# Local Blur Estimation and Super-Resolution

Ming-Chao Chiang

Columbia University
Department of Computer Science
New York, NY 10027
chiang@cs.columbia.edu

Terrance E. Boult

Lehigh University
Department of EECS
Bethlehem, PA 18015
tboult@eecs.lehigh.edu

**Abstract**

*Until now, all super-resolution algorithms have presumed that the images were taken under the same illumination conditions. This paper introduces a new approach to super-resolution—based on edge models and a local blur estimate—which circumvents these difficulties. The paper presents the theory and the experimental results using the new approach.*

## 1 Introduction

We have recently proposed a new algorithm for enhancing image resolution from an image sequence [1] where we compared our super-resolution results with earlier research on this subject [2, 3, 4, 5, 6, 7, 8]. We showed that the integrating resampler [9] can be used to enhance image resolution. We further showed that warping techniques can have a strong impact on the quality of the super-resolution imaging. However, left unaddressed in [1] are several important issues. Of particular interest is lighting variation. The objective of this paper is to address techniques to deal with these issues. The examples herein use text because the high frequency information highlights the difference in super-resolution algorithms and because it allows us to ignore, for the time being, the issues of 3D edges under different views. Clearly, matching is a critical component if this is to be used for fusing 3D objects under different viewing/lighting conditions. This paper, however, concentrates on how well we can combine them presuming good matching information.

In what follows in this paper, we first present the new super-resolution algorithm where we compare the resulting super-resolution images with those presented in [1] in Section 2. We then review the integrating resampler, an efficient method for warping using imaging-consistent restoration/reconstruction algorithms, in Section 3. This algorithm is well suited for today's pipelined micro-processors. In addition, the integrating resampler can allow for modifications of the intensity values to better approximate the warping characteristics of real sensors. The new algorithms for edge localization and local blur estimation are given, respectively, in Sections 4 and 5. These two algorithms are both based on the imaging-consistent restoration/reconstruction algorithms described in [10]. Future work is given in Section 6.

This work is supported in part by NSF PYI IRI-90-57951, NSF grant CDA-9413782, and DOD MURI program ONR N00014-95-1-0601. Several other agencies and companies have also supported parts of this research.

## 2 Edge-Based Super-Resolution

For almost all applications involving an image sequence, the problem of lighting variation arises. It is very unlikely that lighting conditions remain the same for the entire image sequence, even when they are taken consecutively in a well controlled environment. If the images are not from a short time span, we know that variations are often significant. There exist all kinds of possibilities that may cause lighting to vary, e.g., viewpoint variation, temperature of light bulbs, clouds or people passing by, and so on. Thus, the problem is that how do we eliminate this undesirable effect? One of the easiest solutions to this problem is, of course, just to ignore lighting variations and assume that the effect of lighting variations is negligible. However, this is limiting. The idea we propose herein is a simple solution. Rather than obtaining a super-resolution image by fusing all the images together, we may choose one, and only one, of the images from the image sequence and then fuse together all the edges from the other images. This does not necessarily solve the lighting variation problem. But, this effectively mitigates the problem of lighting variation since we are now dealing with a single image as well as the edge positions that are less sensitive to the change of lighting.

However, to fuse all the edges together, it requires that the edges be first detected and then warped. It also requires that the reference image be reestimated and scaled up based on the edge models and local blur estimation. Therefore, we generalize the idea of the imaging-consistent restoration/reconstruction algorithms to deal with discontinuities in an image. The new algorithm for edge detection is given in Section 4. The new algorithm for local blur estimation for each edge point is presented in Section 5. The idea of edge-based super-resolution described herein is to fuse all the edge models together.

We turn now to the problem of super-resolution from an image sequence. The idea of super-resolution is based on the fact that each image in the sequence provides small amount of additional information. By warping all the im-

ages to the reference image (scaling at the same time the images are being warped) and then fusing together all the information available from each image, a super-resolution image can be constructed. Given the image sequence, our super-resolution algorithm is now formulated, as follows:

1. Estimate the motions involved in the image sequence.
2. Estimate the edges using the procedure derived in Section 4.
3. Estimate the blur models for each edge point in the image using the procedure derived in Section 5.
4. Choose a "reference" image to determine the lighting for the output image.
5. Warp all the edge/blur models to the reference image and fuse them.
6. Use the fused edge/blur models and the reference image to compute the super-resolution intensity image.
7. Optional deblurring stage.

The central idea of super-resolution is to bring together small amount of additional information available from each image in the image sequence. Even if the final sampling rate is exactly the same as the original, we can still do super-resolution. In this case, it provides a sharper image of the original size. As an example, we demonstrate this in [1, Figure 1].

We presume that "motion" is computed, which is easy for rigid transformation, although interlacing issues must be properly addressed.

When off-the-shelf lenses and cameras are used, prewarping can be used to remove the distortions (See [11, 12]). In [9], we showed that the integrating resampler can improve the quality of the matching.

We tried several different approaches to fuse the edge/blur models together, including the averaging and the median filters. Our experiments show that the median filter is better, though often not much better than the averaging filter.

The test data was taken using Sony XC-77, attached to a Datacube MV200 System. Figures 1 and 2 show our experimental results. A 32 image sequence of size 77x43 was used to compute the super-resolution image of size 308x172, i.e., a scale-up by a factor of 4.

Figure 1 shows two of the original images with two different illumination conditions blown up by a factor of 4 using, respectively, pixel replication and bi-linear resampling. Figures 1a and b shows the results using pixel replication; Figures 1c and d, the results using bi-linear resampling.

Figure 2 shows the final results of our experiment. Figure 2a shows the resulting super-resolution image using the algorithm presented in [1] without deblurring at the end; Figure 2b shows Figure 2a with deblurring at the end. Figure 2c shows the resulting super-resolution image using the algorithm proposed herein without deblurring at the end; Figure 2d shows Figure 2c with deblurring at the end.

As can be easily seen, the new super-resolution algorithm proposed herein circumvents the difficulties caused by the illumination conditions while all previous algorithms simply ignore the changes in illumination. Furthermore, the running time of our method is often more than two or three times faster than Irani's back-projection method. See [1] for the comparison between our algorithm and Irani's back-projection method.

## 3 Imaging-Consistent Warping

Consider the imaging model in Fig. 3. An algorithm is called *imaging-consistent* if it is the exact solution for some input function, which, according to the sensor model, would have generated the measured input. For image reconstruction, we achieve this by first computing a functional restoration (i.e., $f_2$), then blurring it again by the pixel's PSF. This actually defines a whole class of image restoration/reconstruction techniques, depending on the model for $f_2$. Details can be found in [10]. Only one dimensional image models are presented; higher dimensions are treated separably.

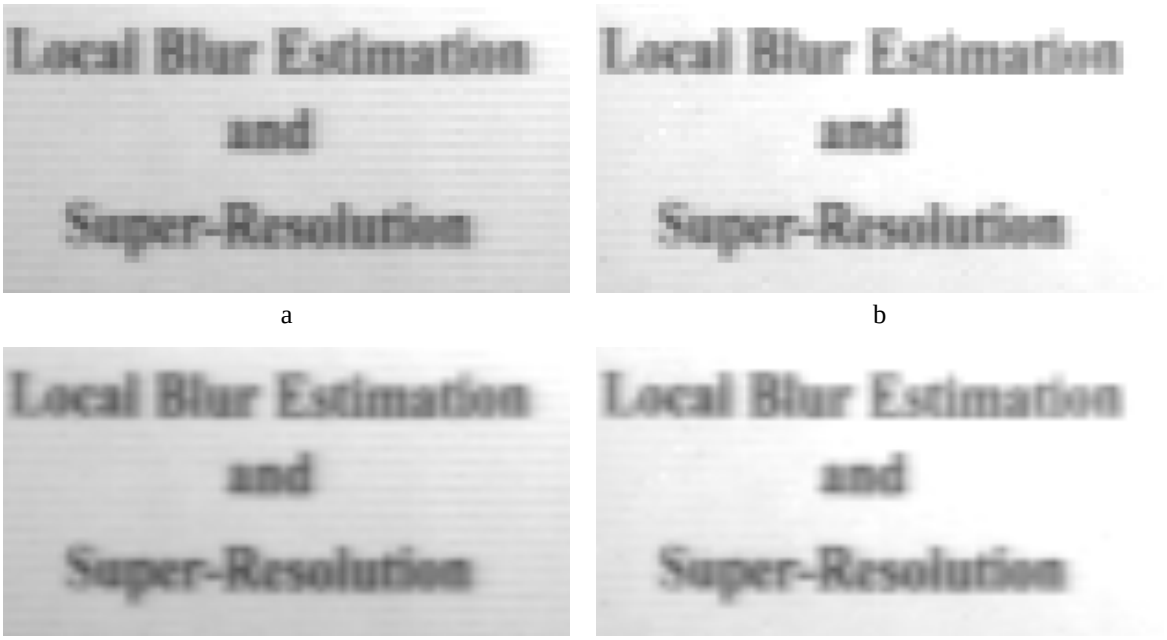
Once the idea of imaging-consistent reconstruction from area samples is accepted, probably the simplest method to consider is based on a piecewise quadratic model for the image. If we assume a Rect PSF filter (1 inside and 0 outside the pixel), the imaging consistent algorithm is easy to derive. To ensure that the function is continuous, and that the method is local, we define the value of the reconstruction at the pixel boundaries $k_i$ and $k_{i+1}$, to be equal to $E_i$ and $E_{i+1}$. Any method of approximation could be used to compute $E_i$, though our examples will only include linear interpolation and cubic convolution.

Given the values $E_i$ at the pixel edges, an additional constraint, that the integral across the pixel must equal $V_i$, results in exactly three constraints. From this, one can determine $f_2$, a quadratic polynomial. This gives the intra-pixel restoration. Reconstruction algorithm can be derived by simply blurring the resulting restoration by a PSF of the same scale as input. For super-resolution, we consider only restoration.

The imaging-consistent algorithms described above and in [10] are linear filters. We designed them for use in what we call the *integrating resampling* approach.

As described before, our model of image formation requires the image to be spatially sampled with a finite area sampler. This is tantamount to a weighted integral being computed on the input function. Because we have a functional form for the restoration, we can simply integrate this function with the PSF for the output area sampler. In this section, we assume that the output sampler has a Rect PSF, though there is no limitation on the degradation models that one can use.

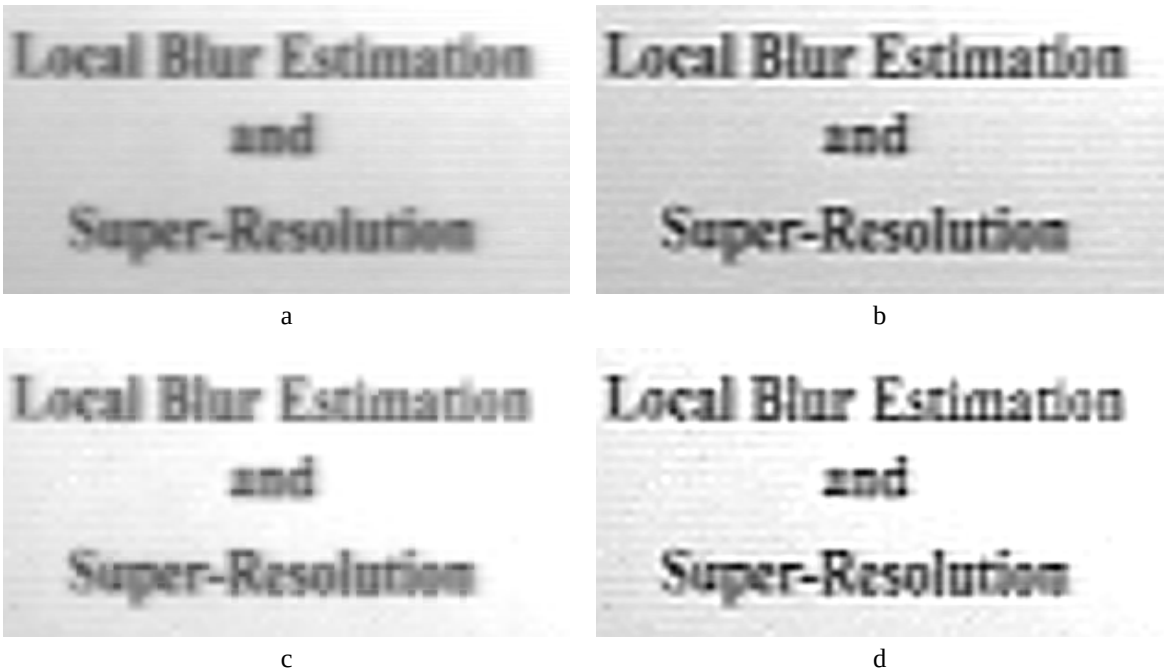
To define the integrating resamplers, we generalize the idea of the imaging-consistent algorithms described above. Whereas imaging-consistent algorithms simply assume the




a b

c d

Figure 1: Two of the original images showing two different illumination conditions blown up by a factor of 4. (a) and (b) using pixel replication; (c) and (d) using bi-linear resampling.




a b

c d

Figure 2: Final results from a 32 77x43 image sequence taken by XC-77. (a) super-resolution using the algorithm proposed in [1] without deblurring at the end; (b) (a) with deblurring at the end; (c) super-resolution using the algorithm proposed herein without deblurring at the end; (d) (c) with deblurring at the end.

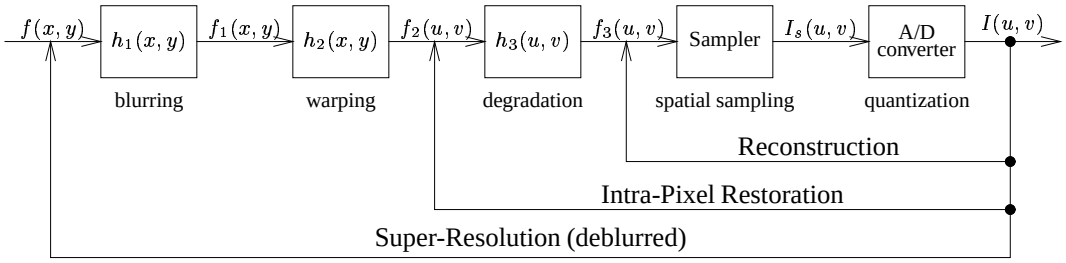



Figure 3: The image formation process and the relationship between restoration, reconstruction, and super-resolution.

degradation models are identical for both input and output; the integrating resamplers go one step further, allowing (1) both input and output to have their own degradation model, and (2) the degradation model to vary its size for each output pixel.

When we are resampling the image and warping its geometry in a nonlinear manner, this new approach allows us to efficiently do both pre-filtering and post-filtering. Because we have already determined a functional form for the input, no spatially-varying filtering is needed, as would be the case if direct inverse mapping were done.

Computing the exact value of the restored function weighted by the PSF could be done in functional form if the mapping function has a functional inverse and the PSF is simple (as in this example). In general, however, it cannot be done in closed form, and numerical integration is required. To reduce the computational complexity, we propose a scheme where for each input pixel, we use a linear approximation to the spatial warping within that pixel, but use the full non-linear warp to determine the location of pixel boundaries. This integrating resampler, proposed in [9], also handles antialiasing of partial pixels in a straightforward manner.

The underlying idea of this integrating resampler can be found in the work of Fant [13] who proposed a warping algorithm for the special case of piecewise constant reconstruction. Our contribution is twofold: the generalization to deal with more advanced reconstruction algorithms, and providing for a modeling of real lens effects by using a modeling of real warps that affects the image radiance.

## 4 Edge Localization

Typically, edge detection involves the estimation of first and second derivatives of the luminance function, followed by selection of zero crossing and extrema. The edge detection algorithm we propose herein uses pretty much the same idea except that it is based on the functional form for the imaging-consistent restoration/reconstruction algorithms. Thus, its sub-pixel accuracy and other parameters are easily combined with the imaging-consistent restoration/reconstruction algorithms to provide for a reconstruction algorithm that supports intensity discontinuities.

In what follows in this section, we present only one dimensional case; higher dimensions are treated separably. Also, we consider only the imaging-consistent reconstruction algorithm QRR derived in [10]. The imaging-consistent reconstruction algorithm QRR is a cubic polynomial that spans from the center of one input pixel to the next and is given by

$$\begin{aligned} Q_i(x) &= (E_{i+2} - E_i - 2(V_{i+1} - V_i))\,x^3 \\ &\quad + (2E_i - E_{i+1} - E_{i+2} + 3(V_{i+1} - V_i))\,x^2 \\ &\quad + (E_{i+1} - E_i)\,x + V_i \end{aligned}$$

where $0 \le x \le 1$.

To simplify our discussions, throughout this section, we use

$$Q(x) = ax^3 + bx^2 + cx + d$$

to represent the imaging-consistent reconstruction algorithm QRR. For convenience, we also drop the subscripts for $Q$, $E$, and $V$.

Given $Q(x)$, our edge detection algorithm is outlined below.

1. *Find the function that gives all the slopes of $Q(x)$.* Since we have a functional form for $Q(x)$, this is nothing more than taking the first derivative of $Q(x)$. Clearly,
$$Q'(x) = 3ax^2 + 2bx + c.$$
For convenience, let us call this quadratic the slope function $S(x)$.
2. *Find edge position with sub-pixel accuracy.* This is exactly the same as finding either zero crossing or the extremum of the slope function $S(x)$. Again, this is easy to derive. Clearly, the first and second derivatives of the slope function $S(x)$ are
$$S'(x) = 6ax + 2b \quad \text{and} \quad S''(x) = 6a.$$
If $a \neq 0$, the extremum (i.e., the step edge) can be found located at $x_0 = -b/3a$, and its value is given by $S(x_0) = -b^2/3a + c$. If $x_0$ is turned out to be located outside the range $(0, 1)$, it is simply assumed to be undefined. If $a = 0$, then there are two cases:
   (a) If $b \neq 0$, then the extremum must occur at one end or the other of that pixel, i.e., $x_0 = 0$ or $x_0 = 1$, Its value is given by either $S(0) = c$ or $S(1) = 3a + 2b + c$.

(b) If $b = 0$, then the extremum must occur everywhere within that pixel. This is exactly the same as saying that there is no extremum.

These two cases are also assumed to be undefined.

3. *Find edge direction and magnitude.* This is easy to determine given that we know the edges in the $x$ and $y$ directions. Let $s_x$ and $s_y$ denote, respectively, the slopes of an edge in the $x$ and $y$ directions. Then, the direction and magnitude of an edge are given, respectively, by

$$E_d = \begin{bmatrix} s_x \\ s_y \end{bmatrix} \quad \text{and} \quad E_m = \sqrt{s_x^2 + s_y^2}.$$

In the case that $x_0$ is located outside the range $(0, 1)$ or $a = 0$, it can simply be assumed to be undefined.

One of the problems with edge detection is that how do we distinguish significant edges from insignificant ones? Our experiments use a predefined threshold to get rid of insignificant edges. Better techniques are under investigation.

## 5 Local Blur Estimation

In [1], we showed that deblurring after image fusion is most effective for super-resolution imaging. However, this presumes that the blur is known. In this section, we propose a new algorithm for local blur estimation. The idea of this algorithm is to model a blurred edge with two components: a step edge and a blur kernel. The blurred edge can be obtained by applying the blur kernel to the step edge.

Given $Q(x)$ and the location of the blurred edge $x_0$, our local blur estimation algorithm is outlined below.

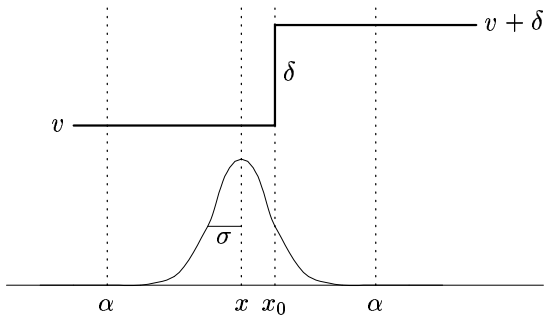



Figure 4: Edge model. See text for details.

1. *Determine the edge model.* Before we can discuss the local blur estimation algorithm, we must specify the edge model. Figure 4 shows our edge model. We model an edge as a step function $v + \delta u(x)$ where $v$ is the unknown intensity value and $\delta$ is the unknown amplitude of the edge. The focal blur of this edge is modeled by a "truncated" Gaussian blur kernel

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

where $\sigma$ is the unknown standard deviation. The complete edge model is thus given by Eq. (1); expanding and simplifying gives Eq. (2) where $\alpha$ is a predefined nonnegative constant, and $x_0 \in [0, 1]$ is the location of the step edge. Since there are three unknowns, $v$, $\delta$, and $\sigma$, we need three constraints to solve it. In what follows, we consider only the edge model defined on the interval $[x_0 - \alpha, x_0 + \alpha]$ because outside the range, they are constant.

2. *Find solutions for the three unknowns $v$, $\delta$, and $\sigma$.* Since we have exactly three constraints,

$$\begin{aligned} Q(x) &= B(x) \\ Q'(x) &= B'(x) \\ Q'''(x) &= B'''(x) \end{aligned}$$

the system of equations can be easily solved. Note that we use the third derivative instead of the second derivative for solving the system of equations. This is to ensure that the general edge model gives exactly the same edge model located at $x_0$. Solving the system of equations for the unknowns $v$, $\delta$, and $\sigma$, and noting that the value of $\sigma$ must be positive, we have

$$\sigma = +\sqrt{\frac{r + \sqrt{s}}{2}}$$

$$\delta = \left(\sqrt{2\pi}\sigma Q'(x)\right) / \exp\left(-\frac{(x - x_0)^2}{2\sigma^2}\right)$$

$$v = \left(Q(x) - \frac{\delta}{2}\,\mathrm{erf}\left(\frac{x - x_0}{\sqrt{2}\sigma}\right)\right) / \mathrm{erf}\left(\frac{\alpha}{\sqrt{2}\sigma}\right) - \frac{\delta}{2}$$

where $r = -Q'(x)/Q'''(x)$ and $s = r^2 - 4r(x - x_0)^2$. Since the value of $\sigma$ must be positive, it follows that $\sigma$ has a solution if and only if $s > 0$ and either $r + \sqrt{s} > 0$ or $r - \sqrt{s} > 0$; $\sigma$ has two solutions if $s > 0$ and both $r + \sqrt{s} > 0$ and $r - \sqrt{s} > 0$. However, to ensure that the general edge model derived herein gives exactly the same edge model located at $x_0$, only $r + \sqrt{s} > 0$ yields a valid solution. Therefore, the solution of $\sigma$ is unique and is given by

$$\sigma = \sqrt{\frac{r + \sqrt{r^2 - 4r(x - x_0)^2}}{2}}$$

The value of $\delta$ may be positive or negative depending on the value of $Q'(x)$. Letting $x = x_0$, we have the edge model located at $x_0$.

The idea of super-resolution is to use the edge models derived above to reestimate the underlying images. Given the edge location $x_0$ as well as its corresponding edge model ($v$, $\delta$, and $\sigma$), the underlying image can be reestimated as follows:

1. Replace the intensity values to the left of $x_0$ by $v$ and the intensity values to the right of $x_0$ by $v + \delta$. The replacement is no more than three pixels to the left and right so that the algorithm remains local.
2. Recompute $E_i$ and $E_{i+1}$ based on the intensity values given in the previous step. Let us call them $\bar{E}_i$ and $\bar{E}_{i+1}$.

$$B(x) = \begin{cases} \int_{x-\alpha}^{x+\alpha} vG(x-z)\,dz, & x < x_0 - \alpha \\ \int_{x-\alpha}^{x_0} vG(x-z)\,dz + \int_{x_0}^{x+\alpha} (v+\delta)G(x-z)\,dz, & x_0 - \alpha \le x \le x_0 + \alpha \\ \int_{x-\alpha}^{x+\alpha} (v+\delta)G(x-z)\,dz, & x > x_0 + \alpha \end{cases} \quad (1)$$

$$B(x) = \begin{cases} v \operatorname{erf}\left(\frac{\alpha}{\sqrt{2}\sigma}\right), & x < x_0 - \alpha \\ (v + \frac{\delta}{2}) \operatorname{erf}\left(\frac{\alpha}{\sqrt{2}\sigma}\right) + \frac{\delta}{2} \operatorname{erf}\left(\frac{x - x_0}{\sqrt{2}\sigma}\right), & x_0 - \alpha \le x \le x_0 + \alpha \\ (v+\delta) \operatorname{erf}\left(\frac{\alpha}{\sqrt{2}\sigma}\right), & x > x_0 + \alpha \end{cases} \quad (2)$$

3. Recompute $\bar{Q}_i(x)$ based on $\bar{E}_i$ and $\bar{E}_{i+1}$ while at the same time the image is being scaled up.

## 6 Future Work

Further work is needed before the super-resolution algorithm is robust enough for general use in applications. We are currently investigating techniques to address the following issues: (1) How do we quantitatively measure the error of the resulting super-resolution image? (2) How do we precisely estimate the motion involved in the image sequence? (3) How do we precisely match 3D edges under different viewing/lighting conditions? (4) How do we use the edge/blur models derived herein to reestimate the underlying image in a better way?

## 7 CONCLUSION

This paper introduces a new algorithm for enhancing image resolution from an image sequence. The new algorithm is based on local blur estimation. Again, the approach we propose herein uses the integrating resampler as the underlying resampling algorithm. We address techniques to deal with lighting variation, edge localization, and local blur estimation. For the purpose of comparison, evaluations are made by comparing the resulting images with those in our previous work.

## References


[1] M. Chiang and T. Boult, "Efficient image warping and super-resolution," *Proceedings of the Third IEEE Workshop on Applications of Computer Vision*, pp. 56–61, Dec 1996.

[2] T. Huang and R. Tsai, "Multi-frame image restoration and registration," *Advances in Computer Vision and Image Processing*, vol. 1, pp. 317–339, 1984.

[3] D. Gross, "Super-resolution from sub-pixel shifted pictures," Master's thesis, Tela-viv University, Oct 1986.

[4] S. Peleg, D. Keren, and L. Schweitzer, "Improve image resolution using subpixel motion," *Pattern Recognition Letter*, pp. 223–226, 1987.

[5] D. Keren, S. Peleg, and R. Brada, "Image sequence enhancement using sub-pixel displacements," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 742–746, Jun 1988.

[6] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models and Image Processing*, vol. 53, pp. 231–239, May 1991.

[7] M. Irani and S. Peleg, "Motion analysis for image enhancement: Resolution, occlusion, and transparency," *Journal of Visual Communication and Image Representation*, vol. 4, pp. 324–335, Dec 1993.

[8] B. Bascle, A. Blake, and A. Zisserman, "Motion deblurring and super-resolution from an image sequence," *Computer Vision—ECCV*, pp. 573–581, Apr 1996.

[9] M. Chiang and T. Boult, "The integrating resampler and efficient image warping," *Proceedings of the ARPA Image Understanding Workshop*, pp. 843–849, Feb 1996.

[10] T. Boult and G. Wolberg, "Local image reconstruction and subpixel restoration algorithms," *CVGIP: Graphical Models and Image Processing*, vol. 55, pp. 63–77, Jan 1993.

[11] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. RA-3, pp. 323–344, August 1987.

[12] M. Chiang and T. Boult, "A public domain system for camera calibration and distortion correction," Tech. Rep. CUCS-038-95, Columbia Univ., Dept. of CS, and Lehigh Univ., Dept. of EECS, Dec 1995.

[13] K. Fant, "A nonaliasing, real-time spatial transform technique," *IEEE Computer Graphics and Applications*, vol. 6, pp. 71–80, January 1986. See also "Letters to the Editor" in Vol.6 No.3, pp. 66-67, Mar 1986 and Vol.6 No.7, pp. 3-8, July 1986.