Image Resampling and Constraint Formulation for Multi-Frame Super-Resolution Restoration

Sean Borman and Robert L. Stevenson[a]

Laboratory for Image and Signal Analysis
Department of Electrical Engineering
University of Notre Dame, Notre Dame, IN 46556, U.S.A.

ABSTRACT

Multi-frame super-resolution restoration algorithms commonly utilize a linear observation model relating the recorded images to the unknown restored image estimates. Working within this framework, we demonstrate a method for generalizing the observation model to incorporate spatially varying point spread functions and general motion fields. The method utilizes results from image resampling theory which is shown to have equivalences with the multi-frame image observation model used in super-resolution restoration. An algorithm for computing the coefficients of the spatially varying observation filter is developed. Examples of the application of the proposed method are presented.

Keywords: Image resampling, image warping, multi-frame restoration, super-resolution, image restoration

1. INTRODUCTION

We consider a method for generalizing the observation models used in multi-frame super-resolution image restoration in which multiple, typically undersampled images of a scene are combined to estimate super-resolved images. In the multi-frame restoration framework, the relative scene/camera motion provides constraints which make super-resolution restoration possible. While the algorithmic basis for multi-frame super-resolution restoration is well known, it has been recognized that the performance of these algorithms is highly depended on the fidelity of the observation model to the true imaging system characteristics. To this end we demonstrate a method for incorporating very general spatially-invariant observation models into the restoration framework. This includes (but is not limited to) optical system characteristics, such as the diffraction-limited point spread function, defocus and lens aberrations, sensor effects such as spatially non-uniform pixel responses and blurring due to temporal integration, as well as a convenient method for representing general motion fields. In essence, any degradation which may be modeled using a spatially varying point spread function may be easily incorporated.

We begin with a discussion of *image resampling* or *image warping* in Section (2) which is a technique commonly used in computer graphics for mapping textures onto objects in image rendering applications. In Section (3) we then examine the multi-frame super-resolution restoration problem with emphasis on the model used to relate the observed imagery to the super-resolution restorations. We show that there are direct and useful parallels between the multi-frame restoration observation model and the process of image resampling (summarized in Section (4)). These equivalences allows us to apply the *resampling filter* approach developed for image resampling to the multi-frame restoration problem. In particular, we show how to represent the observation model in the multi-frame restoration framework using a spatially varying *observation filter* akin to the resampling filter used in image resampling.

We demonstrate these ideas with a fully developed example which models an imaging system consisting of a diffraction-limited optical system (Section (6)) and a focal-plane sensor array with non-ideal sampling (Section (7)), under a projective viewing transformation (Section (9)). In Section (10) we present an algorithm for the computation of the spatially varying observation filter, and in Section (11) we show the results of applying these ideas to simulated imagery. Some extensions to and difficulties associated with the proposed method are discussed in Section (12) and conclusions may be found in Section (13).

[a]Author to whom correspondence should be addressed.
Robert L. Stevenson E-mail: rls@nd.edu Telephone: +1-574-631-8308
Sean Borman E-mail: Sean.Borman.1@nd.edu

2. IMAGE RESAMPLING

Image resampling refers to methods for sampling discrete images under coordinate transformation. Typically one is faced with the problem of having to map a discrete input image, often referred to as a *texture* image in the computer graphics literature, to an output or *warped* image via a given coordinate transformation or *warp*.

Let $f(\mathbf{u})$ with $\mathbf{u} = [u\ v]^T \in \mathbb{Z}^2$ represent the sampled input (texture) image, and let the discrete output (warped) image be given by $g(\mathbf{x})$ with $\mathbf{x} = [x\ y]^T \in \mathbb{Z}^2$. Let the map $H : \mathbf{u} \longmapsto \mathbf{x}$ represent the coordinate transformation from the input to the output spaces. We shall assume that the map is invertible, and is given by $H^{-1} : \mathbf{x} \longmapsto \mathbf{u}$.

A naïve implementation of image resampling might go as follows: for each pixel $\mathbf{x} \in \mathbb{Z}^2$ in the warped image g, let $g(\mathbf{x}) = f(H^{-1}(\mathbf{x}))$. While this may at first glance appear appealing, it has two serious problems, one of which is fundamental, the other somewhat more subtle:

1. $H^{-1}(\mathbf{x})$ need not fall on sample points.

   This is a serious problem since $f(\mathbf{u})$ is only defined at sample points. Thus in order to determine the value of f between sampling points, some form of interpolation is required.

2. $H^{-1}(\mathbf{x})$ may undersample $f(\mathbf{u})$ resulting in aliasing.

   This more insidious problem leads to objectionable artefacts in warped images where the mapping results in minification. In the quest for realism in computer graphics rendering, addressing these artefacts spurred the development of image resampling theory as well as the development of many optimized techniques for fast, high-quality, texture rendering.

2.1. Conceptual image resampling pipeline

Heckbert was one of the first researchers to address image resampling issues in his now-classic work.[1] Following Heckbert's exposition, an accurate image resampling implementation is composed of four processes. In these equations, functions of a continuous variable are denoted with a tilde above the function name and $\circledast$ denotes convolution. This sequence of processes is illustrated in Fig. (1).

1. Reconstruct a continuous version of the discrete texture image using the reconstruction filter $r(\mathbf{u})$ as,

$$\tilde{f}(\mathbf{u}) = f(\mathbf{u}) \circledast r(\mathbf{u}) = \sum_{\mathbf{k} \in \mathbb{Z}^2} f(\mathbf{k}) \cdot r(\mathbf{u} - \mathbf{k}). \tag{1}$$

2. Use the inverse mapping which associates locations in the warped output image with their locations in the texture image to find the continuous warped image as,

$$\tilde{g}(\mathbf{x}) = \tilde{f}\left(H^{-1}(\mathbf{x})\right). \tag{2}$$

3. Pre-filter the warped output image using the anti-alias pre-filter $p(\mathbf{x})$ to prevent aliasing in the sampling step,

$$\tilde{g}'(\mathbf{x}) = \tilde{g}(\mathbf{x}) \circledast p(\mathbf{x}) = \int \tilde{g}(\boldsymbol{\alpha}) \cdot p(\mathbf{x} - \boldsymbol{\alpha})\ \mathrm{d}\boldsymbol{\alpha}. \tag{3}$$

4. Sample the pre-filtered image to produce the discrete output image,

$$g(\mathbf{x}) = \tilde{g}'(\mathbf{x}) \quad \text{for} \quad \mathbf{x} \in \mathbb{Z}^2. \tag{4}$$

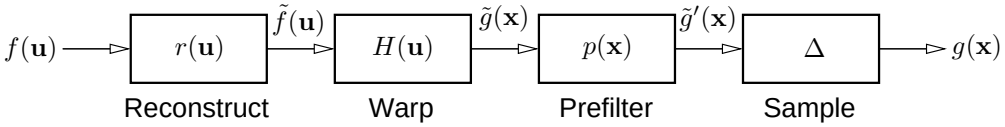

Figure 1. Heckbert's conceptual image resampling pipeline.

2.2. Realizable image resampling pipeline

Although the resampling pipeline discussed above is conceptually useful, it is not practical from an implementation standpoint. While it is instructive to think of the resampling process in terms of reconstructing a continuous version of the texture $\tilde{f}(\mathbf{u})$ in Eqn. (1), warping that texture to produce a continuous function $\tilde{g}(\mathbf{x})$ in Eqn. (2) and pre-filtering this continuous function to produce $\tilde{g}'(\mathbf{x})$ in Eqn. (4), it is neither computationally feasible nor desirable to reconstruct the intermediate continuous functions. Fortunately there is no necessity to do so. Working backward up the resampling pipeline, it is possible to derive expressions for the warped output which do not require reconstruction of the intermediate steps:

$$\begin{aligned} g(\mathbf{x})\Big|_{\mathbf{x}\in\mathbb{Z}^2} &= \tilde{g}'(\mathbf{x}) \\ &= \int \tilde{f}\left(H^{-1}(\boldsymbol{\alpha})\right) \cdot p(\mathbf{x}-\boldsymbol{\alpha})\, \mathrm{d}\boldsymbol{\alpha} \\ &= \int p(\mathbf{x}-\boldsymbol{\alpha}) \sum_{\mathbf{k}\in\mathbb{Z}^2} f(\mathbf{k}) \cdot r\left(H^{-1}(\boldsymbol{\alpha})-\mathbf{k}\right) \mathrm{d}\boldsymbol{\alpha}. \end{aligned} \tag{5}$$

By defining

$$\rho(\mathbf{x},\mathbf{k}) = \int p(\mathbf{x}-\boldsymbol{\alpha}) \cdot r\left(H^{-1}(\boldsymbol{\alpha})-\mathbf{k}\right) \mathrm{d}\boldsymbol{\alpha}, \tag{6}$$

we may write Eqn. (5) in the succinct form,

$$g(\mathbf{x})\Big|_{\mathbf{x}\in\mathbb{Z}^2} = \sum_{\mathbf{k}\in\mathbb{Z}^2} f(\mathbf{k})\rho(\mathbf{x},\mathbf{k}), \tag{7}$$

where $\rho(\mathbf{x},\mathbf{k})$ is a *spatially varying* resampling filter. Thus the warped image $g(\mathbf{x})$ may be computed using only a discrete filtering operation. There is no need to compute the intermediate stages of the pipeline described in Section (2.1).

The resampling filter of Eqn. (6) is described in terms of the warped reconstruction filter r with the integration performed over $\mathbf{x}$-space. With a change of variables $\boldsymbol{\alpha} = H(\mathbf{u})$, and integrating in $\mathbf{u}$-space the resampling filter can be expressed in terms of the warped pre-filter p as,

$$\rho(\mathbf{x},\mathbf{k}) = \int p(\mathbf{x}-H(\mathbf{u})) \cdot r\left(\mathbf{u}-\mathbf{k}\right) \left|\frac{\partial H}{\partial \mathbf{u}}\right| \mathrm{d}\mathbf{u}, \tag{8}$$

where $|\partial H/\partial \mathbf{u}|$ is the determinant of the Jacobian of the transformation H. This form for the resampling filter will be useful in the sections ahead.

3. MULTI-FRAME SUPER-RESOLUTION RESTORATION

Given a sequence of noisy, under-sampled low-resolution images of a scene, the objective of super-resolution restoration is the estimation of super-resolved images corresponding to observed frames. The term super-resolved implies bandwidth extrapolation, not merely increasing the number of image samples (interpolation). Typically the observed image sequence includes relative scene/camera motion which is utilized in the multi-frame super-resolution restoration process in estimating super-resolved frames.

Resolution enhancement is possible as a result of the following factors:

1. Sub-pixel registration of the observed images provides multiple constraints on any given super-resolution restoration pixel. The multiple observation constraints provide far more information than is available in the classic single image restoration framework.

2. An accurate observation model makes it possible to invert the effects of degradations in the image acquisition process. The observation model may take into account diverse effects including lens responses, lens aberrations, defocus, sensor spatial and temporal integration, sensor noise and motion blurring. In this paper we are interested in extending the observation model to easily include space-varying degradations and general motion maps.

3. Inclusion of *a-priori* knowledge improves restoration. Known characteristics of the solution, from very general notions such as positivity or smoothness, to very specific information for example the image being composed of binary valued text scanned from a page, can lead to dramatic improvements in restoration, as well as ensuring that the ill-posed restoration inverse problem has a useful solution.

3.1. Multi-frame observation model

Let the observed low-resolution image sequence consist of N images $g^{(i)}(\mathbf{x}),\ i \in \{1, 2, \ldots, N\}$ derived from an underlying scene $f(\mathbf{u})$. The observed images are related to the scene $f(\mathbf{u})$ via the coordinate transformations $H^{(i)} : \mathbf{u} \longmapsto \mathbf{x}$ which account for the relative motion between the scene and the camera. Associated with each observation is a (possibly spatially varying) PSF $h^{(i)}$, which may differ from observation to observation. The PSFs can model the lens and sensor response, defocus, motion blur and so on. Finally the images are sampled. The multi-frame observation model may be described by,

$$g^{(i)}(\mathbf{x})\Big|_{\mathbf{x}\in\mathbb{Z}^2} = \int h^{(i)}(\mathbf{x}, \boldsymbol{\alpha}) \cdot f\left(H^{(i)^{-1}}(\boldsymbol{\alpha})\right) \mathrm{d}\boldsymbol{\alpha}, \tag{9}$$

and is illustrated in Fig. (2).

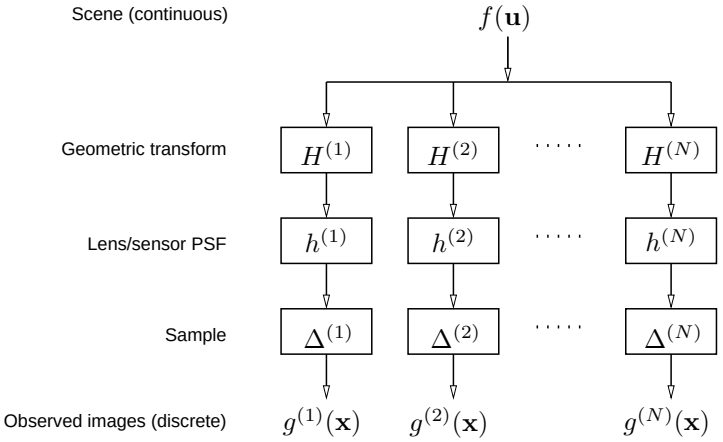

Figure 2. Multi-frame observation model.

3.2. Super-resolution restoration observation model

In the multi-frame super-resolution restoration process we typically wish to obtain super-resolved estimates of the observed images $\boldsymbol{g}^{(i)}$. To do this, we must relate the observations to the image to be restored. Here, however, we will consider the problem of estimating the underlying scene $f(\mathbf{u})$ directly. Estimating super-resolved images coincident with each observed frame may be done in a similar fashion.

Our objective will be to estimate a discretized approximation of the scene $f(\mathbf{u})$ on a high-resolution sampling grid. Typically the ratio of the areal sampling density of a super-resolved restoration to that of the observed

images should be proportional to the number of observed images. This is justified in terms of the total number of constraints provided by the observations compared with the number of unknowns to be estimated. Assume that we increase the number of samples per dimension by a factor of M, which in practice would likely be some integer close to $\sqrt{N}$. We begin by constructing a discretized approximation $f(\mathbf{k})$ of the scene $f(\mathbf{u})$ using an interpolation kernel h_r as,

$$f(\mathbf{u}) \approx \sum_{\mathbf{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\mathbf{k}) \cdot h_r\left(\mathbf{u} - \boldsymbol{V}\mathbf{k}\right) \tag{10}$$

where $\boldsymbol{V}$ is the sampling matrix[2]

$$\boldsymbol{V} = \begin{bmatrix} \frac{1}{M} & 0 \\ 0 & \frac{1}{M} \end{bmatrix}.$$

Combining Eqn. (9) with Eqn. (10) yields the relationship between the observed frames and the discrete approximation of the scene as,

$$\begin{aligned} g^{(i)}(\mathbf{x})\Big|_{\mathbf{x}\in\mathbb{Z}^2} &= \int h^{(i)}(\mathbf{x},\boldsymbol{\alpha}) \cdot f\left(H^{(i)^{-1}}(\boldsymbol{\alpha})\right) \mathrm{d}\boldsymbol{\alpha} \\ &= \int h^{(i)}(\mathbf{x},\boldsymbol{\alpha}) \sum_{\mathbf{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\mathbf{k}) \cdot h_r\left(H^{(i)^{-1}}(\boldsymbol{\alpha}) - \boldsymbol{V}\mathbf{k}\right) \mathrm{d}\boldsymbol{\alpha}. \end{aligned} \tag{11}$$

Comparing Eqn. (11) with the image resampling filter in Eqn. (5) in Section (2.2), which is reproduced below for convenience,

$$g(\mathbf{x})\Big|_{\mathbf{x}\in\mathbb{Z}^2} = \int p(\mathbf{x}-\boldsymbol{\alpha}) \sum_{\mathbf{k}\in\mathbb{Z}^2} f(\mathbf{k}) \cdot r\left(H^{-1}(\boldsymbol{\alpha}) - \mathbf{k}\right) \mathrm{d}\boldsymbol{\alpha},$$

it becomes immediately apparent that there is a parallel between the image resampling equation and the multi-frame observation model. In particular, the PSF $h^{(i)}(\mathbf{x},\boldsymbol{\alpha})$ associated with the observation $g^{(i)}(\mathbf{x})$ replaces the image resampling pre-filter $p(\mathbf{x}-\boldsymbol{\alpha})$ in the resampling equation, and similarly the reconstruction filter $h_r(H^{(i)^{-1}}(\boldsymbol{\alpha}) - \boldsymbol{V}\mathbf{k})$ has a counterpart in the reconstruction filter $r(H^{-1}(\boldsymbol{\alpha}) - \mathbf{k})$ in the resampling equation. Noting these parallels, we are encouraged to proceed as we did in Section (2.2) and define the spatially varying filter,

$$\rho^{(i)}(\mathbf{x},\mathbf{k}) = \int h^{(i)}(\mathbf{x},\boldsymbol{\alpha}) \cdot h_r\left(H^{(i)^{-1}}(\boldsymbol{\alpha}) - \boldsymbol{V}\mathbf{k}\right) \mathrm{d}\boldsymbol{\alpha}, \tag{12}$$

allowing us to write Eqn. (11) as,

$$g^{(i)}(\mathbf{x})\Big|_{\mathbf{x}\in\mathbb{Z}^2} = \sum_{\mathbf{k}\in\mathbb{Z}^2} f(\boldsymbol{V}\mathbf{k}) \cdot \rho^{(i)}(\mathbf{x},\mathbf{k}). \tag{13}$$

As in the resampling derivation, we make the change of variables $\boldsymbol{\alpha} = H^{(i)}(\mathbf{u})$, and integrate in $\mathbf{u}$-space so the observation filter $\rho^{(i)}(\mathbf{x},\mathbf{k})$ can be expressed in terms of the warped PSF $h^{(i)}(\mathbf{x},\boldsymbol{\alpha})$ as,

$$\rho^{(i)}(\mathbf{x},\mathbf{k}) = \int h^{(i)}\left(\mathbf{x}, H^{(i)}(\mathbf{u})\right) \cdot h_r\left(\mathbf{u} - \boldsymbol{V}\mathbf{k}\right) \left|\frac{\partial H^{(i)}}{\partial \mathbf{u}}\right| \mathrm{d}\mathbf{u}. \tag{14}$$

In Eqn. (13), a sample in one of the observed images is described in terms of a filtering operation on the sampled approximation of the scene we hope to estimate. In the super-resolution restoration frameworks[3,4] this observation equation constitutes a single row of the matrix $\boldsymbol{A}$ which relates the observations to the unknowns to be estimated as,

$$\left[\boldsymbol{g}^{(1)^T}\, \boldsymbol{g}^{(2)^T}\, \cdots\, \boldsymbol{g}^{(N)^T}\right]^T = \boldsymbol{A}\boldsymbol{f}. \tag{15}$$

Here we have assumed that the images are lexicographically ordered column vectors. These equations are solved to determine the super-resolution estimate for $\boldsymbol{f}$. Typically regularized solution methods are used since the inverse problem is usually ill-posed. At this point we also note that no changes to the restoration algorithm are required.

4. COMPARING THE RESAMPLING AND RESTORATION MODELS

The parallels between the image resampling and the super-resolution restoration observation models which were discussed in Section (3.2) are tabulated below:

Resampling		Restoration	
Discrete texture	$f(\mathbf{u})$	Discrete scene estimate	$f(\mathbf{u})$
Reconstruction filter	$r(\mathbf{u})$	Interpolation kernel	$h_r(\mathbf{u})$
Geometric transform	$H(\mathbf{u})$	Scene/camera motion	$H^{(i)}(\mathbf{u})$
Anti-alias pre-filter	$p(\mathbf{x})$	Observation SVPSF	$h^{(i)}(\mathbf{x}, \boldsymbol{\alpha})$
Warped output image	$g(\mathbf{x})$	Observed images	$g^{(i)}(\mathbf{x})$
Resampling filter		Observation filter	
$\rho(\mathbf{x}, \mathbf{k}) = \int p(\mathbf{x} - H(\mathbf{u})) \cdot r\left(\mathbf{u} - \mathbf{k}\right) \left\lvert \frac{\partial H}{\partial \mathbf{u}} \right\rvert \mathrm{d}\mathbf{u}$		$\rho^{(i)}(\mathbf{x}, \mathbf{k}) = \int h^{(i)}\left(\mathbf{x}, H^{(i)}(\mathbf{u})\right) \cdot h_r\left(\mathbf{u} - \boldsymbol{V}\mathbf{k}\right) \left\lvert \frac{\partial H^{(i)}}{\partial \mathbf{u}} \right\rvert \mathrm{d}\mathbf{u}$	

5. DETERMINING THE OBSERVATION FILTER

In the sections that follow we consider how to determine the observation filter $\rho(\mathbf{x}, \mathbf{k})$. We have seen that the observation process model is comprised of three main components:

1. The observation PSFs: We will show how a model for the observation PSF may be obtained by considering an imaging system consisting of a diffraction-limited optical system and focal plane sensor array.
2. The coordinate warps: In practice the coordinate warps result from relative scene/camera motion. In our examples we will consider a case where a moving camera views a static planar scene. This viewing geometry allows very simple computation of the mapping between points in each camera view.
3. The interpolation kernel: We use a constant-valued box function for the interpolation kernel $h_r(\mathbf{u})$.

We then present a simple algorithm for computing the coefficients $\rho(\mathbf{x}, \mathbf{k})$ given the above information.

6. OPTICAL SYSTEM MODEL

In this section we present a common model of a diffraction-limited optical system, with the lens system having a circular exit pupil. The lens system f-number, that is, the ratio of focal length to aperture diameter, is given by N. It is assumed that the system operates under incoherent, monochromatic illumination with wavelength λ. No lens aberrations are accounted for. Under these assumptions the point spread function associated with the optical system is radially symmetric and is given by,

$$h_o(r') = \left[2\frac{J_1(r')}{r'}\right]^2, \quad \text{with} \quad r' = (\pi/\lambda N)r \tag{16}$$

where $J_1(\cdot)$ is the Bessel function of the first kind and r' is the normalized radial distance computed from the true radial distance r.

The optical system may be equivalently characterized by the optical transfer function (OTF), which is the normalized frequency response of the optical system. The OTF of the diffraction-limited optical system described above is given by,[5]

$$H_o(\rho') = \begin{cases} \frac{2}{\pi}\left[\cos^{-1}(\rho') - \rho'\sqrt{1-\rho'^2}\right], & \text{for } \rho' \leq 1 \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

where $\rho' = \rho/\rho_c$ is a normalized radial spatial frequency. The normalized radial frequency ρ' is defined in terms of ρ, the true radial spatial frequency and $\rho_c = 1/\lambda N$ the radial spatial frequency cut-off which is a function of the wavelength of the illumination and the lens f-number.

7. FOCAL PLANE SENSOR ARRAY MODEL

The image projected onto the focal plane by the optical system is measured by a focal plane array (FPA) consisting of a regular lattice of independent light-sensing elements called *pixels*. Each pixel integrates the incident illumination over the sensor's *active region* for the duration of the *aperture time*. In charge-coupled devices for example, incident photons striking the active area of a pixel result in the generation of electronic charge which is accumulated in a charge "well" until it is read out.

We model the sensor array as a regular tessellation of square pixels with dimension T. Each pixel integrates the incident illumination over its entire $T \times T$ region of support and the response to photons striking at any point across this region is assumed to be uniform. These assumptions make possible a simple but useful model of an ideal focal plane array pixel. The point spread function of a focal plane array pixel may be modeled as,

$$h_p(x,y) = \frac{1}{T^2}\,\mathrm{rect}\left(\frac{x}{T}\right)\mathrm{rect}\left(\frac{y}{T}\right) = \begin{cases} 1/T^2, & |x| \le T/2 \;\text{ and }\; |y| \le T/2 \\ 0, & \text{otherwise} \end{cases} . \tag{18}$$

The two-dimensional Fourier transform of Eqn. (18) is given by $H_p(\xi_1,\xi_2) = \mathrm{sinc}(T\xi_1)\cdot\mathrm{sinc}(T\xi_2)$. Since the response $H_p(0,0) = 1$, this expression is also the optical transfer function associated with the pixel.

In reality, sensors deviate significantly from the assumptions we have made in this section. Firstly the "fill-factor", that is the ratio of light sensitive area of the pixel as compared with actual size of the pixel T^2 is often not unity. This is because some portion of the pixel area is often required to contain read-out electronics. Additionally, the sensitivity of the pixel to incident illumination is often not uniform across the entire pixel area. This is especially true for devices which utilize micro-lenses in an attempt to artificially increase the pixel fill-factor. Finally, due to variations across the device substrate, the sensitivity of pixels across the device is not uniform. Nevertheless the model we have developed is useful as a first estimate of the sensor response. Later it will become clear that incorporating measured sensor data into the multi-frame observation model does not pose any additional difficulties.

8. COMBINED LENS AND SENSOR MODEL

By convolving the point spread functions associated with the lens and sensor we obtain the PSF for the combined system, $h_c = h_o \circledast h_p$. Plots of a combined lens/pixel PSF are shown in Fig. (3). Notice in the combined lens/pixel response profile in Fig. (3) that there is considerable "leakage" between adjacent pixels. The response for real-world imaging systems will typically be considerably worse than these plots given that we have assumed ideal responses for both the lens and sensor PSFs.

The combined lens and sensor MTF, H_c, is found by multiplying the optical system MTF and the sensor MTF expressions as $H_c = H_o \cdot H_p$.

9. IMAGING GEOMETRY

Next we consider the form of the coordinate transformations $H^{(i)}$ which relate the observed image coordinate systems to the scene. In general these mappings are computed by estimating the motion occurring between observed frames. The warps need not be uniform but they must be locally invertible.

In the example presented later, we consider a 2D "scene" located in the X-Y plane in 3-space. Two camera views of the plane are generated using a custom ray-tracer which models the lens and sensor effects discussed earlier. Since the calibration matrices of the cameras are known, we may easily compute the homography induced by the scene plane. This mapping relates the coordinates of points in each image and is a projectivity. $H^{(i)} : \mathbf{u} \longmapsto \mathbf{x}$ is thus of the form,

$$\begin{array}{rcccl} x & = & \phi(u,v) & = & \frac{h_{11}u + h_{12}v + h_{13}}{h_{31}u + h_{32}v + 1} \\ y & = & \psi(u,v) & = & \frac{h_{21}u + h_{22}v + h_{23}}{h_{31}u + h_{32}v + 1} \end{array} \tag{19}$$

and the determinant of the Jacobian $\left|\frac{\partial H^{(i)}}{\partial \mathbf{u}}\right|$ is given by $\begin{vmatrix} \frac{\partial \phi}{\partial u} & \frac{\partial \phi}{\partial v} \\ \frac{\partial \psi}{\partial u} & \frac{\partial \psi}{\partial v} \end{vmatrix}$.

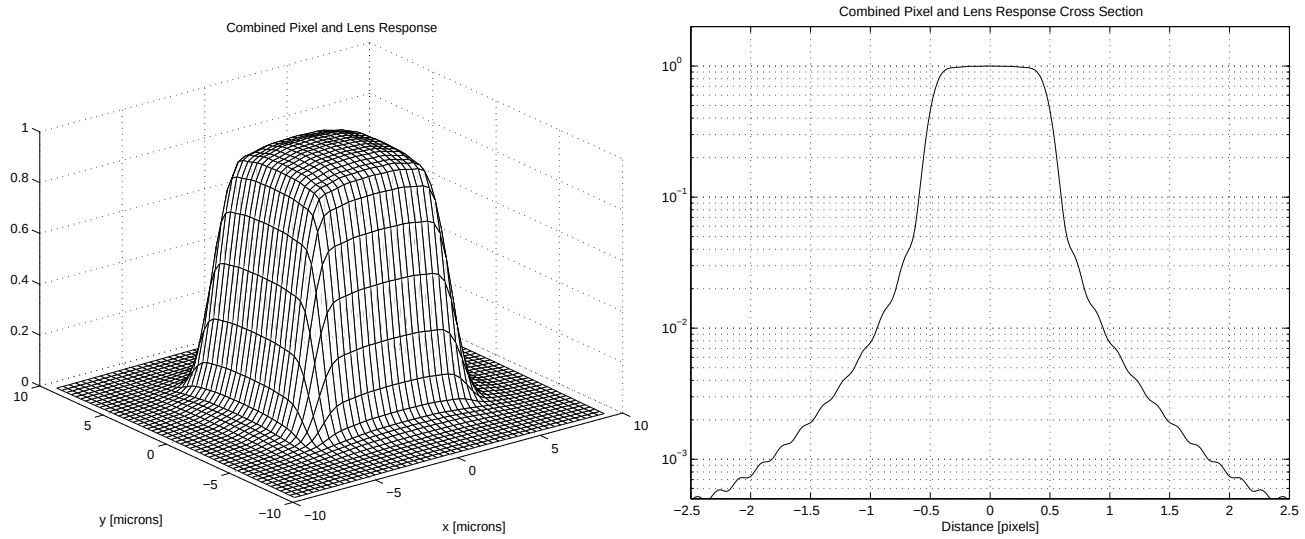

Figure 3. Combined lens/pixel point spread function with focal plane array pixel dimensions $9\mu\text{m} \times 9\mu\text{m}$, illumination wavelength $\lambda = 550\text{nm}$ (green), lens f-number $N = 2.8$.

10. DETERMINING THE WARPED PIXEL RESPONSE

In this section we present an algorithm for computing the coefficients of the spatially varying observation filter of Eqn. (14). In practice, to obtain a reasonably efficient implementation we make the simplifying assumptions that the size of the restoration pixels is sufficiently small that the integral may be approximated by a weighted point sample. Also, we assume that the interpolation kernel is constant over the support of each restoration pixel (box function) so we may omit it evaluation and instead ensure that the sum of all the filter coefficients is unity. Under these assumptions, the following algorithm may be used:

for each observed image $\boldsymbol{g}^{(i)}$ {
  for each pixel $\mathbf{x}$ {
    back-project the boundary of $h^{(i)}(\mathbf{x}, \boldsymbol{\alpha})$ from $g^{(i)}(\mathbf{x})$ to the restored image space using ${H^{(i)}}^{-1}$
    determine a bounding region for the image of $h(\mathbf{x}, \boldsymbol{\alpha})$ under ${H^{(i)}}^{-1}$
    for each pixel indexed by $\mathbf{k}$ in the bounding region {
      set $\rho^{(i)}(\mathbf{x}, \mathbf{k}) = h^{(i)}\left(\mathbf{x}, H^{(i)}(\mathbf{u})\right) \cdot \left|\frac{\partial H^{(i)}}{\partial \mathbf{u}}\right|$ with $\mathbf{u} = \boldsymbol{V}\mathbf{k}$
    }
    normalize $\rho^{(i)}(\mathbf{x}, \mathbf{k})$ so that $\sum_k \rho^{(i)}(\mathbf{x}, \mathbf{k}) = 1$
  }
}

Note that the function of the backprojection step in the algorithm is to reduce the range of $\mathbf{k}$ over which we need to evaluate the warped filter. The backprojection and projection process is illustrated in Fig. (4). In this figure, the support of the PSF associated with an observed pixel is darkly-shaded in the left image. It's image under H^{-1} is shown as the darkly-shaded region in the restored image on the right. The smallest integer-pixel bounding box for this region is shown as the large, lightly shaded region in the right image. Finally the image under H of a high-resolution pixel in the bounding area is is shown as a warped region within the observed pixel PSF, with the ratio of the areas of these regions determined by the determinant of the Jacobian.

In practice it is convenient to build a data structure to store, for each observed pixel, a list of indices $\mathbf{k}$ and the corresponding values of $\rho^{(i)}(\mathbf{x}, \mathbf{k})$ for which $\rho^{(i)}(\mathbf{x}, \mathbf{k})$ is non-zero or above some threshold.

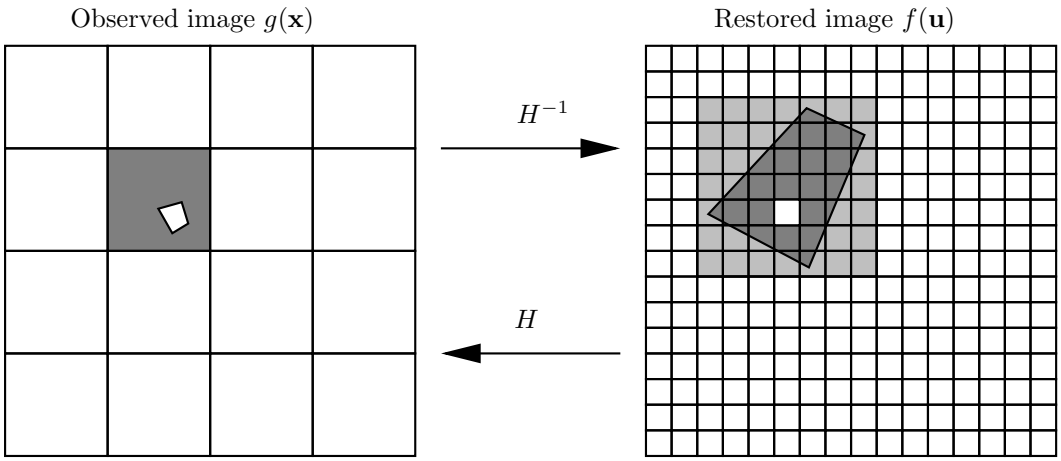

Figure 4. Determining the warped pixel response function.

11. EXAMPLES

In order to demonstrate the application of the observation filter, we have used a simulated imaging environment with known scene and camera geometry. Using a custom ray-tracer, we simulate two camera views $\boldsymbol{g}^{(1)}$ and $\boldsymbol{g}^{(2)}$ of a two dimensional scene placed in the X-Y plane. The ray-tracer simulates the lens and sensor characteristics described earlier. Since the camera calibration matrices are known, with the scene being a a simple plane, we determine the mapping H relating coordinates in $\boldsymbol{g}^{(1)}$ and $\boldsymbol{g}^{(2)}$. This mapping (a projective transformation) is simply the homography induced by the X-Y plane.[6] The imaging geometry is shown in Fig. (5) with the two views $\boldsymbol{g}^{(1)}$ and $\boldsymbol{g}^{(2)}$ shown in Fig. (6) and Fig. (7) respectively.

Assume that we wish to restore an image coincident with the view $\boldsymbol{g}^{(2)}$. The restoration sampling density is, for purposes of illustration, chosen to be four times greater in each dimension than that of the observed images. In the example we find the observation filter which relates the observed image $\boldsymbol{g}^{(1)}$ to the unknown super-resolution image coincident with $\boldsymbol{g}^{(2)}$. In Fig. (8) we show the observation filter which relates the value of the observation $g^{(1)}(0,0)$ to the unknown super-resolution image. On the left of the figure is the PSF associated with the observed pixel $g^{(1)}(0,0)$. On the right is an image of the observation filter coefficients in $\mathbf{u}$ space. Notice the warping of the point spread function which results from the projective transformation that relates the views. Also note that the regular samples on the super-resolution restoration grid map to irregular locations in the observed image PSF. The value of the observed pixel may be computed using Eqn. (13) given an estimate $\hat{\boldsymbol{f}}$ of the super-resolved image f coincident with $\boldsymbol{g}^{(2)}$.

Next we show the application of the observation filter as is typical in a super-resolution restoration algorithm. Recall from the discussion leading to Eqn. (15) that the observation filter coefficients are the non-zero elements in each row of the observation matrix $\boldsymbol{A}$ which relates the observations to the unknowns. Given an estimate $\hat{\boldsymbol{f}}$ of the scene $\boldsymbol{f}$ coincident with $\boldsymbol{g}^{(2)}$, if we compute the projection $\boldsymbol{A}\hat{\boldsymbol{f}}$, we expect that the result should resemble $\boldsymbol{g}^{(1)}$. This is indeed the case, as shown in Fig. (9).

In a typical super-resolution restoration algorithm, a prediction for each observed image $\boldsymbol{g}^{(i)}$ is made in this fashion and the difference between these predictions and the actual observation is used to correct the super-resolution estimate $\hat{\boldsymbol{f}}$. Typically this done via terms which include the application of the transpose of the $\boldsymbol{A}$-matrix. Applying $\boldsymbol{A}^T$ effects a backprojection from the observed image space into the restoration space. To illustrate this, we back-project the observation $\boldsymbol{g}^{(1)}$ using the operation $\boldsymbol{A}^T\boldsymbol{g}^{(1)}$. This is illustrated in Fig. (10). Not surprisingly, the result resembles the super-resolution estimate $\hat{\boldsymbol{f}}$.

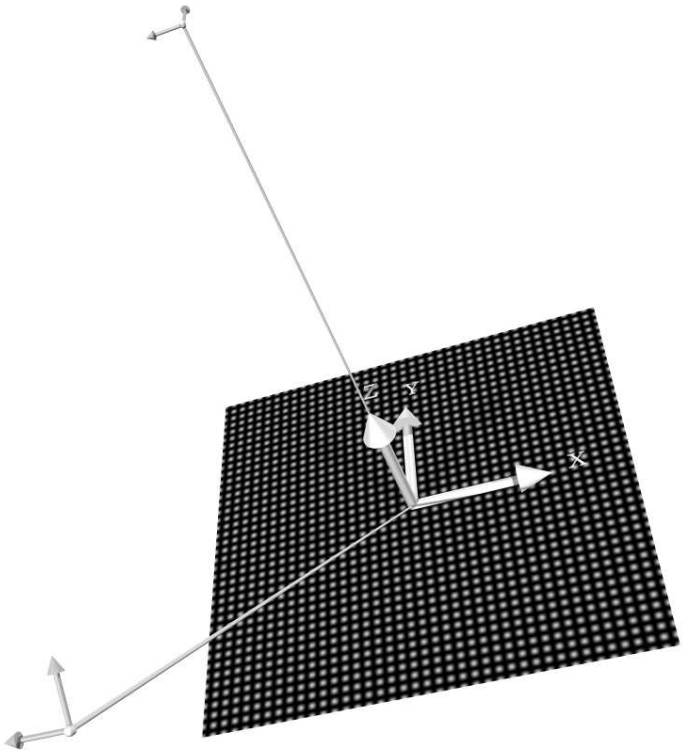

Figure 5. Imaging geometry.

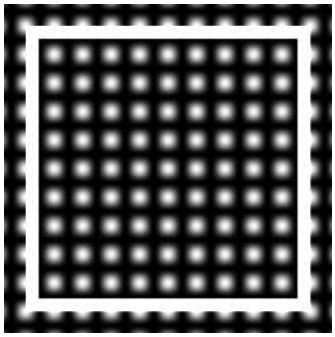

Figure 6. Simulated image $\boldsymbol{g}^{(1)}$.

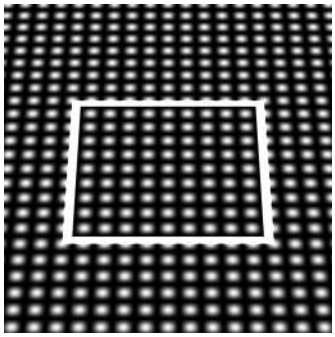

Figure 7. Simulated image $\boldsymbol{g}^{(2)}$.

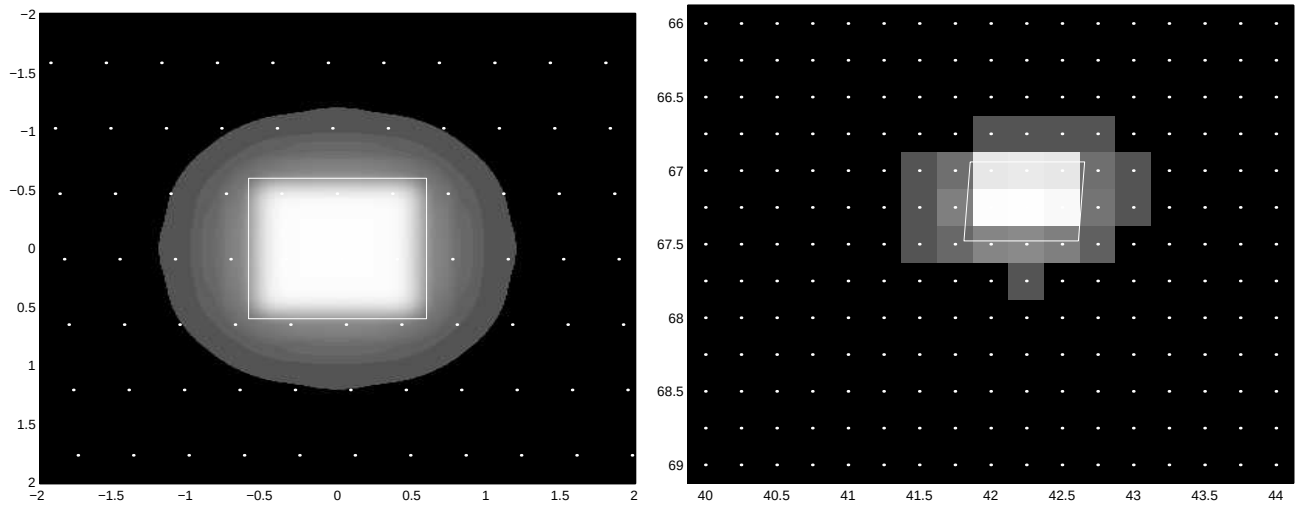

Figure 8. Spatial response of pixel located at $(x, y) = (0, 0)$ (left) and sampled warped response in the restored image u, v-space (right).

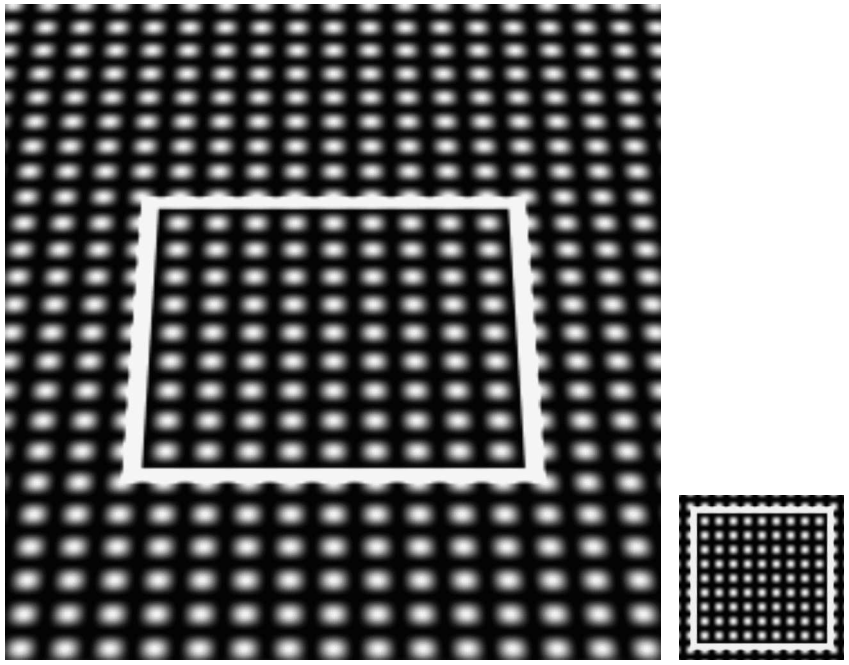

Figure 9. Predicting the observation coincident with $\boldsymbol{g}^{(1)}$ given a super-resolution estimate $\hat{\boldsymbol{f}}$ of the scene $\boldsymbol{f}$ coincident with $\boldsymbol{g}^{(2)}$.

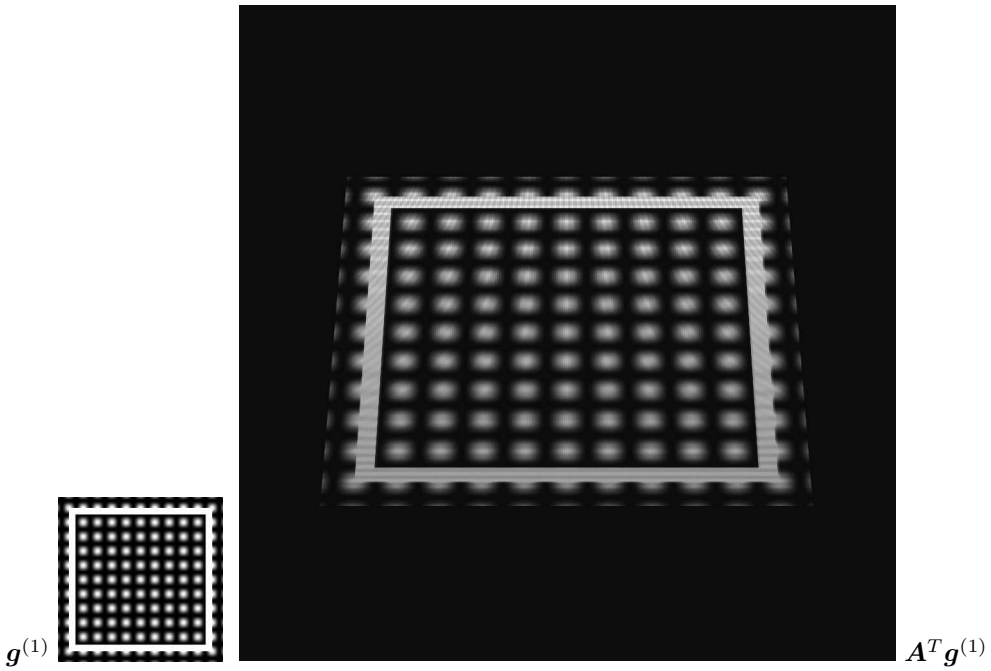

Figure 10. Back-projecting the observation $\boldsymbol{g}^{(1)}$ using $\boldsymbol{A}^T$.

12. EXTENSIONS AND CHALLENGES

Accelerating the computation of $\rho^{(i)}(\mathbf{x}, \mathbf{k})$ is a worthwhile goal since computing the projected PSFs is costly. One immediate savings would be the use of a tighter bounding box on the backprojection of the observed pixel PSF. An approach to achieve this might utilize scan-line algorithms from the computer graphics literature.[7]

Another area of consideration regards the accuracy of the observation model versus the space complexity of storing $\rho^{(i)}(\mathbf{x}, \mathbf{k})$ and time complexity of applying the filter. The space complexity of storing the observation filter coefficients is high, given that for each observed pixel one must store a list of non-zero coefficient weights as well as indices of the super-resolution pixels to which the weights apply. While a larger number of filter coefficients allows for greater fidelity in representing the warped PSF this also results in increased storage as well as computation in applying the filters.

In this presentation we have considered a trivial "motion" map. The techniques discussed should generalize easily to locally invertible parametric motion fields, but interpolation would be required to address non-parametric, sampled motion vector fields. Occlusions are also likely to present some difficulties.

13. CONCLUSIONS

In this paper we introduced a method for generalizing the linear observation model commonly used in multi-frame super-resolution restoration frameworks to account for spatially varying observation PSFs as well as general coordinate transformations induced by scene/camera motion. A close relationship between image resampling theory and the multi-frame image restoration problem was shown. This informed the derivation of a spatially varying observation filter for the multi-frame super-resolution observation model. The coefficients of the observation filter were shown to constitute rows of the observation matrix in the common linear inverse problem formulation of the super-resolution restoration problem. Since only the coefficients of this matrix are changed in the new scheme, there is no necessity to alter the super-resolution restoration algorithm. The observation coefficients need only be computed once after which the restoration algorithm proceeds as usual.

REFERENCES

1. P. Heckbert, "Fundamentals of texture mapping and image warping," Master's thesis, U.C. Berkeley, June 1989.
2. D. Dudgeon and R. Mersereau, *Multidimensional Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1984.
3. R. Schultz and R. Stevenson, "Extraction of high-resolution frames from video sequences," *IEEE Transactions on Image Processing* **5**, pp. 996–1011, June 1996.
4. S. Borman and R. Stevenson, "Simultaneous multi-frame MAP super-resolution video enhancement using spatio-temporal priors," in *Proceedings of the IEEE International Conference on Image Processing*, **3**, pp. 469–473, (Kobe, Japan), Oct. 1999.
5. J. Goodman, *Introduction to Fourier Optics*, McGraw-Hill, 2nd, 1996.
6. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.
7. J. Foley, A. van Dam, S. Feiner, and J. Hughes, *Computer Graphics – Principles and Practice*, Addison-Wesley Publishing Company, Second Edition in C, 1996.