Superresolution Video Reconstruction with Arbitrary Sampling Lattices and Nonzero Aperture Time

Andrew J. Patti, Member, IEEE, M. Ibrahim Sezan, Senior Member, IEEE, and A. Murat Tekalp, Senior Member, IEEE

Abstract- Printing from an NTSC source and conversion of NTSC source material to high-definition television (HDTV) format are some of the recent applications that motivate superresolution (SR) image and video reconstruction from lowresolution (LR) and possibly blurred sources. Existing methods for SR image reconstruction are limited by the assumptions that the input LR images are sampled progressively, and that the aperture time of the camera is zero, thus ignoring the motion blur occurring during the aperture time. Because of the observed adverse effects of these assumptions for many common video sources, this paper proposes i) a complete model of video acquisition with an arbitrary input sampling lattice and a nonzero aperture time, and ii) an algorithm based on this model using the theory of projections onto convex sets to reconstruct SR still images or video from an LR time sequence of images. Experimental results with real video are provided, which clearly demonstrate that a significant increase in the image resolution can be achieved by taking the motion blurring into account especially when there exists large interframe motion.

Index Terms—Superresolution, video stills, video resampling, standards conversion.

I. INTRODUCTION

WITH THE availability of frame grabbers capable of acquiring multiple frames of video, there is a growing interest in superresolution image and video reconstruction (SR reconstruction), whereby multiple frames are used to overcome the inherent resolution limitations of a low-resolution (LR) camera system. SR reconstruction proves useful in many practical applications, including printing SR stills from video, where it is often desirable to enlarge an image and increase the detail. Because video signals are commonly interlaced, creating SR stills requires a combination of deinterlacing and removing acquisition degradations. Some other applications are conversion from NTSC video to a high-definition television (HDTV) standard, and creation of synthetic "video zoom,"

Manuscript received February 21, 1995; revised July 3, 1996. This work was supported in part by an NSF IUCRC grant and a New York State Science and Technology Foundation grant to the Center for Electronic Imaging Systems at the University of Rochester, and a grant by Eastman Kodak Company. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Michael T. Orchard.

A. J. Patti was with the Department of Electrical Engineering, University of Rochester, Rochester, NY. He is now with Hewlett-Packard Laboratories, Palo Alto, CA 94304 USA (e-mail: patti@hpl.hp.com).

M. I. Sezan was with the Electronic Imaging Research Laboratories, Eastman Kodak Company, Rochester, NY 14627 USA. He is now with Sharp Laboratories of America, Camas, WA 98607 USA.

A. M. Tekalp is with the Department of Electrical Engineering and Center for Electronic Imaging Systems, University of Rochester, Rochester, NY 14627 USA.

Publisher Item Identifier S 1057-7149(97)05471-7.

where a region of the video display is enlarged by some factor and then replayed.

SR reconstruction methods consist of three basic components: i) motion compensation, ii) interpolation, and iii) blur and noise removal, which can be implemented separately or simultaneously. Motion compensation is used to map the pixels from all available LR frames to a common reference frame. The motion field can be modeled in terms of a set of individual motion vectors (which can be estimated at each pixel with a technique such as block matching [1]), or by warping mappings such as affine or perspective transformations [2] (which can be estimated as in [3]). The second component, interpolation including sampling lattice conversion, refers to mapping the motion-compensated pixels onto a rectangularly sampled SR grid. The third component, blur and noise removal, is needed to remove the sensor blurring and optical blurring. We note here that the existing methods account for blurring due to nonzero aperture size (sensor blur), but not due to nonzero aperture time (motion blur). This paper demonstrates that motion blurring is a significant source of degradation when there exist large interframe motion (high action video); and then, superresolution and sampling lattice conversion should be handled simultaneously (in a unified framework) with motion-blur modeling. This unified framework is the video formation model proposed in Section II.

When the three components of SR reconstruction are treated in separate processing stages, as in [4] and [5], the interpolation proceeds without regard for the physical degradations in the LR image formation process. These methods are only applicable when the blurs are the same for all LR frames, and can effectively be modeled as a single blur function acting on the resulting SR image. When nonzero aperture time is considered, the motion model would then have to be restricted to constant velocity, uniform translational motion. Further, this method is suboptimal, since the restoration stage is vulnerable to errors made in the interpolation stage. A frequency domain formulation that addressed the interpolation step was originally proposed by Tsai and Huang [6]. This approach, which makes explicit use of the aliasing relationship under the assumption that the SR image is bandlimited, was extended by Kim et al. [7], [8] as a least squares problem where noise and linear shift-invariant (LSI) blurring in the LR images are taken into account, thus simultaneously solving the interpolation and restoration portions of the SR problem. Tom et al. [9], on the other hand, apply the expectation-maximization (EM) algorithm to simultaneously solve the restoration and motion estimation problems. Their algorithm is also implemented in

1057-7149/97\$10.00 © 1997 IEEE

Authorized licensed use limited to: Cliff Reader. Downloaded on October 31,2023 at 03:07:37 UTC from IEEE Xplore. Restrictions apply.

the frequency domain, so like the SR method proposed in [7], [8], it is limited to global translational motion between LR frames and LSI blur functions.

An iterative class of image domain algorithms have also been proposed for SR reconstruction of progressively sampled LR images, that allow for more flexibility in modeling the imaging process, since they can correct for linear shift varying (LSV) degradations. These algorithms simultaneously solve the restoration and interpolation problems by posing a model relating the LR images to the desired SR image, and then using iterative reconstruction techniques to estimate the SR image. In one such algorithm, Stark and Oskoui [10] propose using the method of projections onto convex sets (POCS), and account for the blur caused by the LR sensor geometry. Tekalp, Ozkan, and Sezan [4] also use a POCS formulation and in this case the sensor noise is taken into account in addition to the blur caused by the physical dimensions of the sensor. Both of these POCS-based methods are applied to LR images whose relative motion is described by translations only. Also, they both assume that the aperture time is negligible. A similar formulation of the problem is given by Irani and Peleg [11], in which the method of averaging projections [12] is used to iteratively solve for the SR image. This algorithm is applied to the case of translational and rotational motion between the LR frames, but does not model the noise process. Mann and Picard [13] have extended Irani and Peleg's method by incorporating a perspective motion model into the image acquisition process. Later, Irani and Peleg [14] have posed a slightly different formulation to take into account more general interframe motion modeling, such as an affine model. Here, an iteration is used that can be shown to be equivalent, in certain cases, to the Landweber iteration [15]. Again, this method does not model the noise. The Landweber iteration is also used by Komatsu et al. [16], where they solve the slightly different problem of SR reconstruction from multiple LR cameras. Their motion model is restricted to translations, and they interpolate the LR images to the SR size, then use block matching to compute a vector for every pixel location. Later, they extend their method to the case of multiple cameras with different sensor element sizes [17]. The basic sampling pattern for each sensor, however, is rectangular.

There also exist Bayesian methods for SR image reconstruction that use a statistical a priori model for the SR image, and simultaneously solve the restoration and interpolation problems by using a maximum a posteriori (MAP) or maximum likelihood (ML) formulation. Cheeseman et al. [20] use Gaussian models for all distributions, and estimate both the SR image and the LR image registration parameters (in the general case, they use a six-parameter affine model) with Jacobi's method to iteratively solve the problem. In their method, the sensor point spread function (PSF) can be taken into account; however, the resulting optimization becomes extremely ill-posed. Shultz and Stevenson [19], [20] use a Huber-Markov-Gibbs model for the a priori SR image model, which is intended to preserve edges while providing a global smoothness constraint. In their formulation, the blur function, modeling sensor blur caused by the dimensions of the LR

sensor, is an input to the algorithm. The input data is only considered to be sampled on a progressive lattice.

Although most of the above methods do consider blurring due to nonzero aperture size, they all ignore blurring that occurs during the aperture time, and do not allow for a general description of the sampling lattice used. Sensor aperture time modeling in the framework of the image sequence restoration problem has been proposed by Trussell and Fogel [21], but in this case the sequence of input images are progressively sampled, and the objective is to remove blur degradations, not to increase the spatial sampling density. Since many consumer video cameras use a relatively large aperture time and are interlaced, it is important to consider arbitrary input lattices and correct for motion blurring due to nonzero aperture time. Also, since the motion blur is in general space-varying and varies from one frame to another, resampling over a denser rectangular lattice and deblurring cannot be performed separately. The combination of modeling and POCS formulation we propose will allow simultaneous interpolation from nonrectangular sampling structures and removal of aperture time effects.

To this effect, this paper proposes a novel SR algorithm to process LR imagery sampled on an arbitrary spatio-temporal lattice, and that takes into account a nonnegligible aperture time. SR imagery output on an arbitrary lattice can be computed by appropriately subsampling a sequence of progressively sampled SR images, thus providing the capability to convert from interlaced NTSC sampling lattices to higher resolution HDTV lattices. The remainder of this paper is organized as follows. The model relating the input LR video to a SR version of this video, via an LSV PSF, is delineated in Section II. In Section III, the modeling is used in conjunction with the method of POCS, to derive an algorithm for SR reconstruction. The effectiveness of the proposed algorithm is demonstrated in Section IV by application to real video sequences.

II. MODELING

In this section we present a model that serves to unify the problems of sampling lattice conversion and SR image reconstruction in the presence of nonzero aperture time. Before beginning, we note that because the motion blurring caused by a nonzero aperture time will in general be space and time varying, it cannot be factored out of the SR restoration problem and performed as a separate postprocessing step. Thus, our modeling will directly include the aperture time effect from the beginning.

Conceptually, continuous LR imagery is the spatio-temporal intensity distribution at the focal plane of the camera, where the sensor is placed. The observed, or sampled LR imagery is found at the output of the camera sensor. Continuous SR imagery is defined as the spatio-temporal intensity distribution at the focal plane of the camera, as it would exist if it were not effected by the degradations introduced by the LR camera system. In modeling the LR imaging process, we account for

 motion (caused by movement of the LR camera or changes in the contents of the scene);



Fig. 1. Video formation model.

- 2) a nonzero sensor aperture time;
- nonzero physical dimensions for each individual sensor element (i.e., giving rise to a particular image pixel);
- 4) blurring caused by the imaging optics;
- 5) sensor noise;
- 6) sampling of the continuous scene on an arbitrary space-time lattice.

The proposed model is delineated as follows. First, a video formation model is described. Motion modeling is then included into the video formation model, which results in an LSV system. Next, a discretization is presented to relate a discrete version of the SR image to the observed LR imagery: The result is a discrete LSV system. To preserve generality in the presentation of the overall model, the motion modeling in this section is generic. An important problem, however, is how to actually compute the resulting discrete LSV blur function for a specific motion model, such as an affine model. Therefore, in the Appendix we provide a practical method for computing the discrete LSV blur function when various motion models are considered as a part of the overal modeling presented in this section.

A. Video Formation

The proposed video formation model is depicted in Fig. 1, where the input signal $f(x_1, x_2, t)$ denotes the continuous SR imagery in the focal plane coordinate system (x_1, x_2) . The effects of the physical dimensions of the LR sensor (i.e., blur due to integration over the sensor area) and the optical blur (i.e., out-of-focus blur) are modeled in the first stage of the figure. There, the SR imagery $f(x_1, x_2, t)$ is convolved with the kernels $h_a(x_1, x_2, t)$ and $h_o(x_1, x_2, t)$, which represent the sensor and the focus blurs, respectively. Both kernels are functions of time, but we restrict them to be constant over the aperture time (next stage). The focus blur and aperture dimensions are thus allowed to differ from frame to frame. This is useful for modeling the case of multiple LR cameras, and/or focus changes.

The sensor aperture time is modeled in the second stage of the figure by an integrator whose output is given by

$$g_2(x_1, x_2, t) = \frac{1}{T_a} \int_{t-T_a}^t g_1(x_1, x_2, \tau) d\tau \tag{1}$$

where T_a denotes the sensor aperture time. Note that the first two stages commute, since the first is spatially LSI, and the second is temporally LSI.

The third stage in Fig. 1 models sampling using the arbitrary space-time lattice Λ_s . The output of this stage is denoted by $g_2(m_1, m_2, k)$. As a matter of convention, integer values

 m_1, m_2 , and k, that appear as a function argument, are interpreted as in

$$g_2(m_1, m_2, k) = g_2(\boldsymbol{x}_s, t) |_{\boldsymbol{V}_s[m_1 \, m_2 \, k]^t} \tag{2}$$

where V_s denotes the matrix that specifies the sampling lattice [22], and t denotes the transpose operation. In the last modeling step, additive noise due to the LR sensor is added to the sampled video signal.

B. Including Motion: The LSV System

We now incorporate a motion model into the video formation model to establish an LSV relationship between the LR imagery and the desired SR image at a fixed but arbitrary time t_r . By appropriately setting the value(s) of t_r , a single (still) SR image or a sequence of SR images (i.e., SR video) can be reconstructed.

When a motion model is incorporated into the image formation model, the first two stages in Fig. 1 can be combined to form a single LSV relation. We begin by considering motion as in

$$f(\boldsymbol{x},t) = f(\boldsymbol{M}(\boldsymbol{x},t,t_r),t_r) = f(\boldsymbol{x}_{t_r},t_r)$$
(3)

where \boldsymbol{x} denotes (x_1, x_2) , and $\boldsymbol{M}(\boldsymbol{x}, t, t_r)$ is a transformation relating \boldsymbol{x} , the position of an intensity at time t, to its position \boldsymbol{x}_{t_r} at time t_r . This equation expresses the wellknown assumptions of intensity conservation along motion trajectories. By letting $h_1(\boldsymbol{x}, t) = h_a(\boldsymbol{x}, t) * * h_o(\boldsymbol{x}, t)$, the output of the first modeling stage can be expressed as

$$g_1(\boldsymbol{x},t) = \int h_1(\boldsymbol{x}-\boldsymbol{\chi}) f(\boldsymbol{\chi},t) \, d\boldsymbol{\chi}.$$
 (4)

By making the change of variables $\boldsymbol{x}_{t_r} = \boldsymbol{M}(\boldsymbol{\chi}, t, t_r)$ and using (3), (4) becomes

$$g_{1}(\boldsymbol{x},t) = \int h_{1}(\boldsymbol{x} - \boldsymbol{M}^{-1}(\boldsymbol{x}_{t_{r}},t,t_{r}))f(\boldsymbol{x}_{t_{r}},t_{r})$$
$$\cdot |\boldsymbol{J}(\boldsymbol{M})|^{-1} d\boldsymbol{x}_{t_{r}}$$
(5)

where M^{-1} denotes the inverse transformation, J(M) denotes the Jacobian of M, and $|\cdot|$ denotes the determinant operator. It is evident from (5) that the first stage of the model has been transformed into an LSV operation, acting on a SR image at time t_r . To reflect this fact, we let

$$h_1(\boldsymbol{x}; \boldsymbol{x}_{t_r}; t, t_r) = h_1(\boldsymbol{x} - \boldsymbol{M}^{-1}(\boldsymbol{x}_{t_r}, t, t_r)) |\boldsymbol{J}(\boldsymbol{M})|^{-1}$$
(6)

denote the LSV PSF modeling the effect of the sensor geometry, focus blur, and relative motion. This equation demonstrates



Fig. 2. Effect of the blur function warping is demonstrated.

an effective sensor warping, which is depicted in Fig. 2. In the figure, the picture to the left represents the imaging process at time t, where a sensor element is imposed on the picture. The picture to the right shows the equivalent imaging process at time t_r . Notice the warping applied to the aperture in going from time t to t_r is the inverse of the warping applied to the image. Rewriting (5) in LSV form yields

$$g_1(\boldsymbol{x},t) = \int h_1(\boldsymbol{x};\boldsymbol{x}_{t_r};t,t_r) f(\boldsymbol{x}_{t_r},t_r) d\boldsymbol{x}_{t_r}.$$
 (7)

The second modeling stage can now be expressed as

$$g_2(\boldsymbol{x},t) = \int_{t-T_a}^{t} h_1(\boldsymbol{x};\boldsymbol{x}_{t_r};\tau,t_r) f(\boldsymbol{x}_{t_r},t_r) \, d\boldsymbol{x}_{t_r} \, d\tau. \quad (8)$$

By changing the order of the integrations, the above becomes

$$g_2(\boldsymbol{x},t) = \int h_2(\boldsymbol{x};\boldsymbol{x}_{tr};t,t_r) f(\boldsymbol{x}_{tr},t_r) \, d\boldsymbol{x}_{tr}$$
(9)

where

$$h_2(\boldsymbol{x}; \boldsymbol{x}_{t_r}; t, t_r) = \int_{t-T_a}^t h_1(\boldsymbol{x}; \boldsymbol{x}_{t_r}; \tau, t_r) \, d\tau.$$
(10)

Thus, the first two stages of the model have been combined into a single LSV system, acting on the continuous SR image at time t_r . This allows us to write the observed LR imagery in terms of a continuous SR image at time t_r , as

$$g(m_1, m_2, k) = \int h_2(m_1, m_2; \boldsymbol{x}_{t_r}; k, t_r) f(\boldsymbol{x}_{t_r}, t_r) \, d\boldsymbol{x}_{t_r} + v(m_1, m_2, k) \tag{11}$$

where $h_2(\cdot)$ is the effective LSV PSF, and the integer arguments m_1, m_2 , and k have the same interpretation as in (2).

C. Discretization

It is desirable to discretize the LSV blur relationship in (11), to relate the observed LR images to a discrete version of $f(x_1, x_2, t_r)$. Thus, a discrete superposition summation of the form

$$g(m_1, m_2, k) = \sum_{\substack{(n_1, n_2) \\ + v(m_1, m_2, k)}} f(n_1, n_2, t_r) h_{t_r}(n_1, n_2; m_1, m_2, k)$$
(12)



Fig. 3. Depiction of the discrete LSV PSF for the case of translational motion.

will now be formulated. We assume that the continuous imagery $f(x_1, x_2, t_r)$ is sampled on the two-dimensional (2-D) lattice Λ_{t_r} (i.e., (n_1, n_2) are integers that specify a point in Λ_{t_r}), by a SR sensor, to form $f(n_1, n_2, t_r)$.

An individual SR sensor element (giving rise to a single SR image pixel) is assumed to have physical dimensions, which can be used as a unit cell \mathcal{U}_{t_r} for the lattice Λ_{t_r} , and to have a uniform response over its support. Thus, the space of the focal plane is completely covered by the SR sensor. The term $\mathcal{U}_{t_r}(n_1, n_2)$ is used to denote the unit cell \mathcal{U}_{t_r} shifted to the location specified by (n_1, n_2) . With this definition, and with the assumption that $f(x_1, x_2, t_r)$ is approximately constant over $\mathcal{U}_{t_r}(n_1, n_2)$, (11) can be written as

$$g(m_1, m_2, k) = \sum_{(n_1, n_2)} f(n_1, n_2, t_r) \cdot h_2(m_1, m_2; \boldsymbol{x}_{t_r}; k, t_r) \cdot d\boldsymbol{x}_{t_r} + v(m_1, m_2, k).$$
(13)

By comparing (12) with (13), it is evident that

$$h_{t_r}(n_1, n_2; m_1, m_2, k) = \int_{\mathcal{U}_{t_r}(n_1, n_2)} h_2(m_1, m_2; \boldsymbol{x}_{t_r}; k, t_r) \, d\boldsymbol{x}_{t_r} \quad (14)$$

where the integer arguments m_1, m_2, k, n_1 , and n_2 are interpreted as in (2).

A pictorial example of the discrete LSV PSF formulation, with a rectangular SR lattice Λ_{t_r} , is provided in Fig. 3. In the figure, it is assumed that the motion is purely translational, that a square LR sensor geometry (outlined in bold) is used, and that there is no focal blur. The (x_1, x_2) space is the sensor focal plane at time t_r . The focal plane is shown covered by shifted SR sampling unit cells $\mathcal{U}_{t_r}(n_1, n_2)$. The LR sensor area is shown outlined in bold, and the larger of the two shaded regions shows the region of the focal plane "swept" by the LR sensor during the aperture time T_a . The discrete LSV PSF specified in (14) is formed by computing the duration of time a given area of the LR sensor "dwelled" over the region $\mathcal{U}_{t_r}(n_1, n_2)$, while translating from the dotted outline at the aperture opening time, to the bold outline at the aperture closing time. Note that the result indicated by (14) does not specify a simple area of overlap between the large shaded region of the figure and the SR sampling regions $U_{t_r}(n_1, n_2)$.

Because the blur PSF (14) is LSV when the sampling geometry is not progressive and/or when the motion is not a global translation between frames, the solution method for SR reconstruction should be capable of processing LSV blurs. Both the Landweber iteration and POCS methods have this property. The POCS solution delineated in the next section, however, has a mechanism for adapting to space-varying properties of the additive noise, whereas the Landweber iteration does not because it converges to the inverse of the least squared error solution.

III. THE POCS SOLUTION

We propose a POCS based solution to the SR reconstruction problem. The method of POCS requires the definition of closed convex constraint sets within a well-defined vector space, that contain the actual SR image. An estimate of the SR image is then defined as a point in the intersection of these constraint sets, and is determined by successively projecting an arbitrary initial estimate onto the constraint sets.

Associated with each constraint set is a projection operator P, mapping an arbitrary point within the space to the closest point within the set. Relaxed projection operators, $T \doteq (1 - \lambda)I + \lambda P$; $0 < \lambda < 2$, can also be defined and used in finding an estimate in the intersection set. For a detailed review of POCS, please see [23].

We define the following closed, convex constraint sets, one for each pixel within the LR image sequence $g(m_1, m_2, k)$:

$$C_{t_r}(m_1, m_2, k) = \{ y(n_1, n_2, t_r) \colon |r^{(y)}(m_1, m_2, k)| \\ \leq \delta_0(m_1, m_2, k) \}$$
(15)

where

$$r^{(y)}(m_1, m_2, k) \doteq g(m_1, m_2, k) - \sum_{(n_1, n_2)} y(n_1, n_2, t_r) \\ \cdot h_{t_r}(n_1, n_2; m_1, m_2, k)$$
(16)

is the residual associated with an arbitrary member, y, of the constraint set. We refer to these sets as the data consistency constraint sets. Note that sets $C_{t_r}(m_1, m_2, k)$ can be defined only where the motion information is accurate. It is therefore a simple task to incorporate occlusion and uncovered background knowledge, by only defining sets for appropriate observations. This type of flexibility is an advantage of the POCS-based solution. The quantity $\delta_0(m_1, m_2, k)$ is a bound reflecting the statistical confidence, with which the actual image is a member of the set $C_{t_r}(m_1, m_2, k)$. Since $r^{(f)}(m_1, m_2, k) = v(m_1, m_2, k)$, where f denotes the actual SR image, the local statistics of $r^{(f)}(m_1, m_2, k)$ are identical to those of $v(m_1, m_2, k)$. Hence the bound $\delta_0(m_1, m_2, k)$ is determined from the possibly space and time-varying statistics of the noise process, so that the actual image (i.e., the ideal solution) is a member of the set within a certain statistical confidence. Thus, the POCS solution will be able to model space- and time-varying white noise processes.

The projection $P_{t_r}(m_1, m_2, k)[x(n_1, n_2, t_r)]$ of an arbitrary $x(n_1, n_2, t_r)$ onto $C_{t_r}(m_1, m_2, k)$ can be defined as [24]

$$P_{t_r}(m_1, m_2, k)[x(n_1, n_2, t_r)] = x(n_1, n_2, t_r) + \begin{cases} \frac{(r^{(x)}(\cdot) - \delta_0(\cdot))h_{t_r}(n_1, n_2; \cdot)}{\sum_{o_1}\sum_{o_2}}h_{t_r}^2(o_1, o_2; \cdot) & r^{(x)}(\cdot) > \delta_0(\cdot) \\ 0, & -\delta_0(\cdot) \le r^{(x)}(\cdot) \le \delta_0(\cdot) \\ \frac{(r^{(x)}(\cdot) + \delta_0(\cdot))h_{t_r}(n_1, n_2; \cdot)}{\sum_{o_1}\sum_{o_2}}h_{t_r}^2(o_1, o_2; \cdot) & r^{(x)}(\cdot) < -\delta_0(\cdot) \end{cases}$$
(17)

where the "·" function argument is interpreted as meaning " m_1, m_2, k ."

Additional constraints such as bounded energy, positivity, and limited support can be utilized to improve the results. Here, we also use the amplitude constraint set

$$C_A = \{y(n_1, n_2, t_r) : \alpha \le f(n_1, n_2, t_r) \le \beta\}$$
(18)

with amplitude bounds of $\alpha = 0$ and $\beta = 255$. The projection P_A onto the amplitude constraint set C_A is defined as a clipping operation such that $P_A[x(n_1, n_2, t_r)]$ limits $x(n_1, n_2, t_r)$ to values between α and β .

Given the above projections, an estimate, $\hat{f}(n_1, n_2, t_r)$, of the SR image $f(n_1, n_2, t_r)$, is obtained iteratively from all LR images $g(m_1, m_2, k)$ where constraint sets can be defined, as

Ĵ

$$\hat{f}_{\ell+1}(n_1, n_2, t_r) = \mathbf{T}_A \tilde{\mathbf{T}}[\hat{f}_{\ell}(n_1, n_2, t_r)] \qquad \ell = 0, 1, 2, \cdots$$
(19)

where \hat{T} denotes the composition of the relaxed projection operators projecting onto the family of sets $C_{t_r}(m_1, m_2, k)$. The initial estimate, $\hat{f}_0(n_1, n_2, t_r)$, is obtained by bilinearly interpolating one of the LR images to the SR grid, and then motion compensating. The remaining LR images can be similarly used to estimate the borders of the SR image.

A pictorial depiction of the proposed POCS method is given in Fig. 4. The LSV blur relates a region (shaded) of the current SR image estimate, say $\hat{f}_{\ell}(\cdot)$, to a particular pixel intensity $g(m_1, m_2, k)$ in one of the LR images. The residual term $r^{(\hat{f}_{\ell})}(m_1, m_2, k)$ is then formed, which indicates whether or not the observation could have been formed from the current SR image estimate (within some error bound determined by $\delta_0(m_1, m_2, k)$), and therefore whether the SR estimate belongs to the data consistency set $C_{t_r}(m_1, m_2, k)$. If it is not in the set (i.e., the residual is too large), the projection operator $P_{t_r}(m_1, m_2, k)$ backprojects the residual onto the current SR image estimate [the additive term in (17)], thus forming a new estimate of the SR image that does belong to the set $C_{t_r}(m_1, m_2, k)$, and therefore could have given rise to the observation $g(m_1, m_2, k)$, within the bound $\delta_0(m_1, m_2, k)$. Performing these projections over every LR pixel where a consistency constraint set is defined completes the composite projection $T[f_{\ell}(n_1, n_2, t_r)]$ referred to in (19). Subsequent projection onto the amplitude constraint set completes a single iteration of the POCS algorithm. In theory, the iterations



Fig. 4. Pictorial depiction of the POCS based reconstruction algorithm.

continue until an estimate lies within the intersection of all the constraint sets. In practice, however, iterations are generally terminated according to a certain stopping criterion such as visual inspection of the image quality, or when changes between successive estimates, as measured by some metric (i.e., $||\hat{f}_{\ell} - \hat{f}_{\ell-1}||$, using the L_2 norm), fall below a preset threshold.

One possible implementation of the POCS-based reconstruction algorithm is as follows.

- 1) Choose the reference frame, and thus the reference time t_r .
- 2) Estimate motion to satisfy (3):
 - a. spatially bilinearly interpolate each LR image $g(m_1, m_2, k)$ to the SR image grid;
 - b. estimate motion from each interpolated LR image, to the interpolated LR image at t_r .
- 3) Define sets $C_{t_r}(m_1, m_2, k)$ according to (15), for each pixel site (m_1, m_2, k) where the motion path is accurate.
- 4) Compute the blur $h_{t_r}(n_1, n_2; m_1, m_2, k)$ for every site (m_1, m_2, k) where the sets $C_{t_r}(m_1, m_2, k)$ have been defined.
- 5) Estimate $f_0(n_1, n_2, t_r)$ by motion compensating one of the interpolated images from step 2 (use the other LR images in a similar manner for estimating the borders).
- 6) For all sites (m_1, m_2, k) where the sets $C_{t_r}(m_1, m_2, k)$ have been defined:
 - a. compute the residual $r^{(\hat{f}_{\ell})}(m_1, m_2, k)$ according to (16);
 - b. backproject the residual $r^{(\hat{f}_t)}(m_1, m_2, k)$ using the projection $P_{t_r}(m_1, m_2, k)$ in (17).
- 7) Perform the amplitude projection P_A based on (18).
- 8) If the stopping criterion is satisfied then stop, otherwise go to 6.

IV. RESULTS

We have conducted three experiments to demonstrate the performance of the proposed SR image reconstruction algorithm. The digitized data for each experiment has been acquired using consumer grade imaging and capture devices, and the motion and blur functions will need to be estimated from the data. These experiments show upsampling from diamond and interlaced lattices when motion blur is negligible, and clearly show the importance of modeling the aperture time when the interframe motion is large.

The first experiment uses LR images acquired using a digital camera placed at different positions against a stationary target. This camera uses a color filter array (CFA) that samples the green channel on a nonrectangular diamond-shaped grid. We will process only the green channel in this first experiment. To create a color SR image, we can separately process the red and blue channels as well, and combine these with the result from the green channel processing. In the second experiment, we consider an interlaced LR video sequence obtained by digitizing the S-video output of a Hi-8 tape, recorded and played back by a consumer grade Sony Hi-8 camcorder. The third experiment uses input LR video captured by digitizing the live feed from the S-video output of the same Hi-8 camcorder. In the first two cases, relative motion is not excessive, and the effect of the aperture time is negligible. In the last experiment, motion is sufficiently large and the effect of aperture time is nonnegligible. In both the second and third experiments, we convert the digitized color signal to a luminance signal, and only process the luminance. Although the camcorder may have a CFA pattern, we do not have direct access to the CFA output, and as a result treat the grabbed LR data as if it originates from a luminance charge coupled device (CCD) array sensor. Color images could be obtained by processing the chrominance channels as well, and combining the results.

In all experiments, the proposed algorithm is used to simultaneously resample a reference LR image over a progressive grid, increase the sampling density of this grid by a factor of two in both dimensions, and undo the effects of sensor and optical blurs. Since we have not yet discussed how the motion information that is included in the model should be estimated in practice, that discussion comes next. Then, each of the three experiments are described in detail. In each experiment, the blur PSF $h_{t_r}(\cdot)$ is computed using the methods detailed in the Appendix.

A. Estimating Motion

The complexity of the modeling described in Section II, for computing the blur PSF $h_{t_r}(\cdot)$, is determined by the motion model. In the simplest case, the motion from the LR images to the reference can be modeled as a spatially uniform translation. In practice, however, we have found this model to be inadequate. As a result, this section uses hierarchical block matching (HBM) methods to estimate nonuniform translational motion, and also affine motion models and estimators. In either case, the performance of the proposed POCS-based SR algorithm will ultimately be limited by the effectiveness of the motion estimation and modeling. We use HBM in the first

TABLE I BLOCK MATCHING PARAMETERS FOR THE DIGITAL CAMERA EXPERIMENT

Level	Max Displ.		Window Size		Filter Size		Step Size	SSF	Accuracy
	hor.	vert.	hor.	vert.	hor.	vert.			
1	31	31	128	128	10	10	32	8	1
2	15	15	64	64	10	10	16	8	1
3	7	7	64	64	5	5	8	4	1
4	3	3	28	28	5	5	4	4	1
5	1	1	12	12	3	3	3	2	0.25

experiment to show that when the aperture time is negligible and the nontranslational components of the actual motion are small, this nonparametric motion estimation method is effective. In all cases, the input LR images are bilinearly interpolated over a LR, progressive grid, for the purpose of motion estimation.

In the case of block matching, the motion is assumed to be locally translational. When other warping effects are small, this approximation can be quite effective, as our experiments will demonstrate. The HBM method of Bierling [1] is used to estimate the nonuniform motion field. The matching criterion used is the mean absolute difference (MAD) between measurement blocks. At each level in the hierarchy, a log-D type search is used, where the following parameters are tabulated for each experiment. The maximum horizontal/vertical displacement (Max Disp. hor./vert.) is the displacement used in the first step of the log-D search. The horizontal/vertical measurement window size (Window Size hor./vert.) is the size of the window over which the MAD is computed. The horizontal/vertical filter size (Filter Size hor./vert.) specifies the support of a Gaussian filter, with variance set to one-half of the support size. The step size is the horizontal and vertical distance between neighboring pixels in the reference image for which an estimate of the motion is computed, the subsampling factor (SSF) is the horizontal and vertical subsampling used when computing the MAD over the measurement window, and the accuracy of estimation is in terms of the sampling period of the progressive LR lattice. The parameter values for HBM are shown in Table I. Note that all units for the parameters are relative to the spatial sampling period of a progressive LR lattice (i.e., refinement to one-quarter-pixel accuracy, relative to the progressive LR lattice, is performed in the final level of HBM).

The second and third experiments model the interimage motion using the global affine transformation defined by the parameters c_1 - c_6 in

$$g(x_1 + c_1 + c_2x_1 + c_3x_2, x_2 + c_4 + c_5x_1 + c_6x_2, t)$$

= $g_p(x_1, x_2, t_r).$ (20)

This parametric modeling method is more descriptive of the actual imaging process, so the resulting blur computation will be more accurate, as well as more involved. Also, the spurious inaccurate vectors that can be introduced by HBM are eliminated. The technique we use to estimate the parameters $c_1, c_2, \dots c_6$ is summarized by Bergen *et al.* [3], and is based on a Taylor series expansion of the optical flow equation. This method requires spatial and temporal derivatives to be



Fig. 5. Graphical depiction of the computing for the LSI blur function $h_2^\prime(x_1,x_2,t,t_r).$

estimated. The spatial derivatives are estimated using a 2-D second-order polynomial least-squares fit over a 5 \times 5 window centered at each pixel, while the temporal derivatives are computed using a two-point finite forward difference at each pixel. Prior to estimating these derivatives, the images are blurred using an 11 \times 11 pixel uniform blur to reduce the effects of noise.

B. Digital Camera Experiment

In this experiment, we use a digital camera to acquire multiple LR images. The camera uses a CFA that samples the green channel over a diamond grid. Six LR images of a stationary text target are acquired, with the camera being placed on a table in approximately the same position for each picture. Because of the approximate camera placement, the images contain relative motion. The goal of this experiment is to simultaneously convert one of the LR images from the diamond grid to a progressive one, undo the effects of sensor and optical blurs, and increase the density of the progressive grid by a factor of two in both spatial dimensions. The aperture time is considered to be negligible in this experiment, and LR images of size 130×142 (horizontal by vertical, relative to a rectangular grid) are cut from the same location in the full size images.

The six LR images are shown in Fig. 7, parts "a"–"f," where they have been bilinearly interpolated for display purposes, to fill a rectangular sampling grid. The image in part "a" is taken



Fig. 6. Demonstration of the approximation used for affine and perspective warps.

 TABLE II

 Estimated Affine Motion Parameters for the Second Experiment

image	Affine Parameters							
	c_1	c_2	c_3	c_4	c_5	c_6		
b	1.609	0.0002	-0.0042	0.113	0.0095	0.0013		
с	3.791	0.0007	-0.0059	0.584	-0.0038	0.0031		
d	5.471	0.0037	-0.0141	0.815	-0.0018	0.0087		
е	6.278	-0.0009	-0.0017	0.471	-0.0069	0.0010		
f	6.322	0.0076	-0.0026	0.122	0.0126	-0.0046		
g	6.510	0.0063	-0.0066	0.146	-0.0010	-0.0009		
h	6.013	0.0017	-0.0080	0.572	0.0068	-0.0081		
i	5.810	-0.0078	-0.0051	0.722	-0.0039	-0.0037		
j	5.390	0.0028	-0.0096	1.280	-0.0034	0.0031		
k	5.043	-0.0032	-0.0025	2.059	-0.0004	-0.0185		
1	5.647	-0.0055	-0.0079	1.601	-0.0094	0.0042		

as the reference, and motion is estimated from each LR image to the reference using the parameters shown in Table I. Motion estimation is performed on the bilinearly interpolated images shown in the figure, so the vectors are relative to a rectangular LR grid. The horizontal and vertical dimensions of the LR sensor are taken to be twice the size of the corresponding SR sensor element dimensions, and the sensor is taken to have a square support with uniform response. The optical blur function h_o is assumed to be Gaussian with unity variance, and have a 5 × 5 support, in terms of the sampling period of the SR image.

The POCS algorithm is initialized using the image shown in in Fig. 7, part "g." The relaxation parameter, λ , in (19) is set to 0.1, the parameter $\delta_0(m_1, m_2, k)$ is assumed to be 0.01, and the minimum and maximum allowable grey levels are 0 and 255, respectively. This choice of parameters leads to fairly rapid convergence, which is measured in terms of visual quality. The sensitivity of the POCS algorithm to these parameters has been addressed under similar constraint sets in [25]. The POCS result after 20 iterations is shown in part "h." We chose 20 iterations simply because no visual changes are detectable in the SR estimates at this point. It is clear from the pictures that the proposed algorithm has created a SR image with considerably more resolution than is seen in the LR images. The reason for some of the aliasing artifacts



Fig. 7. Results from the digital camera experiment. Parts a-f are the LR images, converted to a rectangular sampling grid for display using bilinear interpolation. Part g is the SR image computed using bilinear interpolation, and part h is the POCS result.

that are apparent in the resulting SR image is due to only using six LR images; the problem becomes underdetermined with six images, since we need at least eight LR images to have the same number of unknowns as equations.

C. Camcorder Experiment: Negligible Aperture Time

In this experiment, we use a Hi-8 camcorder to acquire interlaced LR images. The camcorder is hand held while aimed at the target scene, and the video is recorded on an Hi-8mm tape. The target is a cardboard containing musical notes, and is not perfectly planar. Six frames (12 fields) of the taped video are captured by frame grabbing the output of the camcorder during playback. The motion in the scene is due to the movement of the hand-held camera and is not excessive. The effect of the aperture time is considered to be negligible. We apply the proposed algorithm to simultaneously convert from an interlaced sampling lattice to a progressive one, increase the sampling density by a factor of two in both horizontal and vertical directions, and reduce the effects of sensor and optical blurs.

The 12 LR fields of size 80×40 are cut from the same location in each of the grabbed fields, and are shown in Fig. 8, parts "a"–"l." The field shown in part "a" of the figure is taken as the reference image, and the affine motion parameters listed in Table II are estimated using the previously discussed method. All of the POCS parameters are set to the same values as in the previous experiment. The initialization obtained using

image	e Cut Offset		Affine Parameters						
	hor.	vert.	<i>c</i> ₁	c_2	c_3	c_4	c_5	c_6	
b	10	-10	-0.027	-0.0062	0.0007	1.319	0.0001	-0.0064	
с	20	-20	-0.091	-0.0128	-0.0001	2.524	0.0053	-0.0089	
d	30	-32	-0.469	-0.0125	-0.0037	0.807	0.0060	-0.0103	
е	41	-42	0.294	-0.0170	-0.0045	0.931	0.0086	-0.0144	
f	50	-52	-0.581	-0.0185	-0.0068	-0.278	0.0128	-0.0128	
g	60	-60	-0.008	-0.0212	-0.0080	0.717	0.0162	-0.0132	
h	69	-68	-0.107	-0.0210	-0.0094	1.083	0.0175	-0.0151	
i	79	-76	1.375	-0.0220	-0.0115	1.289	0.0175	-0.0142	
j	88	-85	2.555	-0.0230	-0.0139	0.401	0.0160	-0.0163	

 TABLE III

 ESTIMATED AFFINE MOTION PARAMETERS FOR THE THIRD EXPERIMENT

bilinear interpolation is shown in Fig. 8, part "m," and the POCS result after ten iterations is shown in part "n." Again, it is clear from the pictures that the POCS result has created an SR image with considerably more resolution than is seen in either the LR images or the bilinear interpolated SR image. This experiment is important, since anyone with access to a consumer camcorder could obtain such LR images.

D. Camcorder Experiment: Nonnegligible Aperture Time

The third experiment uses LR interlaced video that is captured by grabbing the live S-video output from the same Hi-8 camcorder used in the previous experiment. In this case, the motion during the aperture time is nonnegligible, and we demonstrate the use of our algorithm in the presence of interlaced video effected by motion blurring. An SR image is reconstructed over a progressive grid that is two times denser in both dimensions than that of the given frames (so for a given field, the upsampling is by two horizontally and by four vertically).

Motion blur is introduced by imaging a moving poster board containing both text and pictorial information. The camera aperture time is set to $\frac{1}{100}$ seconds, which is its default setting. Five interlaced, digital, LR frames (ten fields) are acquired by connecting the S-video output of the camcorder to a frame grabber. LR fields of size 90×45 are then cut out from the grabbed images to form the inputs to our SR algorithm. The cutouts are chosen to contain approximately the same portion of the scene, so they do not come from the same spatial location in every image. The cutting process accounts for a significant global translational motion, and as a result, when motion is estimated between the cutouts this shift is taken into account. These cutouts are shown deinterlaced by bilinear interpolation in Fig. 9, parts "a"-"j," where deinterlaced images are used to make it easier to gain a feel for the motion blurring. Motion is estimated from each deinterlaced LR cutout image to the reference image, chosen as image "a" in the figure, using the previously discussed affine motion estimation method. The results of the motion estimation are tabulated in Table III, along with the relative global shifts incurred during the cutting process (the coordinate system is anchored at the top left corner of the image, with the horizontal component increasing in the left-right direction, and the vertical component increasing in the top-bottom direction). Using the information furnished in the table, the blur is



Fig. 8. Results from the negligible aperture time Hi-8 camcorder experiment. Parts a–l are the LR fields. Part m is the SR image computed using bilinear interpolation, and part n is the POCS result.

computed as described in the Appendix, for the case of an affine motion model. To initialize the POCS algorithm, the LR image labeled "j" is chosen as the best image.

Three results are shown in Fig. 9, parts "k," "l," and "m." The first SR image, "k," is produced by applying bilinear interpolation to initialize the POCS algorithm. The second SR image, "l," is the result of applying the proposed POCS algorithm when the aperture time is ignored. For this image, the same POCS parameters as in the first experiment are used, and the image is the result of ten iterations. The last SR image, "m," is the result of applying the proposed POCS algorithm when the aperture time is taken into account. The resolution in this SR image is far greater than in the other two images, and since we have shown the POCS method to be effective when the aperture time is indeed negligible, the importance of modeling the aperture time is evident.

To both more fully appreciate this result and provide a more realistic sized image for viewing, a larger section of the same processed image is shown in Fig. 10. In the figure, the color channels have been included. There is no noticeable



Fig. 9. Results from the nonnegligible aperture time Hi-8 camcorder experiment. Parts a-j are the LR images deinterlaced using bilinear interpolation. Part k is the SR image computed using bilinear interpolation of the reference image shown in j. Part l is the SR image obtained if the aperture time is ignored, and part m is the SR image resulting from the proposed POCS algorithm.

difference resulting from applying the SR procedure to the chrominance channels, or simply bilinearly interpolating them, so the bilinearly interpolated versions are included. Notice that in the POCS result the strip of film furthest to the right (top portion of the picture) can be seen to contain a saxophone with a hand on it, and that on the film strip immediately to the left is text with the letters "A" and "L." The source of the ringing artifacts is most likely due to our assumption that the acquired video data came directly from a CCD sensor. This is not the case, since the CCD data within the camera has already been processed and converted to the NTSC format before our digitization. We are forced to make this assumption, since in practice we do not have access to the internal camera CCD data. Given the blur support sizes (as high as 25×25), however, we do not believe this ringing is excessive.

As part of this experiment, we also demonstrate the efficacy of using multiple (as opposed to only a single) LR fields. We have used the POCS projection algorithm exactly as previously described in this experiment, with the exception of only using the visually best field (Fig. 9, part "j") to estimate the SR image. The resulting SR image is shown in Fig. 11, part "a." In part "b" of the same figure, the output of the POCS algorithm when all fields are used (Fig. 9, part "m") is shown for comparison. We can see that even though the other fields are far more affected by the motion blurring than field "j," a noticeable improvement is still produced by using them in the algorithm. Note that using any LR field other than "j" for single field processing would provide an even greater contrast in the results, since all other LR fields are more substantially motion blurred than "j." We have thus demonstrated the lower bound on the improvement increase for multifield processing.

Before leaving this section, a few comments are in order. First, with regard to the interframe motion, although the nontranslational component of the motion appears small in the printed pictures, it is large enough that applying the algorithms under the assumption of global translational motion leads to no improvement over bilinear interpolating a single field. Also of interest is that the LSV motion blur is different in each LR frame. As a result, a formulation using a separate postprocessing step for motion blur removal would not be appropriate for the images processed in the third experiment. Last, the computations required for this algorithm are not trivial, but not prohibitive either. Using nonoptimized code, the images in Figs. 7 and 8 are computed on a SPARC-10 platform in roughly 30 min. The images in Fig. 9 require more processing time, roughly 3 h, due to the larger blur sizes caused by the large interimage motion. We emphasize, however, that no attempts have been made to optimize the algorithm implementation or the code used. For instance, the POCS projections easily lend themselves to a parallel implementation.

V. CONCLUSION

We have proposed a model for video acquisition that takes into account sampling on an arbitrary lattice, a sensor element's physical dimensions, the aperture time, focus blurring, and additive noise. This model relates the observed LR video to discretized SR video. The proposed model is then used in developing a POCS-based algorithm for reconstructing a SR image or video from LR video or imagery containing relative motion. Fractional pixel relative motion is necessary for resampling over a denser lattice. However, when motion is large, resolution improvement is not possible unless blur due to nonzero aperture time is modeled and taken into account. We have demonstrated this fact through examples using real video. treated to serve this goal. In the first case, translational motion is assumed. In the second case, general image warpings are considered. To solve this second case, a general approximation is given, that leads to a blur computation algorithm that is built on top of the algorithm delineated for the translational motion case. Relative to this approximation, we make specific mention

1) Translational Motion: For the case of translational motion, we define piecewise constant velocity motion paths, effective during the kth opening of the aperture (i.e., acquiring the k^{th} LR image at time t_k), as

of the affine and perspective transformations.

$$\boldsymbol{x}_{t_r} = \boldsymbol{M}(\boldsymbol{x}, t, t_r) = \boldsymbol{x} + \boldsymbol{x}_{b_k} + \boldsymbol{v}_k(t - (t_k - T_a))$$
(21)

where the velocities $v_{1,k}$ and $v_{2,k}$, where $\boldsymbol{v}_k = [v_{1,k} \quad v_{2,k}]^t$, are assumed to be constant over the aperture time $T_a, (t_k - T_a)$ is the time of the kth opening of the aperture, and x_{b_k} denotes the relative initial position at the kth opening of the aperture. The quantity \boldsymbol{x}_{b_k} is a function of the times t_k and t_r . If for the moment the focal blur is ignored, then the PSF $h_2(\boldsymbol{x}; \boldsymbol{x}_{t_r}; t, t_r)$ is LSI, and by defining $h'_2(\boldsymbol{x} - \boldsymbol{x}_{t_r}; t, t_r) = h_2(\boldsymbol{x}; \boldsymbol{x}_{t_r}; t, t_r)$ and applying (6) and (10)

$$h_2'(\boldsymbol{x}; t, t_r) = \frac{1}{T_a} \int_0^{T_a} h_a(\boldsymbol{x} + \boldsymbol{x}_{b_k} + \boldsymbol{v}_k \tau) \, d\tau.$$
(22)

If we now assume the aperture response is a 2-D "rect" function given by

$$h_a(x_1, x_2) = \begin{cases} \frac{1}{\Delta M_1 \Delta M_2}; & |x_1| < \frac{\Delta M_1}{2}, |x_2| < \frac{\Delta M_2}{2} \\ 0; & \text{else,} \end{cases}$$

then h'_2 can be computed graphically as depicted in Fig. 5. The coordinate $\boldsymbol{x} + \boldsymbol{x}_{b_k}$ sets the starting point of the line shown in the figure, at time $\tau = 0$. The integral follows the line to its endpoint at $\tau = T_a$, and the result is simply the length of the line segment that intersects the shaded region. Working out this integration shows that the PSF $h_2'(\boldsymbol{x};t,t_r)$ consists of convex regions in the x coordinate system, within each of which $h'_{2}(\cdot)$ is described by the equation of a plane. Computing the discrete PSF $h_{t_r}(n_1, n_2; m_1, m_2, k)$ then requires nothing more than summing the volumes under planar convex regions formed by the intersection of $\mathcal{U}_{t_r}(n_1, n_2)$ [see (14)] and the convex regions that define $h'_2(\cdot)$.

The focus blur $h_o(\mathbf{x}, t)$ can be subsequently taken into account using a discrete approximation by carrying out the convolution

$$h_{t_r}(n_1, n_2; m_1, m_2, k) *_{n_1, n_2} h^d_{o_k}(n_1, n_2)$$
 (23)

where $h_{o_k}^d(n_1, n_2)$ is the discrete representation of the focus blur for the k^{th} LR image, and $*_{n_1,n_2}$ denotes 2-D discrete convolution over the variables (n_1, n_2) . By taking the focal blur into account in this way, we are making the assumption that the blur PSF h_{t_r} , within a region about $\boldsymbol{x}_s(m_1, m_2, k)$, is approximately LSI. This is a reasonable assumption as long as the image has not undergone an extreme nontranslational warping. Handling the focus blur as in (23) is attractive, since h_{t_r} can easily be computed when the focus blur is not considered, and the convolution in (23) is easy to implement.

Fig. 10. Results from the nonnegligible aperture time Hi-8 camcorder experiment, where large regions are processed, and color is included. At top is the POCS initialization, and at bottom is the result after six iterations.

Fig. 11. Results from the nonnegligible aperture time Hi-8 camcorder experiment, where only a single field is used for estimating the SR image. Part a shows the SR image generated using only LR field j from the previous figure, and part b shows the result of using all the fields (same as m in the previous figure).

In addition, experiments with real data have been performed that demonstrate the effectiveness of our algorithm when the input imagery is not sampled progressively, and the input imagery is motion blurred with different blurs in each field.

APPENDIX

A PRACTICAL BLUR COMPUTATION METHOD

This appendix describes a practical method for computing the blur function $h_{t_r}(\cdot)$ given by (14). Two cases will be



2) General Warpings: We now extend this method for computing the blur to the case of more complex warpings, such as those described by affine or perspective transformations. The extension is based on the following observation: The warping between the times t_r and t_k may be significant, however, the nontranslational component of the warping that effects the blur shape will be small during the aperture time. This concept is demonstrated in Fig. 6. The figure is a graphical representation of the computation described in (10), which is rewritten here as

$$h_{2}(\boldsymbol{x};\boldsymbol{x}_{t_{r}};t,t_{r}) = \int_{t-T_{a}}^{t} h_{1}(\boldsymbol{x}-\boldsymbol{M}^{-1}(\boldsymbol{x}_{t_{r}},\tau,t_{r})) \\ \cdot |J(\boldsymbol{M}(\boldsymbol{x}_{t_{r}},\tau,t_{r})|^{-1} d\tau.$$
(24)

The figure depicts the warped blur kernel $h_1(\cdot)$ as it translates across the point x_{t_r} during the integration time from $t-T_a$ to t. The value of $h_2(\cdot)$ is then the "dwell" time over x_{t_r} , weighted by the Jacobian and the amplitude of $h_1(\cdot)$. Computation of (24) is difficult since the translating kernel $h_1(\cdot)$ is continuously warping during the integration period. As previously pointed out, however, the nontranslational component of this warping, during the aperture time, is assumed to be small. This effect is demonstrated in Fig. 6 by showing the dotted outline of the function $h_1(M^{-1}(\boldsymbol{x}_{t_r}, t-T_a, t_r))$ superimposed on the shaded region representing $h_1(M^{-1}(x_{t_r}, t, t_r))$. In terms of (24), the approximation makes the assumptions that i) the Jacobian weighting is a constant, ii) the warping $M(x_{t_r}, \tau - T_a, t_r)$ is maintained throughout the aperture time (i.e., this function only translates as τ changes), and iii) the path of translation during two consecutive frames and, thus, within the aperture time, is linear. With this approximation, (24) can be rewritten as

$$|J(\boldsymbol{M}(\boldsymbol{M}(\boldsymbol{x}, t - T_a, t_r), \tau, t_r))|^{-1} \cdot \int_{t - T_a}^{t} h_1(\ell(\boldsymbol{x}, \tau) - \boldsymbol{M}^{-1}(\boldsymbol{x}_{t_r}, t - T_a, t_r)) d\tau \quad (25)$$

where

$$\ell(\boldsymbol{x},\tau) = \frac{T-\tau}{T} \boldsymbol{x} + \frac{\tau}{T} \boldsymbol{M}^{-1}(\boldsymbol{M}(\boldsymbol{x},t-T_a+T,t_r),t-T_a,t_r)$$
(26)

and T is the time between consecutive frames.

Using this approximation, the same procedure for computing the blur in the case of spatially uniform, temporally piecewise constant-velocity translational motion is used, except that at each point \boldsymbol{x} the blur is computed with the appropriate warping applied to the rectangular function depicted in Fig. 5. To summarize, when the warping is defined by uniform and constant translations, the approximation will result in an exact blur computation. When the warping is affine, the Jacobian does not vary with \boldsymbol{x}_{t_r} , but we have approximated it to be constant over time, while the aperture is open. Additionally, the translation is assumed to be constant velocity, where this may not necessarily the case. In the case of perspective motion, the approximation has the same effects as in the affine case, with the additional approximation that the Jacobian is constant over the spatial blur support of $h_1(\cdot)$. This approximation is used for the affine case to compute the blurs in Section IV, Results.

REFERENCES

- M. Bierling, "Displacement estimation by hierarchical blockmatching," in *Proc. SPIE Visual Communications and Image Processing* '88, pp. 942–951.
- [2] G. Wolberg, *Digital Image Warping*. Los Alamitos, CA: IEEE Comput. Soc. Press, 1990.
- [3] J. Bergen, P. Burt, R. Hingorani, and S. Peleg, "A three-frame algorithm for estimating two-component image motion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, pp. 886–896, Sept. 1992.
- [4] A. M. Tekalp, M. K. Ozkan, and M. I. Sezan, "High-resolution image reconstruction from lower-resolution image sequences and spacevarying image restoration," in *IEEE Int. Conf. Acoustics, Speech, Signal Processing*, San Francisco, CA, Mar. 23–26, 1992, pp. III-169–172.
- [5] H. Ur and D. Gross, "Improved resolution form subpixel shifted pictures," *CVGIP: Graph. Models Image Processing*, vol. 54, pp. 181–186, Mar. 1992.
- [6] R. Y. Tsai and T. S. Huang, "Multiframe image restoration and registration," in Advances in Computer Vision and Image Processing, T. S. Huang, Ed. Greenwich, CT: JAI Press, 1984.
- [7] S. P. Kim, N. K. Bose, and H. M. Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframes," *IEEE Trans. Acoust., Speech Signal Processing*, vol. 38, pp. 1013–1027, June 1990.
- [8] S. P. Kim and W.-Y. Su, "Recursive high-resolution reconstruction of blurred multiframe images," *IEEE Trans. Image Processing*, vol. 2, pp. 534–539, Oct. 1993.
- [9] B. C. Tom, A. K. Katsaggelos, and N. P. Galatsanos, "Reconstruction of a high resolution image from registration and restoration of low resolution images," in *Proc. IEEE Int. Conf. Image Processing*, Austin, TX, Nov. 13–16, 1994.
- [10] H. Stark and P. Oskoui, "High-resolution image recovery from imageplane arrays, using convex projections," J. Opt. Soc. Amer. A, vol. 6, pp. 1715–1726, 1989.
- [11] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graph. Models Image Processing*, vol. 53, pp. 231–239, May 1991.
- [12] M. R. Civanlar and H. J. Trussell, "The Landweber iteration and projection onto convex sets," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1632–1634, Dec. 1985.
 [13] S. Mann and R. W. Picard, "Virtual bellows: constructing high quality
- [13] S. Mann and R. W. Picard, "Virtual bellows: constructing high quality stills from video," in *Proc. IEEE Int. Conf. Image Processing*, Austin, TX, Nov. 13–16, 1994.
- [14] M. Irani and S. Peleg, "Motion analysis for image enhancement: Resolution, occlusion, and transparency," J. Visual Commun. Image Represent., vol. 4, pp. 324–335, Dec. 1993.
- [15] A. J. Patti, "Digital video filtering for standards conversion and resolution enhancement," Ph.D. dissertation, Univ. Rochester, Rochester, NY, 1995.
- [16] T. Komatsu, K. Aizawa, T. Igarashi, and T. Saito, "Signal-processing based method for acquiring very high resolution images with multiple cameras and its theoretical analysis," *Proc. Inst. Electr. Eng.*—*I*, vol. 140, pp. 19–25, Feb. 1993.
- [17] T. Komatsu, T. Igarashi, K. Aizawa, and T. Saito, "Very high resolution imaging sheme with multiple different-aperature cameras," *Signal Processing: Image Commun.*, vol. 5, pp. 511–526, Dec. 1993.
- [18] P. Cheeseman, B. Kanefsky, and R. Hanson, "Super-resolved surface reconstruction from multiple images," Tech. Rep., NASA, Jan. 1993.
 [19] R. R. Schultz and R. L. Stevenson, "A Bayesian approach to image
- [19] R. R. Schultz and R. L. Stevenson, "A Bayesian approach to image expansion for improved definition," *IEEE Trans. Image Processing*, vol. 3, pp. 233–242, May 1994.
- [20] R. R. Schultz and R. L. Stevenson, "Improved definition video frame enhancement," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Detroit, MI, May 1995, pp. 2169–2172.
- [21] H. J. Trussell and S. Fogel, "Identification and restoration of spatially variant motion blurs in sequential images," *IEEE Trans. Image Processing*, vol. 1, pp. 123–126, Jan., 1992.
 [22] E. Dubois, "The sampling and reconstruction of time-varying imagery
- [22] E. Dubois, "The sampling and reconstruction of time-varying imagery with application in video systems," *Proc. IEEE*, vol. 73, pp. 502–522, Apr. 1985.
- [23] M. I. Sezan, "An overview of convex projections theory and its applications to image recovery problems," *Ultramicroscopy*, no. 40, pp. 55–67, 1992.

- [24] H. J. Trussell and M. R. Civanlar, "Feasible solution in signal restoration," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 201–212, 1984.
- [25] M. K. Ozkan, A. M. Tekalp, and M. I. Sezan, "POCS-based restoration of space-varying blurred images," *IEEE Trans. Image Processing*, vol. 3, pp. 450–454, Apr. 1994.



Andrew J. Patti (S'91–M'96) was born in Utica, NY. He received the B.S.E.E degree from Clarkson University, Potsdam, NY, in 1987, and theM.S.E.E. and Ph.D. degrees from the University of Rochester, Rochester, NY, in 1991 and 1995, respectively.

From 1987 to 1990, he worked as an Engineer at the Westinghouse Defense Electronics Center, Baltimore, MD, designing hardware for electro-optical systems. He is currently a Member of Technical Staff at Hewlett Packard Laboratories. His research interests in the area of digital video processing

include motion estimation, standards conversion, resolution enhancement, and restoration and segmentation.



M. Ibrahim Sezan (S'80–M'84–SM'91) received the B.S degrees in electrical engineering and mathematics from Bogazici University, Istanbul, Turkey in 1980, with the highest honors. He received the M.S degree in physics from Stevens Institute of Technology, Hoboken, NJ, and the M.S and Ph.D degrees in electrical computer and systems engineering from Rensselaer Polytechnic Institute (RPI), Troy, NY, in 1982, 1983, and 1984, respectively.

He is currently the Senior Manager of Digital Video Processing, Sharp Laboratories of America,

Camas, WA. From 1984 to 1996, he was with Eastman Kodak Company, Rochester, NY, where he headed the Video and Motion Technology Area, Imaging Research and Advanced Development Laboratories, from 1992 to 1996. He also holds an Adjunct Associate Professor position at the Electrical Engineering Department, University of Rochester.

Dr. Sezan was the co-recipient of the A. B. Du Mont award at RPI in 1984. During 1988–1992, he served as an Associate Editor of the IEEE TRANSACTIONS ON MEDICAL IMAGING. From 1992 to 1994, he was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING. He is a member of the Multidimensional Digital Signal Processing Committee of the IEEE Signal Processing Society. He contributed to the books *Image Recovery: Theory and Application* (New York: Academic, 1987), *Mathematics in Signal Processing* (Oxford, U.K.: Oxford 1987), *Handbook of Signal Processing* (Marcel Dekker, 1988), *Digital Image Restoration* (New York: Academic, 1994). He edited *Selected Papers in Digital Image Restoration* (SPIE Milestone Series, 1992), and co-edited *Motion Analysis and Image Sequence Processing* (Boston, MA: Kluwer, 1993) His current research interests include motion analysis, video processing, content-based compression, and multimedia data bases. He is a member of Sigma Xi.



A. Murat Tekalp (S'80–M'84–SM'91) received the B.S. degree in electrical engineering and mathematics (with highest honors) from Boğazici University, Istanbul, Turkey, in 1980, and the M.S. and Ph.D. degrees in electrical, computer and systems engineering from Rensselaer Polytechnic Institute (RPI), Troy, NY, in 1982 and 1984, respectively.

From December 1984 to August 1987, he was a Research Scientist and then a Senior Research Scientist at Eastman Kodak Company, Rochester, NY. He joined the Electrical Engineering Depart-

ment at the University of Rochester as an Assistant Professor in September 1987, and is currently a Professor. His current research interests are in the areas of digital image and video processing, including image restoration and reconstruction, object-based image/video editing and coding, object-tracking, and image/video indexing for digital libraries.

Dr. Tekalp has served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING (1990-1992), IEEE TRANSACTIONS ON IMAGE PROCESSING (1994–1996), and the Kluwer Journal on Multidimensional Systems and Signal Processing (1994-1996). He has been the Technical Program Chair for the IEEE Signal Processing Society MDSP Workshop (1991), Special Sessions Chair for the IEEE International Conference on Image Processing (1995), and the organizer and first Chairman of the Rochester Chapter of the IEEE Signal Processing Society. He has served as the Chair of the IEEE Rochester Section in the 1994-1995 term of office. At present, he is the Chair of the IEEE Signal Processing Society Technical Committee on Image and Multidimensional Signal Processing, and a member of the IEEE Computer Society Technical Committee on Computer Vision and Image Processing. He is also on the editorial boards of Graphical Models and Image Processing, and Visual Communications and Image Representation. He is the author of Digital Video Processing (Englewood Cliffs, NJ: Prentice-Hall, 1995). He received the NSF Research Initiation Award in 1988, and IEEE Rochester Section Awards in 1989, 1992, and 1994. He is a member of Sigma Xi.