. 817–834, Sept. 1999.
11. C.A. Laroche and M.A. Prescott, "Apparatus and method for adaptive for adaptively interpolating a full color image utilizing chrominance gradients," United States Patent 5,373,322, 1994.
12. R. Kimmel, "Demosaicing: Image reconstruction from color ccd samples," *IEEE Trans. Image Processing*, vol. 8, no. 9, pp. 1221–1228, Sept. 1999.
13. D. Keren and M. Osadchy, "Restoring subsampled color images," *Machine Vision and applications*, vol. 11, no. 4, pp. 197–202, 1999.
14. Y. Hel-Or and D. Keren, "Demosaicing of color images using steerable wavelets," Tech. Rep. HPL-2002-206R1 20020830, HP Labs Israel, 2002.
15. Bahadir K. Gunturk, Yucel Altunbasak, and Russell M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Trans. Image Processing*, vol. 11, no. 9, pp. 997–1013, Sep. 2002.
16. D. Alleysson, S. Süsstrunk, and J. Hrault, "Color demosaicing by estimating luminance and opponent chromatic signals in the fourier domain," in *Proc. IS&T/SID 10th Color Imaging Conf.*, Nov. 2002, pp. 331–336.
17. R. Ramanath, W.E. Snyder, G.L. Bilbro, and W.A. Sander, "Demosaicking methods for the Bayer color arrays," *Journal of Electronic Imaging*, vol. 11, no. 3, pp. 306–315, july 2002.
18. S. C. Pei and I. K. Tam, "Effective color interpolation in ccd color filter arrays using signal correlation," *IEEE Trans. Image Processing*, vol. 13, no. 6, pp. 503–513, June 2003.
19. A. Zomet and S. Peleg, "Multi-sensor super resolution," in *In Proc. of the IEEE Workshop on Applications of Computer Vision*, December 2002, pp. 27–31.
20. N. R. Shah and A. Zakhor, "Resolution enhancement of color video sequences," *IEEE Trans. Image Processing*, vol. 8, no. 6, pp. 879–885, June 1999.
21. B. C. Tom and A. Katsaggelos, "Resolution enhancement of monochrome and color video using motion compensation," *IEEE Trans. Image Processing*, vol. 10, no. 2, pp. 278–287, Feb. 2001.
22. S. M. Kay, *Fundamentals of statistical signal processing:estimation theory*, vol. I, Englewood Cliffs, New Jersey: Prentice-Hall, 1993.
23. W. K. Pratt, *Digital image processing*, New York: John Wiley & Sons, INC., 3rd edition, 2001.
24. P. Golland and A. M. Bruckstein, "Motion from color," *Computer Vision and Image Understanding*, vol. 68, no. 3, pp. 346–362, Dec. 1997.
25. J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, "Hierachical model-based motion estimation," *In Proc. European Conf. on Computer Vision*, pp. 237–252, May 1992.