IMAGE ACQUISITION MODELING FOR SUPER-RESOLUTION RECONSTRUCTION

Murat Gevrekci and *Bahadir K. Gunturk*

Electrical and Computer Engineering
Louisiana State University
Baton Rouge, LA 70803
Email: lgevre1@lsu.com, bahadir@ece.lsu.edu

ABSTRACT

Super-resolution reconstruction is the process of reconstructing a high-resolution image from multiple low-resolution images. Most super-resolution reconstruction methods neglect camera response function, exposure time, white balancing, and external illumination changes. In this paper, we show how to extend traditional super-resolution reconstruction methods to handle these factors. We provide formulations for several different super-resolution reconstruction approaches. Experimental results are also included.

1. INTRODUCTION

Development of visual communications and storage capabilities increase the demand to high-resolution images. High-resolution images are required in digital cameras, surveillance systems, medical imaging, aerial/satellite imaging, and high-definition TV systems. In computer vision systems, degraded images may cause having false decisions. High-resolution imaging is crucial in recognition, detection, and classification applications.

Although pixels can be placed more densely in camera in order to obtain higher resolution, producing denser sensor arrays increase cost significantly. In addition, smaller pixel size results in smaller signal to noise ratio because of shot noise, which is roughly independent of the pixel size.

Alternatively, we can take advantage of the correlation among degraded observations of a scene to construct a higher resolution image. In order to achieve super-resolution, subpixel-accurate motion vectors that warp input images onto a target image must be calculated. Image registration is followed by restoration step, which models point spread function and subsampling. Such a multiframe resolution enhancement process is referred to as super-resolution reconstruction in the literature [1].

Current super-resolution algorithms do not model some of the common process of the image acquisition pipeline, such as exposure time and white balancing. In a real life scenario, images in a data set may posses different information of the dynamic range of a scene due to the change in exposure rates and aperture sizes. In every different lightning condition, camera parameters are automatically adjusted to capture the right portion of the scene's dynamic range by automatic parameter adjustment units. Also more detailed information can be obtained about a scene if images with different camera properties can be combined properly.

The so-called high-dynamic range imaging has been an active research area in the computer vision community. Debevec *et al.* [4], Mann [7], Robertson *et al.* [9], and Candocia [2] have demonstrated how to improve dynamic range by combining images captured with different exposure times. However, the issue of nonlinear sensor response, different exposures, and white balancing has not been addressed extensively in the super-resolution research. Recently, we proposed a projections onto convex sets based super-resolution algorithm that addresses the saturation in pixel measurements and potential changes in illumination [6]. In this paper, we show how to extend traditional super-resolution reconstruction methods with a better imaging model (Section 2). We provide formulations for several different approaches (Section 3). Preliminary experimental results are given in Section 4.

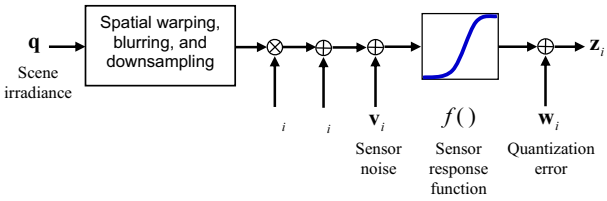

Fig. 1. The proposed super-resolution algorithm uses an imaging model that includes dynamic range and spatial domain effects.

2. IMAGING MODEL

Incorporating the relative motion among observed images, super-resolution algorithms model the imaging process as a linear mapping between a high-resolution input signal $\mathbf{q}$ and low-resolution observations $\mathbf{z}_i$. ($i = 1, ..., N$; N is the total number of observations.) The imaging process is formulated as

$$\mathbf{z}_i = \mathbf{H}_i\mathbf{q}, \quad i = 1, ..., N, \tag{1}$$

where $\mathbf{H}_i$ is the linear mapping that includes motion (of the camera or the objects in the scene), blur (caused by the point spread function of the sensor elements and the optical system), and downsampling. Therefore, super-resolution reconstruction is an inverse problem where $\mathbf{q}$ is estimated from a set of observations $\mathbf{z}_i$. $\mathbf{H}_i$ can be space- and time-varying. In practice, $\mathbf{H}_i$ is implemented in three steps: spatial warping to compensate for motion, convolution with a point spread function (PSF), and downsampling. Details of $\mathbf{H}_i$ modeling can be found in the special issue of the IEEE Signal Processing Magazine [1] and the references therein.

As mentioned earlier, the model in equation (1) is not a complete model. The irradiance of the scene is multiplied by the exposure time and then passed through a non-linear camera response

0-7803-9134-9/05/$20.00 ©2005 IEEE

function. In color images, white balance is achieved by transforming the color space and scaling each component. Exposure time and white balancing can be modeled as gain and offset terms in general. Another factor in imaging is the vignetting effect. The luminance on the sensor array changes by $\cos^4\theta$ with respect to the luminance on the optical center. (Let d_f be the focal length and d_p be the distance of a point to the optical center, then $\tan\theta = d_p/d_f$.) The vignetting effect could be included in $\mathbf{H}_i$ as a space-varying intensity scale factor.

Denoting $\mathbf{v}_i$ as the additive noise term (due to shot noise and thermal noise) and $\mathbf{w}_i$ as the quantization error, the overall imaging process can be formulated as

$$\mathbf{z}_i = f\left(\alpha_i \mathbf{H}_i \mathbf{q} + \beta_i + \mathbf{v}_i\right) + \mathbf{w}_i, \quad i = 1, ..., N, \tag{2}$$

where $f(\cdot)$ is the nonlinear camera response function function, α_i is the gain factor, β_i is the offset factor, and $\mathbf{H}_i$ is the linear mapping that incorporates motion, PSF, vignetting, and downsampling. (See the block diagram in Figure 1.)

3. RECONSTRUCTION METHODS

In this section, we provide several approaches for solving $\mathbf{q}$. Defining $g(\cdot) \equiv f^{-1}(\cdot)$ and using a Taylor series expansion, equation (2) can be written as

$$g\left(\mathbf{z}_i\right) \simeq \alpha_i \mathbf{H}_i \mathbf{q} + \beta_i + \mathbf{v}_i + g'\left(\mathbf{z}_i\right) \mathbf{w}_i. \tag{3}$$

With this linearization, we can easily apply the standard linear estimation techniques.

3.1. Least Squares Estimation

For the least squares (LS) estimation, the following cost function is minimized:

$$C = \frac{1}{2} \sum_i \left\| \frac{g\left(\mathbf{z}_i\right) - \beta_i}{\alpha_i} - \mathbf{H}_i \mathbf{q} \right\|^2. \tag{4}$$

The gradient of the cost function is

$$\nabla C = -\sum_i \mathbf{H}_i^T \left(\left(\frac{g\left(\mathbf{z}_i\right) - \beta_i}{\alpha_i} \right) - \mathbf{H}_i \mathbf{q} \right), \tag{5}$$

from which the direct solution is found to be

$$\hat{\mathbf{q}} = \left(\sum_i {\mathbf{H}_i}^T \mathbf{H}_i \right)^{-1} \sum_i {\mathbf{H}_i}^T \left(\frac{g\left(\mathbf{z}_i\right) - \beta_i}{\alpha_i} \right). \tag{6}$$

This formulation requires taking the inverse of a large matrix. Although it is possible to construct the matrix $\mathbf{H}_i$, a more convenient way is to think of $\mathbf{H}_i$ as a linear operator that applies warping, blurring, and downsampling operations successively [11]. Similarly, ${\mathbf{H}_i}^T$ can be implemented by upsampling the image (with zero padding), convolving with the flipped PSF, and motion warping back to the reference frame. The Landweber iteration technique is suitable for such an implementation. At each iteration step, the current estimate $\hat{\mathbf{q}}$ is updated as follows:

$$\hat{\mathbf{q}}' = \hat{\mathbf{q}} + k \mathbf{H}_i^T \left(\frac{g\left(\mathbf{z}_i\right) - \beta_i}{\alpha_i} - \mathbf{H}_i \hat{\mathbf{q}} \right), \tag{7}$$

where k is a step size, which can be constant or updated adaptively at each iteration.

3.2. Projections Onto Convex Sets - Vector Formulation

Projections onto convex sets (POCS) technique is a well-known reconstruction technique. The POCS technique projects an initial estimate onto constraint sets iteratively to obtain a solution that is consistent with all the constraints [3]. There are different ways of defining the constraint sets. One way is based on considering the whole image as a vector. Defining the residual $\mathbf{r}_i = \frac{g(\mathbf{z}_i) - \beta_i}{\alpha_i} - \mathbf{H}_i \hat{\mathbf{q}}$, each observation sets a constraint on the reconstructed image $\hat{\mathbf{q}}$ such that the residual is in between some lower and upper bounds. That is the constraint set is

$$S_i = \left\{ \hat{\mathbf{q}} : \mathbf{b}_l \le \mathbf{r}_i \le \mathbf{b}_u \right\}, \tag{8}$$

where $\mathbf{b}_l$ and $\mathbf{b}_u$ are the lower and upper bound vectors. (Note that $\mathbf{b}_l$ and $\mathbf{b}_u$ are functions of measured intensity $\mathbf{z}_i$. For saturated regions, the bounds are not as tight as for the non-saturated regions.) The corresponding projection operation can be implemented as follows

$$\hat{\mathbf{q}}' = \hat{\mathbf{q}} + \mathbf{H}_i^T \left(\mathbf{H}_i \mathbf{H}_i^T \right)^{-1} \mathbf{d}_i, \tag{9}$$

where

$$\mathbf{d}_i = \begin{cases} \mathbf{b}_u - \mathbf{r}_i, \text{ for } \mathbf{r}_i > \mathbf{b}_u \\ \mathbf{0}, \text{ for } \mathbf{b}_l \le \mathbf{r}_i \le \mathbf{b}_u \\ \mathbf{b}_l - \mathbf{r}_i, \text{ for } \mathbf{r}_i < \mathbf{b}_l \end{cases} \tag{10}$$

To avoid calculating $\left(\mathbf{H}_i \mathbf{H}_i^T \right)^{-1}$, one may try to restore the image iteratively as follows

$$\hat{\mathbf{q}}' = \hat{\mathbf{q}} + k \mathbf{H}_i^T \mathbf{d}_i, \tag{11}$$

where k is a step size. This is not exactly a projection operation; however, experimental results show that it works well.

3.3. Projections Onto Convex Sets - Pixel Formulation

When the constraint set is defined for the whole image, the projection operation requires taking the inverse of a large matrix. When the constraints are defined pixel-by-pixel, it is easier to implement the projection operation. Let (l_1, l_2) and (n_1, n_2) be the pixel coordinates in low-resolution and high-resolution images, respectively, and $h_i(l_1, l_2; n_1, n_2)$ be the linear mapping. The residual at each pixel is now

$$\mathbf{r}_i(l_1, l_2) = \frac{g\left(\mathbf{z}_i(l_1, l_2)\right) - \beta_i}{\alpha_i} - \sum_{n_1, n_2} h_i(l_1, l_2; n_1, n_2) \hat{\mathbf{q}}(n_1, n_2). \tag{12}$$

Defining

$$\gamma = \frac{h_i(l_1, l_2; n_1, n_2)}{\sum\limits_{n_1, n_2} \left| h_i(l_1, l_2; n_1, n_2) \right|^2}, \tag{13}$$

the projection operation can be formulated as

$$\hat{\mathbf{q}}'(n_1, n_2) = \hat{\mathbf{q}}(n_1, n_2) + \begin{cases} \gamma\left(\mathbf{b}_u(l_1, l_2) - \mathbf{r}_i(l_1, l_2)\right), \text{ for } \mathbf{r}_i(l_1, l_2) > \mathbf{b}_u(l_1, l_2) \\ 0, \text{ for } \mathbf{b}_l(l_1, l_2) \le r_i(l_1, l_2) \le \mathbf{b}_u(l_1, l_2) \\ \gamma\left(\mathbf{b}_l(l_1, l_2) - \mathbf{r}_i(l_1, l_2)\right), \text{ for } \mathbf{r}_i(l_1, l_2) < \mathbf{b}_l(l_1, l_2) \end{cases} \tag{14}$$

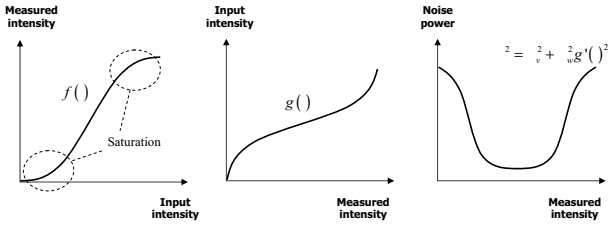

Fig. 2. For a typical sensor response function shown on the left, the total noise variance has a shape similar to the one on the right. For saturated regions, the noise variance is larger.

3.4. Bayesian Estimation

We model $\mathbf{v}_i$ and $\mathbf{w}_i$ as zero mean independent identically distributed (IID) Gaussian noises with variances σ_v^2 and σ_w^2, respectively. This will result in an analytically trackable derivation. It can be shown that the total noise, $\mathbf{v}_i + g'(\mathbf{z}_i)\,\mathbf{w}_i$, is also a zero mean Gaussian noise with variance

$$\sigma^2 = \sigma_v^2 + g'(\mathbf{z}_i)^2\,\sigma_w^2. \tag{15}$$

A critical implication of this result is that the total noise variance σ^2 is a function of the camera response function and measured pixel intensities $\mathbf{z}_i$. Equation (15) indicates that the total noise variance is larger for saturated pixel values. (See Figure 2.)

Denoting $\mathbf{K}$ as the covariance matrix of the total noise, and using a Gaussian prior for $\mathbf{q}$ with mean image $\mu_{\mathbf{q}}$ and covariance matrix $\mathbf{\Lambda}$, the maximum a posteriori (MAP) estimate of $\mathbf{q}$ minimizes the following cost function:

$$\begin{aligned} E(\mathbf{q}) = \tfrac{1}{2}\sum_i \left(\tfrac{g(\mathbf{z}_i)-\beta_i}{\alpha_i} - \mathbf{H}_i\mathbf{q}\right)^T \mathbf{K}^{-1}\left(\tfrac{g(\mathbf{z}_i)-\beta_i}{\alpha_i} - \mathbf{H}_i\mathbf{q}\right) \\ + \tfrac{1}{2}(\mathbf{q}-\mu_{\mathbf{q}})^T\mathbf{\Lambda}^{-1}(\mathbf{q}-\mu_{\mathbf{q}}). \end{aligned} \tag{16}$$

One technique to obtain the MAP estimate in equation (16) is the steepest descent technique. The current estimate $\hat{\mathbf{q}}$ is updated in the direction of the negative gradient of $E(\hat{\mathbf{q}})$:

$$\hat{\mathbf{q}}' = \hat{\mathbf{q}} - \kappa\nabla E(\hat{\mathbf{q}}), \tag{17}$$

where κ is the step size, and $\nabla E(\hat{\mathbf{q}})$ can be found using

$$\begin{aligned} \nabla E(\mathbf{q}) = -\sum_i \mathbf{H}_i^T\mathbf{K}^{-1}\left(\tfrac{g(\mathbf{z}_i)-\beta_i}{\alpha_i} - \mathbf{H}_i\mathbf{q}\right) \\ + \mathbf{\Lambda}^{-1}(\mathbf{q}-\mu_{\mathbf{q}}). \end{aligned} \tag{18}$$

The step size κ in equation (17) can be fixed or updated adaptively during the iterations. Hessian of $E(\mathbf{q})$ can be used for changing κ. (In our experiments we utilized the Hessian to determine κ at each iteration.)

4. EXPERIMENTAL RESULTS

We captured 23 images with a Canon G5 digital camera. (Figure 3 shows six of these images.) Exposure rates and illumination conditions are different for some of these images. Images are first registered using a feature based method. Feature points in the images are extracted using the Harris corner detector. Then these control points are matched using normalized cross correlation. We use RANSAC to eliminate the outliers. We run RANSAC for the second time on the inliers to obtain the homographies.

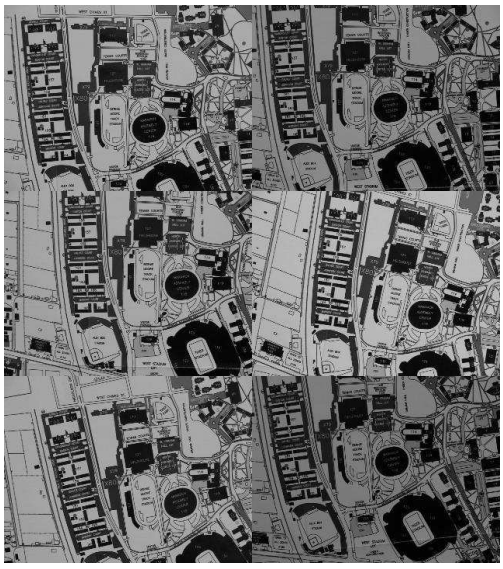

Fig. 3. Images of the same scene captured with a Canon G5 digital camera.

There are various methods available in the literature to estimate camera response function [4, 9, 7, 8, 10, 5]. In our experiments, we use the comparagram method in [8]. (We had captured multiple images with known exposure times and used those to estimate the camera response function. Even if the exposure times were not known, iterative estimation was possible [8]. The camera response function can be determined up to a scale factor; therefore, $f(255)$ is set to one. To obtain a smooth function, a regularization filter of $[1, -2, 1]$ is used.) Estimated camera response function is shown in Figure 4. For the new data set, we used the registered images and previously calculated camera response function to estimate α_i and β_i parameters using robust least squares estimation. Deviation of pixel values from the linear relationship based on α_i and β_i is used to determine the variance σ^2.

Figure 5 shows the results for the LS (based on equation (7)), the POCS (based on equation (11)) and the MAP algorithms. The POCS and MAP algorithms performed similarly; the least squares algorithm performed slightly worse than those. The MAP has produced the least artifacts due to the Gaussian prior. (We used bilinearly interpolated initial estimate as the mean image $\mu_{\mathbf{q}}$, and identity matrix for $\mathbf{\Lambda}$.) In all experiments, the point spread function was chosen to be a 5×5 Gaussian window with standard deviation of 1. The number of iterations were five in all cases.

5. SUMMARY

In this paper, we showed how to model camera response function, exposure time, and white balancing for super-resolution reconstruction. We showed formulations for several approaches, some of these algorithms were applied on a real data set. The improvement in spatial resolution is evident. We should be able to achieve better results as we increase the number of images in the data set.

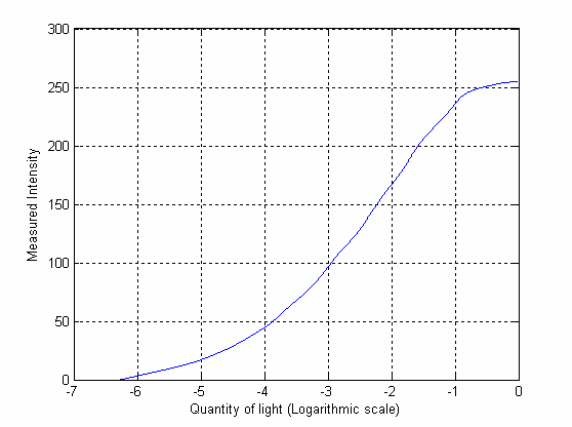

Fig. 4. Estimated camera response function is shown in logarithmic scale.

6. REFERENCES

[1] "Super-resolution image reconstruction," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 21–86, May 2003.

[2] F. M. Candocia, "A least squares approach for the joint domain and range registration of images," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 4, May 2002, pp. 3237–3240.

[3] P. L. Combettes, "The foundations of set theoretic estimation," *Proc. of the IEEE*, vol. 81, no. 2, pp. 182–208, February 1993.

[4] P. E. Debevec and J. Malik, "Modeling and rendering architecture from photographs," in *Proc. of the ACM SIGGRAPH*, 1997, pp. 369–378.

[5] M. D. Grossberg and S. K. Nayar, "Determining the camera response from images: what is knowable?" *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 11, pp. 1455–1467, November 2003.

[6] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, "Multiframe information fusion for gray-scale and spatial enhancement of images," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, September 2003, pp. 319–322.

[7] S. Mann, "Comparametric equations with practical applications in quantigraphic image processing," *IEEE Trans. Image Processing*, vol. 9, no. 8, pp. 1389–1406, August 2000.

[8] S. Mann and R. Mann, "Quantigraphic imaging: estimating the camera response and exposures from differently exposed images," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 842–849.

[9] M. A. Robertson, S. Borman, and R. L. Stevenson, "Dynamic range improvement through multiple exposures," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1999, pp. 159–163.

[10] Y. Tsin, V. Ramesh, and T. Kanade, "Statistical calibration of the ccd imaging process," in *Proc. Int. Conf. Computer Vision*, vol. 1, 2001, pp. 480–487.

[11] A. Zomet, A. Rav-Acha, and S. Peleg, "Robust super-resolution," in *Computer Vision and Patter Recognition*, vol. 1, December 2001, pp. 645–650.

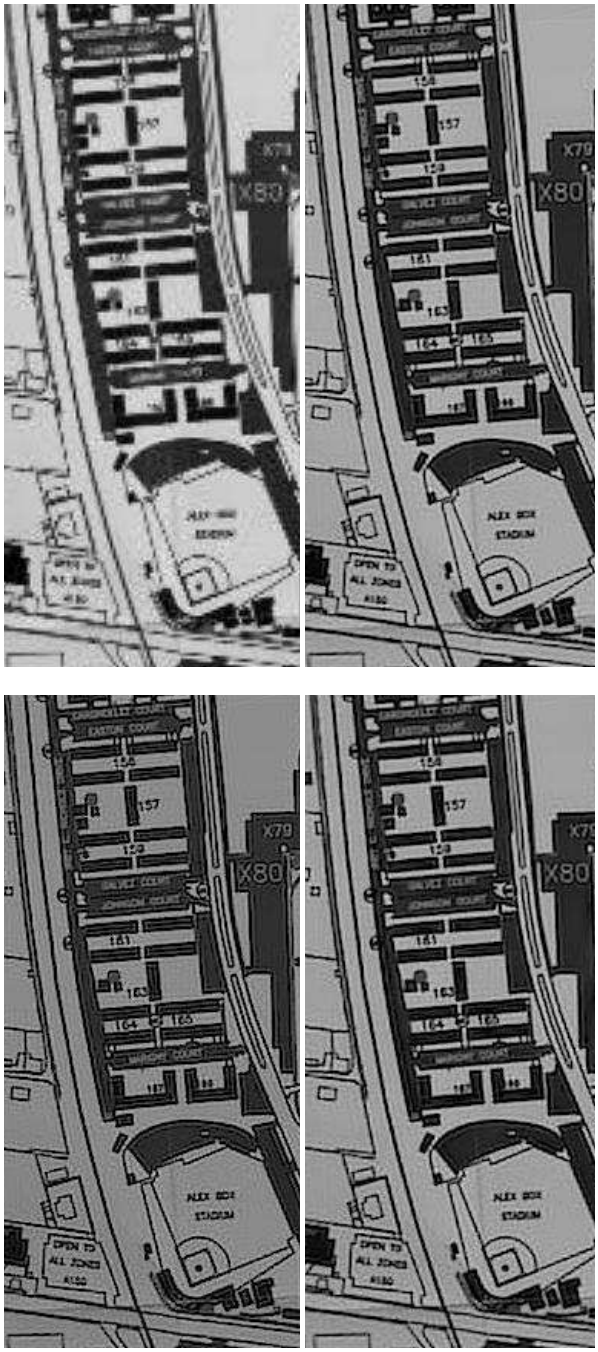

Fig. 5. Image on top is a cropped region from one of the observations. Image on top-right is obtained with the LS algorithm. Image on bottom-left is obtained with the POCS algorithm. Image on bottom-right is obtained with the MAP algorithm.