# Robust Wavelet-Based Super-Resolution Reconstruction: Theory and Algorithm

Hui Ji and Cornelia Fermüller, *Member, IEEE*

**Abstract**—We present an analysis and algorithm for the problem of super-resolution imaging, which is the reconstruction of high-resolution (HR) images from a sequence of low-resolution (LR) images. Super-resolution reconstruction entails solutions to two problems. One is the alignment of image frames. The other is the reconstruction of an HR image from multiple aligned LR images. Both are important for the performance of super-resolution imaging. Image alignment is addressed with a new batch algorithm, which simultaneously estimates the homographies between multiple image frames by enforcing the surface normal vectors to be the same. This approach can handle longer video sequences quite well. Reconstruction is addressed with a wavelet-based iterative reconstruction algorithm with an efficient denoising scheme. The technique is based on a new analysis of video formation. At a high level, our method could be described as a better-conditioned iterative back-projection scheme with efficient regularization criteria in each iteration step. Experiments with both simulated and real data demonstrate that our approach has better performance than existing super-resolution methods. It can remove even large amounts of mixed noise without creating artifacts.

**Index Terms**—Perfect reconstruction filter banks, super-resolution, multiple frame alignment, wavelet denoising.

✦

## 1 OVERVIEW

THE problem of *super-resolution* reconstruction, which is defined as restoring a high-resolution (HR) image from a sequence of low-resolution (LR) images (see Fig. 1), has been studied by many researchers in recent years. Most super-resolution algorithms formulate the problem as a signal reconstruction problem. Essentially, these algorithms differ in two aspects: One is in how the image frames of the sequence are aligned in a common coordinates system; the other is in how the HR image is reconstructed from the aligned LR image frames. The accuracy of the image alignment and the robustness of the signal reconstruction determine the algorithmic performance. The two aspects are related. Thus, the imperfectness of the image alignment has to be considered in the later stage of image restoration.

In this paper, we propose a complete super-resolution reconstruction system, which addresses both the problems of frame alignment and image restoration. Our contributions are twofold. The first is a theoretical analysis of video formation from the view of filter banks, which leads to a new wavelet-based reconstruction scheme for image reconstruction. The second is a batch algorithm for image alignment, which computes the projective flow (planar homographies) across all frames in the sequences.

Using a sequence of similar but not identical image frames makes it is possible to increase the spatial resolution

of one still image. However, it is difficult since this is an ill-posed inverse problem. Despite the difficulties, researchers have made great progress toward stable algorithms. Iterative back-projection methods [14], [23] have been shown to be an effective tool for HR image reconstruction. It is known, however, that the deblurring process, which is part of this approach, makes it very sensitive to the noise. Thus, the requirement of very accurate image alignment estimates limits its practical use. Various regularization methods have been proposed to deal with the noise. However, these methods either are very sensitive to the assumed noise model (Tikhonov regularization) or are computationally expensive (Total Variation regularization). See [10] for more details.

Our contributions to the reconstruction process are twofold. First, we model the image formation procedure from the point of view of filter bank theory. Then, based on this new formulation, we analyze the limits of HR reconstruction. Our conclusion is that in general full recovery is not possible without enforcing some constraints on the recovered images. At best, we could reconstruct the image convolved with a specific low-pass filter (namely, $\frac{1}{4}(1,1) \otimes (1,1)$ for the case of the Box-type PSF). Second, based on our new formulation, we present a robust wavelet-based algorithm for image reconstruction. The iteration scheme in our algorithm is inherently more robust to noise than that of classic back-projection methods [14], [23], because the projection matrix of our back-projection scheme has a better condition number. We will show that, both in theory and experiments, it has better performance in suppressing the error propagation than other back-projection iteration schemes.

Furthermore, our algorithm allows us to include a wavelet-based denoising scheme in each iteration of the reconstruction, which effectively removes the noise without creating smoothing artifacts. The advantage of our denoising scheme over regularization methods is that it is nearly

---

- *H. Ji is with the Department of Mathematics, National University of Singapore, 2, Science Drive 2, Singapore 117543. E-mail: matjh@nus.edu.sg.*
- *C. Fermüller is with the Computer Vision Laboratory, Center for Automation Research, UMIACS, University of Maryland, College Park, MD 20742. E-mail: fer@cfar.umd.edu.*
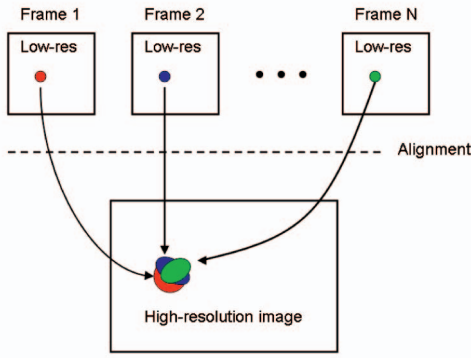
Fig. 1. Illustration of super-resolution imaging.

optimal with respect to the risk bound. That is, it has the theoretical minimal error in removing noise of unknown models. Its effectiveness in removing mixed noise and relatively large amounts of noise is demonstrated in experiments. It is worth mentioning that our denoising scheme adds very little computational burden compared to other complicated regularization methods. Briefly, our method could be described as a generalized iterative back-projection method with fast and optimal regularization criteria in each iteration step.

Wavelet theory has previously been used for image denoising and deblurring from static images [5], [6]. However, it has not been studied much with respect to the super-resolution problem. In recent work, wavelet theory has been applied to this problem [21], but only for the purpose of speeding up the computation. Our contribution lies in an analysis that reveals the relationship between the inherent structure of super-resolution reconstruction and the theory of wavelet filter banks. This relationship is fully exploited by using various techniques from wavelet theory in the iterations of the reconstruction.

For the process of image frame alignment, the flow-based approach is the most popular choice because of its flexibility. The difficulty in accurate estimation of general nonparametric image flow makes the reconstruction of higher resolution images not very meaningful. Thus, some assumption on the underlying flow model has to be enforced. Such assumption, of course, makes the algorithm less general but more practical. The most frequently used flow models are the affine transform and the more general projective transform [2], [9], [14], [27]. The most popular reconstruction procedure is the iterative back-projection method [14], [23], [28].

In this paper, we propose a new algorithm for computing projective flow (planar homographies) simultaneously on all frames in the sequence. The most closely related studies are [12], [26], and [15], which all provide algorithms for the estimation of planar motion between multiple frames based on some rank constraint. However, these studies either are limited to short image sequences [12], [26] or compute relative motion only between neighboring frames [15], which could lead to accumulative errors in the image alignment. Our approach is an extension of [15] to the more general estimation of homographies (as opposed to planar flow). Briefly, the motion between the reference frame and any other frame is modeled as a homography. Then, the shape constraint proposed in [15] is directly incorporated

into the estimation of the homographies. This leads to accurate alignment for longer sequences. As a result, we have more frames available for the reconstruction.

## 2 FORMULATION OF HIGH-TO-LOW IMAGE FORMATION

We first formulate the high-to-low image formation process. To simplify the exposition, in the following, we only discuss 1D signals with resolution enhancement by a factor 2. Later, without much difficulty, the analysis will be extended to the 2D case with arbitrary resolution enhancement. Adopting Farsiu et al.'s [10] notation, the image formation process in the pixel domain can be modeled as

$$y = \sigma[H * X(F(t))] + N, \tag{1}$$

where $t$ is the spatial variable, $X(t)$ is the continuous signal, $y$ is the discrete signal, $H$ is the blurring operator (either optical blurring or motion blurring or both), $F$ is the geometric transform (affine, perspective, and so forth), $N$ is the noise in the LR image, $\sigma$ is the decimation operator, and "$*$" is the convolution operator. Not considering the noise for the moment, the HR signal $x$ and the LR signal $y$ can be defined as

$$x = \sigma[X], \quad y = [\sigma[H * X(F(t))]] \downarrow_2, \tag{2}$$

where $\downarrow_2$ is the downsampling operator with rate 2.

Next, we derive the relationship between the LR signal $y$ and the HR signal $x$. Denote the velocity of the signal as $\epsilon(t) = F(t) - t$, which is also called the *optical flow* in computer vision. For simplicity of notation, here, we assume a subpixel flow model with $0 < \epsilon(t) < 2$ on the denser grid of the HR image $x$. (Larger flow could always be reduced to the case of subpixel flow by reassigning the pixel value.) Thus, in the LR image, the flow values are all subpixel shifts (recall that a one-unit shift on the coarse grid of $y$ is equal to a two-unit shift on the fine grid of $x$).

Let $\{j\}$ be a fine grid for the spatial coordinates $x$. Then, for point $j$ of $y$ on the coarse grid (its coordinate is $2j$ on the fine grid) with $0 \leq \epsilon(2j) < 1$, the first-order Taylor approximation of (2) at point $2j$ can be written as

$$
\begin{aligned}
y(j) &= [H * X(F(t))]_{t=2j} \\
&= [H * X(\epsilon(t) + t)]_{t=2j} \\
&= [H * X(t)]_{t=2j} + \epsilon(2j)[H * X'(t)]_{t=2j} \\
&= [H * X(t)]_{t=2j} + \epsilon(2j)[H' * X(t)]_{t=2j}.
\end{aligned}
$$

For all other points $j'$ of $y$ with $1 \leq \epsilon(2j') < 2$, a similar argument yields

$$y(j') = [H * X(t)]_{t=2j'+1} + (\epsilon(2j') - 1)[H' * X(t)]_{t=2j'+1}.$$

Thus, an LR sequence $y$ could be expressed in the pixel domain as a subsequence of the following two sequences:

$$
\begin{aligned}
&[a * x] \downarrow_2 + \epsilon \cdot *[b * x] \downarrow_2 \\
&a * x(\cdot + 1)] \downarrow_2 + (\epsilon - 1) \cdot *[b * x(\cdot + 1)] \downarrow_2,
\end{aligned} \tag{3}
$$

where $a$ and $b$ are discrete versions of the convolution kernels $H$ and $H'$, respectively, and $\cdot *$ denotes the component-wise multiplication operator. Having available

the optical flow values $\epsilon_k$ for multiple LR images $y_k$, we can obtain the four components by solving an overdetermined system of linear equations:

$$[a * x] \downarrow_2, \quad [a * x(\cdot + 1)] \downarrow_2, \quad [b * x] \downarrow_2, \quad [b * x(\cdot + 1)] \downarrow_2.$$

$$(4)$$

As will be shown in Section 3, the two filters $a$ and $b$ (which are determined by the blurring kernel $H$ and its derivative $H'$) characterize the super-resolution reconstruction.

Let us next look at some examples of filters $a$ and $b$ for different blurring kernels.

**Example 1.** Consider the box-type blurring kernel $H = \frac{1}{2n} \chi_{[-n,n]}$, where $\chi_{[-n,n]}$ is the characteristic function on $[-n, n]$. Let $\epsilon(t) = \epsilon \leq 1$. Then, we have

$$y(j) = \int_{-\infty}^{\infty} H(2j - t) X(F(t)) dt$$

$$= \frac{1}{2n} \int_{2j-n}^{2j+n} X(F(t)) dt = \frac{1}{2n} \int_{2j-n}^{2j+n} X(t + \epsilon) dt.$$

Approximating the integration by quadrature rules, we obtain

$$y(j) = \frac{1}{2n} \left( \frac{1}{2} (1 - \epsilon) x(2j - n) \right.$$

$$\left. + \sum_{i=-n+1}^{n-1} x(2j - i) + \frac{1}{2} (1 + \epsilon) x(2j + n) \right).$$

Equivalently, we can write

$$y = [a * x + \epsilon(b * x)] \downarrow_2,$$

$$(5)$$

where $a$ and $b$ are the following low-pass and high-pass filters, respectively:

$$a = \frac{1}{4n} (1, 2, \ldots, 2, 1), \quad b = \frac{1}{4n} (-1, 0, \ldots, 0, 1).$$

The same observation was also shown in [4], [7], and [8].

**Example 2.** Consider a Gaussian-type blurring kernel $H$. Using the Cubic Cardinal B-spline $B(t)$ as approximation to the Gaussian function [3], we have

$$y(j) = \int_{-\infty}^{\infty} B(2j - t) X(F(t)) dt.$$

Again, by the quadrature rule, we have the approximation

$$y = \sum_i x(2j - i)(a(i) - \epsilon \cdot b(i)),$$

where

$$\begin{cases} a = \frac{1}{96} (1, 8, 23, 32, 23, 8, 1), \\ b = \frac{1}{48} (3, 12, 15, 0, -15, -12, -3). \end{cases}$$

## 3 ANALYSIS OF THE HR RECONSTRUCTION

Given multiple LR signals $y_k$ with different motions $\epsilon_k$, theoretically we can obtain two complete sequences $a * x$ and $b * x$ from (4). An interesting question arises. Without any assumption on the given finite signal $x$, can we reconstruct the signal sequence $x$ exactly from the two sequences $a * x$ and $b * x$?

To answer this question, let us write the sequence in another form, namely, as its $Z$-transform. The $Z$-transform of a signal sequence $x = \{x(i)\}$ is defined as

$$\mathbf{x}(z) = \sum_i x(i) z^{-i}.$$

It is easy to see that this transform is a one-to-one mapping between sequence space and polynomial space. Let $\mathbf{a}(z)$ and $\mathbf{b}(z)$ denote the $Z$-transforms of the filters $a$ and $b$, then the $Z$-transforms of $a * x$ and $b * x$ are $\mathbf{a}(z)\mathbf{x}(z)$ and $\mathbf{b}(z)\mathbf{x}(z)$, respectively.

Now, the question can be addressed by checking whether the polynomial equation

$$(\mathbf{a}(z)\mathbf{x}(z))\mathbf{u}(z) + (\mathbf{b}(z)\mathbf{x}(z))\mathbf{v}(z) = \mathbf{x}(z) \qquad (6)$$

is solvable for the two unknowns $\mathbf{u}(z)$ and $\mathbf{v}(z)$. Eliminating $\mathbf{x}(z)$ from both sides of (6) yields

$$\mathbf{a}(z)\mathbf{u}(z) + \mathbf{b}(z)\mathbf{v}(z) = 1. \qquad (7)$$

From the theory of Diophantine equation [19], we know the following:

**Lemma 3.** *Given two polynomials $\mathbf{a}(z)$ and $\mathbf{b}(z)$, (7) is solvable if and only if the greatest common divisor of $\mathbf{a}(z)$ and $\mathbf{b}(z)$ is a scalar, that is, if $\mathbf{a}(z)$ and $\mathbf{b}(z)$ are coprime.*

It is observed that $\mathbf{a}(z)$ and $\mathbf{b}(z)$ in our two examples (Examples 1 and 2) both have a common divisor

$$\mathbf{c}(z) = (1 + z).$$

This can be seen from the fact that $\mathbf{a}(-1) = \mathbf{b}(-1) = 0$ and, therefore, $z = -1$ is the root of both $\mathbf{a}(z)$ and $\mathbf{b}(z)$. Thus, for these blurring kernels we cannot reconstruct $\mathbf{x}(z)$ from $\mathbf{a}(z)\mathbf{x}(z)$ and $\mathbf{b}(z)\mathbf{x}(z)$ exactly. This observation is not an incident. The same holds true for general blurring kernels, as we will show next.

We follow Baker's modeling of the blurring kernel $H$ [1]. The blurring kernel (Point spread function) can be decomposed into two components:

$$H = \Omega * C,$$

where $\Omega(X)$ models the blurring caused by the optics, and $C(X)$ models the spatial integration performed by the CCD sensor. Typically, $\Omega$ is modeled by a Gaussian-type function and $C$ is modeled by a Box-type function. Notice that

$$H' = \Omega' * C.$$

Thus, we can express the corresponding discrete filters as

$$a = \ell * c; \quad b = \tau * c,$$

where $c$ is the discrete version of the spatial integration kernel $C$, and $\ell$ and $\tau$ are the discrete versions of $\Omega$ and $\Omega'$.

Since $\mathbf{a}(z)$ and $\mathbf{b}(z)$ have a common divisor $\mathbf{c}(z)$, we cannot reconstruct $\mathbf{x}(z)$ for general $\mathbf{x}(z)$, unless $C$ is a Dirac function, which generally is not true. Based on Lemma 3, we then have the following result.

**Theorem 4.** *Given multiple LR finite signals $y_k$, we cannot perfectly reconstruct the HR finite signal $x$ without any assumptions on $x$. At most, we can reconstruct $c * x$ for some low-pass filter $c$. The corresponding Z-transform $\mathbf{c}(z)$ of $c$ is the greatest common divisor of $\mathbf{a}(z)$ and $\mathbf{b}(z)$, which includes the spatial integration filter.*

Notice that $c$ is a low-pass finite impulse response (FIR) filter. To recover $x$ from $c * x$, we have to apply a high-pass filter on $c * x$ and impose some boundary condition on the signal $x$. Such a deblurring process generally is sensitive to the noise and creates artifacts in the recovered image. A good strategy then is to modify our reconstruction goal during the intermediate iterative reconstruction process: Instead of trying to reconstruct $x$, we reconstruct $c * x$ in the iterative process and we leave the recovery of $x$ from $c * x$ to the last step, after finishing the iterative reconstruction.

Thus, the modified HR signal to be reconstructed is $\widetilde{x} = c * x$. The LR sequence $\{y_k\}$ can be expressed as a subset of the following two sequences:

$$[\ell * \widetilde{x}] \downarrow_2 + \epsilon_k [\tau * \widetilde{x}] \downarrow_2 \qquad (8)$$

and

$$[\ell * \widetilde{x}(\cdot + 1)] \downarrow_2 + (\epsilon_k - 1)[\tau * \widetilde{x}(\cdot + 1)] \downarrow_2,$$

where the $Z$-transforms of $\ell$ and $\tau$ are $\mathbf{a}(z)$ and $\mathbf{b}(z)$ divided by their greatest common divisor, $\mathbf{c}(z)$, respectively. It is worth mentioning that we model the blurring procedure from HR to LR by a first-order Taylor approximation. However, our reasoning could easily be extended to a modeling by higher order Taylor approximations, leading to the same conclusion.

# 4 RECONSTRUCTION BASED ON PR FILTER BANKS

## 4.1 Introduction to PR Filter Banks

Before presenting our algorithm, we first give a brief introduction to two-channel *perfect reconstruction* (PR) filter banks, also called wavelet filter banks (see [18] for more details). A two-channel filter bank consists of two parts: an analysis filter bank and a synthesis filter bank. In our case, the signal $\widetilde{x}$ is first convolved with a low-pass filter $\ell$ and a high-pass filter $h$ and then subsampled by 2. In other words, we analyze the signal by an analysis filter bank. Then, a reconstructed signal $\widehat{x}$ is obtained by upsampling the signal by zero interpolation and then filtering it with a dual low-pass filter $g$ and a dual high-pass filter $q$. In other words, we reconstruct the signal by synthesizing the output from the analysis bank with a synthesis filter bank. See Fig. 2 for an illustration.

Such a filter bank is called a PR filter bank if $\widehat{x} = \widetilde{x}$ for any input $\widetilde{x}$. The question then is, what makes $\{\ell, h, q, g\}$ a PR filter bank. It is easy to see that the process illustrated in Fig. 2 could be expressed using $Z$-transforms as follows:

$$\widehat{\mathbf{x}}(z) = \frac{1}{2}(\widetilde{\mathbf{x}}(z) + \widetilde{\mathbf{x}}(-z)) \begin{pmatrix} \mathbf{l}(z) & \mathbf{h}(z) \\ \mathbf{l}(-z) & \mathbf{h}(-z) \end{pmatrix} \begin{pmatrix} \mathbf{g}(z) \\ \mathbf{q}(z) \end{pmatrix}.$$
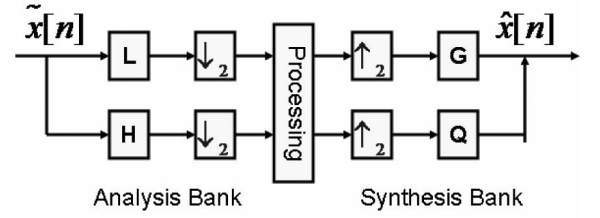


Fig. 2. The two-channel filter bank.

Thus, the sufficient and necessary condition for $\widehat{x}(z) = \widetilde{x}(z)$ is

$$\begin{pmatrix} 2z^m \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{l}(z) & \mathbf{h}(z) \\ \mathbf{l}(-z) & \mathbf{h}(-z) \end{pmatrix} \begin{pmatrix} \mathbf{g}(z) \\ \mathbf{q}(z) \end{pmatrix}$$

or, equivalently,

$$\begin{pmatrix} \mathbf{g}(z) \\ \mathbf{q}(z) \end{pmatrix} = \begin{pmatrix} \mathbf{l}(z) & \mathbf{h}(z) \\ \mathbf{l}(-z) & \mathbf{h}(-z) \end{pmatrix}^{-1} \begin{pmatrix} 2z^m \\ 0 \end{pmatrix}.$$

Then, in order to obtain the FIR filters $g$ and $q$, we require that

$$\det(\mathbf{H}(z)) = \det\left(\begin{pmatrix} \mathbf{l}(z) & \mathbf{h}(z) \\ \mathbf{l}(-z) & \mathbf{h}(-z) \end{pmatrix}\right) = cz^n.$$

In summary, the analysis filters $\{\ell, h\}$ of a perfect reconstruction filter bank have to satisfy the following condition:

$$\mathbf{l}(z)\mathbf{h}(-z) - \mathbf{l}(-z)\mathbf{h}(z) = z^m \quad \text{for some integer } m \qquad (9)$$

and the corresponding synthesis filters amount to

$$\mathbf{g}(z) = \mathbf{h}(-z); \quad \mathbf{q}(z) = -(\mathbf{l}(-z)).$$

Thus, given a low-pass filter $\mathbf{l}(z)$, we can find the corresponding high-pass filter $\mathbf{h}(z)$ such that we have an analysis filter bank and thus a PR filter bank by solving the linear system (9).

**Example 5.** For the well-known "Haar" wavelet filter bank, the synthesis and analysis filters amount to

$$\ell = \frac{1}{\sqrt{2}}(1, 1), \quad h = \frac{1}{\sqrt{2}}(1, -1);$$

$$g = \frac{1}{\sqrt{2}}(1, 1), \quad q = \frac{1}{\sqrt{2}}(-1, 1).$$

## 4.2 Iterative Reconstruction Scheme

We have made available a number of signals $y_k$ and the corresponding estimates of the optic flow values $\epsilon_k$. We also have estimates of the convolution kernels $\ell$ and $\tau$. Obviously, estimating $\ell * x$ and $\tau * x$ directly from (3) would not be wise. For reasons of numerical stability, special attention is necessary. Fortunately, the scheme of PR filter banks provides us with an iterative scheme.

Let $\ell$, which corresponds to the blurring kernel, be the low-pass filter of a PR filter bank. Then, the corresponding high-pass filter $h$ is computed by solving (9). Note, $h$ may be different from $\tau$.

Recall that, for each LR signal $y_k$, we have

$$y_k = [\ell * \widetilde{x}] \downarrow_2 + \epsilon_k \cdot * [\tau * \widetilde{x}] \downarrow_2.$$

(If the original signal is incomplete, this can be overcome by a simple interpolation.) From the above equation, $[\ell * \widetilde{x}] \downarrow_2$ is obtained as

$$[\ell * \widetilde{x}] \downarrow_2 = y_k - \epsilon_k \cdot *[\tau * \widetilde{x}] \downarrow_2 . \tag{10}$$

The process of a signal $\widetilde{x}$ passing through a PR filter bank, as shown in Fig. 2, can be expressed as

$$\widetilde{x} = g * [(\ell * \widetilde{x}) \downarrow_2] \uparrow_2 + q * [(h * \widetilde{x}) \downarrow_2] \uparrow_2 . \tag{11}$$

Combining (10) and (11), we obtain the iterative reconstruction of $\widetilde{x}$ from $K$ LR signals $y_k$ as follows: At step $n + 1$,

$$\widetilde{x}^{n+1} = q * [(h * \widetilde{x}^n) \downarrow_2] \uparrow_2$$
$$+ g * \left( \frac{1}{K} \sum_{k=1}^{K} [y_k - \epsilon_k \cdot *(\tau * \widetilde{x}^n) \downarrow_2] \uparrow_2 \right). \tag{12}$$

**Theorem 6.** *The iteration of (12) converges to the true value $x$ under the condition that*

$$\|g * \tau\| \leq \frac{1}{2}. \tag{13}$$

**Proof.** As in [14], for simplicity, we omit the downsampling and upsampling processes, as well as the fusion process. That is, we write

$$y = \ell * \widetilde{x} + \epsilon \cdot *(\tau * \widetilde{x}).$$

Then, the iteration amounts to

$$\widetilde{x}^{(n+1)} = g * \left( y - \epsilon \cdot * \left( h * \widetilde{x}^{(n)} \right) \right) + q * \left( \tau * \widetilde{x}^{(n)} \right). \tag{14}$$

Subtracting $\widetilde{x}$ on both sides of (14) yields

$$\widetilde{x}^{(n+1)} - \widetilde{x} = g * \left( y - \epsilon \cdot * \left( \tau * \widetilde{x}^{(n)} \right) \right) + q * \left( \tau * \widetilde{x}^{(n)} \right) - \widetilde{x}$$
$$= g * \left( \ell * \widetilde{x} + \epsilon \cdot * \left( \tau * \left( \widetilde{x} - \widetilde{x}^{(n)} \right) \right) \right)$$
$$+ q * \left( \tau * \widetilde{x}^{(n)} \right) - \widetilde{x}.$$

Recall that we have

$$\widetilde{x} = g * (\ell * \widetilde{x}) + q * (h * \widetilde{x}).$$

Then,

$$\widetilde{x}^{(n+1)} - \widetilde{x} = - g * \left( \epsilon \cdot * \left( \tau * \left( \widetilde{x}^{(n)} - \widetilde{x} \right) \right) \right)$$
$$+ q * h * \left( \widetilde{x}^{(n)} - \widetilde{x} \right)$$
$$= (g * (-\epsilon \cdot * \tau *) + q * h *) \left( \widetilde{x}^{(n)} - \widetilde{x} \right).$$

Let $A$ denote the operator that represents

$$g * (-\epsilon \cdot * \tau * + q * h *).$$

Then, the above equation can be rewritten as

$$\widetilde{x}^{(n+1)} - \widetilde{x} = A \left( \widetilde{x}^{(n)} - x \right).$$

Since we have (see [18] for more details)

$$\|q * h\| = \|g * \ell\| = \frac{1}{2},$$

from the fact that $\|\epsilon\|_\infty < 1$, we obtain

$$\|A\| < \|g * \tau\| + \|q * h\| \leq \|g * \tau\| + \frac{1}{2}.$$

Thus, $\|g * \tau\| \leq \frac{1}{2}$ is sufficient for the convergence of the iteration. □

### 4.3 Relationship to Other Back-Projection Methods

Applying (11), we can rewrite (12) in the form

$$\widetilde{x}^{n+1} = \widetilde{x}^n - g * [\ell * (\widetilde{x}^n) \downarrow_2] \uparrow_2$$
$$+ g * \left( \frac{1}{K} \sum_{k=1}^{K} [y_k - \epsilon_k \cdot *(\tau * \widetilde{x}^n) \downarrow_2] \uparrow_2 \right)$$
$$= \widetilde{x}^n + g * \left( \frac{1}{K} \sum_{k=1}^{K} \left[ y_k - (\ell * \widetilde{x}^n + \epsilon_k \cdot *(\tau * \widetilde{x}^n)) \downarrow_2 \right] \uparrow_2 \right).$$

From this, we can see that the iteration scheme presented here falls in the class of back-projection methods. However, it has advantages over the usual back-projection iterations. Consider the well-known method by Irani and Peleg [14]. Its iteration can be described as

$$x^{n+1} = x^n + \frac{1}{K} \sum_{k=1}^{K} T_k^{-1} (((y_k - [\ell * T_k(x^n)] \downarrow_2) \uparrow_2) * p), \tag{15}$$

where $T_k$ is the geometric transform between $y_k$ and $\widetilde{x}$ and the high-pass filter $p$ is the deblurring kernel. Notice that the two methods differ in the deblurring kernel: one uses $g$ with $g(z) = h(-z)$ defined in (9); the other uses $p$ in (15), the approximate inverse filter of $\ell$.

The requirement on $p$ in (15) is

$$\|\delta - \ell * p\| < 1, \tag{16}$$

where $\delta$ is the ideal unit impulse response filter. In other words, $p$ should be a good approximation for the inverse of $\ell$. In comparison, $g$ in our iteration only needs to be a companion filter for the smooth filter $\ell$ with sufficient decay, such that condition (13) holds. This difference makes $g$ more desirable than $p$. Let us investigate this in more detail.

The noise will be propagated exponentially as $O(\|p\|^n)$ in (15) and as $O(\|g\|^n)$ in (12). Generally, the flexibility of $g(z)$ makes it possible to design a $g$ that has much smaller norm than $p$. This leads to much better resistance to noise propagation. Here is an example.

**Example 7.** Consider $\ell = \frac{1}{4}(1, 2, 1)$. Then,

$$g = (-1/8, -1/4, 3/4, -1/4, -1/8)$$

is a dual PR filter for $\ell$ with

$$\mathbf{l}(z)\mathbf{g}(-z) = -1 + 9z^{-2} + 16z^{-3} + 9z^{-4} - z^{-6}.$$

It is easy to check that $\|g\|_2$ is around 0.85. The minimum for the norm of all filters with the same length as $g$ is around 1.1. The corresponding $p$ is

$$p = \left( \frac{1}{2}, -\frac{2}{3}, \frac{4}{3}, -\frac{2}{3}, \frac{1}{2} \right).$$

In order to make the norm of $p$ close to the norm of $g$, a lengthy $p$ with much slower decay is necessary. Such a filter

is not desirable since it causes artifacts, like the "ring" effect. This clearly indicates that our iteration scheme is more robust to noise and causes fewer artifacts in the reconstructed image.

# 5 ROBUST ALGORITHM ON 2D IMAGES WITH DENOISING

Next, we generalize the algorithm to 2D images. Then, we introduce a denoising process during the iterative reconstruction to suppress the noise in the optical flow estimation. Furthermore, the algorithm is adjusted to handle outliers.

## 5.1 Extension to 2D Images with a Built-In Denoising Process

All the previous analysis can be generalized using the tensor product. By an argument similar to the 1D case, we approximate the LR image $I^{LR}$ with the HR image $I^{HR}$ as follows:

$$I^{LR} = [(a \otimes a) * I^{HR} + u \cdot *((a \otimes b) * I^{HR}) + v \cdot *((b \otimes a) * I^{HR})] \downarrow_2,$$

where "$\otimes$" is the Kronecker tensor product and $(u, v)$ is the 2D optical flow vector. Then, the 2D analysis bank is

Low-pass filter : $\quad L = \ell \otimes \ell,$

High-pass filters : $\quad H_1 = \ell \otimes h, H_2 = h \otimes \ell, H_3 = h \otimes h,$

and the 2D synthesis filter bank is

Low-pass filter : $\quad G = g \otimes g,$

High-pass filters : $\quad Q_1 = g \otimes q, Q_2 = q \otimes g, Q_3 = q \otimes q.$

It is easy to verify that the 2D filter bank defined above is a perfect reconstruction filter bank with the analysis filter bank $\{L, H_i\}$ and the reconstruction filter bank $\{G, Q_i\}$. Then, generalizing (12), the iterative equation for the reconstruction of the HR image $\widetilde{I}$ from LR images $I_k^{LR}$ amounts to

$$\widetilde{I}^{n+1} = \sum_{i=1}^{3} Q_i * \left[ \left( H_i * \widetilde{I}^{(n)} \right) \downarrow_2 \right] \uparrow_2$$
$$+ G * \frac{1}{K} \left( \sum_{k=1}^{K} \left[ \widetilde{I}_k^{LR} - u \cdot *\left( (\ell \otimes \tau) * \widetilde{I}^n \right) \downarrow_2 \right. \right.$$
$$\left. \left. - v \cdot *\left( (\tau \otimes \ell) * \widetilde{I}^{(n)} \right) \downarrow_2 \right] \uparrow_2 \right).$$

Recall that, here, $\widetilde{I}$ is the blurred version of the true $I$ with $\widetilde{I} = (c \otimes c) * I$, where $c \otimes c$ is the spatial integration kernel. A similar iterative equation has been derived in the work of Chan et al. [6] on static imaging arrays. Here, we push the algorithm further by including the high-frequency information from other frames.

There is always noise in the estimated flow $u, v$ and the deconvolution operator could make the HR image reconstruction very sensitive to this noise. It is known that the noise variance of the solution will have hyperbolic growth when the blurring low-pass filter has zeros at high frequency. Thus, denoising is necessary to suppress the error propagation during the iterative reconstruction.

To suppress the noise, we introduce a wavelet denoising scheme, which subtracts some high-frequency components

from $\widetilde{I}^n$. Briefly, we first do a wavelet decomposition of the high-pass response, then apply a shrinkage of wavelet coefficients to the decomposition, and then reassemble the signal.

Our iteration scheme with built-in denoising operator amounts to

$$\widetilde{I}^{n+1} = \sum_{i=1}^{3} Q_i * \left[ \Psi \left( H_i * \widetilde{I}^{(n)} \right) \downarrow_2 \right] \uparrow_2$$
$$+ G * \frac{1}{K} \left( \sum_{k=1}^{K} \left[ \widetilde{I}_k^{LR} - u \cdot *\left( (\ell \otimes \tau) * \widetilde{I}^n \right) \downarrow_2 \right. \right.$$
$$\left. \left. - v \cdot *\left( (\tau \otimes \ell) * \widetilde{I}^{(n)} \right) \downarrow_2 \right] \uparrow_2 \right).$$

The denoising operator $\Psi$ defined in the equation above is

$$\Psi(H_i * \widetilde{I}^n) = G * \left[ \left( L * (H_i * \widetilde{I}^n) \right) \downarrow_2 \right] \uparrow_2$$
$$+ \sum_{i=1}^{3} \left[ Q_i * \left( \Gamma \left[ H_i * (H_i * \widetilde{I}^n) \right] \right) \downarrow_2 \right] \uparrow_2,$$

where $\Gamma$ is the threshold operator.

## 5.2 Shrinkage Operator and Robust Regression

The basic idea of wavelet denoising is to reduce the noise by shrinking the wavelet coefficients where typically most noise exists. Two popular shrinkage schemes are *hard* shrinkage and *soft* shrinkage. The hard threshold operator removes all wavelet coefficients below a threshold value $\mu$ as follows:

$$\Gamma(\nu) = \begin{cases} \nu & \text{if } |\nu| \geq \mu; \\ 0 & \text{Otherwise.} \end{cases} \quad (17)$$

The soft threshold operator shrinks the wavelet coefficients above the threshold $\mu$ and removes the wavelet coefficients below the threshold $\mu$ as follows:

$$\Gamma(\nu) = \begin{cases} \nu - \mu \cdot sgn(\nu) & \text{if } |\nu| \geq \mu; \\ 0 & \text{Otherwise.} \end{cases} \quad (18)$$

Here, we take a hybrid shrinkage approach. The hybrid threshold operator $\Gamma$ is defined as

$$\Gamma(\nu) = \begin{cases} \nu & \text{if } |\nu| \geq \mu_1; \\ sign(\nu)\mu_2 \frac{|\nu| - \mu_1}{\mu_2 - \mu_1} & \text{if } \mu_1 > \nu \geq \mu_2; \\ 0 & \text{Otherwise.} \end{cases} \quad (19)$$

This hybrid shrinkage scheme is used to remove the noise efficiently by combining both hard shrinkage and soft shrinkage. The heuristic goes as follows: First, in order to keep the edges and remove noise in image formation, we keep the large wavelet coefficients and remove the small wavelet coefficients using hard shrinkage. Then, a soft shrinkage is applied on wavelet coefficients in the middle range to reduce the noise introduced by the displacement error. Outliers will be handled by the median operator when fusing multiple frames.

In summary, our algorithm is given as follows: Given an initial HR image $I^{(0)}$, we have

$$I^{(n+1)} = G * median_k \left\{ \left[ I_k^{LR} - u_k \cdot * \left( \ell \otimes \gamma * I^{(n)} \right) \downarrow_2 \right. \right.$$
$$\left. - v_k \cdot * \left( \gamma \otimes \ell * I^{(n)} \right) \downarrow_2 \right] \uparrow_2 \right\} \tag{20}$$
$$+ \left( \sum_{i=1}^{3} Q_i * \left[ \Psi \left( H_i * I^{(n)} \right) \downarrow_2 \right] \uparrow_2 \right).$$

The algorithm above could easily be adapted to different blur filters. We only need to adjust the dual filters $G$, $Q_i$ to make a new perfect reconstruction filter bank. In addition, here we only considered a doubling of the image resolution. However, any other resolution increase could be achieved by changing the two-channel perfect reconstruction filter bank to an $M$-channel perfect reconstruction filter bank.

### 5.3  Relation to Regularization Methods

One popular denoising technique used for robust reconstruction is regularization [24]. Recall that back-projection methods basically find $\widetilde{x}$ by minimizing $\sum_{k=1}^{K} \|y_k - \widetilde{y}_k(\widetilde{x})\|_2^2$, where $\widetilde{y}_k(\widetilde{x})$ is the LR signals derived from our estimated $\widetilde{x}$. Such a least squares estimation problem usually is ill-conditioned. One way to increase the stability is to enforce a regularization term and solve

$$\min_{\widetilde{x}} \sum_{k=1}^{K} \|y_k - \widetilde{y}_k(\widetilde{x})\|_2^2 + \alpha \|\Phi(\widetilde{x})\|,$$

where $\Phi$ is some regularization function and $\alpha$ is some predefined smoothing factor. If the regularization is a least squares problem, we call it a Tikhonov-type regularization. The advantage is its simplicity and efficiency, the disadvantage is its relatively poor performance. A nonlinear diffusion regularization, like Total Variation regularization, usually performs better but is computationally expensive.

Wavelet denoising is closely related to nonlinear diffusion regularization. Mrazek et al. [20] discussed the relationship of wavelet denoising to Total Variation regularization for two simple cases. More specifically, consider a wavelet denoising scheme based on Haar wavelets ($\ell = \frac{1}{\sqrt{2}}(1,1)$, $h = \frac{1}{\sqrt{2}}(1,-1)$). Then, for the case of the threshold operator in the wavelet denoising being a *soft thresholding* operator defined as in (18), it was shown that the wavelet denoising process is equivalent to Total-Variation-based nonlinear diffusion ($\Phi(\widetilde{x}) = \|\widetilde{x}\|_1$) for a two-pixel signal.

Although Mrazek et al. [20] only showed the equivalence between Total Variation regularization and a simplistic wavelet denoising scheme for a signal with two pixels, these results still demonstrate that the wavelet denoising process in our reconstruction is comparable to some nonlinear diffusion regularization schemes in its ability to suppress the error propagation. However, it does not have the computational burden of most nonlinear diffusion regularizations, since it only needs a linear wavelet decomposition over one level. In comparison, nonlinear regularizations need to solve a nonlinear optimization.

## 6  FLOW ESTIMATION FOR SUPER-RESOLUTION

### 6.1  Basic Notations

We consider here the planar motion model. In other words, we assume that the underlying 3D structures of the interesting image regions are planar surfaces. Let $I_0, I_1, I_2, \ldots, I_K$ be the image frames in the sequence and fix frame $I_0$ be the reference image. We need to estimate the homographies between the reference frame $I_0$ and the frames $I_k$. There is the following constraint on the planar homography $P_k$ from reference frame $I_0$ to frame $I_k$:

$$P_k = R_k + \vec{v}_k \vec{n}^t, \tag{21}$$

where $R_k$ is a rotation matrix, $\vec{v}_k$ is the translation, and $\vec{n}$ is the normal of the plane. For two frames close by, the optical flow $\vec{u}$ at a point $\vec{r}$ is constrained by the brightness consistency constraint [13]

$$\left( \frac{dI}{d\vec{r}} \right)^t \vec{u}(\vec{r}) = -\frac{dI}{dt}. \tag{22}$$

Let $\vec{p}_k = vec[P_k] = (p_{1_k}, p_{2_k}, \ldots, p_{9_k})^t$ be the vectorized version of the homography $P_k$. Assume $I_k$ to be very close to $I_0$, then under the small motion assumption, the brightness consistency constraint becomes

$$\left( \frac{dI}{d\vec{r}} \right)^t (\vec{p}_k(\vec{r}) - \vec{r}) = \frac{dI}{dt}, \tag{23}$$

with

$$\vec{p}_k(\vec{r}) = \begin{pmatrix} \frac{p_1 x + p_2 y + p_3}{p_7 x + p_8 y + p_9} \\ \frac{p_4 x + p_5 y + p_6}{p_7 x + p_8 y + p_9} \end{pmatrix}.$$

$\vec{p}_k(\vec{r})$ is a 2D linear rational polynomial. Thus, multiplying the denominator of $\vec{p}_k(\vec{r})$ on both sides of (23) yields a linear homogeneous equation system on $\vec{p}_k$:

$$A_k \vec{p}_k = 0.$$

### 6.2  Multiframe Homography Estimation

We need to simultaneously estimate all homographies between the reference image $I_0$ and the frames $I_k$. The $P_k$s are not independent. They share the same plane normal. Using the expression for homographies from [22], we have the following expressions for all homographies, which share the same plane normals:

$$P_k = R_k + \vec{v}_k \vec{n}^t, \quad \text{for} \quad k = 1, \ldots, K.$$

In order to improve the estimation of the $P_k$s, this constraint on the surface normal has to be incorporated into a batch algorithm. Furthermore, we need to deal with frames $I_k$ with large displacement to the reference frame $I_0$.

Here, we take an iterative approach to estimate the homographies $P_k$. Suppose that, at the $j$th step, we are given the approximate solution $P_k^j = R_k^j + v_k^j \cdot (n^j)^t$ for the true homography $P_k$. Then, (23) can be applied to $P_k$ as follows:

$$\left( \frac{dI}{d\vec{r}} \right)^t \left( \vec{p}_k(\vec{r}) - \vec{p}_k^j(\vec{r}) \right) = I_k(\vec{r}) - I_0 \left( p_k^j(\vec{r}) \right).$$

In other words, we apply the differential brightness consistency constraint between the frame $I_0(p_k^j(\vec{r}))$ (the image obtained by warping $I_0$ from the homography $p_k^j$) and $I_k(\vec{r})$. The differential motion is due to the difference between the actual homography and its estimation in the current stage.

We write these linear equations on $\vec{p}_k$ as

$$A_k\left(P_k^j\right)\vec{p}_k = 0.$$

Then, the minimization across all homographies $P_k$ can be written as

$$\min_{R_k,\vec{n},v_k} \sum_k \left\|A_k\left(P_k^j\right)vec[R_k + \vec{v}_k\vec{n}^t]\right\|^2 \tag{24}$$

subject to the constraints that the $R_k$s are rotation matrices and $\|\vec{n}\| = 1$. This is a constrained minimization, bilinear in $R_k$, $\vec{v}_k$, and $\vec{n}$. In the remainder of this section, we show how to robustly solve the minimization (24) using an alternative two-step optimization.

Given $P_k^j$ at step $j$, we compute $P_k^{j+1} = R^{j+1} + \vec{v}_k^{j+1}(\vec{n}^{j+1})^t$ at step $j+1$.

Given $P_k^j = R_k^j + \vec{v}_k^j(\vec{n}^j)^t$, we first update $\vec{n}^{j+1}$. This minimization of (24) is just a regular least squares minimization over the sphere of $\vec{n}^{j+1}$ and can be written as

$$\min_{\vec{n}^{j+1}} \sum_k \left\|A_k\left(P_k^j\right)vec\left[R_k^j + \vec{v}_k^j(\vec{n}^{j+1})^t\right]\right\|^2 \tag{25}$$

subject to $\|\vec{n}^{j+1}\| = 1$. The minimization of (25) can easily be solved by SVD decomposition. The algorithm is given as follows: Given $A$ and $\vec{b}$, the following procedure computes a vector $\vec{n}$ such that $\|A\vec{n} - \vec{b}\|_2$ is minimum, subject to the constraint $\|\vec{n}\| = 1$. Run SVD on $A$ such that $A = U\Sigma V^t$ and save

$$V = \{v_1, v_2, \dots, v_n\}, b = U^t b, \Sigma = diag(\sigma_i).$$

Find $\lambda_*$ such that

$$\sum_i \left(\frac{\sigma_i b_i}{\sigma_i^2 + \lambda_*}\right)^2 = 1.$$

Then,

$$\vec{n} = \sum_i \left(\frac{\sigma_i b_i}{\sigma^2 + \lambda_*}\right)v_i.$$

Then, given $P_k^j$, $\vec{n}^{j+1}$, we estimate $\{R_k^{j+1}, v_k^{j+1}\}$ for $k = 1, \dots, K$. We need to solve the following minimization: For each $k$,

$$\min_{R_k^{j+1}, v_k^{j+1}} \left\|A_k\left(P_k^j\right)vec\left(R_k^{j+1} + \vec{v}_k^{j+1}(\vec{n}^{i+1})^t\right)\right\|^2,$$

subject to

$$\left(R_k^{j+1}\right)^t R_k^{j+1} = I_3.$$

This is not a trivial task. Here, we present a fast linear method to obtain an approximate solution. Let $R_k^{j+1}$ be approximated by $(I + [\omega]_\times)R_k^j$, where $[\omega]_\times$ is the skew-symmetric matrix of the rotation vector $\vec{\omega}$. Using the observation that

$$P_k^{j+1} = R_k^{j+1} + \vec{v}_k^{j+1}(\vec{n}^{j+1})^t,$$

we perform the following decomposition:

$$P_k^{j+1} = \left(I + \left[\omega_k^j\right]_\times + \Delta v_k^j \cdot \left(R_k^j n^{j+1}\right)^t\right)P_k^j,$$

with

$$v_k^{j+1} = \left(I + [\omega]_x\right)v_k^j + \left(1 + \left(R_k^j n^{j+1}\right)^t v_k^j\right)\Delta v_k^j. \tag{26}$$

Then, the minimization (26) is simplified to a standard least squares minimization on $\omega_k^i$ and $\Delta v_k^i$:

$$\min_{\omega_k^j, \Delta v_k^j}\left\|A_k\left(P_k^j\right)vec\left[\left(I + [\omega]_\times + \Delta v_k^j \cdot \left(R_k^j\vec{n}^{j+1}\right)^t\right)P_k^j\right]\right\|^2. \tag{27}$$

Thus, $R_k^{j+1}$ and $v_k^{j+1}$ can be derived from $\Delta v_k^j$ and $\omega_k^i$ after solving the least squares minimization of (27).

We adopt the procedure in [12] to compute initial values for the homographies $P_k^0$ from multiple relative motions $P_{k_1,k_2}$ between frames $I_{k_1}$ and $I_{k_2}$ related by a small displacement. Briefly, the $P_k^0$s are determined by an over-determined linear system:

$$P_{k_1,k_2}P_{k_1}^0 - P_{k_2}^0 = 0.$$

(See [12] for more details.) The iteration algorithm is given as follows: Given $P_k^j = R_k^j + \vec{v}_k^j(\vec{n}^j)^t$ at step $j$:

1. The minimization (25) is solved by SVD to obtain $\vec{n}^{j+1}$.
2. The minimization (27) is solved by least squares to obtain $\omega_k^j$, $\Delta_k^j$.
3. $\vec{v}_k^{j+1}$ is obtained from (26) and the rotation matrix is obtained as

$$R_k^{j+1} = I + \left[\omega_k^j\right]_\times + \left(1 - \left(1 - \|\omega\|^2\right)^{\frac{1}{2}}\right)\left(\left[\omega_k^j\right]_\times\right)^2.$$

The iteration is terminated when the $p_k^{j+1}$s are close enough to the $p_k^j$s.

At a quick glance, it seems that decomposing $P_k$ is an overkill since we do not need motion and structure. However, actually, in terms of computational cost, it does not make much difference. It is known that the decomposition of $P = R + \vec{t} \cdot \vec{n}^t$ is unique up to two solutions. By enforcing consistency between the $P_k$s, the decomposition becomes unique and it is not difficult to obtain this decomposition.

## 7 EXPERIMENTS AND CONCLUSION

We compared our algorithm's HR reconstruction to standard methods using both simulated and real data.

### 7.1 Simulated Data

The first experiment evaluates the wavelet-based reconstruction. We simulated four LR images ($16 \times 16$) from an HR image by shifting, blurring, and downsampling. The blurring filter is
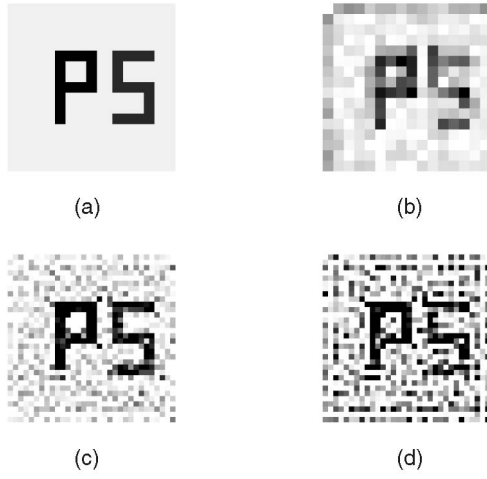
Fig. 3. The HR images (c) and (d) are reconstructed from four LR images by five iterations. (c) is reconstructed by our method. (d) is reconstructed by the back-projection method with Tikhonov regularization. The motion noise is local Gaussian noise with $\sigma = 0.2$. The image formation noise is Gaussian noise with $\gamma = 0.01$. The approximation $\hat{\ell}$ in (28) is used in the reconstruction instead of the true PSF $\ell$. (a) Original image. (b) Noisy LR image. (c) Wavelet method. (d) Tikhonov regularization.

$$\ell = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}.$$

Three kinds of noise were simulated:

1. Error in motion estimation. This error is modeled by local Gaussian white noise with parameter $\sigma$. The local covariance matrix is due to the magnitudes of the image gradients.
2. Noise in pixel formation. We added Gaussian white noise with parameter $\gamma$ to the pixel values.
3. Error in PSF modeling. We also checked how error in the PSF modeling influences the performance. The approximated PSF $\hat{\ell}$ used in the reconstruction was

$$\hat{\ell} = \frac{1}{16} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 8 & 1 \\ 1 & 1 & 1 \end{pmatrix}. \tag{28}$$

We compared our wavelet-based method to the popular "POCS" back-projection method [25] enforced by Tikhonov regularization (see Fig. 3). It may be possible that another scheme, namely Total Variation regularization, would give a bit better results. However, this would require solving a nonlinear minimization over each iterative step during the reconstruction, which is computationally expensive. In our implementation, the regularization term is the two-norm of the Laplacian smoothness constraint with roughly tuned regularization parameter $\alpha$.

Fig. 4 demonstrates the performance of the wavelet-based method for various noise settings. Performance is measured by the Signal-to-Noise ratio (SNR) of the reconstructed image to the true image, which is defined as

$$\text{SNR} = 20 \log_{10} \frac{\|x\|_2}{\|x - \hat{x}\|_2},$$
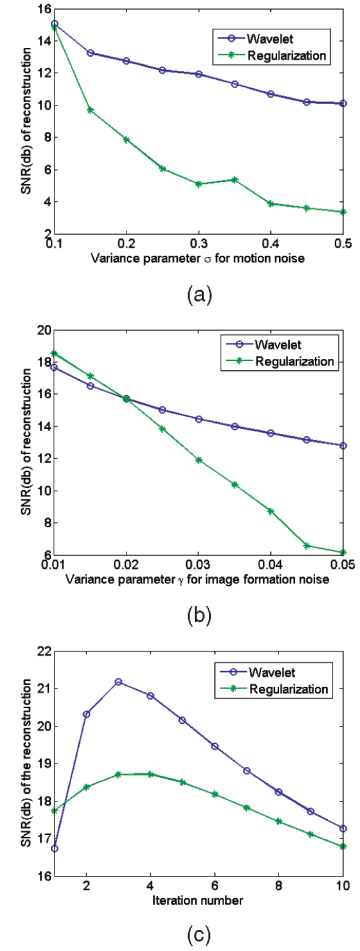


Fig. 4. Comparison between our method and POCS with Tikhonov regularization for various amounts of noise. The reconstructed image has been obtained in five iterations. The x-axis denotes the variance of the noise and the y-axis denotes the SNR of the reconstruction. (a) Comparison for motion noise. (b) Comparison for image formation noise. (c) Comparison for PSF error.

where $\hat{x}$ is the estimate for the true image $x$. Fig. 4 clearly indicates the advantage of our wavelet-based method in suppressing the noise. Especially when noise is large, the boost in performance is significant.

## 7.2 Real Data

We used an indoor sequence depicting 13 image frames of a paper box (Fig. 5). The sequence was taken with a handheld camera. An interesting planar region was chosen manually. Figs. 6 and 7 show a comparison of the results from four different methods for different regions. Here, the reconstructed HR images double the resolution of the LR images. In Figs. 6 and 7a, we upscaled all LR images using bilinear interpolation, followed by averaging all aligned images using an affine motion model. In Figs. 6 and 7b, we used the POCS method and estimated the flow field using an affine motion model. In Figs. 6 and 7c, we used the POCS method and estimated the flow field with the homography model. Figs. 6 and 7d show the results from our reconstruction scheme. The difference can be evaluated visually. As can be seen, there is an improvement from Figs. 6b to 6c and from Figs. 6c to 6d, also from Figs. 7b to 7c, and from Figs. 7c to
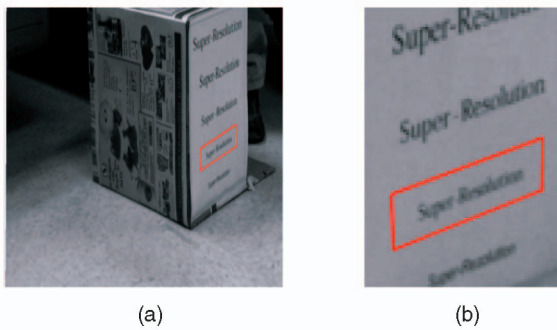
Fig. 5. Reference image frame of first indoor video and its selected region. (a) Reference frame. (b) LR planar image region.
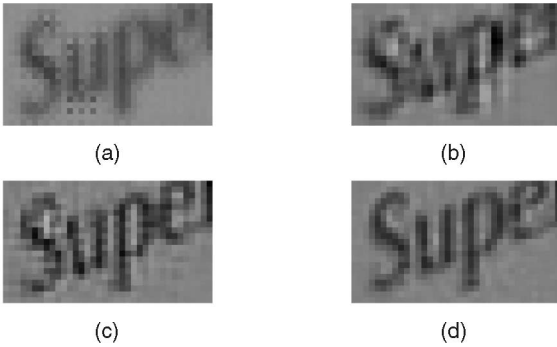


Fig. 6. Comparison of one reconstructed HR region for various methods. (a) Interpolation + affine. (b) POCS + affine. (c) POCS + homography. (d) Wavelet + homography.
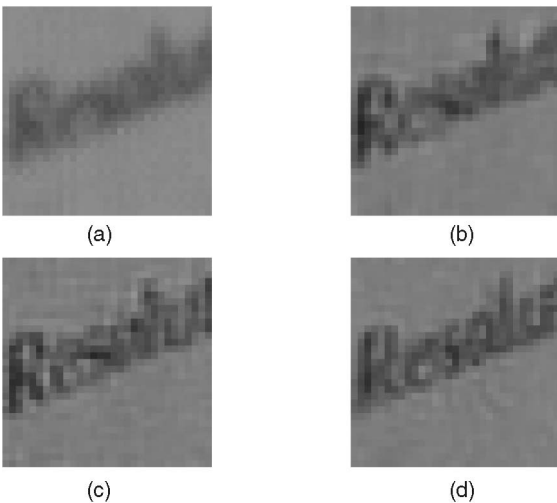


Fig. 7. Comparison of another reconstructed HR region from Fig. 5 for various methods. (a) Interpolation + affine. (b) POCS + affine. (c) POCS + homography. (d) Wavelet + homography.

7d. The letters in Figs. 6d and 7d are the clearest, and there are minimal artifacts around the edges.

A second indoor sequence depicting a box wrapped in newspaper (Fig. 8) was tested. We compared our method against Irani's method [14]. See Fig. 9 for a visual comparison. The same conclusion holds as for the previous experiment; both our homography-based motion estimation and our reconstruction method lead to improved results.

In the third indoor sequence (Fig. 10), we simultaneously captured both HR frames and LR frames of a moving box on a table. (Our digital camera averages the CCD charge in



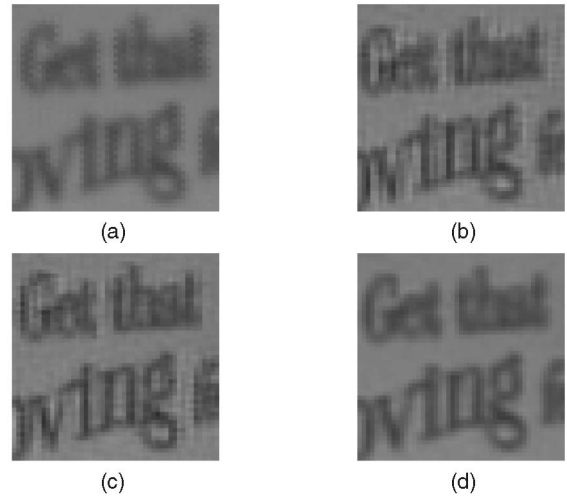Fig. 8. Reference frame from the second indoor video.



Fig. 9. Comparison of one reconstructed HR region from Fig. 8 for various methods. (a) Interpolation + affine. (b) Irani's + affine. (c) Irani's + homography. (d) Wavelet + homography.
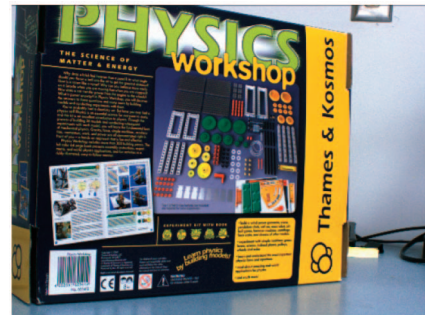


Fig. 10. Reference frame from the third indoor video.

a 2 × 2 pixel neighborhood in the HR image to obtain a pixel in the LR image.) In total, 19 frames were used in our reconstruction algorithm. It is seen that the image reconstructed from our super-resolution method can recover the lost details of LR image in most cases. Furthermore, the reconstruction HR image even has some visual improvement over the true HR image (Fig. 11).

We also used an outdoor sequence of 11 frames containing a warning sign in the scene. We compared the reconstructions from different methods for a manually selected region. Since the regions are too small to provide enough information for the homography flow model, here instead we used an affine flow model. See Fig. 12 for a visual comparison.

## 7.3 Summary

In recent years, an effort started combining geometric constraints with signal processing [11], [16], [17]. Along

(a)  (b)  (c)

Fig. 11. Comparison of various regions from Fig. 10. Images in column (a) are from the LR frames, images in column (b) are from our super-resolution method, and images in column (c) are true HR images.
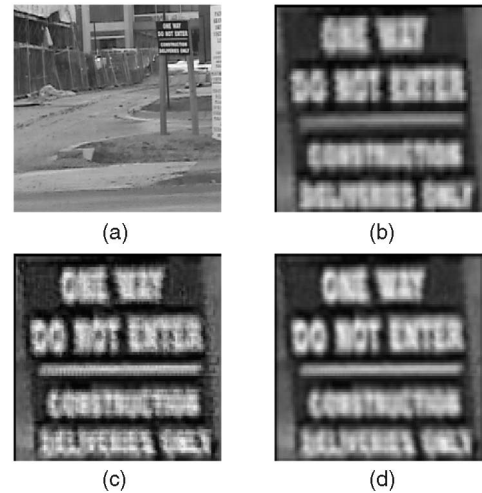


(a)  (b)

(c)  (d)

Fig. 12. (a) The key frame in the video. (b) The reconstruction from the interpolation. (c) The reconstruction from Irani's method using affine flow. (d) The reconstructed image from the wavelet method with denoising using affine flow.

these lines, we have presented an algorithm for the problem of super-resolution reconstruction. We have presented a new method for estimating the homography between multiple frames in a sequence and a new wavelet-based reconstruction algorithm. We demonstrated both in theory and experiments that the proposed method is very robust to noise without sacrificing efficiency. The reconstruction scheme allows for super-resolution reconstruction from general video sequences, even when the estimated optical flow is very noisy, and it outperforms existing methods.

## ACKNOWLEDGMENTS

## REFERENCES

[1] S. Baker and T. Kanade, "Limits on Super-Resolution and How to Break Them," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 372-379, 2000.

[2] B. Bascle, A. Blake, and A. Zisserman, "Motion Deblurring and Super-Resolution from an Image Sequence," *Proc. Fourth European Conf. Computer Vision,* vol. 2, pp. 573-582, 1996.

[3] C.D. Boor, *A Practical Guide to Splines,* first hardcover printing ed. Spring, 2001.

[4] N. Bose and K. Boo, "High-Resolution Image Recognition with Multisensors," *J. Imaging Systems and Technology,* vol. 9, pp. 294-304, 1998.

[5] A. Chambolle, R. Devore, N. Lee, and B. Lucier, "Nonlinear Wavelet Image Processing: Variational Problems, Compression and Noise Removal through Wavelets," *IEEE Trans. Image Processing,* vol. 7, 1998.

[6] R. Chan, T. Chan, L. Shen, and Z. Shen, "Wavelet Deblurring Algorithms for Spatially Varying Blur from High-Resolution Image Reconstruction," *Linear Algebra and Its Applications,* vol. 366, pp. 139-155, 2003.

[7] R. Chan, S. Riemenschneider, L. Shen, and Z. Shen, "High-Resolution Image Reconstruction with Displacement Errors: A Framelet Approach," *Int'l J. Imaging System and Technology,* vol. 14, pp. 91-104, 2004.

[8] R. Chan, S. Riemenschneider, L. Shen, and Z. Shen, "Tight Frame: An Efficient Way for High-Resolution Image Reconstruction," *Applied and Computational Harmonic Analysis,* vol. 17, pp. 91-115, 2004.

[9] M. Eland and A. Feuer, "Restoration of a Signal Super-Resolution Image from Several Blurred, Noisy and Undersampled Measured Images," *IEEE Trans. Image Processing,* pp. 1646-1658, 1997.

[10] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Robust Shift and Add Approach to Super-Resolution," *Proc. Int'l Soc. Optical Eng. (SPIE),* 2003.

[11] C. Fermüller, P. Baker, and Y. Aloimonos, "Visual Space Time Geometry: A Tool for Perception and Imagination," *Proc. IEEE,* vol. 90, no. 5, pp. 1113-1135, 2002.

[12] V.M. Govindu, "Combining Two-View Constraints for Motion Estimation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2001.

[13] B.K.P. Horn, *Robot Vision.* MacGraw-Hill, 1986.

[14] M. Irani and F. Peleg, "Motion Analysis for Image Enhancement: Resolution, Occlusion and Transparency," *J. Visual Comm. and Image Representation,* vol. 4, pp. 324-335, 1993.

[15] H. Ji and C. Fermüller, "A 3D Shape Constraint on Video," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 6, pp. 1018-1023, June 2006.

[16] A. Makadia and K. Daniilidis, "Rotation Estimation from Spherical Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, pp. 1170-1175, 2006.

[17] A. Makadia, C. Geyer, and K. Daniilidis, "Correspondenceless Structure from Motion," *Int'l J. Computer Vision,* 2007.

[18] S. Mallat, *A Wavelet Tour of Signal Processing.* Academic Press, 1999.

[19] L.J. Mondell, *Diophantine Equations.* Academic Press, 1969.

[20] P. Mrazek, J. Weickert, and G. Steidl, "Correspondences between Wavelet Shrinkage and Nonlinear Diffusion," *Proc. Scale-Space Methods in Computer Vision,* pp. 101-116, 2003.

[21] N. Nguyen and N.P. Milanfar, "A Wavelet-Based Interpolation-Restoration Method for Superresolution," *Circuits, Systems, and Signal Processing,* vol. 19, pp. 321-338, 2000.

[22] A. Shashua and S. Avidan, "The Rank 4 Constraint in Multiple ($\geq 3$) View Geometry," *Proc. Fourth European Conf. Computer Vision,* 1996.

[23] A. Tekalp, M. Ozkan, and M. Sezan, "High-Resolution Image Reconstruction from Low-Resolution Image Sequences and Space-Varying Image Restoration," *Proc. Int'l Conf. Acoustics, Speech, and Signal Processing,* pp. 169-172, 1992.

[24] J. Weickert, *Anisotropic Diffusion in Image Processing.* Teubner, 1998.

[25] C. Youla, "Generalized Image Restoration by the Method of Alternating Orthogonal Projections," *IEEE Trans. Circuits and System,* 1978.

[26] L. Zelnik-Manor and M. Irani, "Multi-Frame Estimation of Planar Motion," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, pp. 1105-1114, 2000.
[27] W. Zhao and H.S. Sawhney, "Is Super-Resolution with Optical Flow Feasible," *Proc. Seventh European Conf. Computer Vision,* pp. 599-613, 2002.
[28] A. Zomet, A. Rav-Acha, and S. Peleg, "Robust Super-Resolution," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2001.

**Hui Ji** received the BSc degree from Nanjing University, the MSc degree in mathematics from the National University of Singapore, and the PhD degree in computer science from the University of Maryland, College Park. Since 2006, he has been an assistant professor in the Department of Mathematics at the National University of Singapore. His research interests are in human and computer vision, image processing, and computational harmonic analysis.

**Cornelia Fermüller** received the MS degree in applied mathematics from the University of Technology, Graz, Austria, in 1989 and the PhD degree in computer science from the Technical University of Vienna, Vienna, in 1993. Since 1994, she has been with the Computer Vision Laboratory, Institute for Advanced Computer Studies, University of Maryland, College Park, where she is currently an associate research scientist. Her research has been in the areas of computational and biological vision centered around the interpretation of the scene geometry from multiple views. Her work is published in 35 journal articles and numerous book chapters and conference articles. Her current research focuses on visual navigation and recognition, which she studies using the tools of robotics, signal processing, and visual psychology. She is a member of the IEEE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.