Linear Super-Resolution Through Translational Motion

Rémy Abergel[1], Andrés Almansa[1], Lionel Moisan[1], Camille Noûs[2]

[1] Université Paris Cité, CNRS, MAP5, F-75006 Paris, France
[2] Laboratoire Cogitamus, https://www.cogitamus.fr/
({Remy.Abergel,Andres.Almansa,Lionel.Moisan}@u-paris.fr, Camille.Nous@cogitamus.fr)

Abstract

The problem of computing a single super-resolved image from multiple shifted images of the same scene has been addressed in several ways in the literature, with different levels of assumptions on the image to reconstruct, but a frequent core step is the resolution of the linear system derived from the image formation model. We here propose a detailed description of a direct (non-iterative) algorithm that computes the least-squares solution of the super-resolution problem in the general case of possibly non-integer super-resolution factors. We also provide a detailed analysis of the reconstruction error, both in the image and Fourier domains. In order to handle potential outliers, we build a robust (L1) variant of the algorithm, and illustrate its efficiency on synthetic and real data. We finally show how this linear and well-controlled process can be supplemented with a simple (still linear) spectral enhancement filter in order to better reveal super-resolved details without creating artifacts.

Source Code

This article has been submitted for peer-review to a scientific journal with sources codes in C language and Matlab language, including documentation, example dataset, demonstration scripts, and scripts specifically dedicated to the reproduction of most experimental results discussed in this article. This additional content will be released soon.

Keywords: image super-resolution, aliasing, spectral unfolding, image sequence, least-squares estimate

1 Introduction

1.1 Resolution and Super-resolution

Among the characteristics of an imaging system (may it be a camera phone, a microscope, a telescope, etc.), resolution is generally considered as of utmost importance, as it is intuitively related to the fineness of details that will be perceptible in the output image. The term resolution, however, is ambiguous, because depending on the context it can either refer to the number of pixels of the output image (e.g., 4096×2160 for a 4K digital camera) or to the actual resolving power of the imaging device, that is, the minimum angular distance between two points (or lines) of the scene that can be distinguished from one another in the output image. In this paper, the term "image resolution" will be used for this second acceptation, and we will use the term "image size" (or "image dimensions") to refer to the number of pixels.

In most digital cameras, the number of pixels of the image sensor is adapted to the resolution of the optical system, so that image size and image resolution are strongly linked together. Indeed, knowing the resolution of the optical system as determined by diffraction, one can compute from Shannon sampling theorem a critical (that is, maximal) pixel size for the image sensor, and deduce from the sensor area the corresponding critical (that is, minimal) image size. Replacing this ideal sensor with another one covering the same area with more pixels would increase the image size, but would certainly not change the image resolution.

Super-resolution, which has been an active research area in Image Processing for a long time, is the idea of increasing the resolution of one or several images to a level that would correspond to an optical device with a smaller critical pixel size, by means of an appropriate processing. Hence, denoising a single image, or averaging several images of the same scene to produce a single image with better signal-to-noise ratio, are both processes that increase the resolution, but they do not achieve super-resolution since they do not require the use of a finer sampling grid (that is, an increase of the image size).

Now let us distinguish two cases, depending on the number (one versus several) of low-resolution input images used to produce the high-resolution output image.

1.2 Single Image Super-resolution

Achieving super-resolution from a single image is a very appealing idea, and it has been addressed by the research community under various names: image zooming, image upscaling, or single image super-resolution (SISR). One important consideration for SISR is the nature of the input image relatively to sampling. If the available low-resolution image has been well-sampled (that is, obtained with a sensor with pixel size below the critical size), then it can be perfectly interpolated with the cardinal sine kernel (in practice, by using zero-padding in the frequency domain, or approximations with splines [67]). Of course, zooming an image with such a process is hardly a super-resolution method: since new (high-frequency) Fourier coefficients are set to zero, no real high-resolution content is created.

On the contrary, if the input low-resolution image has been undersampled relatively to its optical content, some high-frequency components are still present in the image data, but they are mixed with low-frequency ones (this is the so-called aliasing phenomenon). Unmixing these high-frequency and low-frequency coefficients is an ill-posed problem, so it is generally addressed with Bayesian priors or, equivalently, with regularization terms in variational methods. These image priors can be based on total variation [39, 3, 40, 2], on sparsity [60], on the self-similarity principle [54] or its multiscale variant [42], or on learning methods (either "internal", by means of a prior on the patches of the high-resolution image learned from the patches of the low-resolution sequence [72, 61], or "external", by means of a prior learned from a large database of high-resolution images [80]). External learning

methods recently became mainstream due to the surge of convolutional neural networks, and continue to improve the image quality of state-of-the-art SISR algorithms [18, 15, 76, 75, 77, 28, 37, 47, 48]. It is interesting to notice that these models can be used as well in the case of a well-sampled image in order to perform spectrum extrapolation (in that case, the high-frequency components are not assumed to be zero anymore but are considered as unknown in the model).

An interesting example is the so-called *face hallucination* problem, which consists in achieving SISR from small (typically 16×16 pixels) face images. This problem is generally addressed with learning methods trained on huge face databases. In favorable situations, these methods are able to generate very realistic high-resolution face images with a magnification factor up to 8 in both image directions, but, as often mentioned, age or gender inconsistencies are frequent: *"[...] due to the inherently ill-posed nature of SR problem, these methods are prone to produce unfaithful results such as face rejuvenation and gender reversal to the ground truth"* [79]. Even with the introduction of additional semantic attributes [73, 74] or image features [79] to reduce ill-posedness, the term "hallucination" continues to make sense, as many details of the computed high-resolution images, as plausible as they can be, remain arbitrary.

There are some domains, however, where an accurate high-resolution image can be reconstructed from a single low-resolution image, because the possible reconstructions are intrinsically constrained. For example, if the high-resolution image is known to be composed of a limited number of point light sources (for example, when imaging a sample with a low-density fluorescent labeling in microscopy), the reconstuction can be theoretically exact under appropriate hypotheses (see for instance [17] and references therein). Another example can be found in the area of video games, where the game engine is modified to produce a low-resolution video stream (which is much faster) and an upscaling is applied using strong priors built from the image synthesis model (and in particular texture maps) used in the initial high-resolution game engine [57, 19, 45].

Nevertheless, apart from these very constrained cases, the general goal of SISR methods is not really to try and retrieve the "true" high-resolution image, but rather to build a plausible high-resolution image compatible with the low-resolution observation. There are several situations where this objective is satisfactory, e.g. image enhancement in photography or real-time upscaling of video streams. However, many applications cannot settle such a "plausible reconstruction" objective, like medicine, forensics, surveillance, research supported by image evidence, metrology, ...

1.3 Multiple Image Super-resolution

A natural idea to avoid the ill-posedness of SISR is to acquire several low-resolution snapshots of the same scene from slightly different locations and combine them to obtain a single high-resolution image. This general framework is called *multi-image super-resolution* (MISR), and it is a very active area of research (see, e.g., [70, 33], and the recent CVPR NTIRE [10] and PROBA-V[1] challenges). If the input images are well-sampled, their information contents are essentially equivalent (up to noise), and the problem is similar to SISR with a better input signal-to-noise ratio, which leads us back to the limited ambition of producing a plausible high-resolution image. If the input images are aliased, however, high-frequency information are mixed in low-frequency components, with different weights on each image, and it is generally possible to achieve a good unmixing of these components and thus recover true high-resolution details of the observed scene.

Under reasonable hypotheses (in particular the absence of occlusions), the acquisition model for MISR can be written under the general form

$$u_0 = Au + \varepsilon,$$

[1] https://kelvins.esa.int/proba-v-super-resolution

where u_0 is the observed low-resolution image sequence, u is the (unknown) high-resolution image of the scene to be recovered, A is a linear operator which depends on the camera motion between the snapshots, and ε represent the additive noise model. Hence, MISR can be naturally formulated as an inverse problem, and many algorithms from the literature would propose variational solutions obtained from the minimization of an energy

$$E(u) = \|Au - u_0\|^2 + F(u), \tag{1}$$

where $F(u)$ is a regularization term which favors plausible reconstructions (see for instance [23, 22, 66, 65, 71, 40]). The minimization of $E(u)$ is generally addressed with iterative algorithms (gradient descent, conjugate gradient, proximal methods, ...) in reason of the very high-dimensional setting (u_0 and u typically contain millions of pixels).

1.4 The Least-squares Minimizer

In the case $F = 0$ (no regularization), the minimizer of E is simply the least-squares estimates associated to the linear system $Au = u_0$, which can be written

$$u_{\text{LS}} := \operatorname*{argmin}_u \|Au - u_0\|^2. \tag{2}$$

This minimizer is interesting for several reasons. Firstly, as it defines a linear operator $u_0 \mapsto u_{\text{LS}}(u_0)$, its behavior (and in particular its robustness to measurement errors) is easier to study and understand. Secondly, in the more general case $F \neq 0$ (regularized estimate), the minimization of (1) can be achieved with iterative two-steps algorithms, where the first step is a least-squares minimization essentially equivalent to the case $F = 0$, while the second step is independent of A but related to the regularization function F.

The complexity of the linear operator A depends on the nature of the relative motion between the frames (which can range from simple translations to affine transforms or even homographic transforms), and also on the interpolation model used to link the high-resolution image u to the low-resolution sequence u_0. In general, u_{LS} cannot be directly computed in reason of the high-dimensionality of the matrix associated to A, and iterative algorithms (or sub-optimal estimates) must be used.

Recent works on MISR [4, 36, 35] use least-squares minimization as a step in the whole process and solve (2) suboptimally with a limited number of iterations of (conjugated or not) gradient descent. In the case of Shannon interpolation, the operator $T = A^*A$ has been shown by Gröchenig and Strohmer [27] to have a Toeplitz-block structure in a Fourier basis, and the nonuniform FFT algorithm [52] provides an efficient way to compute its entries with arbitrary precision. This combined use of conjugate gradients with the Toeplitz structure of T and the nonuniform FFT is called ACT algorithm, and was successfully applied to MISR in [21, 4]. Alternatively, one can substitute the matrix A by an approximate matrix $\tilde{A}$ that uses high-order splines instead of Shannon interpolation. In that case the operator $\tilde{T} = \tilde{A}^*\tilde{A}$ becomes quite sparse in a suitable basis, and the least-squares minimizer can be estimated with an iterative algorithm like the conjugate gradient or a gradient descent. This is a popular approach for variational MISR methods in the literature [30, 40, 36], and it was recently shown in [4] to provide results almost as precise as the ACT method, provided that a sufficiently high-order spline is used for interpolation.

There is a special case, however, where these iterative or approximate algorithms can be avoided and the least-squares minimizer u_{LS} can be computed exactly, as we shall see in the next section.

1.5 Case of a Translational Motion

The MISR problem in the case of translational motion has been the topic of numerous works, see for instance [64, 23, 22, 68, 58, 9]. This pure translation case is important by itself, since some acquisition systems are built in a way that they can take bursts of images of the same scene, with a purely translational motion (or almost so). Examples can be found among push-frame satellites; for example, in the SkySat constellation from Planet [46], the translational motion model is —at least locally— quite accurate. Another example of a translational motion-based super-resolution camera is the *jitter camera* prototype described in [9]. This camera is specifically equipped with micro-actuators allowing accurate and controlled shifts of the camera sensor. Industrial realizations of similar ideas can be found among high-end digital cameras, in particular with the Hasselblad H6D-400c, which includes a shifting sensor.

The reconstruction of a high-resolution image from shifted low-resolution images has been addressed in several ways in the literature. In [9], the bursts of low-resolution images are processed with the *iterated back projection* algorithm of [31], a blockwise processing approach being used to handle in-scene motions and occlusions. In [23, 22], the shifts are considered as known (e.g., resulting from a pre-estimation step) and the high-resolution image is obtained by solving a regularized problem like (1), where the ℓ^2 square distance of the data-fidelity term $\|Au - u_0\|^2$ is replaced by the ℓ^1 distance $\|Au - u_0\|_1$ to improve the robustness to modeling errors (including errors on the shift parameters). Notice that the authors of [23] assume that the shift parameters are integer multiples of the high-resolution sampling step in order to reduce the complexity of the algorithm. Other works like [68, 58] address the problem of estimating the shift motion parameters from the stack of low-resolution images and rely on repeated computations of the least-squares estimator u_{LS} which can be considered as just one simple step in a more complex estimation process.

In the framework of a pure translational motion, and when Shannon (discrete cardinal sine) interpolation is used, the matrix A^*A is block-circulant in Fourier domain, and one can compute exactly the least-squares minimizer u_{LS}, without the need for an iterative algorithm (provided that non-periodicity issues are handled properly). The block-circulant structure of A^*A in Fourier domain was mentioned in [58] (and also in [9]) in the case of integer super-resolution factors, but we shall see later that an explicit description of the blocks can be made in the more general case of non-integer super-resolution factors.

1.6 Contributions

The aim of this paper is to provide a comprehensive study of the MISR problem in the case of translational motion and Shannon interpolation. As mentioned earlier, in that case the least-squares minimization problem (2) defines a linear operator $u_0 \mapsto u_{\text{LS}}(u_0)$ which can be formulated in Fourier domain, and computed exactly without need for an iterative algorithm. This fast algorithm can be used as such to solve (1), or as a fast step in an iterative algorithm using a regularizer F.

The paper is organized as follows. Section 2 introduces our model for the MISR problem, based on discrete cardinal sine Shannon interpolation, and presents practical ways to avoid periodization artifacts inherent to this model. In Section 3, we study the block-diagonal structure of A^*A in the Fourier basis, and show how it can be used to obtain efficient computations. In particular we prove there are no more than four different blocks in this matrix, so that preliminary computations with these four templates bring significant speed-up to the computation of u_{LS}. Section 4 builds upon the previous one, and provides exact error predictions when the measurements u_0 are affected by a zero-mean Gaussian noise of known variance. We discuss how these predictions are related to the condition number of A^*A, and show in particular that although it is often used in the literature [7, 27, 59, 12, 66], the condition number yields a very pessimistic error bound, and much better

estimates of the reconstruction error can be found. In Section 5, we present several experiments on synthetic and real data, which let us assess the quality of the least-squares estimator and the error predictions introduced in the previous section. In particular, we closely compare the Fourier transforms of the initial and reconstructed images, which is very informative (albeit notably absent in the literature). As a remedy to the well-known sensitivity to outliers of least-squares estimators, in Section 6 we present a variational frame selection algorithm, implemented with the iteratively reweighted least-squares (IRLS) algorithm. We show that it considerably increases the robustness of the classical least-squares estimator when outliers are present. The last part, Section 7, is devoted to the description of a post-processing step, which, staying in line with the linear pipeline, simply consists in a one-parameter linear deconvolution filter applied to the high-resolution reconstructed image. This post-processing filter manages to take advantage of the noise reduction effect of the least-squares estimator and significantly increases the sharpness of the reconstructed images.

2 The Least-squares Model for Multiframe Super-resolution

In this section, we derive the equations that explicitly describe the link between the high-resolution scene and the observed low-resolution image stack. This involves Shannon interpolation of discrete images, and its consequences in terms of subpixellic translation and subsampling. The important issue of implicit periodization (inherent to the discrete Fourier transform), and the apodization process required to mitigate its effects, are also discussed.

2.1 Shannon Interpolation of Discrete Images

Let M and N denote two positive integers and let $u : \Omega \to \mathbb{R}$ denote a discrete image with size $M \times N$ and discrete domain $\Omega = \{0, \dots, M-1\} \times \{0, \dots, N-1\}$. We call Shannon interpolate of u the continuous image $\mathcal{U} : \mathbb{R}^2 \to \mathbb{R}$ defined by

$$\forall (x, y) \in \mathbb{R}^2 \quad \mathcal{U}(x, y) = \sum_{(k,\ell)\in\Omega} u(k, \ell)\, \mathrm{sincd}_M(x - k)\, \mathrm{sincd}_N(y - \ell)\,, \tag{3}$$

where sincd_K denotes the discrete cardinal sine function with order K defined by

$$\forall t \in \mathbb{R} \setminus K\mathbb{Z}\,, \quad \mathrm{sincd}_K(t) = \begin{cases} \dfrac{\sin(\pi t)}{K \sin\left(\frac{\pi t}{K}\right)} & \text{if } K \text{ is odd,} \\ \dfrac{\sin(\pi t)}{K \sin\left(\frac{\pi t}{K}\right)} \cdot \cos\left(\frac{\pi t}{K}\right) & \text{if } K \text{ is even,} \end{cases} \tag{4}$$

using the continuity-preserving convention $\mathrm{sincd}_K(t) = 1$ for all $t \in K\mathbb{Z}$ (note that sincd_K is simply the K-periodization of the classical cardinal sine function $\mathrm{sinc} : t \mapsto \frac{\sin(\pi t)}{\pi t}$). Interpolation formula (3) can be directly derived from Shannon sampling theorem under the assumption that the discrete image u corresponds to the sampling of a periodic continuous image [2, 34]. However, the practical computation of (3) as a direct convolution between the image samples and the 2D-separable product of discrete cardinal sine functions is barely used since an equivalent (but numerically more efficient) computation of (3) is possible in the Fourier domain. In the following, we denote by $\widehat{u} : \mathbb{Z}^2 \to \mathbb{C}$ the discrete Fourier transform (DFT) of u, defined by

$$\forall (\alpha, \beta) \in \mathbb{Z}^2\,, \quad \widehat{u}(\alpha, \beta) = \sum_{(k,\ell)\in\Omega} u(k, \ell)\, e^{-2i\pi\left(\frac{\alpha k}{M} + \frac{\beta \ell}{N}\right)}\,. \tag{5}$$

The DFT of u is a (M,N)-periodic two-dimensional discrete complex-valued image. Besides, since u is real-valued, $\widehat{u}$ exhibits Hermitian symmetry, that is, $\widehat{u}(-\alpha,-\beta)=\widehat{u}(\alpha,\beta)^*$ for any $(\alpha,\beta)\in\mathbb{Z}^2$, where z^* denotes the complex conjugate of z. Therefore, it is quite natural and convenient to consider the period of $\widehat{u}$ centered on zero[2],

$$\widehat{\Omega}=\left[-\frac{M}{2},\frac{M}{2}\right)\times\left[-\frac{N}{2},\frac{N}{2}\right)\cap\mathbb{Z}^2\,, \tag{6}$$

which is also called the canonical discrete frequency domain associated to Ω. Notice that $\widehat{\Omega}$ is symmetrical if and only if both M and N are odd. When M is even for instance, the frequency $(-M/2,\beta)$ (with $-\frac{N}{2}\leq\beta<\frac{N}{2}$) belongs to $\widehat{\Omega}$ but its opposite $(M/2,-\beta)$ does not. In the following, we denote by $\widehat{\Omega}_s$ the symmetrized canonical frequency domain defined by

$$\widehat{\Omega}_s=\left[-\frac{M}{2},\frac{M}{2}\right]\times\left[-\frac{N}{2},\frac{N}{2}\right]\cap\mathbb{Z}^2\,. \tag{7}$$

It is shown in [2, Proposition 5] (see also [11]) that the discrete Shannon interpolate of u can be rewritten as the trigonometrical polynomial

$$\forall(x,y)\in\mathbb{R}^2\,,\quad \mathcal{U}(x,y)=\frac{1}{MN}\sum_{(\alpha,\beta)\in\widehat{\Omega}_s}\varepsilon\left(\frac{\alpha}{M}\right)\varepsilon\left(\frac{\beta}{N}\right)\widehat{u}(\alpha,\beta)\,e^{2i\pi\left(\frac{\alpha x}{M}+\frac{\beta y}{N}\right)}\,, \tag{8}$$

where the frequency weighting function ε is defined by

$$\forall\xi\in\mathbb{R}\,,\quad \varepsilon(\xi)=\begin{cases}1/2 & \text{if } |\xi|=1/2\\ 1 & \text{otherwise.}\end{cases} \tag{9}$$

In some situations, it may be more convenient to consider the complex interpolate $\mathcal{U}_c$ defined by

$$\forall(x,y)\in\mathbb{R}^2\,,\quad \mathcal{U}_c(x,y):=\frac{1}{MN}\sum_{(\alpha,\beta)\in\widehat{\Omega}}\widehat{u}(\alpha,\beta)\,e^{2i\pi\left(\frac{\alpha x}{M}+\frac{\beta y}{N}\right)}\,. \tag{10}$$

When both M and N are odd, one has $\mathcal{U}_c=\mathcal{U}$ (thus, $\mathcal{U}_c$ is real-valued in this case), while without assumption on M and N one can always write

$$\mathrm{Re}(\mathcal{U}_c(x,y))=\frac{1}{MN}\sum_{(\alpha,\beta)\in\widehat{\Omega}_s}\varepsilon'\left(\frac{\alpha}{M},\frac{\beta}{N}\right)\widehat{u}(\alpha,\beta)\,e^{2i\pi\left(\frac{\alpha x}{M}+\frac{\beta y}{N}\right)}\,,$$

where

$$\varepsilon'(\xi_1,\xi_2)=\begin{cases}1/2 & \text{if } (\xi_1,\xi_2)=\pm(1/2,1/2)\\ 0 & \text{if } (\xi_1,\xi_2)=\pm(1/2,-1/2)\\ 1 & \text{otherwise.}\end{cases}$$

Thus, $\mathrm{Re}(\mathcal{U}_c)=\mathcal{U}$ as soon as either M or N is odd. When both M and N are even, $\mathcal{U}$ is to be preferred to $\mathrm{Re}(\mathcal{U}_c)$ because it is separable and commutes with 90° rotations and horizontal/vertical axis mirroring, which is not the case for $\mathrm{Re}(\mathcal{U}_c)$. However, the difference is negligible for most images, since

$$\mathcal{U}(x,y)-\mathrm{Re}(\mathcal{U}_c(x,y))=\frac{1}{MN}\,\widehat{u}(M/2,N/2)\,\sin(\pi x)\sin(\pi y)\,,$$

[2] the choice of closing the intervals on their left-hand sides is an arbitrary convention, somehow compliant with that used in the famous GNU FFTW library [24]

and the high-frequency Fourier coefficient $\widehat{u}(M/2, N/2)$ is expected to be very small compared to most other coefficients of $\widehat{u}$ for natural images. A more thorough discussion on this subject can be found in [2] and [11].

In the present case of multiframe super-resolution using the least-squares estimator, we shall use the complex interpolation formula $\mathcal{U}_c$ (and take the real part at the end of the process) for several reasons. First, it simplifies the study, by avoiding a particular handling of the frequencies located on the border of the frequency domain. More importantly, it will allow us to design a fast estimator, which could not be similarly obtained with $\mathcal{U}$. Note that in practice, the slight approximation introduced by $\mathcal{U}_c$ (compared to the canonical choice $\mathcal{U}$) produces visually unnoticeable differences on the result images.

Another approximation, which will be discussed in Section 2.3, is required in reason of the (unrealistic) periodicity assumption made in Equation (3). For image transforms or image reconstruction purposes, the periodic plus smooth decomposition [43] can be efficiently used, but in the case of multiframe super-resolution, the use of a windowing function yields a greater inter-frame coherence.

2.2 Modeling Translational Motion and Subsampling

Let $u : \Omega \to \mathbb{R}$ denote the (unobserved) high-resolution image with size $M \times N$. Instead of observing u, we assume that we observe a stack containing L low-resolution images $u_0 = (u_0^{(1)}, \dots, u_0^{(L)})$ with discrete domain $\omega = \{0, \dots, m-1\} \times \{0, \dots, n-1\}$, each of them corresponding to the resampling of u over a translated and subsampled grid. More precisely, we assume that

$$\forall j \in \{1, \dots, L\}\,,\ \forall (k, \ell) \in \omega\,, \quad u_0^{(j)}(k, \ell) = A_j u\,(k, \ell) := \mathcal{U}_c(z_x\,(k + \delta_x^{(j)}), z_y\,(\ell + \delta_y^{(j)}))\,, \tag{11}$$

where $z_x = M/m$ and $z_y = N/n$ (with $m \leq M$, $n \leq N$) are the horizontal and vertical zoom factors, and $(\delta_x^{(j)}, \delta_y^{(j)}) \in \mathbb{R}^2$ denotes the translation vector associated to the j-th low-resolution image $u_0^{(j)}$. In the limit case $z_x = z_y = 1$, we can see from (11) that each image $u_0^{(j)}$ corresponds to the resampling of u over the shifted grid $(\delta_x^{(j)}, \delta_y^{(j)}) + \Omega$. In the case $z_x > 1$ or $z_y > 1$, the translation is combined with a subsampling with factor z_x in the horizontal direction and factor z_y in the vertical direction. By linearity of $u \mapsto \mathcal{U}_c$, the low-resolution image $u_0^{(j)}$ linearly depends on u through the linear operator $A_j : \mathbb{C}^\Omega \to \mathbb{C}^\omega$ defined in (11). Consequently, the input stack u_0 also linearly depends on u through the linear operator $A : \mathbb{C}^\Omega \to (\mathbb{C}^\omega)^L$ defined by

$$\forall u \in \mathbb{C}^\Omega\,, \quad Au = (A_1 u, \dots, A_L u)\ , \tag{12}$$

and (11) can be reformulated as the linear equation $u_0 = Au$.

2.3 Dealing with Periodization Artifacts

2.3.1 Need for Apodization

Due to the periodicity of the Shannon interpolate $\mathcal{U}$ (and its variant $\mathcal{U}_c$), the translations involved in Equation (11) lead to unrealistic values as soon as $\mathcal{U}_c$ is evaluated outside the domain $[0, M-1] \times [0, N-1]$. Indeed, for a potential high-resolution image $u \in \mathbb{R}^\Omega$, the corresponding low-resolution images $u_0^{(j)} = A_j u$ $(1 \leq j \leq L)$ are obtained with a periodicity assumption on u, so that they differ from what we could expect to be the sampling of a real-life high-resolution scene over the shifted and subsampled low-resolution grids

$$\{z_x\left(k + \delta_x^{(j)}\right),\ k = 0, \dots, m-1\} \times \{z_y\left(\ell + \delta_y^{(j)}\right),\ \ell = 0, \dots, n-1\}$$

(see Figure 1 (a)). This means that, in practice, the operator A defined in (11) does not faithfully describe (through the relation $Au = u_0$) the relation between the observed data u_0 and the latent high-resolution image u. Even if the differences are concentrated along the border of the image domain, solving the inverse problem $Au = u_0$ will unfortunately propagate and amplify these modeling errors in general, as illustrated in Figure 1 (c). In order to overcome this issue, we propose to compute from u_0 an apodized sequence of low-resolution images $\widetilde{u}_0$ that is expected to accurately correspond to the application of the operator A to an image $\widetilde{u}$, an apodized version of the latent high-resolution image. The role of this apodization is to strongly reduce the impact of the unrealistic periodicity assumption by attenuating the values of the low-resolution images near the border of the image domain, while preserving other values as much as possible. An example of such an apodized sequence is displayed in Figure 1 (b).

2.3.2 Detail of the Apodization Process

We here describe a way to build the apodized low-resolution stack $\widetilde{u}_0$ from u_0 and the sequence of displacement vectors. For that purpose, let us consider a slightly modified Tukey apodization profile $\mathcal{T}_{r,d} : [0,1] \to [0,1]$ defined by

$$\forall s \in [0,1]\,, \quad \mathcal{T}_{r,d}(s) = \begin{cases} 0 & \text{if } s \in [0,d) \\ \frac{1}{2} - \frac{1}{2} \cdot \cos\left(\frac{2\pi}{r}(s-d)\right) & \text{if } s \in \left[d, d+\frac{r}{2}\right) \\ 1 & \text{if } s \in \left[d+\frac{r}{2}, 1-d-\frac{r}{2}\right) \\ \frac{1}{2} - \frac{1}{2} \cdot \cos\left(\frac{2\pi}{r}(1-d-s)\right) & \text{if } s \in \left[1-d-\frac{r}{2}, 1-d\right) \\ 0 & \text{if } s \in [1-d,1]\ , \end{cases} \tag{13}$$

where r and d are two parameters in $(0,1)$. As illustrated in Figure 2, the parameter r can be viewed as a *smoothness* parameter that controls how fast the profile increases from zero to one or decreases from one to zero, and the parameter d controls the length of the intervals where $\mathcal{T}_{r,d}$ vanishes (the standard Tukey apodization profile corresponds to the setting $d = 0$).

Let us now consider the separable apodization function $\gamma_r : [0, M-1] \times [0, N-1] \to \mathbb{R}$ defined by

$$\forall (k,\ell) \in [0,M-1] \times [0,N-1]\,, \quad \gamma_r(k,\ell) = \mathcal{T}_{r,D_x}\left(\frac{k}{M-1}\right) \cdot \mathcal{T}_{r,D_y}\left(\frac{\ell}{N-1}\right)\ , \tag{14}$$

where

$$D_x = \max_{1 \le j \le L} \left| \frac{z_x \delta_x^{(j)}}{M-1} \right| \quad \text{and} \quad D_y = \max_{1 \le j \le L} \left| \frac{z_y \delta_y^{(j)}}{N-1} \right|$$

correspond to the horizontal and vertical fractions of the image domain that should undergo apodization since they may be affected by the implicit periodization. In all the following, we will assume that $D_x < 1$ and $D_y < 1$, which amounts to say that the maximum translation is less than the image size along both dimensions. Let $\widetilde{u}_0$ be the apodized low-resolution stack defined by

$$\forall j \in \{1,\dots,L\}\,, \quad \forall (k,\ell) \in \omega\,, \quad \widetilde{u}_0^{(j)}(k,\ell) = u_0^{(j)}(k,\ell) \cdot \gamma_r\left(z_x(k+\delta_x^{(j)}), z_y(\ell+\delta_y^{(j)})\right)\ . \tag{15}$$

Also, given any potential high-resolution image u, we shall consider the apodized high-resolution image $\widetilde{u}$ defined by

$$\forall (k,\ell) \in \Omega\,, \quad \widetilde{u}(k,\ell) = u(k,\ell) \cdot \gamma_r\,(k,\ell)\ . \tag{16}$$

By construction, $\gamma_r\,(k,\ell)$ vanishes at locations $(k,\ell) \in \Omega$ that can fall outside from the continuous high-resolution domain $[0, M-1] \times [0, N-1]$ when a high-resolution image u is shifted using the displacement vectors $(z_x\delta_x^{(j)}, z_y\delta_y^{(j)})$ involved in (11). Consequently, the apodized high-resolution image

(a) u_0^{r}: realistic low-resolution sequence

(b) $\widetilde{u}_0$: apodized sequence computed from u_0^{r} using (15)

(c) least-squares reconstruction from u_0^{r}

(d) least-squares reconstruction from $\widetilde{u}_0$

Figure 1: **Avoiding boundary artifacts when processing realistic sequences**. We display in (a) three images taken from a realistic low-resolution sequence u_0^{r} obtained by shifting and sub-sampling the underlying high-resolution image u (not represented here). Since the realistic shifting is not periodic (what goes out of the image from one side does not re-enter from the other side), this sequence does not exactly match the model $u_0^{\mathrm{r}} = Au$, which leads to undesired artifacts when the high-resolution image u is estimated from the realistic sequence u_0^{r}. We can observe in (c) that these artifacts are not strictly limited to the image borders, but propagate inside the image domain, so they cannot be discarded by a simple post-processing like cropping. We display in (b) the apodized sequence $\widetilde{u_0}$ computed from u_0^{r} using (15). The corresponding least-squares estimate, displayed in (d), does not suffer from the artifacts of (c) and provides an accurate estimate of $\widetilde{u}$, the apodized reference image defined in (16).

$\widetilde{u}$ and the apodized low-resolution sequence $\widetilde{u}_0$ also vanish at locations impacted by the unrealistic periodization effects. Finally, the apodization process leaves us with an inverse problem $A\widetilde{u} = \widetilde{u}_0$ that is much more realistic than the original model $Au = u_0$, since the apodization filter strongly reduces the undesired effects of the periodicity of $\mathcal{U}_c$. Within this framework, the calculation of u from u_0 is abandoned in favor of computing $\widetilde{u}$ from $\widetilde{u}_0$. We illustrate the benefit of this approach in Figure 1, where we can see that the least-squares estimation of u obtained from a realistic low-resolution stack u_0 suffers from strong artifacts, while the same process applied to the sequence $\widetilde{u}_0$

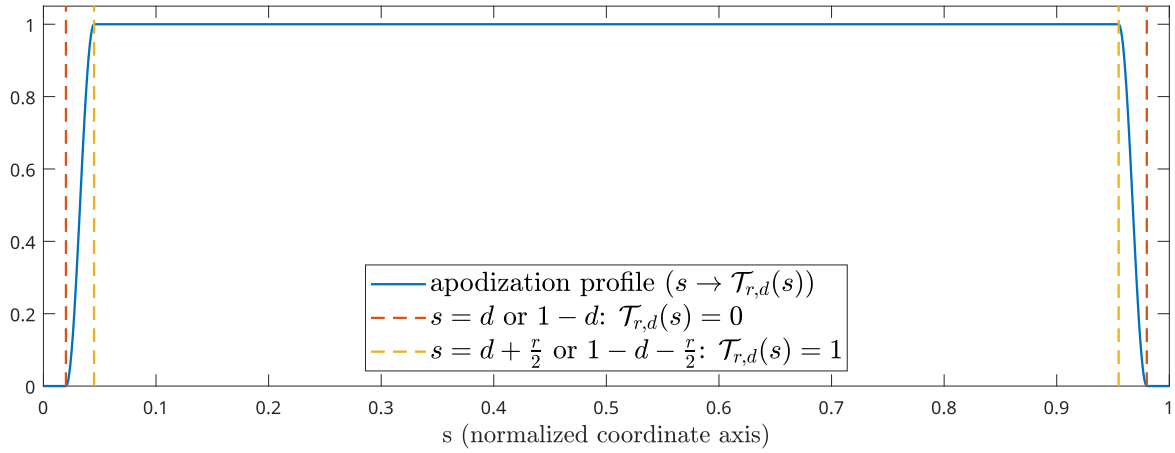


Figure 2: **Graph of the one-dimensional apodization profile** $\mathcal{T}_{r,d}$, here represented for $r = 5\%$ and $d = 2\%$. This function smoothly increases from 0 to 1 over the set $[d, d + \frac{r}{2}]$, and symmetrically, decreases from 1 to 0 over the set $[1 - d - \frac{r}{2}, 1 - d]$.

yields a satisfactory estimation of the apodized high-resolution image $\widetilde{u}$. In the following, we shall give up the $\widetilde{...}$ notation and consider that u_0 and u denote the apodized stack of low-resolution images and the associated reconstruction respectively.

2.4 Least-squares Estimator

The least-squares estimator associated to the linear equation $Au = u_0$ is defined as the unique image $u_{\text{LS}} \in \text{argmin}_{u \in \mathbb{C}^\Omega} \|Au - u_0\|_2^2$ that has minimal norm $\|u_{\text{LS}}\|$. It is given by

$$u_{\text{LS}} = (A^*A)^\dagger A^* u_0 \,, \tag{17}$$

where $A^* : (\mathbb{C}^\omega)^L \to \mathbb{C}^\Omega$ is the adjoint of the operator A, and $(A^*A)^\dagger$ the Moore-Penrose pseudo-inverse of (A^*A) [44, 50]. When A is injective, A^*A is invertible and we have $(A^*A)^\dagger = (A^*A)^{-1}$, so that $u_{\text{LS}} = (A^*A)^{-1}A^*u_0$ which is, in that case, the unique minimizer of $\|Au - u_0\|_2^2$. As we discussed in Section 2.1, the image u_{LS} defined in (17) can take complex (non real) values, in reason of the use of $\mathcal{U}_c$ instead of $\mathcal{U}$ in the definition of A. Thus, we will be in practice more interested in the use of $\text{Re}(u_{\text{LS}})$ as the final estimator.

3 Efficient Implementations in the Fourier Domain

The computation of $u_{\text{LS}} = (A^*A)^\dagger A^* u_0$ is not as straightforward as one could hope, because the matrix A^*A and its pseudo-inverse are too big to be stored in standard matrix form in a computer program. For such a high-dimensional problem, the least-squares estimator $u_{\text{LS}} = (A^*A)^\dagger A^* u_0$ is usually computed using standard optimization algorithms, such as gradient descent, conjugate gradient, proximal algorithms, etc. All these algorithms rely on iterative schemes and u_{LS} is eventually obtained when the convergence of the scheme is achieved. However, in the case of the operator A defined in (12), we will show that the computation of u_{LS} can be realized explicitly, because the dependency between the Fourier transforms of A^*u_0 and u_{LS} is block-diagonal as in [9, 58, 65].

In this section, we first rewrite the link between the high-resolution scene and the low-resolution image stack in the Fourier domain, which leads us to an explicit block-diagonal formulation of the linear operator A^*A in the Fourier domain. We finally show that the solution u_{LS} (in the least-squares sense) of the high-dimensional system $Au = u_0$ can be directly obtained by solving many

independent low-dimensional linear systems in the Fourier domain, and we provide the associated explicit equations.

3.1 Subpixellic Translation Operator and its Adjoint

Let us first describe the effect of the subpixellic translation operator (and its adjoint) in the Fourier domain. For any $\delta = (\delta_x, \delta_y) \in \mathbb{R}^2$ and any image $u \in \mathbb{C}^\Omega$, let $T_\delta u$ denote the discrete image obtained after applying to u (through the interpolate $\mathcal{U}_c$) a translation with vector δ. We have

$$\forall (k,\ell) \in \Omega\,, \quad T_\delta u(k,\ell) = \mathcal{U}_c(k-\delta_x, \ell-\delta_y)\,. \tag{18}$$

Using the explicit form of $\mathcal{U}_c$ with respect to $\widehat{u}$ given in (10), we obtain

$$\forall (k,\ell) \in \Omega\,, \quad T_\delta u(k,\ell) = \frac{1}{MN} \sum_{(\alpha,\beta)\in\widehat{\Omega}} \widehat{u}(\alpha,\beta)\, \varphi_\delta(\alpha,\beta)\, e^{2i\pi\left(\frac{\alpha k}{M}+\frac{\beta \ell}{N}\right)}\,, \tag{19}$$

$$\text{where} \qquad \varphi_\delta(\alpha,\beta) = \exp\left(-2i\pi\left(\frac{\alpha\delta_x}{M}+\frac{\beta\delta_y}{N}\right)\right) \tag{20}$$

is the so-called *ramp-phase* function. We can see in (19) that $T_\delta u$ is the inverse discrete Fourier transform (IDFT) of the image $\widehat{u}\cdot\varphi_\delta$, hence we have

$$\forall (\alpha,\beta) \in \widehat{\Omega}\,, \quad \widehat{T_\delta u}(\alpha,\beta) = \widehat{u}(\alpha,\beta)\cdot\varphi_\delta(\alpha,\beta)\,. \tag{21}$$

This means that the operator T_δ is diagonal in the Fourier basis, so its adjoint in this basis can be obtained by taking the complex conjugate of the diagonal elements, that is,

$$\widehat{T_\delta^* v}(\alpha,\beta) = \widehat{v}(\alpha,\beta)\cdot\varphi_\delta^*(\alpha,\beta) = \widehat{v}(\alpha,\beta)\cdot\varphi_{-\delta}(\alpha,\beta).$$

In other terms, we simply have $T_\delta^* = T_{-\delta}$.

3.2 Subsampling Operator with Rational Factors and its Adjoint

Now let us describe the effect of the subsampling operator and its adjoint (in the case of rational subsampling factors) in the Fourier domain. Let $z = (z_x, z_y) = \left(\frac{M}{m}, \frac{N}{n}\right) \in \mathbb{Q}^2$ and let $S_z u$ denote the discrete image obtained after applying to u (through the interpolate $\mathcal{U}_c$) a subsampling with step (z_x, z_y). We have

$$\forall (k,\ell) \in \omega\,, \quad S_z u(k,\ell) = \mathcal{U}_c(z_x\, k, z_y\, \ell)\,. \tag{22}$$

We first compute the effect of the S_z operator in the Fourier domain. As it is well known in the Signal Processing literature, sampling a continuous signal is equivalent to periodizing its Fourier Transform (this summation of *aliases* is responsible for the so-called *aliasing* phenomenon). This remains true when subsampling the Shannon interpolate of a discrete signal, provided that summation domains are restricted appropriately. As we did for Ω and $\widehat{\Omega}$, let us define $\widehat{\omega}$, the canonical discrete frequency domain associated to ω, by

$$\widehat{\omega} = \left[-\frac{m}{2}, \frac{m}{2}\right) \times \left[-\frac{n}{2}, \frac{n}{2}\right) \cap \mathbb{Z}^2\,. \tag{23}$$

Thanks to (22), and since $M = z_x m$ and $N = z_y n$, for all $(k,\ell) \in \omega$, we have

$$S_z u(k,\ell) =$$

$$\frac{1}{MN} \sum_{(\alpha,\beta)\in\widehat{\Omega}} \widehat{u}(\alpha,\beta)\, e^{2i\pi\left(\frac{\alpha z_x k}{M}+\frac{\beta z_y \ell}{N}\right)} = \frac{1}{mn} \sum_{(a,b)\in\widehat{\omega}} \left(\frac{1}{z_x z_y} \sum_{\substack{(p,q)\in\mathbb{Z}^2 \\ (a+pm,b+qn)\in\widehat{\Omega}}} \widehat{u}(a+pm, b+qn) \right) e^{2i\pi\left(\frac{ak}{m}+\frac{b\ell}{n}\right)}\,,$$

which we can equate term by term to the IDFT of $\widehat{S_z u}$ at position (k, ℓ), yielding

$$\forall (a,b) \in \widehat{\omega}\,, \quad \widehat{S_z u}(a,b) = \frac{1}{z_x z_y} \sum_{\substack{(p,q)\in\mathbb{Z}^2 \\ (a+pm,b+qn)\in\widehat{\Omega}}} \widehat{u}(a+pm, b+qn)\,. \tag{24}$$

The relation (24) means that for each frequency position $(a,b) \in \widehat{\omega}$, the DFT coefficient $\widehat{S_z u}(a,b)$ is, up to the normalization factor $z_x z_y$, the sum of all coefficients of $\widehat{u}$ whose position differs from (a,b) by a multiple (along each coordinate) of the sampling frequency. Hence, this sum is made of all coefficients of the form $\widehat{u}(a+pm, b+qn)$ (for relevant values of $(p,q) \in \mathbb{Z}^2$), including the coefficient $\widehat{u}(a,b)$ itself. Inversely, each coefficient $\widehat{u}(\alpha,\beta)$ of the DFT of the high-resolution image contributes to a single coefficient $\widehat{S_z u}(a,b)$, and we write $(a,b) = \pi_{\widehat{\omega}}(\alpha,\beta)$ the recipient of this contribution (that is, the alias of (α,β) in $\widehat{\omega}$), as stated in Proposition 1 below. The projection operator $\pi_{\widehat{\omega}}$ (called *alias function* in the following) is illustrated in Figure 3.

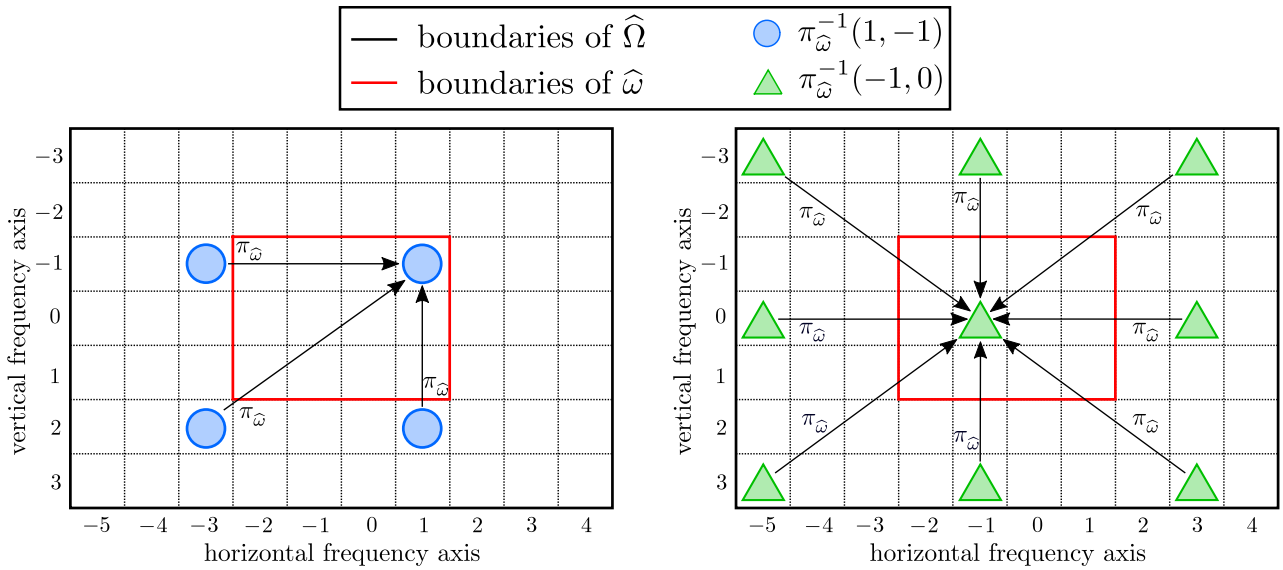


Figure 3: **Frequency aliasing with fractional aliasing factors**. We here illustrate on a very small image ($M \times N = 10 \times 7$) the effect of subsampling in the Fourier domain. *Left:* the large black rectangle ($M \times N$) delineates $\widehat{\Omega}$, the Fourier domain of a high-resolution image, u_{HR}, while the small ($m \times n = 4 \times 3$) red rectangle delineates $\widehat{\omega}$, the Fourier domain of the low-resolution image $u_{\mathrm{LR}} = S_z u_{\mathrm{HR}}$ (here we have $z = (z_x, z_y) = (\frac{10}{4}, \frac{7}{3})$). Blue points represents all frequencies of the high-resolution image that $\pi_{\widehat{\omega}}$ maps onto the low-frequency $(a,b) = (1,-1)$. In other terms, the effect of subsampling a high-resolution image u_{HR} into a low-resolution image u_{LR} would result, in the Fourier domain, in

$$\widehat{u_{\mathrm{LR}}}(1,-1) = \tfrac{1}{z_x z_y}\left(\widehat{u_{\mathrm{HR}}}(-3,-1) + \widehat{u_{\mathrm{HR}}}(1,-1) + \widehat{u_{\mathrm{HR}}}(-3,2) + \widehat{u_{\mathrm{HR}}}(1,2)\right).$$

Right: same figure, with green triangles for frequencies mapped by $\pi_{\widehat{\omega}}$ onto $(a,b) = (-1,0)$. We can see on this example that when the zoom factors z_x, z_y are not integer, the number of high-resolution frequencies that are aliased onto a given frequency (a,b) depends on (a,b) (here, 4 for $(a,b) = (1,-1)$, and 9 for $(a,b) = (-1,0)$). This number, which is denoted by $Z^{a,b}$ later in this article, is constant equal to $z_x \cdot z_y$ when z_x and z_y are integers, and can take at most four different values in the general case (see Proposition 5).

Proposition 1 (and definition of $\pi_{\widehat{\omega}}$). *For any high-resolution frequency $(\alpha,\beta) \in \widehat{\Omega}$, there exists a unique low-resolution frequency $(a,b) \in \widehat{\omega}$ such that $(\alpha,\beta) = (a+pm, b+qn)$ for some $(p,q) \in \mathbb{Z}^2$. It is called the alias of (α,β) in $\widehat{\omega}$, and we note $(a,b) = \pi_{\widehat{\omega}}(\alpha,\beta)$.*

Proof. Since $I = \mathbb{Z} \cap [-\frac{m}{2}, \frac{m}{2})$ is made of m contiguous integers, the set $\{\alpha - pm, p \in \mathbb{Z}\} \cap I$ is made of a unique integer (written a), and $p = \frac{\alpha - a}{m} \in \mathbb{Z}$. We conclude with a similar reasoning applied to (β, n) instead of (α, m). □

Remark 1. *Writing $r = x \bmod q$ the remainder $r \in \{0, 1, \ldots, q-1\}$ in the Euclidean division of $x \in \mathbb{Z}$ by $q \in \mathbb{N}^*$, we have the closed-form $\pi_{\widehat{\omega}}(\alpha, \beta) = (a, b)$, with*

$$a = \begin{cases} (\alpha \bmod m) \text{ if } (\alpha \bmod m) < m/2 \\ (\alpha \bmod m) - m \text{ otherwise} \end{cases} \quad \text{and} \quad b = \begin{cases} (\beta \bmod n) \text{ if } (\beta \bmod n) < n/2 \\ (\beta \bmod n) - n \text{ otherwise.} \end{cases}$$

Remark 2. *With the alias function $\pi_{\widehat{\omega}}$, Equation (24) can be rewritten as*

$$\forall (a, b) \in \widehat{\omega}\,, \quad \widehat{S_z u}(a, b) = \frac{1}{z_x z_y} \sum_{\substack{(\alpha,\beta)\in\widehat{\Omega} \\ \pi_{\widehat{\omega}}(\alpha,\beta)=(a,b)}} \widehat{u}(\alpha, \beta)\,. \tag{25}$$

As we shall see now, the alias function $\pi_{\widehat{\omega}}$ plays a central role in the formulation of S_z^*, the adjoint of the S_z operator.

Proposition 2. *The adjoint of the subsampling operator $S_z : \mathbb{C}^\Omega \to \mathbb{C}^\omega$ is the operator $S_z^* : \mathbb{C}^\omega \to \mathbb{C}^\Omega$ that satisfies*

$$\forall v \in \mathbb{C}^\omega\,, \quad \forall (\alpha, \beta) \in \widehat{\Omega}\,, \quad \widehat{S_z^* v}(\alpha, \beta) = \widehat{v}(\pi_{\widehat{\omega}}(\alpha, \beta))\,.$$

Proof. Let us write $\langle a, b \rangle_{\mathbb{C}^d} = \sum_{i=1}^d a_i b_i^*$ the canonical Hermitian inner product in $\mathbb{C}^d$. For any $(u, v) \in \mathbb{C}^\Omega \times \mathbb{C}^\omega$ we have, by Parseval's identity,

$$\langle S_z u, v \rangle_{\mathbb{C}^\omega} = \frac{1}{mn} \langle \widehat{S_z u}, \widehat{v} \rangle_{\mathbb{C}^{\widehat{\omega}}} = \frac{1}{mn} \sum_{(a,b)\in\widehat{\omega}} \widehat{S_z u}(a, b)\, \widehat{v}(a, b)^*\,.$$

Thanks to (25), we get

$$\begin{aligned} \langle S_z u, v \rangle_{\mathbb{C}^\omega} &= \frac{1}{mn} \sum_{(a,b)\in\widehat{\omega}} \frac{1}{z_x z_y} \sum_{\substack{(\alpha,\beta)\in\widehat{\Omega} \\ \pi_{\widehat{\omega}}(\alpha,\beta)=(a,b)}} \widehat{u}(\alpha, \beta)\, \widehat{v}(a, b)^* = \frac{1}{MN} \sum_{(\alpha,\beta)\in\widehat{\Omega}} \widehat{u}(\alpha, \beta)\, \widehat{v}(\pi_{\widehat{\omega}}(\alpha, \beta))^* \\ &= \frac{1}{MN} \langle \widehat{u}, \widehat{v} \circ \pi_{\widehat{\omega}} \rangle_{\mathbb{C}^{\widehat{\Omega}}}\,, \end{aligned}$$

but since $\langle S_z u, v \rangle_{\mathbb{C}^\omega} = \langle u, S_z^* v \rangle_{\mathbb{C}^\Omega} = \frac{1}{MN} \langle \widehat{u}, \widehat{S_z^* v} \rangle_{\mathbb{C}^{\widehat{\Omega}}}$ (again, thanks to Parseval's identity), we get by identification $\widehat{S_z^* v} = \widehat{v} \circ \pi_{\widehat{\omega}}$ as announced. □

3.3 Least-squares Operator

Using the results obtained in the two previous sections, we will now derive explicit formulations, in the Fourier domain, of A, A^*, A^*A and finally $(A^*A)^\dagger A^*$, the least-squares operator we are interested in. By definition of A in (12), one can easily check that, for any $u_0 = (u_0^{(1)}, \ldots, u_0^{(L)}) \in (\mathbb{C}^\omega)^L$ and for any $u \in \mathbb{C}^\Omega$, we have

$$A^* u_0 = \sum_{j=1}^L A_j^* u_0^{(j)} \quad \text{and} \quad A^* A u = \sum_{j=1}^L A_j^* A_j u\,. \tag{26}$$

In the following, we shall denote $z\delta^{(j)} = (z_x \delta_x^{(j)}, z_y \delta_y^{(j)})$ the j-th high-resolution translation, and adopt the following abbreviated notations,

$$T_j := T_{-z\delta^{(j)}} \quad \text{and} \quad \varphi_j := \varphi_{-z\delta^{(j)}}\,,$$

and similarly

$$T_j^* := T^*_{-z\delta^{(j)}} = T_{z\delta^{(j)}} \quad \text{and} \quad \varphi_j^* := \varphi^*_{-z\delta^{(j)}} = \varphi_{z\delta^{(j)}} \,.$$

Since each linear operator A_j defined in (11) satisfies $A_j = S_z \circ T_j$, for any $u \in \mathbb{C}^\Omega$, by combining the Fourier formulation (21) of the translation operator T_j with the Fourier formulation (25) of the subsampling operator S_z we obtain

$$\forall (a,b) \in \widehat{\omega}\,, \quad \widehat{A_j u}(a,b) = \frac{1}{z_x z_y} \sum_{(\alpha,\beta)\in\pi_{\widehat{\omega}}^{-1}(a,b)} \varphi_j(\alpha,\beta)\cdot \widehat{u}(\alpha,\beta)\,. \tag{27}$$

As for the adjoint of A_j, we have $A_j^* = T_j^* \circ S_z^*$, hence from Proposition 2 we get that

$$\forall v \in \mathbb{C}^\omega\,,\ \forall (\alpha,\beta)\in\widehat{\Omega}\,, \quad \widehat{A_j^* v}(\alpha,\beta) = \varphi_j^*(\alpha,\beta)\cdot \widehat{v}(\pi_{\widehat{\omega}}(\alpha,\beta))\,. \tag{28}$$

Then, for any $w \in \mathbb{C}^\Omega$, by combining (27) and (28), we obtain

$$\forall (\alpha,\beta)\in\widehat{\Omega}\,, \qquad \widehat{A_j^* A_j w}(\alpha,\beta) = \sum_{(\alpha',\beta')\in\pi_{\widehat{\omega}}^{-1}(\pi_{\widehat{\omega}}(\alpha,\beta))} \frac{\varphi_j^*(\alpha,\beta)\,\varphi_j(\alpha',\beta')}{z_x z_y}\cdot \widehat{w}(\alpha',\beta')\,, \tag{29}$$

and finally, by summing over j,

$$\forall (\alpha,\beta)\in\widehat{\Omega}\,, \qquad \widehat{A^*A w}(\alpha,\beta) = \sum_{(\alpha',\beta')\in\pi_{\widehat{\omega}}^{-1}(\pi_{\widehat{\omega}}(\alpha,\beta))} \left(\sum_{j=1}^{L} \frac{\varphi_j^*(\alpha,\beta)\,\varphi_j(\alpha',\beta')}{z_x z_y}\right)\cdot \widehat{w}(\alpha',\beta')\,. \tag{30}$$

We can see that there is a natural way to group these equations, by considering together all high-resolution frequencies (α,β) that are aliased to the same low-resolution frequency $\pi_{\widehat{\omega}}(\alpha,\beta)$, and hence induce the same set of summation indexes $\pi_{\widehat{\omega}}^{-1}(\pi_{\widehat{\omega}}(\alpha,\beta))$ in (30). To simplify, let us consider, for any $(a,b)\in\widehat{\omega}$,

$$\pi_{\widehat{\omega}}^{-1}(a,b) = \left\{ \left(\alpha_\ell^{a,b}, \beta_\ell^{a,b}\right) \right\}_{1\le \ell \le Z^{a,b}}, \tag{31}$$

a lexicographic indexing of the set $\pi_{\widehat{\omega}}^{-1}(a,b)$, where $Z^{a,b} = |\pi_{\widehat{\omega}}^{-1}(a,b)|$. We shall see later that $Z^{a,b}$, the number of high-resolution frequencies that are aliased to the low-resolution frequency (a,b), can only take at most four different values (depending on a and b), and is constant equal to $z_x \cdot z_y$ if z_x and z_y are integers. We now have the following

Proposition 3. *Let $w \in \mathbb{C}^\Omega$, then $v = (A^*A)w$ can be computed in the Fourier domain using the matrix equations*

$$\forall (a,b)\in\widehat{\omega}\,, \quad \begin{pmatrix} \vdots \\ \widehat{v}\left(\alpha_k^{a,b}, \beta_k^{a,b}\right) \\ \vdots \end{pmatrix}_{1\le k\le Z^{a,b}} = \mathcal{M}(a,b) \begin{pmatrix} \vdots \\ \widehat{w}\left(\alpha_\ell^{a,b}, \beta_\ell^{a,b}\right) \\ \vdots \end{pmatrix}_{1\le \ell\le Z^{a,b}}, \tag{32}$$

where $\mathcal{M}(a,b)$ is the $Z^{a,b}\times Z^{a,b}$ matrix defined by

$$\mathcal{M}_{k,\ell}(a,b) = \sum_{j=1}^{L} \frac{1}{z_x z_y} \exp\left(\frac{2i\pi(\alpha_\ell^{a,b} - \alpha_k^{a,b})\delta_x^{(j)}}{m}\right) \exp\left(\frac{2i\pi(\beta_\ell^{a,b} - \beta_k^{a,b})\delta_y^{(j)}}{n}\right). \tag{33}$$

Proof. Thanks to Equation (30), we have, for all $(a,b) \in \widehat{\omega}$,

$$\forall k \in \{1,2,\dots,Z^{a,b}\}, \quad \widehat{v}\left(\alpha_k^{a,b}, \beta_k^{a,b}\right) = \sum_{\ell=1}^{Z^{a,b}} \left(\sum_{j=1}^{L} \frac{\varphi_j^*(\alpha_k^{a,b}, \beta_k^{a,b})\, \varphi_j(\alpha_\ell^{a,b}, \beta_\ell^{a,b})}{z_x z_y} \right) \cdot \widehat{w}(\alpha_\ell^{a,b}, \beta_\ell^{a,b}),$$

which can be rewritten as (32) provided that

$$\mathcal{M}_{k,\ell}(a,b) = \frac{1}{z_x z_y} \sum_{j=1}^{L} \varphi_j^*(\alpha_k^{a,b}, \beta_k^{a,b})\, \varphi_j(\alpha_\ell^{a,b}, \beta_\ell^{a,b}).$$

Replacing $\varphi_j = \varphi_{-z\delta^{(j)}}$ and $\varphi_j^* = \varphi_{z\delta^{(j)}}$ using (20) in this formula, then using $M = z_x m$ and $N = z_y n$, we obtain Equation (33) as expected. □

Let us comment the result of Proposition 3. Indeed, it is of major importance in the present paper, since it basically shows that the original high-dimensional linear system $(A^*A)w = v$ (with $v = A^*u_0$) can be rewritten as a collection of independent low-dimensional linear systems (32) in the Fourier domain. This completely changes the perspective, as we replace the original system (with typically 10^4 to 10^5 unknowns) with small linear systems with typically 4 to 20 unknowns (the value of $Z^{a,b}$, which is roughly $z_x \cdot z_y$). Solving the original linear system would require a costly iterative algorithm, but solving each small linear system can be done quickly and exactly using standard matrix inversion. Moreover, as we shall see later in Proposition 5, these small linear systems are built with the same matrices (more precisely, there can be at most 4 different matrices), so the inversion cost is negligible.

Remark 3. *We can rewrite Equation (33) as $\mathcal{M}(a,b) = \sum_{j=1}^{L} \mathcal{M}^j(a,b)$, where*

$$\mathcal{M}^j(a,b) = \left(\exp\left(2i\pi\left(-\frac{\alpha_k^{a,b}\delta_x^{(j)}}{m} - \frac{\beta_k^{a,b}\delta_y^{(j)}}{n}\right)\right)\right)^{\mathsf{T}}_{1\le k\le Z^{a,b}} \left(\exp\left(2i\pi\left(\frac{\alpha_\ell^{a,b}\delta_x^{(j)}}{m} + \frac{\beta_\ell^{a,b}\delta_y^{(j)}}{n}\right)\right)\right)_{1\le \ell\le Z^{a,b}}$$

is a $Z^{a,b} \times Z^{a,b}$ matrix with rank one. Therefore, the matrix $\mathcal{M}(a,b)$ cannot be invertible if $L < Z^{a,b}$, and may be invertible if $L \ge Z^{a,b}$. When all matrices $\mathcal{M}(a,b)$ involved in the systems (32) are invertible, we can invert them to recover $\widehat{w}$ from $\widehat{v}$, and thus recover $w = (A^*A)^{-1}v$ from v. In particular, for $v = A^*u_0$, we obtain $w = (A^*A)^{-1}A^*u_0 = u_{\text{LS}}$. More generally, the invertibility of $\mathcal{M}(a,b)$ strongly depends on the set of displacement vectors $\{(\delta_x^{(j)}, \delta_y^{(j)})\}_{1\le j\le L}$ involved in the definition of A. However, the important issue is not the invertibility of $\mathcal{M}(a,b)$ per se (when $L \ge Z^{a,b}$, it is very likely that $\mathcal{M}(a,b)$ is invertible, and if not we can replace $(A^*A)^{-1}$ by $(A^*A)^\dagger$ anyway), but the stability of the reconstruction operator $u_0 \mapsto (A^*A)^\dagger A^*u_0$ with respect to noise. This question is partially addressed in [65, 66] and will be more thoroughly discussed in sections 4 and 5.3.

Proposition 4. *Let $v \in \mathbb{C}^\Omega$, then $w = (A^*A)^\dagger v$ can be computed in the Fourier domain using*

$$\forall (a,b) \in \widehat{\omega}, \quad \begin{pmatrix} \vdots \\ \widehat{w}\left(\alpha_\ell^{a,b}, \beta_\ell^{a,b}\right) \\ \vdots \end{pmatrix}_{1\le \ell\le Z^{a,b}} = \mathcal{M}(a,b)^\dagger \begin{pmatrix} \vdots \\ \widehat{v}\left(\alpha_k^{a,b}, \beta_k^{a,b}\right) \\ \vdots \end{pmatrix}_{1\le k\le Z^{a,b}}, \tag{34}$$

where $\mathcal{M}(a,b)^\dagger$ is the pseudo-inverse of $\mathcal{M}(a,b)$. In particular, when choosing $v = A^*u_0$, we obtain $w = (A^*A)^\dagger A^*u_0 = u_{\text{LS}}$.

Proof. The proof is given in Appendix A.

3.4 Implementation of the Least-squares Estimator

Let us finally describe how we can efficiently implement the computation of $u_{\text{LS}} = (A^*A)^\dagger A^* u_0$. Thanks to Proposition 4, we can obtain the DFT of the desired image u_{LS} by computing, for each $(a,b) \in \widehat{\omega}$, the matrix $\mathcal{M}(a,b)$, its pseudo-inverse $\mathcal{M}(a,b)^\dagger$ and the matrix-vector product involved in the right-hand side of (34). This way we obtain $\hat{w}(\alpha,\beta) = \widehat{u_{\text{LS}}}(\alpha,\beta)$ for all $(\alpha,\beta) \in \pi_{\widehat{\omega}}^{-1}(a,b)$, but in order to obtain all coefficients $\hat{w}(\alpha,\beta)$ we need to repeat this pseudo-inverse computation for each $(a,b) \in \widehat{\omega}$. In practice, the number of such pseudo-inverse computations can be greatly reduced: we shall see in this section that with an appropriate indexing of the set $\pi_{\widehat{\omega}}^{-1}(a,b)$ (see earlier remark after Equation (31)), the number of distinct matrices $\mathcal{M}(a,b)$ for $(a,b) \in \widehat{\omega}$ drops to 4 in the general case of rational zoom factors, and to only one in the particular case of integer zoom factors.

Let us first examine more closely the structure of the sets $\pi_{\widehat{\omega}}^{-1}(a,b)$. In all the following, $\lfloor t \rfloor$ and $\lceil t \rceil$ respectively denote the lower integer part and the upper integer part of a real number t.

Lemma 1. *For all $(a,b) \in \widehat{\omega}$, we have*

$$\pi_{\widehat{\omega}}^{-1}(a,b) = \Big(a + mP(a)\Big) \times \Big(b + nQ(b)\Big)\,, \tag{35}$$

where

$$P(a) = \big[p_{\min}(a), p_{\max}(a)\big] \cap \mathbb{Z}\,, \qquad Q(b) = \big[q_{\min}(b), q_{\max}(b)\big] \cap \mathbb{Z}\,, \tag{36}$$

and

$$\begin{aligned} p_{\min}(a) &= \left\lceil -\frac{M}{2m} - \frac{a}{m} \right\rceil\,, \quad p_{\max}(a) = \left\lceil \frac{M}{2m} - \frac{a}{m} \right\rceil - 1\,, \\ q_{\min}(b) &= \left\lceil -\frac{N}{2n} - \frac{b}{n} \right\rceil\,, \quad q_{\max}(b) = \left\lceil \frac{N}{2n} - \frac{b}{n} \right\rceil - 1\,. \end{aligned} \tag{37}$$

Proof. The proof is given in Appendix B. □

Thanks to Lemma 1, we can now explicitly define the lexicographic ordering of the sets $\pi_{\widehat{\omega}}^{-1}(a,b)$ mentioned in Equation (31).

Definition 1. *For all $(a,b) \in \widehat{\omega}$, the lexicographic indexing of the set $\pi_{\widehat{\omega}}^{-1}(a,b)$ is defined by*

$$\begin{aligned} &\pi_{\widehat{\omega}}^{-1}(a,b) = \Big\{(\alpha_\ell^{a,b}, \beta_\ell^{a,b})\Big\}_{1 \le \ell \le Z^{a,b}} \qquad \text{with} \qquad \alpha_\ell^{a,b} = a + p_\ell m\,, \qquad \beta_\ell^{a,b} = b + q_\ell n, \\ &\text{and} \qquad p_\ell = p_{\min}(a) + \left\lfloor \frac{\ell - 1}{|Q(b)|} \right\rfloor\,, \qquad q_\ell = q_{\min}(b) + \big((\ell - 1) \bmod |Q(b)|\big)\,. \end{aligned} \tag{38}$$

Lemma 2. *The matrix $\mathcal{M}(a,b)$ only depends on $|P(a)|$ and $|Q(b)|$.*

Proof. Combining (33) with (38), we obtain, for all $(k,\ell) \in \{1,2,\dots,Z^{a,b}\}^2$,

$$\mathcal{M}_{k,\ell}(a,b) = \frac{1}{z_x z_y} \sum_{j=1}^{L} e^{2i\pi\left(\left\lfloor \frac{\ell-1}{|Q(b)|} \right\rfloor - \left\lfloor \frac{k-1}{|Q(b)|} \right\rfloor\right)\delta_x^{(j)} + 2i\pi\Big(\big((\ell-1) \bmod |Q(b)|\big) - \big((k-1) \bmod |Q(b)|\big)\Big)\delta_y^{(j)}}\,. \tag{39}$$

Hence, the coefficients of the matrix $\mathcal{M}(a,b)$ only depend on $|Q(b)|$ and $|P(a)|$, and so does its order $Z^{a,b} = |P(a)| \cdot |Q(b)|$ (that is, $\mathcal{M}(a,b)$ is a square matrix with size $Z^{a,b} \times Z^{a,b}$). □

Proposition 5. *The order $Z^{a,b}$ of the square matrix $\mathcal{M}(a,b)$ is, depending on $(a,b) \in \widehat{\omega}$, one of the four values $\lfloor z_x \rfloor \cdot \lfloor z_y \rfloor$, $\lfloor z_x \rfloor \cdot \lceil z_y \rceil$, $\lceil z_x \rceil \cdot \lfloor z_y \rfloor$, and $\lceil z_x \rceil \cdot \lceil z_y \rceil$. Moreover, the number of distinct matrices $\mathcal{M}(a,b)$ is*

- *1 if both zoom factors z_x and z_y are integers;*
- *2 if one of the zoom factors z_x and z_y is an integer and the other is not;*
- *4 if none of the zoom factors z_x and z_y is an integer.*

Proof. The set $P(a)$ is the intersection of the one-dimensional lattice $a + m\mathbb{Z}$ (with step m) and the set of M consecutive integers $[-\frac{M}{2}, \frac{M}{2}) \cap \mathbb{Z}$. According to the value of a, this intersection will contain $\lfloor \frac{M}{m} \rfloor$ or $\lceil \frac{M}{m} \rceil$ elements, that is, $\lfloor z_x \rfloor$ or $\lceil z_x \rceil$ since $z_x = \frac{M}{m}$. Likewise, the possibilities for $|Q(b)|$ are $\lfloor z_y \rfloor$ and $\lceil z_y \rceil$. The order of $\mathcal{M}(a,b)$ being $Z^{a,b} = |P(a)| \cdot |Q(b)|$, we hence obtain at most 4 different orders. Lastly, since all the sets $\pi_{\widehat{\omega}}^{-1}(a,b)$ are indexed in lexicographic order, we know from Lemma 2 that each matrix $\mathcal{M}(a,b)$ only depends on $|P(a)|$ and $|Q(b)|$, which proves the second part of the proposition. □

A toy example is illustrated in Figure 3 where $(m,n) = (4,3)$ and $(M,N) = (10,7)$. In that case, we have $z_x = \frac{10}{4} = 2.5$, $z_y = \frac{7}{3} \simeq 2.33$, so we will obtain 4 different matrices $\mathcal{M}(a,b)$: one with order 4, two with order 6, and one with order 9.

We can consider another example where $(m,n) = (3,3)$ and $(M,N) = (8,9)$. In that case we have $z_x = \frac{8}{3} \simeq 2.667$ and $z_y = \frac{9}{3} = 3$, so we will obtain 2 different matrices $\mathcal{M}(a,b)$: one with order 9, and one with order 6.

We end this section with the explicit description of two algorithms: Algorithm 1, which simply computes the block matrix $\mathcal{M}(a,b)$, and Algorithm 2, which computes the super-resolved image u_{LS} from a sequence of low-resolution images.

Algorithm 1: Computation of a block matrix (routine `blockmatrix`)

Input: *a sequence of displacements* $\delta = \left((\delta_x^{(j)}, \delta_y^{(j)})\right)_{1 \le j \le L}$, *a normalization coefficient* γ *and two integers* `size_p` *and* `size_q`.

Output: *a square complex matrix* $\mathcal{M}$ *with size* $Z \times Z$, *denoting* $Z =$ `size_p` $\cdot$ `size_q`.

Initialization: *set all entries of* $\mathcal{M}$ *to* 0.

foreach $(j,k,\ell) \in \{1,\dots,L\} \times \{1,\dots,Z\} \times \{1,\dots,Z\}$ **do**

$\mu_x \longleftarrow \lfloor (\ell - 1)/\texttt{size\_q} \rfloor - \lfloor (k-1)/\texttt{size\_q} \rfloor$

$\mu_y \longleftarrow ((\ell - 1) \bmod \texttt{size\_q}) - ((k-1) \bmod \texttt{size\_q})$

$\mathcal{M}_{k,\ell} \longleftarrow \mathcal{M}_{k,\ell} + \gamma \cdot \exp\left(2i\pi \left(\mu_x \, \delta_x^{(j)} + \mu_y \, \delta_y^{(j)}\right)\right)$

return $\mathcal{M}$

Algorithm 2: Efficient implementation of the least-squares estimator

Input: *the stack* $u_0 = (u_0^{(1)}, \dots, u_0^{(L)}) \in (\mathbb{R}^\omega)^L$ *of* L *low-resolution images with size* $m \times n$, *the sequence of displacements* $\delta = ((\delta_x^{(j)}, \delta_y^{(j)}))_{1 \le j \le L}$ *and two integers* M *and* N *defining the super-resolution factors* $z_x = M/m$ *and* $z_y = N/n$.

Output: *the real part of the image* $u_{\text{LS}} = (A^*A)^\dagger A^* u_0 \in \mathbb{C}^\Omega$ *defined in* (17).

Core of the module:

compute $\widehat{v} = \widehat{A^* u_0}$ *using* (26) *and* (28)
compute $\mathcal{M}_1 = \texttt{blockmatrix}(\delta, 1/(z_x z_y), \lfloor z_x \rfloor, \lfloor z_y \rfloor)$ *using Algorithm 1*
compute $\mathcal{M}_2 = \texttt{blockmatrix}(\delta, 1/(z_x z_y), \lceil z_x \rceil, \lfloor z_y \rfloor)$ *using Algorithm 1*
compute $\mathcal{M}_3 = \texttt{blockmatrix}(\delta, 1/(z_x z_y), \lfloor z_x \rfloor, \lceil z_y \rceil)$ *using Algorithm 1*
compute $\mathcal{M}_4 = \texttt{blockmatrix}(\delta, 1/(z_x z_y), \lceil z_x \rceil, \lceil z_y \rceil)$ *using Algorithm 1*
compute the pseudo-inverses matrices $\mathcal{M}_1^\dagger$, $\mathcal{M}_2^\dagger$, $\mathcal{M}_3^\dagger$ *and* $\mathcal{M}_4^\dagger$
foreach $(a, b) \in \widehat{\omega}$ **do**

$$i \longleftarrow \begin{cases} 1 & \text{if } |P(a)| = \lfloor z_x \rfloor \text{ and } |Q(b)| = \lfloor z_y \rfloor \\ 2 & \text{if } |P(a)| = \lceil z_x \rceil \text{ and } |Q(b)| = \lfloor z_y \rfloor \\ 3 & \text{if } |P(a)| = \lfloor z_x \rfloor \text{ and } |Q(b)| = \lceil z_y \rceil \\ 4 & \text{if } |P(a)| = \lceil z_x \rceil \text{ and } |Q(b)| = \lceil z_y \rceil \end{cases}$$

compute the values of $\widehat{u_{\text{LS}}}$ *at frequency locations* $\{(\alpha_\ell^{a,b}, \beta_\ell^{a,b})\}_{1 \le \ell \le Z^{a,b}}$ *(given by* (38)*) using*

$$\begin{pmatrix} \vdots \\ \widehat{u_{\text{LS}}}(\alpha_\ell^{a,b}, \beta_\ell^{a,b}) \\ \vdots \end{pmatrix}_{1 \le \ell \le Z^{a,b}} = \mathcal{M}_i^\dagger \begin{pmatrix} \vdots \\ \widehat{v}(\alpha_k^{a,b}, \beta_k^{a,b}) \\ \vdots \end{pmatrix}_{1 \le k \le Z^{a,b}} .$$

return the real part of IDFT($\widehat{u_{\text{LS}}}$)

Remark that when z_x is integer, we have $\mathcal{M}_1 = \mathcal{M}_2$ and $\mathcal{M}_3 = \mathcal{M}_4$. Likewise, when z_y is integer, we have $\mathcal{M}_1 = \mathcal{M}_3$ and $\mathcal{M}_2 = \mathcal{M}_4$. Finally, when z_x and z_y are both integers, we have $\mathcal{M}_1 = \mathcal{M}_2 = \mathcal{M}_3 = \mathcal{M}_4$ which simplifies the algorithm.

4 Prediction of the Reconstruction Quality

A benefit of the least-squares approach we proposed is that we are able to predict the quality of the reconstruction through the analysis of the reconstruction error. In this section, we provide several estimates and discuss their usefulness.

4.1 Variance Maps in the Fourier or Image Domain

Proposition 6. *Assume that the observed sequence u_0 is corrupted with a white Gaussian noise with variance σ^2, that is, u_0 is a realization of the random image sequence $U_0 = \overline{u_0} + \sigma\varepsilon$ where $\varepsilon = (\varepsilon^{(1)}, \dots, \varepsilon^{(L)}) \in (\mathbb{R}^\omega)^L$ is an image sequence made of i.i.d. $\mathcal{N}(0,1)$ random variables. Then, the reconstruction $u_{\text{LS}} = (A^*A)^\dagger A^* u_0$ is a realization of the random image*

$$U_{\text{LS}} = \overline{u_{\text{LS}}} + \sigma\varepsilon',$$

where $\overline{u_{\text{LS}}} = (A^*A)^\dagger A^* \overline{u_0}$ is the reconstruction that would be obtained without noise and ε' is a Gaussian noise with zero mean which satisfies

$$\forall (\alpha,\beta) \in \widehat{\Omega}\,, \qquad \sqrt{\frac{\mathbb{E}|\widehat{\varepsilon'}(\alpha,\beta)|^2}{MN}} = \mathcal{A}(\alpha,\beta), \tag{40}$$

where for all $(a,b) \in \widehat{\omega}$ and $1 \le \ell \le Z^{a,b}$,

$$\mathcal{A}(\alpha_\ell^{a,b}, \beta_\ell^{a,b}) := \sqrt{\frac{1}{z_x z_y} \sum_{j=1}^{L} |c_\ell^{(j)}(a,b)|^2} \tag{41}$$

$$\textit{and} \qquad c_\ell^{(j)}(a,b) = \sum_{k=1}^{Z^{a,b}} \left[\mathcal{M}(a,b)^\dagger\right]_{\ell,k} \exp\left(-2i\pi\left(\frac{\alpha_k^{a,b}\delta_x^{(j)}}{m} + \frac{\beta_k^{a,b}\delta_y^{(j)}}{n}\right)\right). \tag{42}$$

Proof. By linearity, we have

$$U_{\text{LS}} = (A^*A)^\dagger A^* U_0 = (A^*A)^\dagger A^*(\overline{u_0} + \sigma\varepsilon) = \overline{u_{\text{LS}}} + \sigma\varepsilon',$$

so the reconstruction error $\varepsilon' = (A^*A)^\dagger A^*\varepsilon$ is a Gaussian vector with zero mean. From Equation (28) we have

$$\forall j \in \{1,\dots,L\},\ \forall (\alpha,\beta) \in \widehat{\Omega}\,, \qquad \widehat{A_j^*\varepsilon^{(j)}}(\alpha,\beta) = \varphi_j^*(\alpha,\beta) \cdot \widehat{\varepsilon}^{(j)}(\pi_{\widehat{\omega}}(\alpha,\beta))$$

with, according to Equation (20) and the relation $(M,N) = (z_x m, z_y n)$,

$$\varphi_j^*(\alpha,\beta) = \varphi_{z\delta^{(j)}}(\alpha,\beta) = \exp\left(-2i\pi\left(\frac{\alpha\delta_x^{(j)}}{m} + \frac{\beta\delta_y^{(j)}}{n}\right)\right).$$

Hence, for all $(a,b) \in \widehat{\omega}$ and $1 \le k \le Z^{a,b}$,

$$\widehat{A^*\varepsilon}(\alpha_k^{a,b}, \beta_k^{a,b}) = \sum_{j=1}^{L} \widehat{A_j^*\varepsilon^{(j)}}(\alpha_k^{a,b}, \beta_k^{a,b}) = \sum_{j=1}^{L} \varphi_j^*(\alpha_k^{a,b}, \beta_k^{a,b}) \cdot \widehat{\varepsilon}^{(j)}(a,b).$$

Furthermore, since $\varepsilon' = (A^*A)^\dagger A^*\varepsilon$, we obtain from Proposition 4

$$\begin{aligned}
\forall \ell \in \{1,2,\dots,Z^{a,b}\}, \quad \widehat{\varepsilon'}(\alpha_\ell^{a,b},\beta_\ell^{a,b}) &= \sum_{k=1}^{Z^{a,b}} \left[\mathcal{M}(a,b)^\dagger\right]_{\ell,k} \widehat{A^*\varepsilon}(\alpha_k^{a,b},\beta_k^{a,b}) \\
&= \sum_{k=1}^{Z^{a,b}} \left[\mathcal{M}(a,b)^\dagger\right]_{\ell,k} \sum_{j=1}^{L} \varphi_j^*(\alpha_k^{a,b},\beta_k^{a,b}) \cdot \widehat{\varepsilon}^{(j)}(a,b) \\
&= \sum_{j=1}^{L} \left(\sum_{k=1}^{Z^{a,b}} \left[\mathcal{M}(a,b)^\dagger\right]_{\ell,k} \varphi_j^*(\alpha_k^{a,b},\beta_k^{a,b}) \right) \cdot \widehat{\varepsilon}^{(j)}(a,b) \\
&= \sum_{j=1}^{L} c_\ell^{(j)}(a,b) \cdot \widehat{\varepsilon}^{(j)}(a,b) \qquad (43)
\end{aligned}$$

where $c_\ell^{(j)}(a,b)$ is defined in Equation (42). Now let us write, for a, b and j fixed, $X = \mathrm{Re}(\widehat{\varepsilon}^{(j)}(a,b))$ and $Y = \mathrm{Im}(\widehat{\varepsilon}^{(j)}(a,b))$. Since $\varepsilon^{(j)}$ is a $m \times n$ Gaussian white noise image (with variance 1), the random variables X and Y are independent, Gaussian-distributed with zero mean and satisfy

$$\mathbb{E}X^2 + \mathbb{E}Y^2 = mn$$

(the repartition is $\mathbb{E}X^2 = \mathbb{E}Y^2 = \frac{mn}{2}$ for most coefficients, but $\mathbb{E}X^2 = mn$ and $\mathbb{E}Y^2 = 0$ for some of them, in the case $a = b = 0$ for example). Consequently, for any $(\alpha,\beta) \in \mathbb{R}^2$, we have

$$\mathbb{E}|(\alpha+i\beta)(X+iY)|^2 = \mathbb{E}(\alpha X - \beta Y)^2 + \mathbb{E}(\alpha Y + \beta X)^2 = (\alpha^2+\beta^2)(\mathbb{E}X^2 + \mathbb{E}Y^2) = mn|\alpha+i\beta|^2.$$

Applying this to the previous computation, we obtain, thanks to the independence of the random variables $\widehat{\varepsilon}^{(j)}(a,b)$ for $1 \le j \le L$,

$$\frac{1}{MN}\mathbb{E}\left|\widehat{\varepsilon'}(\alpha_\ell^{a,b},\beta_\ell^{a,b})\right|^2 = \frac{mn}{MN}\sum_{j=1}^{L}|c_\ell^{(j)}(a,b)|^2 = \frac{1}{z_x z_y}\sum_{j=1}^{L}|c_\ell^{(j)}(a,b)|^2, \qquad (44)$$

which corresponds to the square of $\mathcal{A}(\alpha_\ell^{a,b},\beta_\ell^{a,b})$ according to the definition given in (41). Therefore, taking the square root of (44) for all $(a,b) \in \widehat{\omega}$ and all $\ell \in \{1,2,\dots,Z^{a,b}\}$ yields (40) as announced. □

Proposition 6 allows us to compute the expected level of noise in the reconstructed image u_{LS}, through the Fourier amplification map $\mathcal{A}$, which only depends on the translations and the chosen zoom factors z_x and z_y. The normalization used in Equation (40) ensures that if $\mathcal{A} = 1$ everywhere, then the noise variance in u_{LS} and in u_0 are the same.

Remark 4. *By introducing an appropriate complex factor with modulus 1 in Equation (42), one can prove*[3] *that* $|c_\ell^{(j)}(a,b)| = |c_\ell^{(j)}(a',b')|$ *as soon as* $\mathcal{M}(a,b) = \mathcal{M}(a',b')$. *A consequence is that the Fourier amplification map* $\mathcal{A}$ *is piecewise constant (see first row of Figure 4). In the case of integer zoom factors* z_x, z_y, *all* $\mathcal{M}(a,b)$ *matrices are identical so* $|c_\ell^{(j)}(a,b)|$ *is independent of* a *and* b *and the amplification map* $\mathcal{A}$ *takes at most* $Z^{a,b} = z_x z_y$ *different values (the number of values of the* ℓ *index in Equation (41)). In the more general case of non-integer zoom factors, the number of values is not greater than*

$$(\lfloor z_x \rfloor + \lceil z_x \rceil) \cdot (\lfloor z_y \rfloor + \lceil z_y \rceil)\ .$$

[3] More precisely, Equation (38) and Equation (42) imply that when $|P(a)| = |P(a')|$ and $|Q(b)| = |Q(b')|$, we have $c_\ell^j(a',b') = c_\ell^j(a,b) \cdot e^{-2i\pi\zeta}$ with $\zeta = \delta_x^{(j)}\left(\frac{a'-a}{m} + p_{\min}(a') - p_{\min}(a)\right) + \delta_y^{(j)}\left(\frac{b'-b}{n} + q_{\min}(b') - q_{\min}(b)\right)$.

As shown by Proposition 6, the reconstruction error is not a white noise but a non-stationary Gaussian random field. It can be observed with two maps:

- the (non-constant) variance map in the Fourier domain, $(\alpha, \beta) \mapsto \mathbb{E}|\widehat{\varepsilon'}(\alpha, \beta)|^2$, which is directly related to the amplification map $\mathcal{A}$ (see Figure 4) and is responsible for correlations between the random variables $\varepsilon'(k, \ell)$;
- the (non-constant) variance map in the image domain, $(k, \ell) \mapsto \mathbb{E}\big(\varepsilon'(k, \ell)^2\big)$ (see last row of Figure 5), which is responsible for correlations between the random variables $\widehat{\varepsilon'}(\alpha, \beta)$.

The fluctuations of the variance map in the Fourier domain give the image ε' the aspect of a texture. This texture is Gaussian, but contrary to most classical Gaussian models of textures, it is not stationary. Indeed, we can see from Equation (43) that the coefficients $\widehat{\varepsilon'}(\alpha, \beta)$ are correlated, which implies that the variance map $(k, \ell) \mapsto \mathbb{E}\big(\varepsilon'(k, \ell)^2\big)$ is not constant in general (see Figure 5).

The textured aspect of the reconstruction error can be considered as a reconstruction artifact, since it may lead to the appearance of small spurious oscillating patterns in the reconstructed image. This phenomenon mainly occurs when L is close to $z_x z_y$; as we can see in Figure 4 and 5, the fluctuations of both variance maps quickly vanish as L increases. In most practical cases, one has $L \gg z_x z_y$ and the reconstruction error ε' is visually similar to a Gaussian white noise.

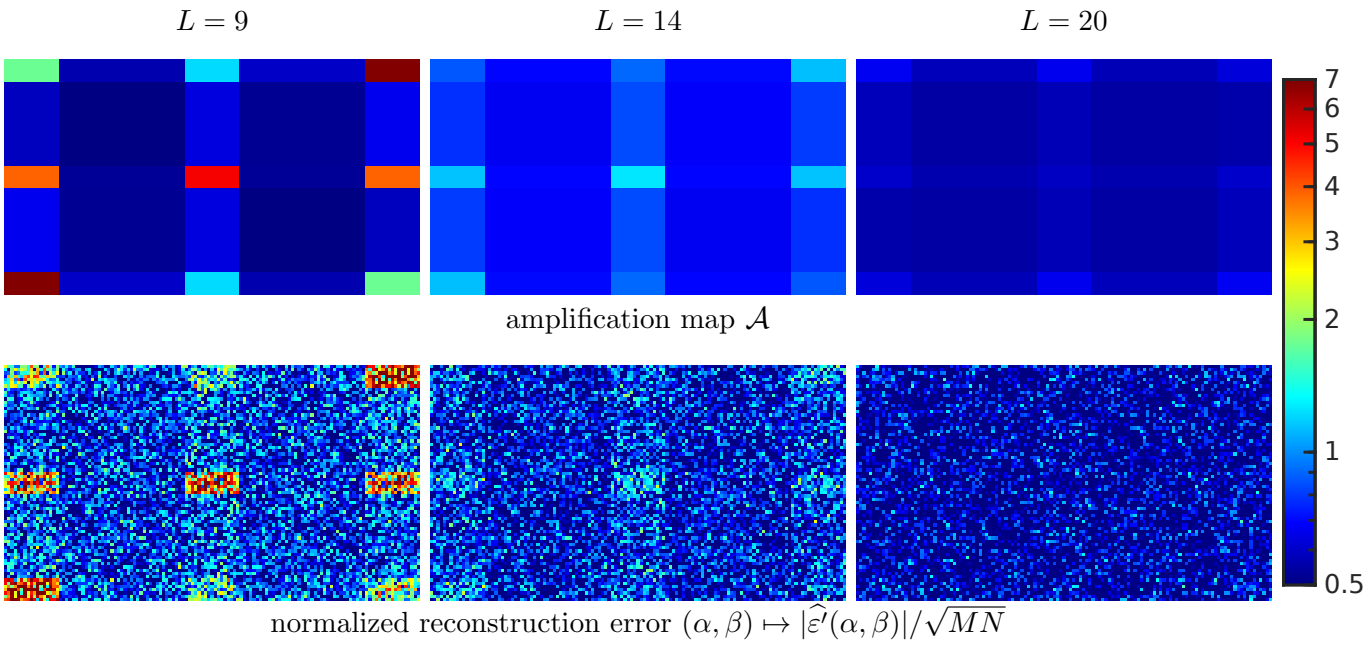


Figure 4: **Amplification map and reconstruction error in the Fourier domain.** First row: the amplification map $\mathcal{A}$ defined in Proposition 6, for different values of L, $\sigma = 1$, random translations uniformly and independently distributed over $[-5, 5]^2$, and super-resolution factors equal to $z_x = M/m = 129/56 \approx 2.3$ and $z_y = N/n = 73/33 \approx 2.2$ respectively. Second row: the corresponding normalized reconstruction error $|\widehat{\varepsilon'}|/\sqrt{MN}$ in the Fourier domain (for one realization of ε). We can see that the amplification map gives a good prediction of the fluctuations of $|\widehat{\varepsilon'}|$, and that the amplitude of these fluctuations quickly decreases when L increases. When L is close to $z_x z_y$ (left column), the amplification map may present strong variations from one region to another, which leads to the appearance of some textured error patterns in the reconstructed image (see Figure 5). In most practical cases however, one has $L \gg z_x z_y$ and the amplification map $\mathcal{A}$ is almost constant (and takes small values).

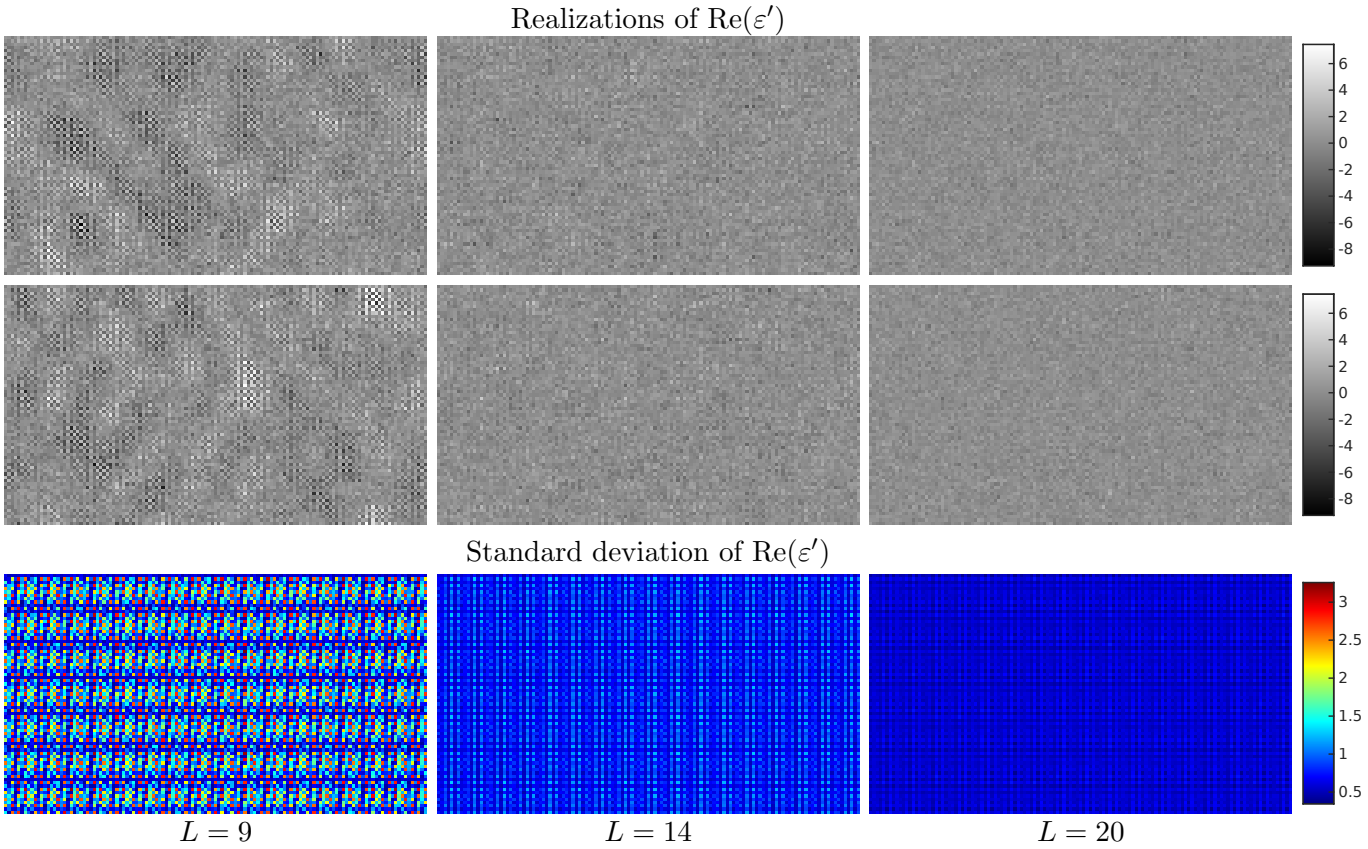


Figure 5: **Reconstruction error in the spatial domain.** The conditions are the same as in Figure 4, but we now examine the effect of noise in the spatial domain through the error image ε'. First and second row: error images $\mathrm{Re}(\varepsilon')$ obtained for two different realizations of ε. Third row: standard deviation of $\mathrm{Re}(\varepsilon')$, estimated from 10^3 realizations of ε. The most visible patterns in the error images are not localized but correspond to a texture behavior (the two realizations show that the patterns do not occur at specific locations, and are barely related to the variations of the standard deviation displayed in the third row). Hence, the visual structure of the reconstruction error is mostly due to the variations of the amplification map displayed in Figure 4, whereas the spatial inhomogeneity shown in the third row plays a minor role. We can also notice that the amplitude and the amount of structure in the error images quickly decreases as L grows, which confirms the observation made in Figure 4 for the Fourier domain.

4.2 Predicted Error

Proposition 6 is very interesting in terms of prediction, because the computation of the amplification map $\mathcal{A}$ only requires the knowledge of the translations $(\delta^{(j)})_{1\leq j\leq L}$, and is independent of the data coming from the low-resolution image sequence. From the number of images (L), the translations $(\delta^{(j)})_{1\leq j\leq L}$, the dimension of the low-resolution images (m, n), and the chosen super-resolution factors (z_x, z_y), we can compute the amplification map $\mathcal{A}$ and predict, for a given level of noise (σ) in the input sequence, the reconstruction error due to the presence of noise. In the following we keep the notations of Proposition 6.

It is a well-known fact in the image processing community that the norm of a white noise image (say, with 0 mean and unit variance) is almost independent of the realization of the noise. This can be proven in several ways, either by computing the exact law (in the case of a Gaussian noise for example), or by invoking asymptotically the strong law of large numbers or the Central Limit Theorem (which also provides an estimate of the norm variance). Numerically, we observed that this stability was also satisfied by the norm of the reconstruction error image $U_{\mathrm{LS}} - \overline{u_{\mathrm{LS}}}$, but since this random image is neither white nor homoscedatic (in other terms, pixels are not independent and all

pixels do not follow the same law), we first provide a theoretical justification with

Lemma 3. *If z_x, z_y, L and $\delta = ((\delta_x^{(j)}, \delta_y^{(j)}))_{1\leq j\leq L}$ are fixed, then one has, almost surely as $mn \to +\infty$,*

$$\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 \;\sim\; \mathbb{E}\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2.$$

Proof. The proof is done in Appendix C. □

Corollary 1. *Given a noisy input image sequence u_0 (a realization of U_0), the reconstruction mean square error (MSE) and the associated peak signal-to-noise ratio (PSNR) defined by*

$$\text{MSE} = \frac{1}{MN}\|u_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 \qquad \textit{and} \qquad \text{PSNR} = 10\,\log_{10}\left(\frac{d^2}{\text{MSE}}\right) \tag{45}$$

(denoting d the peak value of the reconstructed image $u_{\text{LS}} = (A^*A)^\dagger A^* u_0$, e.g. $d = 255$ for images with 8-bit dynamic) can be accurately predicted (independently from the input sequence u_0) from the amplification map $\mathcal{A}$ by

$$\text{MSE} \approx \frac{\sigma^2\|\mathcal{A}\|_2^2}{MN} \qquad \textit{and} \qquad \text{PSNR} \approx 10\,\log_{10}\left(\frac{d^2 MN}{\sigma^2\|\mathcal{A}\|_2^2}\right). \tag{46}$$

Proof. Due to the high dimension of $\mathbb{R}^\Omega$, with very good accuracy and high confidence level we have, thanks to Lemma 3,

$$\|u_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 \approx \mathbb{E}\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2,$$

so that $\text{MSE} = \dfrac{1}{MN}\|u_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 \approx \dfrac{1}{MN}\mathbb{E}\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 = \dfrac{1}{(MN)^2}\mathbb{E}\|\widehat{U_{\text{LS}}} - \widehat{\overline{u_{\text{LS}}}}\|_2^2 = \dfrac{\sigma^2\|\mathcal{A}\|_2^2}{MN}$. The formula for the PSNR directly follows. □

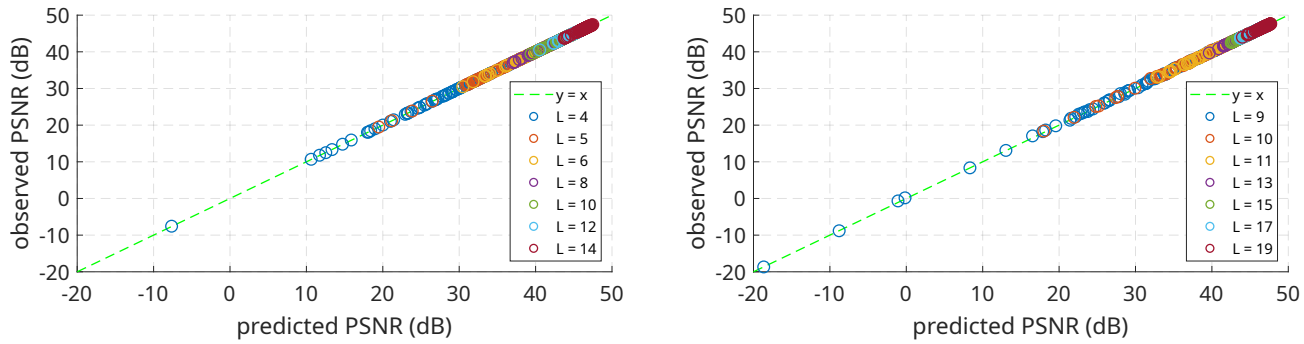
The prediction given by Equation (46) is actually very accurate, as we can check in Figure 6. If the acquisition setup leads to an invertible matrix A (which holds for almost all translations sequences as soon as $L \geq \lceil z_x\rceil \cdot \lceil z_y\rceil$), then the reconstruction $\overline{u_{\text{LS}}}$ is theoretically exact, and Equation (45) exactly measures the reconstruction error. In practice, however, there are other sources of error (in particular, the translations $(\delta^{(j)})_{1\leq j\leq L}$ are not perfectly known), so that Equation (46) is a prediction of the part of the reconstruction error caused by the noise in the input image sequence.

Beyond the overall prediction of the reconstruction quality given by the PSNR, the amplification map $\mathcal{A}$ defined in Proposition 6 could be used to take into account more precisely the nature of the reconstruction noise in post-processing algorithms. Such information is often used in image processing to build restoration filters, in particular Wiener filters. Note that his would be useful only when $L \simeq z_x z_y$, since the variations of the amplification map become negligible when $L \gg z_x z_y$. For that reason, we shall not use the information provided by the amplification map $\mathcal{A}$ later in Section 7, when we shall describe a simple post-processing filter to improve the contrast of the reconstructed high-resolution image.

4.3 Worst-case Scenario with Respect to Noise

We can observe that the MSE estimate given in Equation (46) is simply the expected MSE, and it is very different from the MSE upper bound that would be obtained from the condition number $\kappa(A) = \|A\|_2\|A^{-1}\|_2$ using the inequality

$$\frac{\|u_{\text{LS}} - \overline{u_{\text{LS}}}\|_2}{\|\overline{u_{\text{LS}}}\|_2} \leq \kappa(A)\frac{\|u_0 - \overline{u_0}\|_2}{\|\overline{u_0}\|_2}. \tag{47}$$



(a) $z_x = 2$, $z_y = 2$ (b) $z_x = 2.1$, $z_y = 2.3$

Figure 6: **Prediction of the reconstruction PSNR.** We here illustrate the effectiveness of the error prediction allowed by Corollary 1. For 7 values of L, we simulate several low-resolution sequences (100 per value of L, with random i.i.d. translations uniformly sampled over $[-5, 5]^2$, and noise standard deviation $\sigma = 2$) and for each of them we compute the associated super-resolution image with super-resolution factors equal to $z_x = M/m = 450/225 = 2$, $z_y = N/n = 324/162 = 2$ (case a, left column) or $z_x = M/m = 450/214 \approx 2.1$, $z_y = N/n = 324/141 \approx 2.3$ (case b, right column). Then we plot, in each case, a point with x coordinate equal to the PSNR predicted by Equation (46) from the amplification map $\mathcal{A}$ and y coordinate equal to the PSNR measured by comparing the reconstruction u_{LS} to the ideal (apodized) high-resolution image $\overline{u_{\text{LS}}}$. We can observe the high accuracy of the prediction, as all these points are almost perfectly located on the line $y = x$. Note that the observed PSNR was calculated after removing the small apodization regions on the border of the image domain (otherwise, the apodization would artificially increase the PSNR).

Indeed, assuming that $\frac{1}{MN}\|\overline{u_{\text{LS}}}\|_2^2 \approx \frac{1}{mnL}\|\overline{u_0}\|_2^2$ (which means that $\overline{u_0}^2$ and $\overline{u_{\text{LS}}}^2$ have similar spatial averages), Equation (47) yields

$$\frac{\|u_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2}{MN} \lesssim \kappa(A)^2 \frac{\|u_0 - \overline{u_0}\|_2^2}{mnL} \approx \kappa(A)^2\sigma^2,$$

$$\text{so that} \qquad \frac{MN}{\|u_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2} \gtrsim \frac{1}{\kappa(A)^2\sigma^2},$$

and using (45) we end up with the lower bound

$$\text{PSNR} \gtrsim 10 \log_{10}\left(\frac{d^2}{\kappa(A)^2\sigma^2}\right). \tag{48}$$

Note that this (approximate) lower bound, which roughly corresponds to worst-case realizations of the noise, is very pessimistic (as we shall see later in Table 1). The reason is that Inequality (47) is determined by the worst case, but does not account for average situations that are encountered in practice. Nonetheless, this kind of lower bound was successfully used in [66] and [65, chapter 5] to locally estimate the ill-posedness of the super-resolution problem, and to adjust the amount of regularization accordingly.

4.4 Expected Error for Optimal Translations

Aside from the pessimistic bound (48) we just discussed, we can obtain an optimistic bound on the expected MSE, which corresponds to the minimal expected error we can achieve with respect to the sequence of translations $(\delta^{(j)})_{1\le j\le L}$. In the following, we shall assume that the operator A has full rank. This is not a limitation in practice, for several reasons:

- if A has not full rank, the reconstruction u_{LS} is still defined (thanks to the use of the Moore-Penrose pseudo-inverse $A^\dagger = (A^*A)^\dagger A^*$), but the lack of information will cause artifacts in general. The full-rank property is hence highly desirable;

- if $L \geq \lceil z_x \rceil \cdot \lceil z_y \rceil$ and the translations $(\delta^{(j)})_{1 \leq j \leq L}$ follow a random distribution having a density with respect to Lebesgue's measure, then A has full rank almost surely.

Proposition 7. *If A has full rank, then the expected value of the reconstruction MSE satisfies*

$$\mathbb{E}(\text{MSE}) \geq \frac{z_x z_y}{L} \sigma^2. \tag{49}$$

Proof. 1) Let us first notice that if $X = (X_i)_{1 \leq i \leq d}$ is a vector of i.i.d. $\mathcal{N}(0,1)$ random variables and B a complex $d \times d$ matrix, one has

$$\begin{aligned}
\mathbb{E}\|BX\|_2^2 &= \mathbb{E}\big(X^T B^* B X\big) \\
&= \mathbb{E}\big(\,\text{trace}\,\big(X^T B^* B X\big)\big) \\
&= \mathbb{E}\big(\,\text{trace}\,\big(B X X^T B^*\big)\big) \\
&= \text{trace}\,\big(B \mathbb{E}(X X^T) B^*\big) \\
&= \text{trace}(B B^*) = \|B\|_F^2,
\end{aligned}$$

where $\|B\|_F := \sqrt{\text{trace}(B^* B)} = \sqrt{\text{trace}(B B^*)}$ is the Frobenius norm of B.

2) We can use this result to compute the Frobenius norm of the matrix associated to A in the canonical basis (in this proof, we shall use the same notation A for the operator and its associated matrix). Indeed, if U is a $M \times N$ image made of i.i.d. $\mathcal{N}(0,1)$ random variables, then each random value $(AU)^{(j)}(k,\ell)$ has zero mean and satisfies

$$\text{Var}\left(\,\text{Re}\,\big((AU)^{(j)}(k,\ell)\big)\right) + \text{Var}\left(\,\text{Im}\,\big((AU)^{(j)}(k,\ell)\big)\right) = 1,$$

so that $\mathbb{E}\|AU\|_2^2 = \sum_{j,k,\ell} 1 = mnL$. From 1), we deduce that $\|A\|_F = \sqrt{mnL}$.

3) Let us now estimate the Frobenius norm of $A^\dagger$. Since A has full rank, it admits MN positive singular values $\sigma_1, \sigma_2, \ldots, \sigma_{MN}$, and we have

$$\|A\|_F = \sqrt{\sum_{i=1}^{MN} \sigma_i^2}.$$

Now, the singular values of $A^\dagger$ are $\sigma_1^{-1}, \sigma_2^{-1}, \ldots, \sigma_{MN}^{-1}$, so that

$$\|A^\dagger\|_F = \sqrt{\sum_{i=1}^{MN} \sigma_i^{-2}},$$

and by Cauchy-Schwarz Inequality we obtain

$$\|A\|_F \|A^\dagger\|_F = \sqrt{\sum_{i=1}^{MN} \sigma_i^2} \sqrt{\sum_{i=1}^{MN} \sigma_i^{-2}} \geq \sum_{i=1}^{MN} \sigma_i \cdot \sigma_i^{-1} = MN.$$

Since we know from 2) that $\|A\|_F = \sqrt{mnL}$, we deduce that

$$\|A^\dagger\|_F \geq \frac{MN}{\sqrt{mnL}}.$$

4) Denoting by $\varepsilon = (\varepsilon^{(1)}, \dots, \varepsilon^{(L)}) \in (\mathbb{R}^\omega)^L$ an image sequence made of i.i.d. $\mathcal{N}(0,1)$ random variables (as in Proposition 6), we get

$$\begin{aligned}
\mathbb{E}(\text{MSE}) &= \frac{1}{MN}\mathbb{E}\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 \\
&= \frac{1}{MN}\mathbb{E}\|A^\dagger(\sigma\varepsilon)\|_2^2 \\
&= \frac{\sigma^2}{MN}\|A^\dagger\|_F^2 \\
&\geq \frac{\sigma^2}{MN}\frac{(MN)^2}{mnL} \\
&\geq \frac{\sigma^2 z_x z_y}{L}.
\end{aligned}$$

□

Notice that the right-hand term of Equation (49) can be interpreted with a very simple reasoning. We estimate the MN intensity values of the high-resolution image from the mnL intensity measurements coming from the LR sequence, so we have a ratio of $\lambda = \frac{mnL}{MN} = \frac{L}{z_x z_y}$ observed values for each reconstructed value. Hence, we can at best expect to reduce the initial variance σ^2 by this ratio λ and obtain a reconstruction MSE equal to $\frac{\sigma^2}{\lambda} = \frac{z_x z_y}{L}\sigma^2$. Note that this lower bound is actually attained for perfectly distributed translations, that is, when z_x and z_y are integers, L is a multiple of $z_x z_y$, and the translations $\delta^{(j)}$ are uniformly distributed on the grid

$$\left\{0, \frac{1}{z_x}, \frac{2}{z_x}, \dots, \frac{z_x - 1}{z_x}\right\} \times \left\{0, \frac{1}{z_y}, \frac{2}{z_y}, \dots, \frac{z_y - 1}{z_y}\right\}.$$

From (49), we cannot directly derive an inequality for the expected PSNR (because the expectation operator does not commute with the logarithm), but we still can infer the approximate inequality

$$\text{PSNR} \lesssim 10 \log_{10}\left(\frac{d^2 L}{z_x z_y \sigma^2}\right), \tag{50}$$

where the right-hand term is an optimistic PSNR bound.

4.5 Usage of Error Predictions and Bounds

In the previous section, we derived the bounds

$$10 \log_{10}\left(\frac{d^2}{\kappa(A)^2\sigma^2}\right) \lesssim \text{PSNR} \lesssim 10 \log_{10}\left(\frac{d^2 L}{z_x z_y \sigma^2}\right), \tag{51}$$

in Equation (48) and (50), as well as the estimate

$$\text{PSNR} \approx 10 \log_{10}\left(\frac{d^2 MN}{\sigma^2 \|\mathcal{A}\|_2^2}\right). \tag{52}$$

in Equation (46). The left-hand term of (51) is a very pessimistic lower bound; it is never really attained in practice because it corresponds to extremely rare events. On the contrary, it is possible to obtain PSNR values quite close to the right-hand term of (51) by choosing appropriate translations. Once the translations have been chosen (or measured), the estimate (52) gives a pretty accurate estimate of the actual PSNR. Note that this actual PSNR depends on the noise level σ but is almost

independent of the noise realization in reason of the high-dimensionality of the problem. These remarks will be illustrated in Table 1 of Section 5.

Being able to predict the reconstruction PSNR from the displacements with Equation (52) opens the way to several applications. It can be used to choose between different reconstruction strategies (for instance to decide whether regularization should be used or not), or to optimize the choice of δ when the sequence of displacements can be controlled during the acquisition, as it is the case for the so-called jitter camera [8, 9].

5 Least-squares Reconstruction Experiments over Synthetic and Real Data

In sections 5.1–5.4, we evaluate the ability of the least-squares model to reconstruct an accurate high-resolution image u_{LS} from a sequence of low-resolution images u_0 and the associated sequence of displacements δ. Using synthetic experiments, we study the influence of several important parameters, such as the number L of images in u_0, the level of noise corrupting the sequence u_0, the configuration of the displacements δ, and the accuracy of the estimation of these displacements. To simplify the study, we focus on the case of a 2×2 super-resolution factor ($z_x = z_y = 2$). Then, we consider in Section 5.5 two real sequences: one issued by a 2-photon microscope, and the other by a thermal infrared camera. We completely process these sequences (including the estimation of the unknown displacements), and analyze the obtained super-resolved images.

5.1 Design of the Synthetic Experiments

To build a synthetic low-resolution image sequence such as the one displayed in Figure 1 (a), we draw random displacements and apply the associated operator A to a high resolution reference image. Since the translations are implemented in A in a periodic setting, the low-resolution image sequence exhibits unnatural behavior (for example, details lost on the left side re-enter the image on the right side), so we have to remove several lines and columns of each image to obtain a realistic image sequence (otherwise, we would be in the situation of an "inverse crime" evaluation [29]).

Let us recall that in the super-resolution algorithm we propose, we systematically pre-process the input image sequence with the apodization procedure described in Section 2.3. The consequence is that the high-resolution image we aim to reconstruct is not exactly the reference image used to build the synthetic image, but an apodized version of it, as displayed in Figure 1 (d).

To be able to assess the average performance of the algorithm, we introduce some randomness in the generation of the sequences of interest. Thus, to build a sequence u_0 containing L low-resolution images $\{u_0^{(j)}\}_{1 \le j \le L}$, we use the following procedure:

(i) each low-resolution image $u_0^{(j)}$ is corrupted by an additive Gaussian noise with zero-mean and standard deviation $\sigma = 2$ (the reference image u having a 8-bits dynamic, the peak signal-to-noise ratio (PSNR) of a low-resolution image is roughly equal to 42 dB);

(ii) all vectors of the displacement sequence $\delta = \left(\delta_x^{(j)}, \delta_y^{(j)}\right)\right)_{1 \le j \le L}$ are drawn uniformly and independently in $[-5, 5] \times [-5, 5]$;

(iii) when computing u_{LS}, instead of the exact displacement sequence δ used to generate the input sequence u_0, we may consider a perturbed displacement sequence

$$\delta' = \left((\delta_x^{(j)} + n_x^{(j)}, \delta_y^{(j)} + n_y^{(j)})\right)_{1 \le j \le L}, \tag{53}$$

where the perturbations $(n_x^{(j)}, n_y^{(j)})_{1 \leq j \leq L}$ are independently drawn from a Gaussian random variable with zero mean and standard deviation σ_δ.

The use of a perturbed sequence δ' instead of the actual sequence δ for the reconstruction is justified by the fact that, in practice, the displacements cannot be measured with an infinite accuracy, and the parameter σ_δ is precisely a way to account for this limited accuracy. As we will see in Section 5.4, even small inaccuracies in the displacements can cause important artifacts in the reconstruction.

5.2 A Synthetic Example of Super-resolution Reconstruction

As a first synthetic experiment, we applied the super-resolution Algorithm 2 with factor 2×2 to a synthetic sequence u_0 containing $L = 20$ low-resolution images, with no perturbation of the displacements ($\sigma_\delta = 0$). The result is displayed in Figure 7, and compared to the ground truth (apodized reference image) both in the image and in the Fourier domains. The reconstruction quality is very good, as the reconstructed image is almost identical to the reference.

The fact that we can almost perfectly recover the super-resolved image with factor 2×2 from a low-resolution sequence u_0 containing $L = 20$ images (and with no perturbation of the displacements) is due to the highly favorable ratio $\frac{L}{z_x z_y} = 5$, which is significantly larger than 1. Not only this high ratio improves the "optimistic" expected PSNR of the reconstruction (see Equation (50)), but as we shall see now it also decreases a lot the probability of observing bad displacement configurations.

5.3 Influence of the Displacement Configuration over the Quality of the Reconstruction

A legitimate question about the reconstruction example displayed in Figure 7 is: "How representative of the possible behavior of the reconstruction algorithm is this particular example?". As mentioned in Section 4.5, the variations of the reconstruction due to the noise on the low-resolution sequence are quite small, because the high-dimensionality prevents large deviations to appear with significant probability. However, the variations due to the displacement sequence δ (which is also randomly chosen), can be very large, especially when the "redundancy ratio" $\frac{L}{\lceil z_x \rceil \cdot \lceil z_y \rceil}$ is close to 1. We now analyze this phenomenon more precisely.

In Figure 8, we study the relation between the number L of images and the distribution of the predicted PSNR with respect to the displacements. We observe two effects, both in favor of using a number L of images significantly greater than 4 (more generally, greater than $\lceil z_x \rceil \cdot \lceil z_y \rceil$):

1. as L grows, the average predicted PSNR increases;
2. as L grows, the fluctuations of the predicted PSNR decrease.

In other terms, increasing L improves the average reconstruction quality, and, more importantly, decreases the chance of encountering an "unlucky" configuration of displacements yielding a poor quality reconstructed image.

This phenomenon is illustrated in Figure 9. We observe that for a 2×2 super-resolution factor, using $L = 4$ or $L = 5$ images is rather unwise, since for at least 10% of the possible displacement sets, notable artifacts appear in the reconstructed image. With $L = 10$ images, this phenomenon is much less frequent and most reconstructions look the same. We can now answer the question that motivated this section: the example displayed in Figure 7 ($L = 20$) is highly representative of the reconstruction that would be achieved on any such random data.

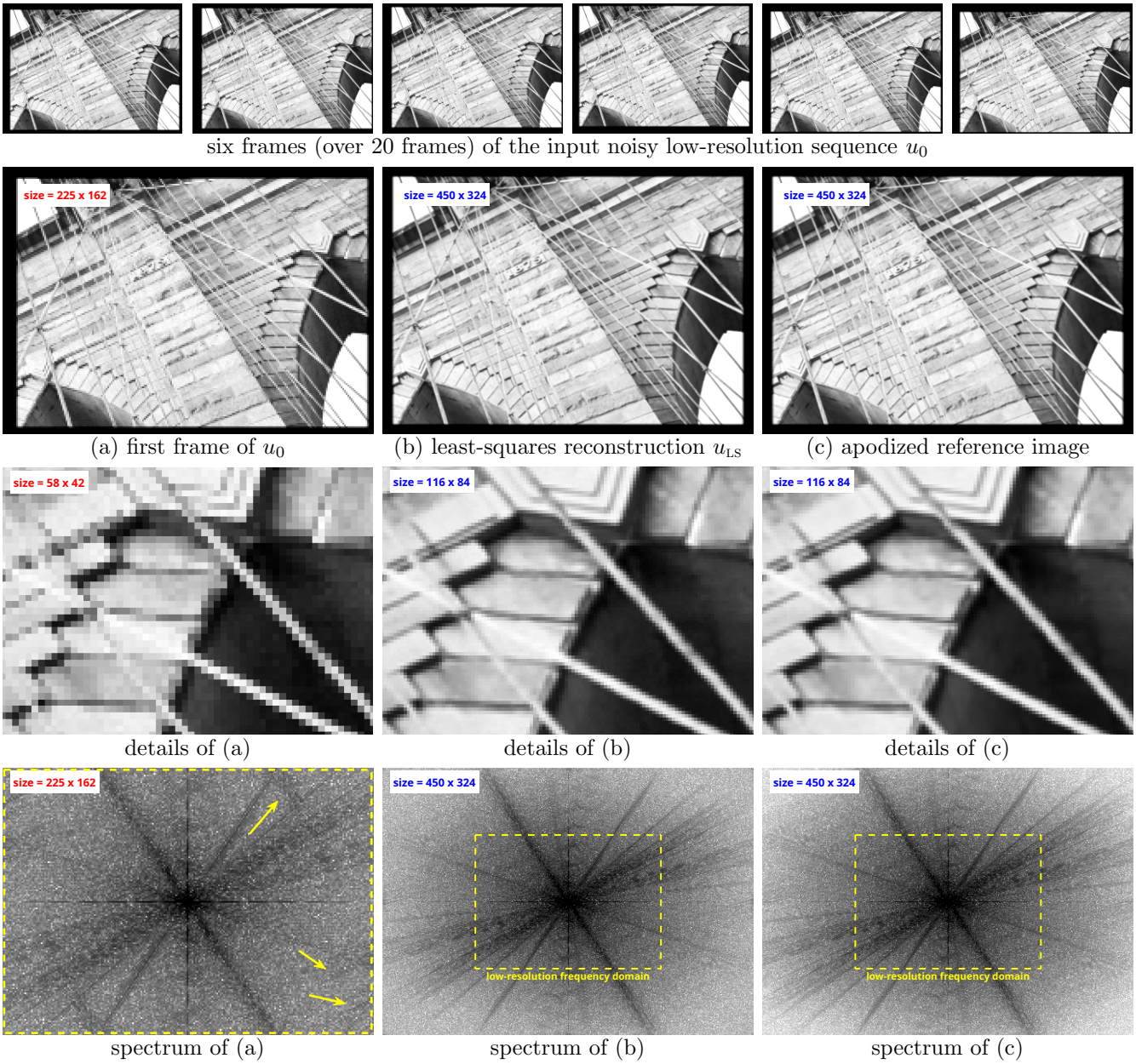


Figure 7: **Super-resolution with factor 2×2 using Algorithm 2.** In the first row, we display six frames of an apodized synthetic sequence u_0 containing $L = 20$ noisy low-resolution images. The first frame of u_0 is also displayed in (a). We display in (b) the high-resolution image u_{LS} computed using Algorithm 2, which can be compared to the apodized high-resolution reference image displayed in (c). In the third row, we display some close-up views of the images (a), (b) and (c), and in the last row, we display their Fourier spectra in logarithmic scale (with high values in dark). The low-resolution image (a) is badly sampled, as attested by the aliasing artifacts appearing in its spectrum (yellow arrows). Comparing (b) and (c), both in the image and the frequency domain, we can see that Algorithm 2 succeeds in reconstructing the high-resolution details of the reference image. Indeed, u_{LS} is very close to the apodized reference image (c), with a relative PSNR equal to 49.0 dB. This experiment illustrates the ability of the least-squares reconstruction model to efficiently recover the high-resolution image from u_0 when L is significantly higher than the targeted total super-resolution factor (here, $4 = 2 \times 2$), and in the (idealized) case of perfectly known displacements ($\sigma_\delta = 0$).

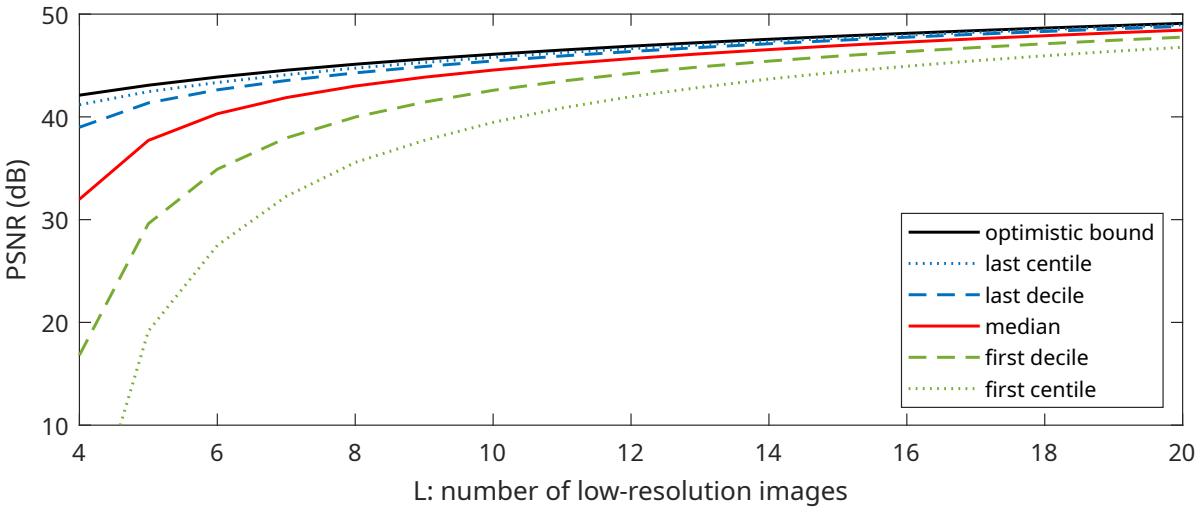
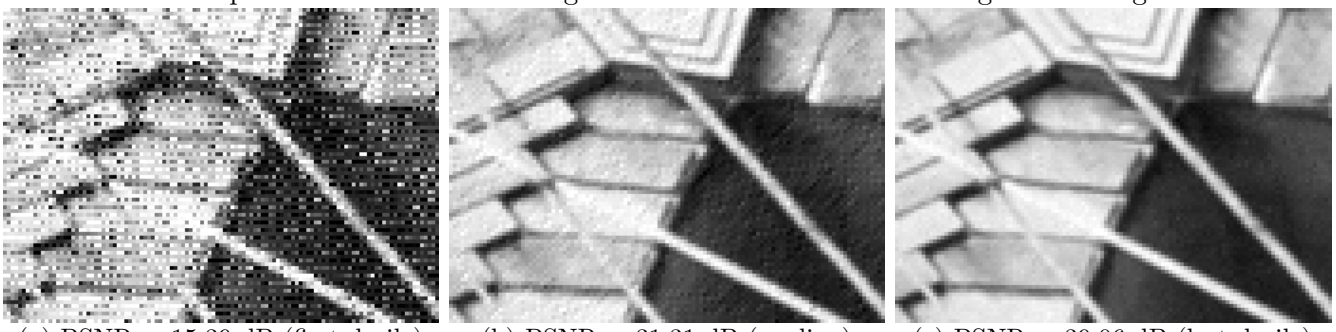


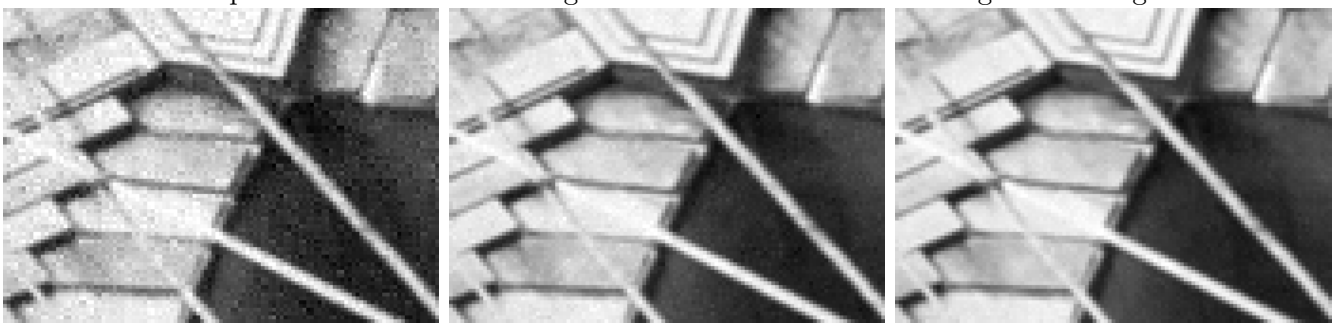
Figure 8: **Variability of the reconstruction quality with respect to the number L of low-resolution images**. We compute the PSNR of the super-resolved image u_{LS} obtained from a sequence u_0 of L synthetic low-resolution images (corrupted by an additive Gaussian noise with standard deviation $\sigma = 2$ and generated using random displacements δ). For each value of L, we perform 10^5 simulations and display, as a function of L, the first centile, the first decile, the median, the last decile and the last centile of the computed PSNR values. As the system is (almost surely) overdetermined when $L > 4$ (that is, the operator A has full rank), we know from Proposition 7 that we can expect to reduce the level of noise up to a factor $\lambda_L = \sqrt{L/(z_x z_y)} = \sqrt{L/4}$, leading to an upper bound for the expected PSNR of $10 \cdot \log_{10}\left(255^2/(\sigma/\lambda_L)^2\right)$, which is represented by the dotted curve (optimistic bound). This PSNR improvement is confirmed by the experiment, especially for large values of L, and it shows the ability of the least-squares to provide a substantial denoising in overdetermined situations.

example of reconstructions using low-resolution stacks containing $L = 4$ images



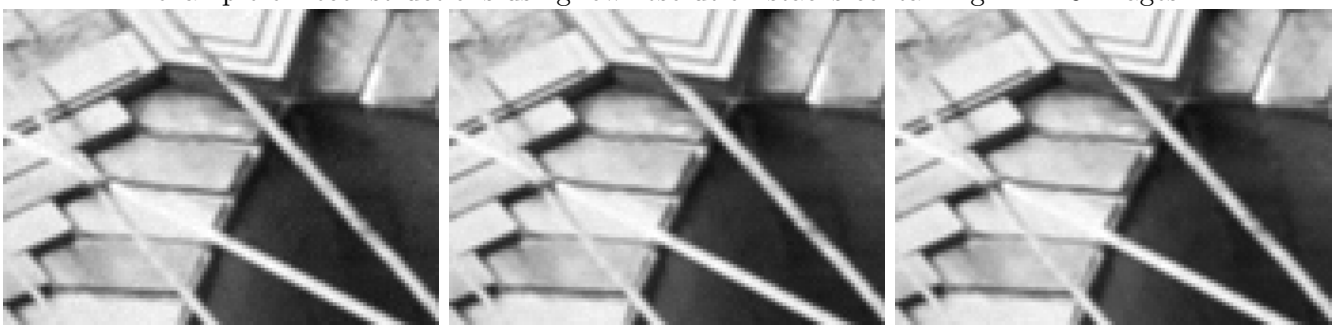
(a) PSNR = 15.39 dB (first decile) (b) PSNR = 31.31 dB (median) (c) PSNR = 39.06 dB (last decile)

example of reconstructions using low-resolution stacks containing $L = 5$ images



(d) PSNR = 29.58 dB (first decile) (e) PSNR = 37.59 dB (median) (f) PSNR = 41.25 dB (last decile)

example of reconstructions using low-resolution stacks containing $L = 10$ images



(g) PSNR = 42.55 dB (first decile) (h) PSNR = 44.61 dB (median) (i) PSNR = 45.43 dB (last decile)

Figure 9: **Prediction of the reconstruction quality and influence of L.** In this experiment, we display some close-up views of super-resolved images reconstructed from low-resolution sequences containing $L = 4$ images (first row), $L = 5$ images (second row) or $L = 10$ images (last row). For each value of L, we repeated 10^3 simulations (with additive Gaussian noise and random displacements) and systematically computed the predicted PSNR according to Equation (46); on the corresponding row, we display the images corresponding to the first decile, the median, and the last decile of the predicted PSNR. We can see that when L is close to the targeted super-resolution factor $z_x \cdot z_y$, bad configurations of displacements often arise, leading to a poor PSNR value and noticeable artifacts, such as those observed in (a) and (d). When L increases, the average value of the PSNR increases and its spread decreases (as shown in Figure 8), thus the reconstruction quality is improved and more stable (last row).

In Table 1, we can check the very little influence of the image noise on the reconstruction (small values of σ_{PSNR} in column 2), as announced earlier. On the contrary, we can see that the PSNR highly depends on the configuration of the displacements, especially when L is small.

Displacement sequence	**PSNR (dB)**			
	average $\pm\, \sigma_{\text{PSNR}}$	pessimistic	predicted	optimistic
$L = 4$, same as Figure 9 (a)	15.44 ± 0.03	5.324	15.39	42.11
$L = 4$, same as Figure 9 (b)	31.31 ± 0.03	22.38	31.31	42.11
$L = 4$, same as Figure 9 (c)	39.06 ± 0.02	32.84	39.06	42.11
$L = 5$, same as Figure 9 (d)	29.60 ± 0.03	19.82	29.58	43.08
$L = 5$, same as Figure 9 (e)	37.60 ± 0.02	29.66	37.59	43.08
$L = 5$, same as Figure 9 (f)	41.23 ± 0.02	34.69	41.25	43.08
$L = 10$, same as Figure 9 (g)	42.53 ± 0.03	31.95	42.55	46.09
$L = 10$, same as Figure 9 (h)	44.56 ± 0.02	35.10	44.60	46.09
$L = 10$, same as Figure 9 (i)	45.40 ± 0.02	37.56	45.43	46.09

Table 1: **Accuracy of the PSNR prediction**. For each displacement sequence considered in Figure 9 (a)-(i), we reconstructed high-resolution images for 10^3 simulated low-resolution sequences corrupted with different noise realizations. We display in the second column the average values of the PSNR, as well as the empirical standard deviation of those PSNR values (σ_{PSNR}). We also display on the three last columns the values of the three PSNR estimates computed in Section 4 (and summarized in Section 4.5) : the (pessimistic) PSNR lower bound (48), the PSNR prediction (46), and the (optimistic) PSNR upper bound (50). We can see that the PSNR prediction is always very close to the actual PSNR achieved in all our simulations. The PSNR lower bound, based on the condition number of the operator A, is indeed very pessimistic (with roughly 7 to 10 dB difference with the actual PSNR values), while the PSNR upper bound, which corresponds to ideal translations, becomes more achievable when the redundancy factor $L/(z_x z_y)$ grows.

5.4 Least-squares Reconstruction Using Erroneous Displacements

In real-life applications, we cannot hope to measure the displacements with infinite accuracy. In the present section, we study the influence of the estimation error on the quality of the reconstruction. We can first distinguish between two different situations:

(i) the displacements are measured independently of the low-resolution sequence u_0, using a dedicated system (see for instance [41, 53, 55, 62, 63]);

(ii) the displacements are measured from the low-resolution sequence u_0 (or said differently, we are only given u_0 and we need to estimate δ before computing u_{LS}).

In the case (i), the estimation of the displacements can be very accurate, provided that the quality of the dedicated system (which may include a specific sensor) is high enough. In the case (ii), the accuracy is inherently limited by the spatial aliasing of the low-resolution sequence u_0. Indeed, most registration algorithms rely on interpolation models, which are much less accurate on aliased images. Several works, such as [32, 58, 68, 69], focus on the problem of motion estimation from aliased images, but this problem is quite difficult and, up to our knowledge and simulations in realistic situations, the typical accuracy of such techniques ranges from an average error of $\sigma_\delta \approx 0.05$ in favorable situations to $\sigma_\delta \geq 0.2$ in more difficult cases. This is not to mention computational difficulties that can make the estimation of the displacement intractable in practical situations [58, 68].

We believe that, more than the super-resolution reconstruction model itself (here, the least-squares), the accurate estimation of the displacements is the most crucial and challenging part of the super-resolution procedure as a whole. In Figure 10, we can observe that the least-squares model provides a very efficient reconstruction when the displacements are perturbed by a Gaussian noise with standard deviation of $\sigma_\delta \leq 0.01$, but some artifacts occur when $\sigma_\delta \geq 0.05$, and those artifacts become rather important when $\sigma_\delta \geq 0.2$. Thus, achieving a very accurate estimation of the displacements should be a major concern of any system aiming at a high quality super-resolved reconstruction. Designing algorithms to refine the estimation of the displacements while computing the super-resolved image would be interesting, but this would deserve a dedicated study that is beyond the scope of the present paper.

Note incidentally that no matter how much we improve the average quality of the displacement estimation (in both situations (i) and (ii) discussed above), in practice we can reasonably expect to observe outliers, that is, low-resolution images associated with a significantly poor estimation of the displacement which will negatively impact the reconstruction. We will address that particular issue in Section 6, where we propose a *lucky-imaging* strategy that attempts to improve the reconstruction quality by removing from the low-resolution sequence u_0 the images that are not compliant with the generative model (images associated with a poor estimation of the displacement, images undergoing a more complicated deformation than the assumed translational motion, images undergoing illumination changes, etc).

5.5 Reconstruction Examples over Real Data

In Figure 11 and 12, we provide examples of super-resolution reconstructions obtained from two different real data sequences. The first sequence (Figure 11) was acquired using a 2-photons microscope, and the second (Figure 12) using a thermal infrared camera. Both sequences exhibit a good SNR, but are genuinely aliased.

In the case of the 2-photon sequence, the pixels of each image are obtained with a scanning system, and a compromise has to be found in the scanning process between a coarse scan grid (which allows for high image acquisition rates, but produces aliased images) and a fine scan grid (which limits or even prevents aliasing, but cannot capture high-speed phenomena). In the sequence we consider in Figure 11, the scan step could not be set below the Nyquist rate considering the observed cell movements, thus all images are aliased.

As concerns thermal infrared cameras, there also is a compromise to be found between the number of pixels (high-resolution sensors are very expensive) and the aperture of the camera (a large aperture collects more light, but requires smaller pixels to avoid aliasing). Thus, thermal infrared cameras often deliver aliased images, and as such are good providers of low-resolution data for super-resolution algorithms.

In both cases, we thus are in a situation where super-resolution is highly desirable not only to remove unwanted aliasing artifacts, but also to improve the true resolution of the observed scene. For each sequence, we first estimated the displacements using the registration algorithm of Keren et al. [32], which relies on a Gaussian pyramid-based multi-scale registration procedure. This algorithm was retained for its accuracy, speed and simplicity. An implementation, provided by the authors of [69], is available at https://github.com/edwardtoday/PolyU_MScST/tree/master/COMP5422/Lab/superresolution_v_2.0. We then applied Algorithm 2, and compared the obtained super-resolution image with the pixelwise average (or median) of the registered low-resolution sequence (the so-called *shift-and-add* or *shift-and-median* method [5, 14, 13]). As we can see in Figure 11 and 12, not only the SNR is improved by the frame fusion process (which is also the case for shift-and-add), but the resolution is clearly increased.

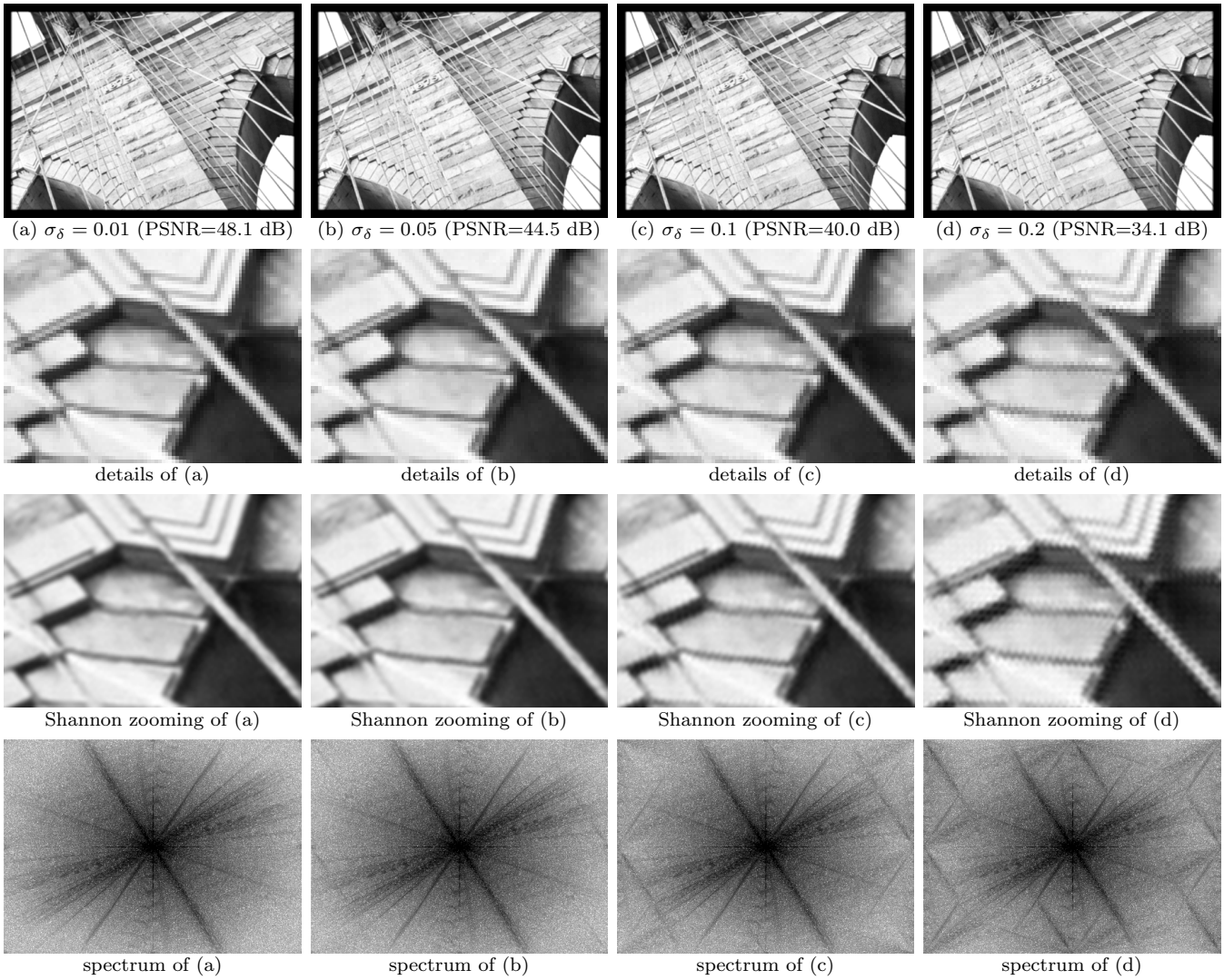
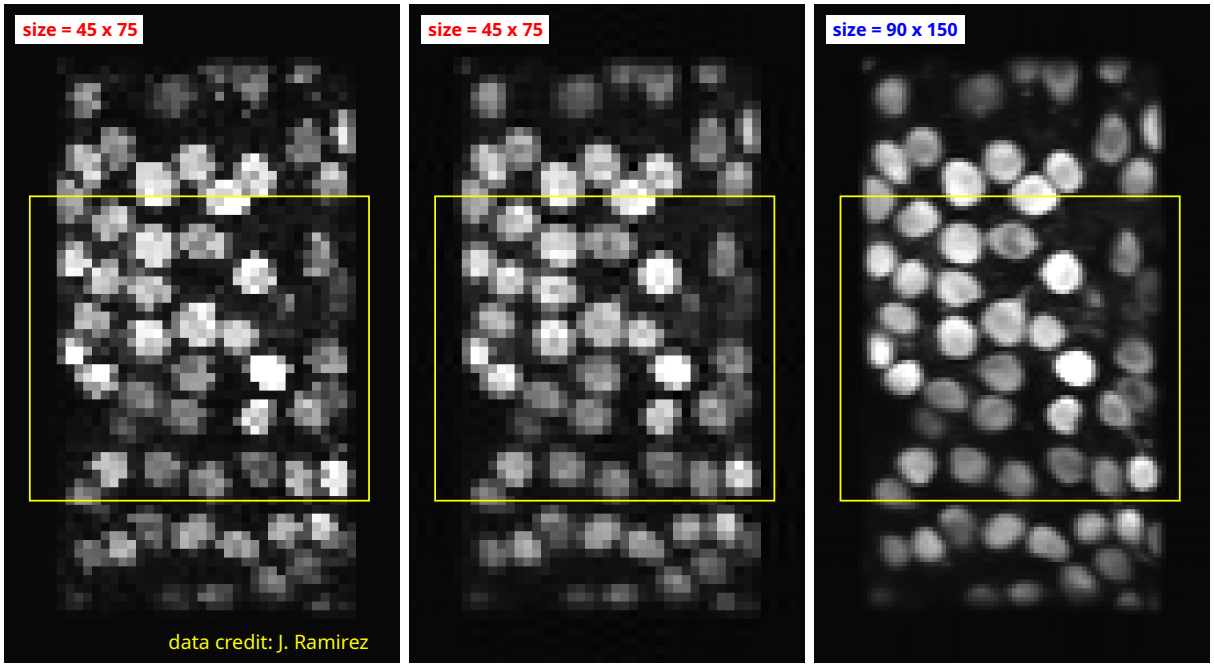


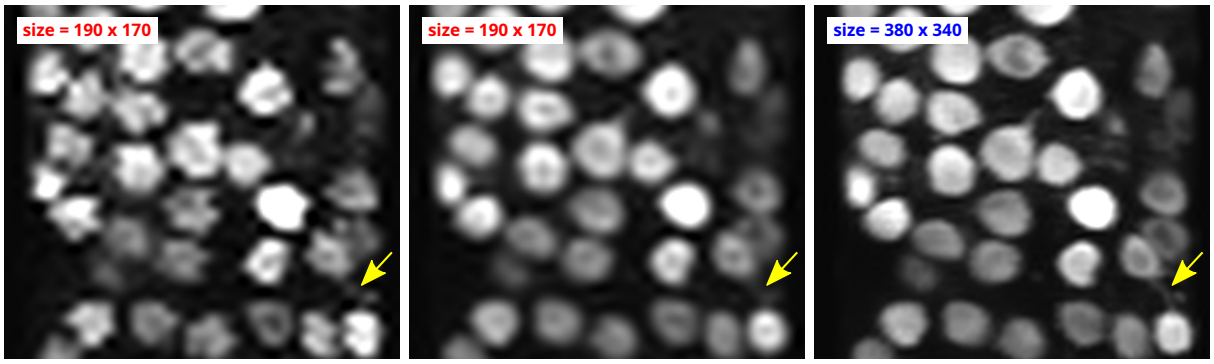
Figure 10: **Least-squares reconstruction using erroneous displacements**. For each value of σ_δ considered in this figure, we simulated 10^3 synthetic low-resolution sequences in the following way. Each sequence u_0 contains $L = 20$ low-resolution images (corrupted by an additive Gaussian noise with standard deviation $\sigma = 2$) and was generated using a random displacement sequence δ (drawn with independent and uniform samples in $[-5, 5] \times [-5, 5]$). For each simulated sequence, we computed the least-squares reconstruction u_{LS} obtained from u_0 and a perturbed displacement sequence $\delta' = (\delta^{(j)} + (n_x^{(j)}, n_y^{(j)}))_{1 \leq j \leq L}$ with i.i.d. $\mathcal{N}(0, \sigma_\delta^2)$ perturbations $(n_x^{(j)}, n_y^{(j)})$ $(1 \leq j \leq L)$. In the first row, we display, for several values of σ_δ, the reconstruction u_{LS} achieving the median PSNR with respect to the reference high-resolution image. A close-up is displayed in the second and third rows (with Shannon interpolation on the third row). The last row represents the spectra (in log scale) of the high-resolution images (a)-(d). Looking closely at the images at the pixel scale (second row), we can observe some spurious structures in the reconstruction (c) and (d). Looking at the sub-pixel scale (third row), artifacts are clearly visible in (c) and (d), and slightly visible in (b). On the spectra of the reconstructed images (last row), artifacts are clearly visible in (b), (c), (d), but barely in (a). This experiment shows that, for a 2×2 super-resolution factor and a sequence of $L = 20$ low-resolution images, artifacts are barely noticeable in the image domain when $\sigma_\delta \leq 0.05$, but $\sigma_\delta \leq 0.01$ is required to draw the same conclusion in the Fourier domain.



(a) first image of the apodized low-resolution sequence u_0

(b) shift-and-add: mean of the registered low-resolution sequence

(c) least-squares reconstruction

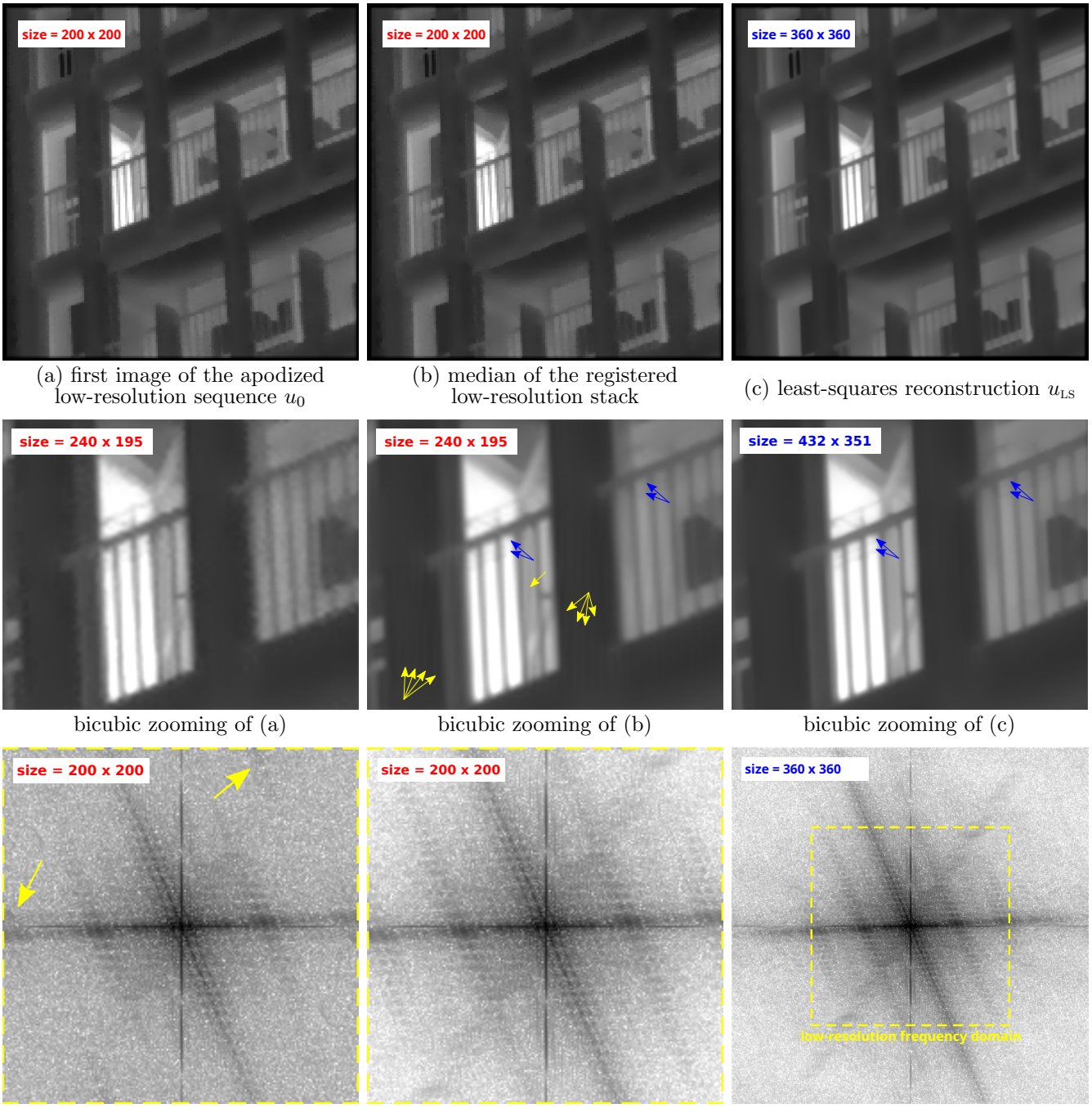


bicubic zooming of (a)

bicubic zooming of (b)

bicubic zooming of (c)

Figure 11: **Super-resolution from a sequence of 2-photons microscopic images**. In this experiment, we used an in-vivo recording of Purkinje cells of a living rat, which was acquired with a 2-photons microscope. This sequence was kindly provided to us by J. Ramirez, author of [56], who performed this challenging video acquisition in order to study the temporal spiking activity of in-vivo brain cells. Achieving a satisfactory temporal sampling (FPS) rate for this sequence constrained the spatial sampling rate to be set large, leading to strongly aliased images. The sequence of interest u_0 contains $L = 2886$ images with size 45×65 (acquired at $4\,\mu m$/pixel). We display in (a) the first image of u_0, and in (b) an enhanced low-resolution image corresponding to the pixelwise averaging of the registered sequence (following the so-called *shift-and-add* methodology [5, 14]). In (c), we can see the high-resolution image u_{LS} computed from u_0. In the second row, we display magnified versions (with a bicubic zoom of factor 5) of the central part (yellow frame) of the upper images. Looking at the images at the pixel scale (first row), we can clearly see in (c) the shape of several cell nuclei, which were not visible in (a) and (b). At the sub-pixel scale (second row), we can see that, due to aliasing, the zoom of image (a) yields very distorted cell shapes. Those distortions are significantly reduced in (b) thanks to the shift-and-add procedure. However, this procedure is not able to restore the high-frequency information, contrary to the super-resolution algorithm used in (c). This is particularly visible on the thin structure (an axone) indicated by the yellow arrow, which is nicely reconstructed in (c).



(a) first image of the apodized low-resolution sequence u_0

(b) median of the registered low-resolution stack

(c) least-squares reconstruction u_{LS}

bicubic zooming of (a)

bicubic zooming of (b)

bicubic zooming of (c)

spectrum of (a)

spectrum of (b)

spectrum of (c)

Figure 12: **Super-resolution from a sequence of thermal infrared images**. We acquired a sequence u_0 containing $L = 600$ images with a thermal infrared FLIR® T640 camera. We display in (a) the first image of u_0, in (b) the enhanced low-resolution image obtained by computing the median of the registered sequence, and in (c) the high-resolution image u_{LS} computed from u_0 (with super-resolution factors $z_x = z_y = 1.8$). Close-ups of these images (obtained with a bicubic zoom of factor 3) are displayed in the second row, and their spectra (in log scale) in the third row. Aliasing artifacts of u_0 are clearly visible on the spectrum of (a) (see the yellow arrows), but strongly reduced in (b) thanks to the multiframe filtering. However, looking at (b) in the image domain (second row), we can see that it suffers from oscillatory artifacts (yellow arrows), which is a characteristic of interpolated aliased images. The super-resolved image (c) is not only much satisfactory in terms of artifacts, but also exhibits high-frequency details (see blue arrows) that are not present in images (a) and (b).

5.6 Complexity and Execution Times

Since Algorithm 2 is very simple, its complexity analysis is equally straightforward. The algorithm primarily involves four dominant operations:

(i) the computation of the $\widehat{u_0}^{(j)}$ (for $1 \leq j \leq L$), which is typically performed using the FFT algorithm in practice, resulting in a complexity of $\mathscr{O}(Lmn \log(mn))$;

(ii) the computation of $\widehat{A^* u_0}$ from the sequence $(\widehat{u_0}^{(j)})_j$ and using (28), leading to a complexity of $\mathscr{O}(LMN) = \mathscr{O}(Lmnz_x z_y)$;

(iii) the computation of the high-resolution DFT coefficients $\widehat{u_{\text{LS}}}(\alpha_\ell^{a,b}, \beta_\ell^{a,b})$ (for $1 \leq \ell \leq Z^{a,b}$ and $(a,b) \in \widehat{\omega}$) by means of matrix-vector products, leading to a complexity of roughly $\mathscr{O}(mn(z_x z_y)^2)$;

(iv) the final computation of $u_{\text{LS}} = \text{IDFT}(\widehat{u_{\text{LS}}})$, which is also handled using the FFT algorithm, resulting in a complexity of $\mathscr{O}(MN \log(MN)) = \mathscr{O}(mnz_x z_y \log(mnz_x z_y))$.

Moreover, since we are considering the case where $z_x z_y < L$, we have $mn\,(z_x z_y)^2 < L\,mn\,z_x z_y$, so the contribution of (iii) to the overall complexity is dominated by that induced by (ii). Similarly, we have

$$mn\, z_x z_y \log(mnz_x z_y) < Lmn\,(\log(mn) + z_x z_y)$$

so the contribution of (iv) to the whole complexity is dominated by that induced by (i) and (ii). Finally, the complexity of Algorithm 2 with respect to L, m, n, z_x and z_y is of

$$\mathscr{O}(Lmn \cdot \lambda(m,n,z_x,z_y)) \quad \text{where} \quad \lambda(m,n,z_x,z_y) = \max(\log(mn), z_x z_y)\,.$$

The factor Lmn involved above corresponds to the number of low-resolution pixels contained in the sequence u_0. It thus represents the number of input data, and its presence in the overall complexity is expected.

Remark 5. *In realistic scenarios–typically involving real-world data–the targeted super-resolution factors are modest (on the order of* 2*), often leading to* $z_x z_y < \log(mn)$*, and thus, to* $\lambda(m,n,z_x,z_y) = \log(mn)$*. In this case, the complexity of Algorithm 2 is driven by (i), i.e., by the computation of the DFT of the low-resolution images.*

We end up this section by reporting in Table 2 the typical execution times required to reconstruct high-resolution images from the various sequences previously considered, as well as their ratios with the reference complexity.

Type	**Setting (L, m, n, $\mathbf{z_x}$, $\mathbf{z_y}$)**	**CPU time (s)**	$\frac{\textbf{CPU Time (s)}}{\textbf{Lmn}\cdot\lambda(\mathbf{m},\mathbf{n},\mathbf{z_x},\mathbf{z_y})}$
Synthetic (bridge)	$L=20$, $m=180$, $n=128$, $z_x=z_y=4$	0.15	$20 \cdot 10^{-9}$
Synthetic (bridge)	$L=20$, $m=360$, $n=256$, $z_x=z_y=2$	0.15	$7.1 \cdot 10^{-9}$
Thermal infrared	$L=600$, $m=200$, $n=200$, $z_x=z_y=1.8$	1.21	$4.8 \cdot 10^{-9}$
2-photons	$L=3186$, $m=45$, $n=75$, $z_x=z_y=2$	0.90	$10 \cdot 10^{-9}$

Table 2: **Execution times.** For several low-resolution sequences considered in the above experiments, we display in the third column the average CPU time (computed over 100 runs) required to reconstruct the high-resolution image using our C implementation of Algorithm 2. The ratios between the average CPU times and the reference complexity $Lmn \cdot \lambda(m,n,z_x,z_y)$ are displayed in the last column and provide an indication about the range of the multiplicative constant hidden in the overall complexity.

6 A Lucky-imaging Strategy Based on Iteratively Reweighted Least-squares

We illustrated in Figure 9 that, assuming a perfect knowledge of the displacements and as long as the sequence u_0 contains enough low-resolution images, the reconstruction of the high-resolution image can be efficiently realized using the least-squares model. However, we saw in Figure 10 that even small errors in the estimation of displacements lead to inaccurate reconstructions. In this section, we propose a *lucky-imaging* strategy, which selects a subset of low-resolution images from the stack u_0, leading to a better high-resolution reconstruction than that obtained from the whole stack u_0. Said differently, we would like to discard outliers, that is, low-resolution images that are in contradiction with the generative model and produce artifacts in the final reconstruction. The proposed strategy is based on the identification of the outliers in u_0 by means of a robust ℓ^1 minimization,

$$u_{\ell^1\ell^2} = \operatorname*{argmin}_{u:\Omega\to\mathbb{R}} \|Au - u_0\|_{1,2}\,, \quad \text{with} \quad \|Au - u_0\|_{1,2} = \sum_{j=1}^{L} \|A_j u - u_0^{(j)}\|_2\,. \tag{54}$$

The use of the $\|\cdot\|_{1,2}$ norm of the residual $Au-u_0$ is motivated by the desire to reject some images of the stack u_0, which will be considered (as a whole) as outliers, while the more classical choice of the standard ℓ^1 norm of the residual, that is, $\|Au-u_0\|_1 := \sum_{j=1}^{L}\sum_{(x,y)\in\omega} |A_j u(x,y) - u_0^{(j)}(x,y)|$, would be more appropriate to reject some pixel values in u_0. Besides, another important reason for the use of the $\|\cdot\|_{1,2}$ norm is that $u_{\ell^1\ell^2}$ can be efficiently computed using iteratively reweighted least-squares (IRLS) algorithms (see [16] and references therein). As explained in [38] (see also [1] and references therein), the idea behind reweighted algorithms is that, for any $r\in\mathbb{R}$, we have $|r| = \inf_{\eta>0} \frac{r^2}{2\eta} + \frac{\eta}{2}$, and thus, the ℓ^1 norm can be rewritten as an infimum over a family of quadratic functions (see [25] for a more general formulation and [26, 49] for interesting variants). Using this identity for each $r_j = \|A_j u - u_0^{(j)}\|_2$, we can rewrite (54) as

$$u_{\ell^1\ell^2} = \operatorname*{argmin}_{u:\Omega\to\mathbb{R}} \inf_{\eta=(\eta_1,\dots,\eta_L)\in(\mathbb{R}_+^*)^L} \mathcal{H}(u,\eta) := \sum_{j=1}^{L} \frac{\|A_j u - u_0^{(j)}\|_2^2}{2\eta_j} + \frac{\eta_j}{2}\,, \tag{55}$$

where the function $\mathcal{H}$ is biconvex (that is, convex with respect to u and convex with respect to η). The minimization can be addressed using the following alternating scheme:

$$\begin{cases} u^{k+1} = \operatorname*{argmin}_{u:\Omega\to\mathbb{R}} \mathcal{H}(u,\eta^k)\,, & \text{(56a)} \\ \eta_j^{k+1} = \max\left(\varepsilon, \|A_j u^{k+1} - u_0^{(j)}\|_2\right) \quad \text{for } j\in\{1,\dots,L\}. & \text{(56b)} \end{cases}$$

Step (56a) yields a least-squares problem similar to (17), provided that we replace A_j and $u_0^{(j)}$ by $A_j/\sqrt{\eta_j}$ and $u_0^{(j)}/\sqrt{\eta_j}$ respectively. With these modifications of the operator A, Proposition 3 and Proposition 4 remain valid if we replace $\mathcal{M}(a,b) = \sum_{j=1}^{L}\mathcal{M}^j(a,b)$ by $\mathcal{M}_\eta(a,b) := \sum_{j=1}^{L}\mathcal{M}^j(a,b)/\eta_j$, so that the computation of u^{k+1} can be handled with Algorithm 2.

Step (56b) should be the minimization of $\mathcal{H}(u^{k+1},\eta)$ with respect to η, which would then write $\eta_j^{k+1} = \|A_j u^{k+1} - u_0^{(j)}\|_2$. We here introduce a threshold parameter ε (a small positive number) in the computation of η_j^{k+1}, in order to prevent some coefficients η_j to vanish, because in that case $\mathcal{H}(u,\eta)$ would not be defined anymore. This issue is common in typical compressed sensing applications, where IRLS schemes are used with undetermined linear systems, which can lead to zero residuals. Several other solutions exist to handle that issue (see [38, 6, 51] for example), but in the case we

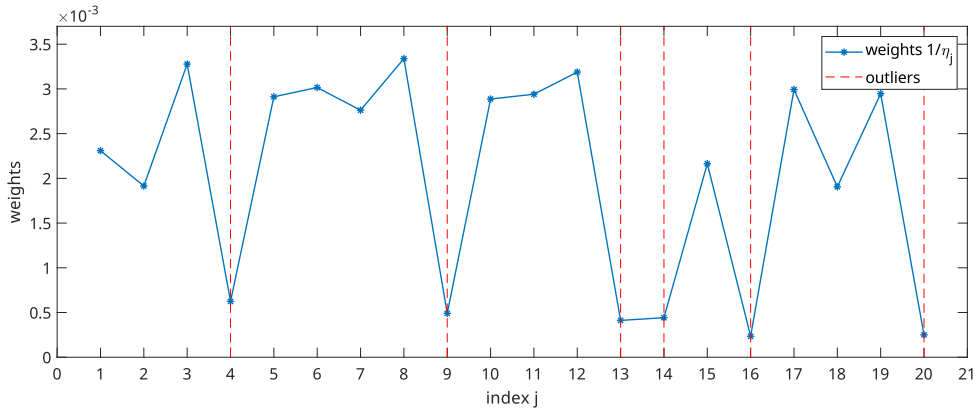


Figure 13: **Weights of the IRLS procedure**. We synthesized a sequence u_0 containing $L = 20$ low-resolution images using a sequence of displacements δ. Then, we computed a perturbed sequence of displacements δ' using two levels of perturbations: 70% of the displacements were perturbed using a Gaussian noise with a *small* standard deviation $\sigma_1 = 0.01$ while 30% of them were perturbed using a Gaussian noise with a *large* standard deviation $\sigma_2 = 0.3$. We display using blue-marks the weights $w_j = 1/\eta_j$ $(1 \leq j \leq L)$ obtained using the IRLS procedure (56) over the sequences u_0 and δ'. We indicate using red-dashed lines the indexes of the *outliers* in u_0, that is, the indexes j of the elements of δ' undergoing a large perturbation. As we can see, those outliers are associated to significantly low values of w_j compared to the others.

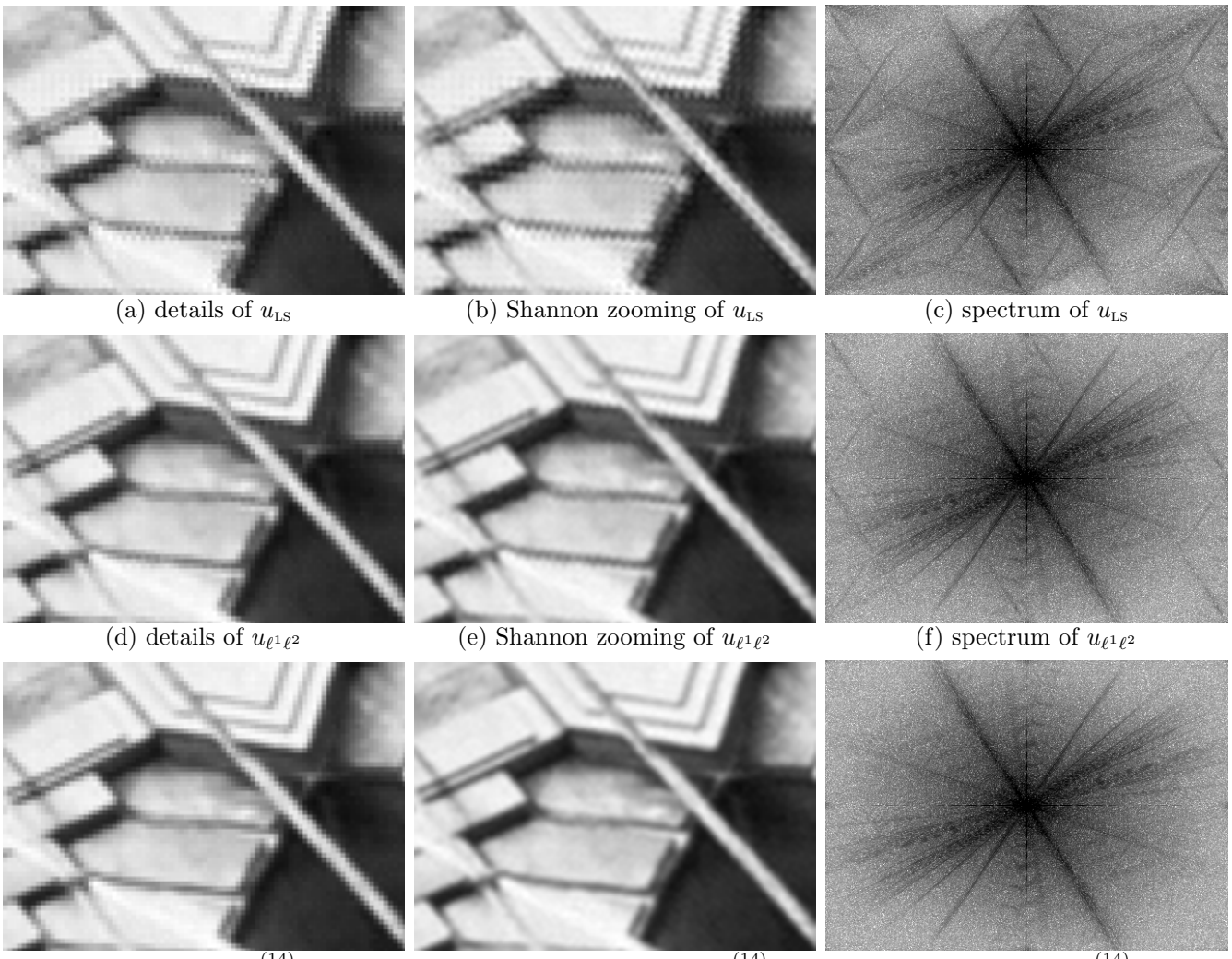
consider here, the situation is much simpler: the linear system $Au = u_0$ is overdetermined, so it is extremely unlikely to observe a zero residual $\|A_j u - u_0^{(j)}\|_2$. In all the experiments we conducted, we used the parameter setting $\varepsilon = 0$ in (56) and never encountered any numerical issue, so this discussion is mostly rhetorical.

As illustrated in Figure 13, although the scheme (56) successfully performs the minimization (54) after a surprisingly low number of iterations (typically ten in the setting of Figure 13) and although the weights $w_j = 1/\eta_j$ associated to outliers in u_0 are significantly lower than the others, they do not vanish. Therefore, those outliers are still responsible for artifacts in the image $u_{\ell^1\ell^2}$, as illustrated in the second row of Figure 14. To further improve the quality of the reconstruction, we propose to remove from the original sequence u_0 a certain number of images corresponding to the highest values of η, and to compute a new super-resolved image from this "outlier-free" sequence. In the end, the full lucky-imaging strategy we here propose can be summarized in three steps:

1. Compute the residuals η_j by iterating (56) until convergence;
2. Select from u_0 the $\mathcal{N}$ images (for some value of $\mathcal{N} \leq L$ discussed later) associated to the $\mathcal{N}$ highest values of η_j;
3. Process these $\mathcal{N}$ images with the least-squares model (17) using Algorithm 2, to obtain the output high-resolution image.

In a fully automatic super-resolution system, it would be interesting to have a specific algorithm that could find the "optimal" value of $\mathcal{N}$. In most super-resolution situations, however, image quality is the primary concern, and manual selection is not a problem, considering the time already spent for setup and data acquisition. In the code associated to this paper, an option proposes to compute the images $u_{\text{lucky}}^{(\mathcal{N})}$ for all possible values of $\mathcal{N}$, so that the manual selection of the optimal value of $\mathcal{N}$ can be realized very quickly by a simple visual inspection of the whole image sequence $(u_{\text{lucky}}^{(\mathcal{N})})_{\mathcal{N}}$. The benefit of this lucky-imaging procedure is illustrated on a synthetic experiment in Figure 14, and also on real data in Figure 15.

Notice that the idea of using the IRLS algorithm to perform robust super-resolution was already proposed in [65], but using the ℓ^1-norm instead of $\|\cdot\|_{1,2}$ in (54), and without the re-estimation procedure (steps 2 and 3) described above. In such situation, η has as many entries as the stack u_0 (that is, $\eta = (\eta_1, \dots, \eta_L)$ and $\eta_j : \omega \to \mathbb{R}_+^*$) instead of lying in $(\mathbb{R}_+^*)^L$ and the update of u cannot be efficiently computed using Algorithm 2, essentially because Proposition 3 and Proposition 4 do not hold anymore when changing $A_j u$ into the pointwise division between $A_j u$ and $\sqrt{\eta_j}$.

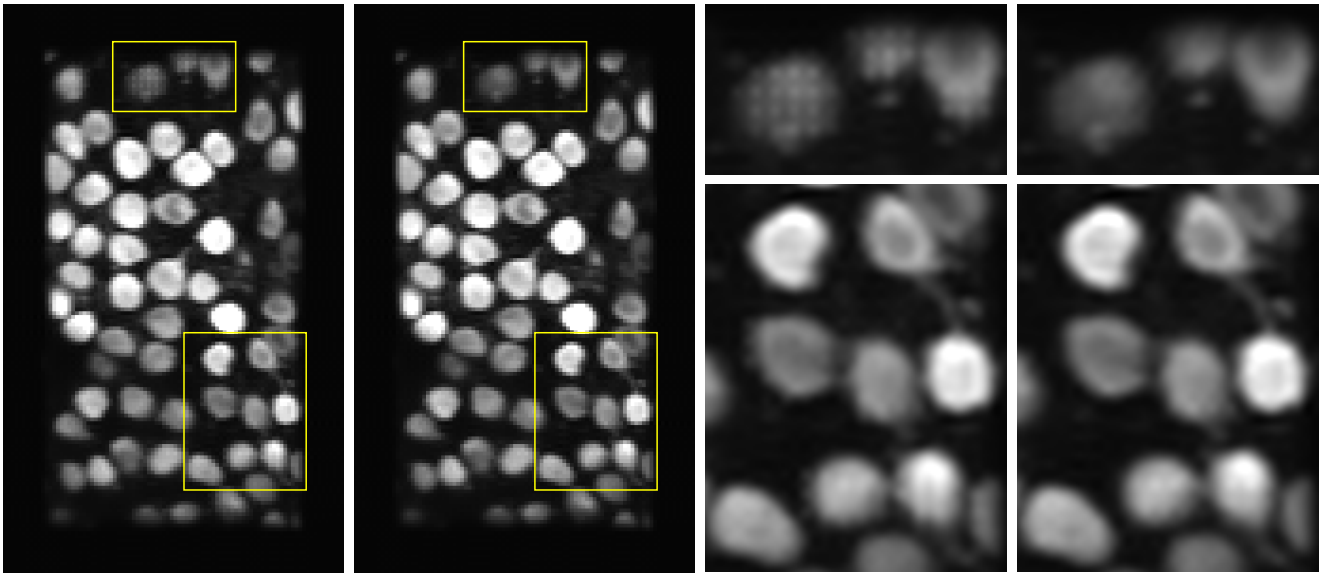


(a) details of u_{LS} (b) Shannon zooming of u_{LS} (c) spectrum of u_{LS}

(d) details of $u_{\ell^1\ell^2}$ (e) Shannon zooming of $u_{\ell^1\ell^2}$ (f) spectrum of $u_{\ell^1\ell^2}$

(g) details of $u_{\text{lucky}}^{(14)}$ (h) Shannon zooming of $u_{\text{lucky}}^{(14)}$ (i) spectrum of $u_{\text{lucky}}^{(14)}$

Figure 14: **Super-resolution in presence of outliers**. We processed the sequence u_0 using the perturbed displacements δ' described in Figure 13. Due to the presence of strongly inaccurate displacements in δ', the least-squares reconstruction u_{LS} obtained from u_0 and δ' exhibits severe artifacts at the pixel scale (a), at the sub-pixel scale (b) and in the frequency domain (c). As we can see on the second row, those artifacts are attenuated in the image $u_{\ell^1\ell^2}$ obtained with the IRLS scheme (56). Unfortunately they remain slightly visible at the pixel scale (d), and significantly visible at the sub-pixel scale (e) and in the frequency domain (f). When removing 6 well-chosen images from the low-resolution sequence u_0 with the proposed lucky-imaging procedure, we obtain the image $u_{\text{lucky}}^{(14)}$ (third row), which is free of artifacts (g)-(i). This experiment illustrates that, although the minimization (54) is more robust than the least-squares minimization (17), it does not completely get rid of the artifacts caused by the presence of outliers in u_0. Fortunately, the IRLS procedure provides, through the weights that characterize outlier images, a very efficient way to avoid these artifacts with the proposed lucky-imaging procedure.



(a) least-squares over the whole sequence u_0 | (b) $u_{\text{lucky}}^{(2886)}$: least-squares after discarding 300 images from u_0 | bicubic zoomings of (a) | bicubic zoomings of (b)

Figure 15: **Lucky-imaging over the 2-photons microscopic sequence**. The original sequence contains $L = 3186$ low-resolution images, and the lucky-imaging super-resolution result (b) (also displayed in Figure 11) actually corresponds to the image $u_{\text{lucky}}^{(2886)}$ obtained after removing 300 images from the original sequence. We can notice the quality improvement brought by the lucky-imaging procedure, by comparing (b) to the image (a) obtained with the whole original sequence. In particular, looking at the close-ups on the two rightmost columns, we can observe (first row) the disappearance of some spurious patterns (which were probably due to the incorrect registration of several images of u_0), and (second row) that the shapes of the cells in (b) are more regular than in (a). These improvements, though not as spectacular as those observed on the synthetic experiment presented in Figure 14, remain valuable, especially considering the simplicity and low computational cost of the proposed method, and the complexity of changes in the original sequence (illumination changes, non-rigid deformations, irregular sampling due to line scanning, ...)

7 Super-resolution and Deconvolution

Super-resolution aims at recovering the high-frequency content of the scene u_{HR} from a low-resolution sequence $u_0 \approx Au_{\text{HR}}$. The inversion of A corresponds to a de-aliasing operation that unfolds from u_0 the high resolution coefficients $\widehat{u}_{\text{HR}}(\widehat{\Omega} \setminus \widehat{\omega})$ and put them back to their original positions in the high-resolution domain $\widehat{\Omega}$. However, in practice, the high-frequency content of the image u_{HR} is not only aliased in u_0 but also attenuated because of the filtering operated by the imaging system. The physical sources of this filtering are manifold: beyond expected contributions like optical diffraction and sensor integration, more complicated distortions may also be involved, such as motion blur, defocus, atmospheric turbulences, or optical aberrations for example. Denoting by $B : \mathbb{R}^{\Omega} \to \mathbb{R}^{\Omega}$ the associated filtering operator, we can write $u_0 \approx ABu_{\text{HR}}$, and the least-squares inversion of A applied to u_0 yields an estimate of the filtered image $u_{\text{HR}}^{\text{filtered}} := Bu_{\text{HR}}$. If we assume that the operator B is linear and shift-invariant, then B corresponds to a convolution operator and its inversion can be addressed using standard deconvolution models. If B is unknown, or partially known, blind deconvolution methods can be used.

When the ratio $\frac{L}{\lceil z_x \rceil . \lceil z_y \rceil}$ is significantly greater than 1, this deconvolution is even more relevant, because the signal-to-noise ratio is increased between u_0 and u_{LS}. Note that we know from Section 4 that in this case the variance of the reconstruction error in the Fourier domain is essentially constant

(see Figure 4 and discussion in the end of Section 4.2), so we here choose not to use the information provided by the amplification map $\mathcal{A}$ in the deconvolution process.

There are plenty of blind deconvolution algorithms in the literature, and some of them specifically studied in the context of super-resolution (e.g. [4, 20]), but the aim of this section is not to obtain the best enhanced super-resolution image but rather to try and assess as objectively as possible the quality of the high-frequency content reconstructed by the algorithm we proposed. Hence, instead of using a sophisticated algorithm (which could legitimately claim some credit for the quality of the deconvolved image), we shall use the simplest possible deconvolution method, that is, a direct frequency-wise amplification of the Fourier transform of the image.

In the following, we demonstrate the usefulness of such a simple linear enhancement filter, as it significantly improves the sharpness of the reconstructed high-resolution image u_{LS}. Since the map $u_0 \mapsto u_{\text{LS}}$ is itself linear, it makes sense to keep using a linear (and translation-invariant) operator to enhance u_{LS}, in particular because we know precisely the reconstruction artifacts that are expected on u_{LS} (trough the analysis of the expected error in the Fourier domain, see Section 4). Using a non-linear enhancement operator could lead to a better enhancement of u_{LS} but would break the linearity of the whole reconstruction chain and would certainly limit our ability to understand and control artifacts.

The enhancement filter we propose consists in the convolution of u_{LS} with the kernel $\psi_\lambda : \Omega \to \mathbb{R}$, defined in the Fourier domain by

$$\forall (\alpha,\beta) \in \widehat{\Omega}\,, \quad \widehat{\psi}_\lambda(\alpha,\beta) = 1 + \lambda(1 - e^{-\rho(\alpha,\beta)})\,, \quad \text{denoting} \quad \rho(\alpha,\beta) = \sqrt{(2\alpha/M)^2 + (2\beta/N)^2}\,, \tag{57}$$

where $\lambda > 0$ is an amplification parameter. In other terms, the enhanced super-resolved image u_λ is obtained by

$$\forall (\alpha,\beta) \in \widehat{\Omega}\,, \quad \widehat{u}_\lambda(\alpha,\beta) = \widehat{\psi}_\lambda(\alpha,\beta) \cdot \widehat{u}_{\text{LS}}(\alpha,\beta)\,. \tag{58}$$

The radial profile of $\widehat{\psi}$ is displayed in Figure 16. Note that $\rho(\alpha,\beta)$ takes values in $[0,1]$ along each frequency axis, and attain its maximum value $\sqrt{2}$ at the corners of the frequency domain. In the experiment below, we took $\lambda = 5$ but this value (which corresponds to the slope of the amplification profile at $\rho = 0$) may be adapted in practice to the value of $\frac{L}{\lceil z_x \rceil . \lceil z_y \rceil}$ to avoid an excessive amplification of the reconstruction error.

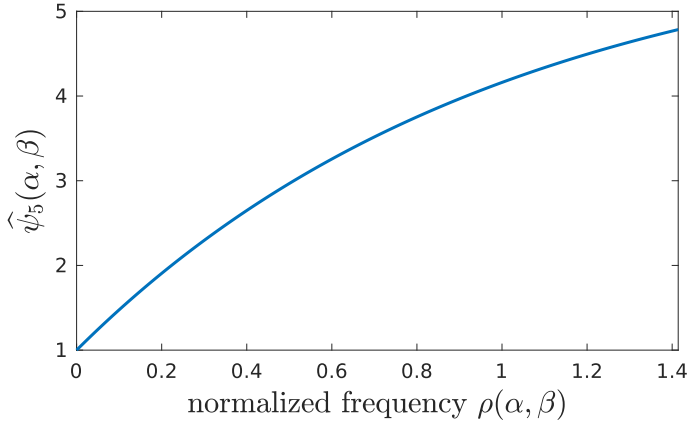


Figure 16: **Radial profile (in the Fourier domain) of the enhancement filter $\widehat{\psi}_5$**

Note that (58) corresponds in spatial domain to a periodic (or circular) convolution, which could in general produce unwanted ringing artifacts near the image boundaries. Here, these artifacts are avoided because u_{LS} vanishes near the boundaries of Ω thanks to the apodization procedure applied to u_0 (see Section 2.3).

In Figure 17, we illustrate on real (thermal infrared) data how the enhancement filter ψ_5 improves the sharpness of the super-resolved image u_{LS}, without producing noticeable artifacts.

In this section, for simplicity, we performed super-resolution and deconvolution (or, more exactly, image enhancement) as two separate operations. Another possibility would be to perform both operations jointly in a single step. Zhao *et al* [78] showed how to obtain a fast closed form solution for this joint deconvolution-super-resolution problem in the single-image case, but the idea could possibly be generalized to the multi-image case.

8 Conclusion and Perspectives

In this paper we described and analyzed the direct least-squares estimator for the MISR problem in the case of translational motion and Shannon interpolation. The solution is computed in linear time with respect to the size of the input image sequence, using Fourier transforms and inversion of at most four low-dimensional matrices. The spectral unfolding realized by the algorithm is directly observed with Fourier transform images, on both synthetic and real data, which permits to better distinguish between true super-resolved details and interpolated features.

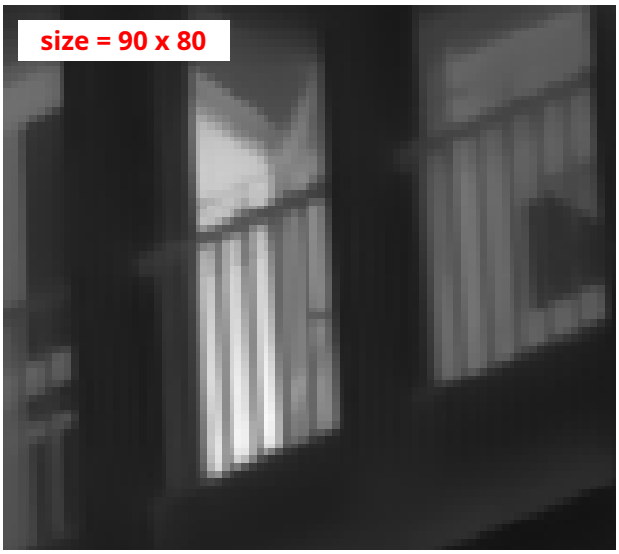
A quantitative analysis of the reconstruction error was provided, both in space and Fourier domains, and it appeared to be much more accurate than those based on the condition number criterion classically used in the literature. We guess that such a precise control of the accuracy of the result could be particularly useful for metrology applications.

All implementation details were given, in particular the important issue raised by the implicit periodicity of the discrete Fourier transform, which may cause severe propagating artifacts at the image boundaries. The case of non-integer zoom factors was also fully detailed, which is an important feature because optimal trade-offs between resolution and noise are rarely attained for integer values.

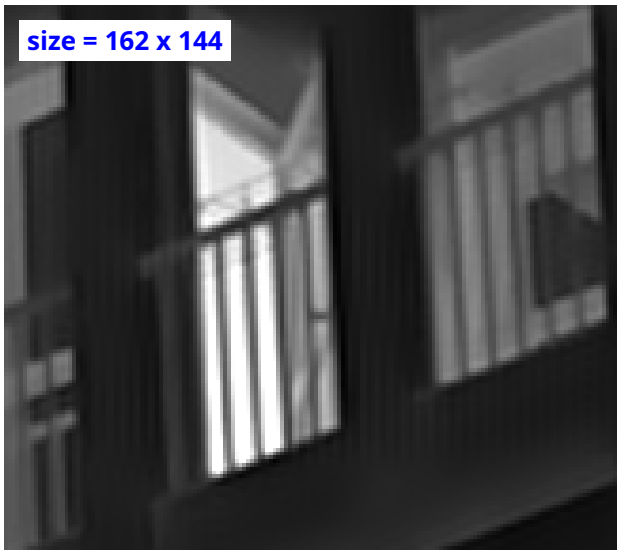
Keeping a linear processing pipeline not only yields a direct algorithm (without need for GPU acceleration, or "regularization by early stopping" as mentioned in [4]), but also guarantees a perfect control of the image sampling. A consequence is that the reconstructed image can be reinterpolated or spectrally enhanced without producing artifacts, which is an important feature for industrial applications.

The proposed algorithm was validated on synthetic data, and then tested on two real image sequences (processed directly, without any simulation). On these real data, apart from the spectral enhancement post-processing, we found that the $\ell^{1,2}$ variant of the proposed method (which, as we showed, can be efficiently implemented with an IRLS algorithm) brings a very useful feature in its ability to remove outliers, which may significantly increase the quality of the reconstructed image.

Throughout this study we assumed that the motion was estimated beforehand (and used the efficient algorithm of [32] to process real data), but it would certainly be beneficial to integrate this estimation step into a more elaborated model. Indeed, since the estimation of the super-resolution image u_{LS} is very fast, it could be an internal step in an iterative scheme that would jointly estimate the motion and the super-resolved image by progressively refining the motion estimation so as to improve the reconstruction quality. More generally, since estimating the proximal operator associated to a ℓ^2 data fidelity term in the MISR problem is exactly equivalent (up to a tiny modification) to the estimation of u_{LS}, the algorithm we provided could be easily used as a basic component in more sophisticated systems, allowing for example joint super-resolution and deblurring, or super-resolution regularized with advanced image priors like neural networks.



(a) median of the registered low-resolution sequence



(b) bicubic zooming of (a) and enhancement

(c) u_{LS}: least-squares super-resolution

(d) enhancement of (c)

Figure 17: **Effect of the enhancement filter ψ_5 on the super-resolved image**. We consider again the thermal infrared sequence u_0 of $L = 500$ low-resolution images with size 200×200 (see Section 5.5 and Figure 12). We display in (a) a 90×80 close-up of the shift-and-median reconstruction (that is, the pointwise median of the registered sequence of low-resolution images). By magnifying this reconstruction with a factor 1.8 in both directions (bicubic zooming) we get an image with size 360×360 on which we apply the enhancement filter defined by ψ_5, leading to the close-up image (b). We can see that the enhancement yields an image with better contrast and a better level of details, but at the same time, amplified artifacts. We display in (c) and (d) equivalent close-ups for the least-squares super-resolution image u_{LS} obtained from u_0 (zoom factors: $z_x = z_y = 1.8$). It is interesting to see that the enhancement of u_{LS}, in addition to its ability to better reveal super-resolution details, does not produce any noticeable artifact.

Acknowledgment

This work was partially funded by the Centre National d'Études Spatiales (CNES), through the Research & Technology project "Traitement de super-résolution d'images spatiales THR", from September 2017 to September 2018. The authors are very grateful to J. Ramirez and his colleagues I. Llano, B. Stell, M. Kassa from the Saints-Pères Paris Institute for the Neurosciences (Université Paris Cité, CNRS UMR 8003) for having shared with us their two-photons microscopy images and for the fruitful discussions we had. Many thanks also to our colleague C. Pouzat, for his participation to those interesting discussions, and to V. Goepp, for the helpful references and advice about reweighted least-squares algorithms.

A Proof of Proposition 4

Since $\pi_{\widehat{\omega}} : \widehat{\Omega} \to \widehat{\omega}$ is onto, the sets $\pi_{\widehat{\omega}}^{-1}(a,b)$ form, for $(a,b) \in \widehat{\Omega}$, a partition of $\widehat{\Omega}$, and the linear relations (32) form, up to a proper ordering of the canonical basis of $\widehat{\Omega}$, a block-diagonal linear operator relating $\widehat{v}$ to $\widehat{w}$. Thus, the relation $v = (A^*A)w$ can be written, in matrix form,

$$R\widehat{v} = DR\widehat{w}, \tag{59}$$

where R is a permutation matrix (corresponding to the reordering of the canonical basis of $\widehat{\Omega}$) and D is a block-diagonal matrix formed by the $\mathcal{M}(a,b)$ matrices. Writing F the matrix of the Fourier Transform from Ω to $\widehat{\Omega}$, we obtain $RFv = DRFw$, so that $A^*A = F^*R^{-1}DRF$. Now, recall that if X is a unitary $n \times n$ matrix (that is, satisfying $X^*X = I$) and Y any $n \times n$ matrix, then $(XY)^\dagger = Y^\dagger X^{-1}$ and $(YX)^\dagger = X^{-1}Y^\dagger$. Since F^*, R^{-1}, R and F are unitary, we obtain

$$(A^*A)^\dagger = (F^*R^{-1}DRF)^\dagger = F^*R^{-1}D^\dagger RF. \tag{60}$$

Consequently, if two images v and w satisfy $w = (A^*A)^\dagger v$, then

$$R\widehat{w} = D^\dagger R\widehat{v}, \tag{61}$$

which is exactly (34) since the pseudo-inverse $D^\dagger$ of the block-diagonal matrix D is simply obtained by taking the pseudo-inverse of each block $\mathcal{M}(a,b)$. □

B Proof of Lemma 1

Proof. Let $(a,b) \in \widehat{\omega}$. Thanks to Proposition 1, we have

$$\pi_{\widehat{\omega}}^{-1}(a,b) = \Big((a + m\mathbb{Z}) \times (b + n\mathbb{Z})\Big) \cap \widehat{\Omega}\,,$$

which can be rewritten

$$\pi_{\widehat{\omega}}^{-1}(a,b) = \big(a + mP(a)\big) \times \big(b + nQ(b)\big)\,,$$

with

$$P(a) = \left\{p \in \mathbb{Z}\,,\ -\frac{M}{2} \le a + pm < \frac{M}{2}\right\} \quad \text{and} \quad Q(b) = \left\{q \in \mathbb{Z}\,,\ -\frac{N}{2} \le b + qn < \frac{N}{2}\right\}.$$

Since

$$P(a) = \left\{ p \in \mathbb{Z}\,,\ -\frac{M}{2m} - \frac{a}{m} \leq p < \frac{M}{2m} - \frac{a}{m} \right\},$$

we get

$$P(a) = \big[p_{\min}(a), p_{\max}(a)\big] \cap \mathbb{Z} \quad \text{with} \quad p_{\min}(a) = \left\lceil -\frac{M}{2m} - \frac{a}{m} \right\rceil \text{ and } p_{\max}(a) = \left\lceil \frac{M}{2m} - \frac{a}{m} \right\rceil - 1\,,$$

and symmetrically

$$Q(b) = \big[q_{\min}(b), q_{\max}(b)\big] \cap \mathbb{Z} \quad \text{with} \quad q_{\min}(b) = \left\lceil -\frac{N}{2n} - \frac{b}{n} \right\rceil \text{ and } q_{\max}(b) = \left\lceil \frac{N}{2n} - \frac{b}{n} \right\rceil - 1$$

as announced in Lemma 1. □

C Proof of Lemma 3

Assume that m and n are odd, and that z_x and z_y are integers. We consider the partition of $\widehat{\omega}$ given by

$$\widehat{\omega} = D_1 \cup D_2 \cup \{(0,0)\} \quad \text{with} \quad D_1 = \{(a,b) \in \widehat{\omega},\ a > 0 \text{ or } (a = 0 \text{ and } b > 0)\},$$

and prove using Equation (43) that for fixed ℓ, the random variables $\left(\widehat{\varepsilon'}(\alpha_\ell^{a,b}, \beta_\ell^{a,b})\right)_{(a,b)\in D_1}$ are i.i.d. Indeed, as the Fourier coefficients of a Gaussian white noise, the random variables

$$\left(\widehat{\varepsilon}^{(j)}(a,b)\right)_{(a,b)\in D_1, 1\leq j\leq L}$$

are i.i.d. Moreover, from Remark 4 we know that $|c_\ell^{(j)}(a,b)|$ does not depend on (a,b) (there is only one matrix $\mathcal{M}(a,b)$), so we can write this number $d_\ell^{(j)}$. Consequently, since the random variables $\widehat{\varepsilon}^{(j)}(a,b)$ follow a phase-invariant distribution, the law of

$$\widehat{\varepsilon'}(\alpha_\ell^{a,b}, \beta_\ell^{a,b}) = \sum_{j=1}^{L} c_\ell^{(j)}(a,b) \cdot \widehat{\varepsilon}^{(j)}(a,b)$$

and the law of

$$\sum_{j=1}^{L} d_\ell^{(j)} \cdot \widehat{\varepsilon}^{(j)}(a,b)$$

are identical, which means that for ℓ fixed, all random variables $\left(\left|\widehat{\varepsilon'}(\alpha_\ell^{a,b}, \beta_\ell^{a,b})\right|^2\right)_{(a,b)\in D_1}$ are i.i.d. Consequently, if we set

$$S_1(\ell) := \frac{1}{m^2 n^2} \sum_{(a,b)\in D_1} \left|\widehat{\varepsilon'}(\alpha_\ell^{a,b}, \beta_\ell^{a,b})\right|^2,$$

we have that

$$\mathrm{Var}(S_1(\ell)) = \frac{1}{m^4 n^4} \sum_{(a,b)\in D_1} \mathrm{Var}\left(\left|\widehat{\varepsilon'}(\alpha_\ell^{a,b}, \beta_\ell^{a,b})\right|^2\right),$$

$$\text{with} \quad \operatorname{Var}\left(\left|\widehat{\varepsilon'}(\alpha_\ell^{a,b},\beta_\ell^{a,b})\right|^2\right) = \operatorname{Var}\left(\left|\sum_{j=1}^{L} d_\ell^{(j)} \cdot \widehat{\varepsilon}^{(j)}(a,b)\right|^2\right)$$
$$= \sum_{j=1}^{L} (d_\ell^{(j)})^4 \operatorname{Var}\left(\left|\widehat{\varepsilon}^{(j)}(a,b)\right|^2\right)$$
$$= 4m^2n^2 \sum_{j=1}^{L} (d_\ell^{(j)})^4,$$

$$\text{so that} \quad \operatorname{Var}(S_1(\ell)) = \frac{2(mn-1)}{m^2n^2} \sum_{j=1}^{L} (d_\ell^{(j)})^4.$$

Since each coefficient $d_\ell^{(j)}$ can be bounded independently of m and n, and L is fixed, we obtain that $\operatorname{Var}(S_1(\ell)) \to 0$ as $mn \to +\infty$. If we set

$$S_2(\ell) := \frac{1}{m^2n^2} \sum_{(a,b)\in D_2} \left|\widehat{\varepsilon'}(\alpha_\ell^{a,b},\beta_\ell^{a,b})\right|^2,$$

we can prove similarly that $\operatorname{Var}(S_2(\ell)) \to 0$ as $mn \to +\infty$. Note that D_1 and D_2 must be handled separately because the Hermitian symmetry (entailed by the fact that ε is a real-valued image) prevent the variables $\widehat{\varepsilon}^{(j)}(a,b)$ and $\widehat{\varepsilon}^{(j)}(-a,-b)$ from being independent. We now examine the variance of

$$\begin{aligned} E &= \frac{1}{mn}\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 \\ &= \frac{1}{mn}\frac{\sigma^2}{MN} \sum_{(\alpha,\beta)\in\widehat{\Omega}} \left|\widehat{\varepsilon'}(\alpha,\beta)\right|^2 \\ &= \frac{\sigma^2}{z_x z_y} \sum_{\ell=1}^{z_x z_y} \frac{1}{m^2n^2} \sum_{(a,b)\in\widehat{\omega}} \left|\widehat{\varepsilon'}(\alpha_\ell^{a,b},\beta_\ell^{a,b})\right|^2 \\ &= \frac{\sigma^2}{z_x z_y} \sum_{\ell=1}^{z_x z_y} \left(S_1(\ell) + S_2(\ell) + S_3(\ell)\right), \qquad \text{with} \quad S_3(\ell) = \frac{1}{m^2n^2}\left|\widehat{\varepsilon'}(\alpha_\ell^{0,0},\beta_\ell^{0,0})\right|^2. \end{aligned}$$

Since we also have

$$\operatorname{Var}(S_3(\ell)) = \frac{2mn}{m^4n^4} \underset{mn\to+\infty}{\longrightarrow} 0,$$

we obtain that $\operatorname{Var}(E) \to 0$ as $mn \to +\infty$. Considering the fact that $\mathbb{E}(E)$ stay bounded away from 0 as $mn \to +\infty$, this proves that, almost surely,

$$\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2 \underset{mn\to+\infty}{\sim} \mathbb{E}\|U_{\text{LS}} - \overline{u_{\text{LS}}}\|_2^2.$$

The cases when m and/or n is even can be handled similarly by considering separately, like $(0,0)$ in the case above, the frequency coefficients that are constrained to be real-valued.

When z_x and z_y are not necessarily integers, the reasoning is the same as the one above, but in that case the partition of $\widehat{\omega}$ is more complex because there can be up to 4 different matrices $\mathcal{M}(a,b)$, which implies that for each ℓ and each j there can be up to 4 different values for $|c_\ell^{(j)}(a,b)|$. □

References

[1] R. Abergel, O. Bouaziz, and G. Nuel, *A Review on the Adaptive-Ridge Algorithm with several extensions*, Statistics and Computing, 34 (2024). https://doi.org/10.1007/s11222-024-10440-6.

[2] R. Abergel and L. Moisan, *The Shannon Total Variation*, Journal of Mathematical Imaging and Vision, 59 (2017), pp. 341–370. https://doi.org/10.1007/s10851-017-0733-5.

[3] A. Almansa, S. Durand, and B. Rougé, *Measuring and Improving Image Resolution by Adaptation of the Reciprocal Cell*, Journal of Mathematical Imaging and Vision, 21 (2004), pp. 235–279. https://doi.org/10.1023/B:JMIV.0000043739.51886.01.

[4] J. Anger, T. Ehret, C. de Franchis, and G. Facciolo, *Fast and accurate multi-frame super-resolution of satellite images*, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, V-1-2020 (2020), pp. 57–64. https://doi.org/10.5194/isprs-annals-V-1-2020-57-2020.

[5] N. Baba, S. Isobe, Y. Norimoto, and M. Noguchi, *Stellar speckle reconstruction by the shift-and-add method.*, Applied Optics, 24 (1985), pp. 1403–1405. https://doi.org/10.1364/AO.24.001403.

[6] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, *Optimization with sparsity-inducing penalties*, Foundations and Trends in Machine Learning, 4 (2012), pp. 1–106. https://doi.org/10.1561/2200000015.

[7] S. Baker and T. Kanade, *Limits on super-resolution and how to break them*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (2002), pp. 1167–1183. https://doi.org/10.1109/TPAMI.2002.1033210.

[8] M. Ben-Ezra, A. Zomet, and S. K. Nayar, *Jitter Camera: High Resolution Video from a Low Resolution Detector*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, 2004, pp. 135–142. https://doi.org/10.1109/CVPR.2004.1315155.

[9] M. Ben-Ezra, A. Zomet, and S. K. Nayar, *Video Super-Resolution Using Controlled Subpixel Detector Shifts*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 27 (2005), pp. 977–987. https://doi.org/10.1109/TPAMI.2005.129.

[10] G. Bhat, M. Danelljan, R. Timofte, K. Akita, W. Cho, H. Fan, L. Jia, D. Kim, B. Lecouat, Y. Li, S. Liu, Z. Liu, Z. Luo, T. Maeda, J. Mairal, C. Micheloni, X. Mo, T. Oba, P. Ostyakov, J. Ponce, S. Son, J. Sun, N. Ukita, R. Muhammad, U. Y. Yan, L. Yu, M. Zhussip, and X. Zou, *NTIRE 2021 challenge on burst super-resolution: Methods and results*, IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, (2021), pp. 613–626. https://doi.org/10.1109/CVPRW53098.2021.00073.

[11] T. Briand and J. Vacher, *How to Apply a Filter Defined in the Frequency Domain by a Continuous Function*, Image Processing On Line, 6 (2016), pp. 183–211. https://doi.org/10.5201/ipol.2016.116.

[12] F. Champagnat, G. Le Besnerais, and C. Kulcsár, *Statistical performance modeling for superresolution: a discrete data-continuous reconstruction framework*, Journal of the Optical Society of America A, 26 (2009), pp. 1730–1746. https://doi.org/10.1364/JOSAA.26.001730.

[13] M.-C. Chiang and T. Boult, *Efficient super-resolution via image warping*, Image and Vision Computing, 18 (2000), pp. 761–771. https://doi.org/10.1016/S0262-8856(99)00044-X.

[14] J. C. Christou, *Image Quality, Tip-Tilt Correction, and Shift-and-Add Infrared Imaging*, Publications of the Astronomical Society of the Pacific, 103 (1991), p. 1040. https://doi.org/10.1086/132922.

[15] V. Cornillère, A. Djelouah, W. Yifan, O. Sorkine-Hornung, and C. Schroers, *Blind image super-resolution with spatially variant degradations*, in ACM Transactions on Graphics, 2019. https://doi.org/10.1145/3355089.3356575.

[16] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, *Iteratively reweighted least squares minimization for sparse recovery*, Communications on Pure and Applied Mathematics, 63 (2010), pp. 1–38. https://doi.org/10.1002/cpa.20303.

[17] Q. Denoyelle, V. Duval, G. Peyré, and E. Soubies, *The sliding frank–wolfe algorithm and its application to super-resolution microscopy*, Inverse Problems, 36 (2019), p. 014001. https://doi.org/10.1088/1361-6420/ab2a29.

[18] C. Dong, C. C. Loy, K. He, and X. Tang, *Learning a deep convolutional network for image super-resolution*, in Proceedings of European Conference on Computer Vision (ECCV), D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds., Springer International Publishing, 2014, pp. 184–199. https://doi.org/10.1007/978-3-319-10593-2_13.

[19] T. Dong, H. Yan, M. Parasar, and R. Krisch, *RenderSR: A lightweight super-resolution model for mobile gaming upscaling*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2022, pp. 3087–3095. https://doi.org/10.1109/CVPRW56347.2022.00348.

[20] T. Eboli, J.-M. Morel, and G. Facciolo, *Breaking down Polyblur: Fast Blind Correction of Small Anisotropic Blurs*, Image Processing On Line, 12 (2022), pp. 435–456. https://doi.org/10.5201/ipol.2022.405.

[21] G. Facciolo, A. Almansa, J.-F. Aujol, and V. Caselles, *Irregular to Regular Sampling, Denoising, and Deconvolution*, Multiscale Modeling & Simulation, 7 (2009), pp. 1574–1608. https://doi.org/10.1137/080719443.

[22] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, *Advances and challenges in super-resolution*, International Journal of Imaging Systems and Technology, 14 (2004), pp. 47–57. https://doi.org/10.1002/ima.20007.

[23] S. Farsiu, M. Robinson, M. Elad, and P. Milanfar, *Fast and robust multiframe super resolution*, IEEE Transactions on Image Processing, 13 (2004), pp. 1327–1344. https://doi.org/10.1109/TIP.2004.834669.

[24] M. Frigo and S. G. Johnson, *The design and implementation of FFTW3*, Proceedings of the IEEE, 93 (2005), pp. 216–231. Special issue on "Program Generation, Optimization, and Platform Adaptation". https://doi.org/10.1109/JPROC.2004.840301.

[25] D. Geman and G. Reynolds, *Constrained restoration and the recovery of discontinuities*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 14 (1992), p. 367–383. https://doi.org/10.1109/34.120331.

[26] D. Geman and C. Yang, *Nonlinear image recovery with half-quadratic regularization*, IEEE Transactions on Image Processing, 4 (1995), pp. 932–946. https://doi.org/10.1109/83.392335.

[27] K. Gröchenig and T. Strohmer, *Numerical and Theoretical Aspects of Nonuniform Sampling of Band-Limited Images*, Springer US, 2001, ch. 6, pp. 283–324. https://doi.org/10.1007/978-1-4615-1229-5_6.

[28] J. Gu, H. Lu, W. Zuo, and C. Dong, *Blind super-resolution with iterative kernel correction*, in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 1604–1613. https://doi.org/10.1109/CVPR.2019.00170.

[29] M. Guerquin-Kern, L. Lejeune, K. P. Pruessmann, and M. Unser, *Realistic analytical phantoms for parallel magnetic resonance imaging*, IEEE Transactions on Medical Imaging, 31 (2012), pp. 626–636. https://doi.org/10.1109/TMI.2011.2174158.

[30] M. Irani and S. Peleg, *Super resolution from image sequences*, in Proceedings of the 10th International Conference on Pattern Recognition, vol. ii, 1990, pp. 115–120. https://doi.org/10.1109/ICPR.1990.119340.

[31] ——, *Improving resolution by image registration*, CVGIP: Graphical Models and Image Processing, 53 (1991), pp. 231–239. https://doi.org/10.1016/1049-9652(91)90045-L.

[32] D. Keren, S. Peleg, and R. Brada, *Image sequence enhancement using sub-pixel displacements*, in Proceedings CVPR '88: The Computer Society Conference on Computer Vision and Pattern Recognition, June 1988, pp. 742–746. https://doi.org/10.1109/CVPR.1988.196317.

[33] J. Lafenetre, G. Facciolo, and T. Eboli, *Implementing Handheld Burst Super-Resolution*, Image Processing On Line, 13 (2023), pp. 227–257. https://doi.org/10.5201/ipol.2023.460.

[34] S. Lanzavecchia and P. L. Bellon, *A moving window Shannon reconstruction algorithm for image interpolation*, Journal of Visual Communication and Image Representation, 5 (1994), pp. 255–264. https://doi.org/10.1006/jvci.1994.1023.

[35] B. Lecouat, T. Eboli, J. Ponce, and J. Mairal, *High dynamic range and super-resolution from raw image bursts*, ACM Transactions on Graphics, 41 (2022). https://doi.org/10.1145/3528223.3530180.

[36] B. Lecouat, J. Ponce, and J. Mairal, *Lucas-Kanade Reloaded: End-to-End Super-Resolution From Raw Image Bursts*, in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2021, pp. 2370–2379. https://doi.org/10.1109/ICCV48922.2021.00237.

[37] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, *SwinIR: Image Restoration Using Swin Transformer*, in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, October 2021, pp. 1833–1844. https://doi.org/10.1109/ICCVW54120.2021.00210.

[38] J. Mairal, F. Bach, and J. Ponce, *Sparse modeling for image and vision processing*, Foundations and Trends in Computer Graphics and Vision, 8 (2014), pp. 85–283. https://doi.org/10.1561/0600000058.

[39] F. Malgouyres and F. Guichard, *Edge direction preserving image zooming: a mathematical and numerical analysis*, SIAM Journal on Numerical Analysis, 39 (2001), pp. 1–37. https://doi.org/10.1137/S0036142999362286.

[40] A. Marquina and S. J. Osher, *Image Super-Resolution by TV-Regularization and Bregman Iteration*, Journal of Scientific Computing, 37 (2008), pp. 367–382. https://doi.org/10.1007/s10915-008-9214-8.

[41] A. Materne, O. Puig, and P. Kubik, *Capteur d'images avec correction de bougé par recalage numérique de prises de vues fractionnées*. CNES Patent, FR2976754, 21/12/2012.

[42] T. Michaeli and M. Irani, *Nonparametric Blind Super-resolution*, in (ICCV) International Conference on Computer Vision, IEEE, 2013, pp. 945–952. https://doi.org/10.1109/ICCV.2013.121.

[43] L. Moisan, *Periodic Plus Smooth Image Decomposition*, Journal of Mathematical Imaging and Vision, 39 (2011), pp. 161–179. https://doi.org/10.1007/s10851-010-0227-1.

[44] E. H. Moore, *On the reciprocal of the general algebraic matrix*, Bulletin of the American Mathematical Society, 26 (1920), pp. 394–395.

[45] C. Murphy, S. Mudur, D. Holden, M.-A. Carbonneau, D. Ghafourzadeh, and A. Beauchamp, *Artist guided generation of video game production quality face textures*, Computers & Graphics, 98 (2021), pp. 268–279. https://doi.org/10.1016/j.cag.2021.06.004.

[46] K. Murthy, M. Shearn, B. D. Smiley, A. H. Chau, J. Levine, and D. Robinson, *SkySat-1: very high-resolution imagery from a small satellite*, in Proceedings of Sensors, Systems, and Next-Generation Satellites XVIII, R. Meynart, S. P. Neeck, and H. Shimoda, eds., vol. 9241, 2014. https://doi.org/10.1117/12.2074163.

[47] D. Mylonopoulos, P. Cascarano, L. Calatroni, and E. L. Piccolomini, *Constrained and Unconstrained Inverse Potts Modelling for Joint Image Super-Resolution and Segmentation*, Image Processing On Line, 12 (2022), pp. 92–110. https://doi.org/10.5201/ipol.2022.393.

[48] N.-L. Nguyen, *A Brief Analysis of the SwinIR Image Super-Resolution*, Image Processing On Line, 12 (2022), pp. 582–589. https://doi.org/10.5201/ipol.2022.430.

[49] M. Nikolova and M. K. Ng, *Analysis of Half-Quadratic Minimization Methods for Signal and Image Recovery*, SIAM Journal on Scientific Computing, 27 (2005), pp. 937–966. https://doi.org/10.1137/030600862.

[50] R. Penrose, *A generalized inverse for matrices*, in Mathematical proceedings of the Cambridge philosophical society, vol. 51, Cambridge University Press, 1955, pp. 406–413. https://doi.org/10.1017/S0305004100030401.

[51] S. Peter, *Algorithms for Robust and Fast Sparse Recovery*, dissertation, Technische Universität München, München, 2016. https://mediatum.ub.tum.de/1295426.

[52] D. Potts and G. Steidl, *Fast Summation at Nonequispaced Knots by NFFT*, SIAM Journal on Scientific Computing, 24 (2003), pp. 2013–2037. https://doi.org/10.1137/S1064827502400984.

[53] S. P. Poupart, A. Materne, C. Thiebaut, J. M. Delvit, C. Latry, and B. Mithieux, *On ground demonstrator of digital stabilization for high-resolution Earth observation time of delay integration imaging*, in International Conference on Space Optics — ICSO 2018, Z. Sodnik, N. Karafolas, and B. Cugny, eds., vol. 11180, International Society for Optics and Photonics, SPIE, 2019, pp. 1777 – 1788. https://doi.org/10.1117/12.2536089.

[54] M. Protter, M. Elad, H. Takeda, and P. Milanfar, *Generalizing the nonlocal-means to super-resolution reconstruction*, in IEEE Transactions on Image Processing, vol. 18, jan 2009, pp. 36–51. https://doi.org/10.1109/TIP.2008.2008067.

[55] M. Rais, J.-M. Morel, C. Thiebaut, J.-M. Delvit, and G. Facciolo, *Improving wavefront sensing with a Shack–Hartmann device*, Applied Optics, 55 (2016), pp. 7836–7846. https://doi.org/10.1364/AO.55.007836.

[56] J. Ramirez, *Somatic Calcium imaging reveals simple spike activity of cerebellar Purkinje Cells: applications and limitations to in vivo research*, PhD thesis, 2018. (Published doctoral dissertation). Director: Brandon Stell. Bio-Sorbonne Paris Cité. Université Paris Descartes. Paris, France. https://theses.hal.science/tel-02493058.

[57] D. Ren, Y. Zhao, Y. Chen, Y. Ma, and X. Zhang, *Finite-Exemplar-Based Super-Resolution for Specific Gaming Videos*, in 2022 the 5th International Conference on Image and Graphics Processing (ICIGP), ICIGP 2022, New York, NY, USA, 2022, Association for Computing Machinery, p. 234–241. https://doi.org/10.1145/3512388.3512423.

[58] D. Robinson, S. Farsiu, and P. Milanfar, *Optimal registration of aliased images using variable projection with applications to super-resolution*, The Computer Journal, 52 (2007), pp. 31–42. https://doi.org/10.1093/comjnl/bxm007.

[59] D. Robinson and P. Milanfar, *Statistical performance analysis of super-resolution*, IEEE Transactions on Image Processing, 15 (2006), pp. 1413–1428. https://doi.org/10.1109/TIP.2006.871079.

[60] Y. Romano, M. Protter, and M. Elad, *Single Image Interpolation via Adaptive Non-Local Sparsity-Based Modeling*, IEEE Transactions on Image Processing, 23 (2014), pp. 3085–3098. https://doi.org/10.1109/TIP.2014.2325774.

[61] A. Shocher, N. Cohen, and M. Irani, *Zero-Shot Super-Resolution Using Deep Internal Learning*, in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, jun 2018, pp. 3118–3126. https://doi.org/10.1109/CVPR.2018.00329.

[62] C. Thiebaut, L. Perraud, J. M. Delvit, and C. Latry, *Fast gradient-based algorithm on extended landscapes for wave-front reconstruction of Earth observation satellite*, in Adaptive Optics Systems V, E. Marchetti, L. M. Close, and J.-P. Véran, eds., vol. 9909, International Society for Optics and Photonics, SPIE, 2016, pp. 2022 – 2029. https://doi.org/10.1117/12.2233355.

[63] C. Thiebaut, S. Petit-Poupart, J. M. Delvit, C. Latry, E. Bousquet, and G. Laurent, *Real-time implementation of digital stabilization for high-resolution Earth observation*

imaging, in High-Performance Computing in Geoscience and Remote Sensing VII, B. Huang, S. López, and Z. Wu, eds., vol. 10430, International Society for Optics and Photonics, SPIE, 2017, pp. 15 – 22. https://doi.org/10.1117/12.2278377.

[64] B. C. Tom and A. K. Katsaggelos, *Reconstruction of a high-resolution image by simultaneous registration, restoration, and interpolation of low-resolution images*, IEEE International Conference on Image Processing, 2 (1996), pp. 539–542. https://doi.org/10.1109/icip.1995.537535.

[65] Y. Traonmilin, *Relations between image model and number of measures for a high fidelity super-resolution*, theses, Télécom ParisTech, July 2014. https://tel.archives-ouvertes.fr/tel-01135196.

[66] Y. Traonmilin, S. Ladjal, and A. Almansa, *On the amount of regularization for super-resolution interpolation*, in 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO), 2012, pp. 380–384. https://hal.science/hal-00824670/.

[67] M. Unser, *Sampling–50 years after Shannon*, Proceedings of the IEEE, 88 (2000). https://doi.org/10.1109/5.843002.

[68] P. Vandewalle, L. Sbaiz, J. Vandewalle, and M. Vetterli, *Super-Resolution From Unregistered and Totally Aliased Signals Using Subspace Methods*, IEEE Transactions on Signal Processing, 55 (2007), pp. 3687–3703. https://doi.org/10.1109/TSP.2007.894257.

[69] P. Vandewalle, S. Süsstrunk, and M. Vetterli, *A frequency domain approach to registration of aliased images with application to super-resolution*, EURASIP Journal on Advances in Signal Processing, 2006 (2006). https://doi.org/10.1155/ASP/2006/71459.

[70] B. Wronski, I. Garcia-Dorado, M. Ernst, D. Kelly, M. Krainin, C.-K. Liang, M. Levoy, and P. Milanfar, *Handheld multi-frame super-resolution*, ACM Transactions on Graphics, 38 (2019). https://doi.org/10.1145/3306346.3323024.

[71] K.-H. Yap, Y. He, Y. Tian, and L.-P. Chau, *A Nonlinear L_1-Norm Approach for Joint Image Registration and Super-Resolution*, IEEE Signal Processing Letters, 16 (2009), pp. 981–984. https://doi.org/10.1109/LSP.2009.2028106.

[72] G. Yu, G. Sapiro, and S. Mallat, *Solving Inverse Problems With Piecewise Linear Estimators: From Gaussian Mixture Models to Structured Sparsity*, IEEE Transactions on Image Processing, 21 (2012), pp. 2481–2499. https://doi.org/10.1109/TIP.2011.2176743.

[73] X. Yu, B. Fernando, R. Hartley, and F. Porikli, *Super-resolving very low-resolution face images with supplementary attributes*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 908–917. https://doi.org/10.1109/CVPR.2018.00101.

[74] ——, *Semantic face hallucination: Super-resolving very low-resolution face images with supplementary attributes*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 42 (2020), pp. 2926–2943. https://doi.org/10.1109/TPAMI.2019.2916881.

[75] K. Zhang, J. Liang, L. Van Gool, and R. Timofte, *Designing a Practical Degradation Model for Deep Blind Image Super-Resolution*, in IEEE International Conference on Computer Vision, 2021, pp. 4791–4800. https://doi.org/10.1109/ICCV48922.2021.00475.

[76] K. Zhang, L. Van Gool, and R. Timofte, *Deep Unfolding Network for Image Super-Resolution*, in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 3214–3223. https://doi.org/10.1109/CVPR42600.2020.00328.

[77] K. Zhang, W. Zuo, and L. Zhang, *Learning a Single Convolutional Super-Resolution Network for Multiple Degradations*, in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 3262–3271. https://doi.org/10.1109/CVPR.2018.00344.

[78] N. Zhao, Q. Wei, A. Basarab, N. Dobigeon, D. Kouame, and J.-Y. Tourneret, *Fast Single Image Super-Resolution Using a New Analytical Solution for $\ell 2$–$\ell 2$ Problems*, IEEE Transactions on Image Processing, 25 (2016), pp. 3683–3697. https://doi.org/10.1109/TIP.2016.2567075.

[79] C. Zhuang, M. Li, K. Zhang, Z. Li, and J. Lu, *Multi-level landmark-guided deep network for face super-resolution*, Neural Networks, 152 (2022), pp. 276–286. https://doi.org/10.1016/j.neunet.2022.04.026.

[80] D. Zoran and Y. Weiss, *From learning models of natural image patches to whole image restoration*, in (ICCV) International Conference on Computer Vision, IEEE, nov 2011, pp. 479–486. https://doi.org/10.1109/ICCV.2011.6126278.