IRWIN AND JOAN JACOBS
CENTER FOR COMMUNICATION AND INFORMATION TECHNOLOGIES

Superresolution of self-similar textures

Ido Zachevsky and Yehoshua Y. Zeevi

CCIT Report #838
August 2013

DEPARTMENT OF ELECTRICAL ENGINEERING
TECHNION - ISRAEL INSTITUTE OF TECHNOLOGY, HAIFA 32000, ISRAEL

CCIT REPORT #838 August 2013

Superresolution of self-similar textures

Ido Zachevsky and Yehoshua Y. Zeevi

August 22, 2013

Abstract

Texture enhancement presents an ongoing challenge, in spite of the considerable progress made in recent years. Whereas most of the effort has been devoted so far to enhancement of regular textures, stochastic textures, which exhibit fine details, are encountered in most natural images wherein they still present an outstanding problem insofar as superresolution enhancement is concerned. In this work, a texture model, based on fractional Brownian motion (fBm), is proposed. The model is based on our observation that, contrary to previous findings that images are not characterized by Gaussian distributions, natural stochastic textures (NST) are Gaussian. The model is global and does not entail using image patches. The fBm is a self-similar stochastic process. The self-similarity is known to characterize a large class of natural textures. The fBm-based model is evaluated and a single-image regularized superresolution algorithm is derived. The algorithm is useful for a wide range of textures. Its performance is compared with state-of-the-art single-image superresolution methods and its advantages are highlighted.

1 Introduction

Single-image superresolution (SR) has attracted considerable attention in recent years and still considered to be one of the most outstanding problems in advanced image processing [1–6]. This is a challenging task, since the original (source) image has to be recovered using only the degraded, subsampled, image. While traditional approaches to image enhancement in terms of denoising, deblurring and contour emphasis result in sharper images, they often yield an unnatural cartoon-like image, compromising on the quality of, and almost even eliminating, some textures. This compromise in image fidelity highlights the observation that textures are an important ingredient of image structure, that must be considered in the context of image enhancement tasks.

Common methods for image enhancement may not work on stochastic textures, and in many cases, other approaches are required for texture enhancements. Image enhancement algorithms, used in deblurring and denoising methods, generally

attempt to solve the following inverse problem:

$$\hat{X} = \arg\min_{X \in \mathcal{X}} \|X * h - Y\|_2^2 + \lambda g(X), \tag{1.1}$$

for a degraded image, $Y \in \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^{N \times N}$, a solution $X \in \mathcal{X}$ and a blur kernel $h \in \mathbb{R}^{n \times n}$. The parameter λ is a Lagrange coefficient and $g(X)$ is a regularization function. Since h is usually a low-pass type filter, and the measurement is noisy, the problem is ill-posed, and a regularization function is needed.

In Wiener filtering, or other linear methods, the regularizing function, $g(X)$, is a quadratic function of the image gradient. In $\mathcal{L}_1$-based methods, such as total variation (TV), $g(X)$ is the $\mathcal{L}_1$ norm of the image gradient.

PDE-based approaches generally use $g(X) = G(|\nabla X|^2)$. A gradient descent minimization then yields the following scheme:

$$X_t = \tilde{h} * (X * h - Y) - \lambda \nabla (G'(|\nabla X|^2) \nabla X)), \tag{1.2}$$

where $\tilde{h}(\eta_1, \eta_2) = h(-x, -y)$ and $G'(|\nabla X|^2)$ is the diffusivity or edge detection function. This function is chosen so that high gradients are preserved and low gradients are smoothed.

When these methods are applied to images that are comprised of stochastic textures, they do not yield the desired results. This is due to the common assumption that low gradient areas in an image are originated by noise or optimization artifacts (such as ringing or aliasing) and not by a valuable texture. This is due to the assumption that images reside in a bounded variation (BV) space, which often lends itself to the wrong choice of the regularization function. This assumption has been challenged in recent years [7] and deblurring schemes, developed under more suitable spaces, have yielded more successful results [8].

1.1 Texture representation and enhancement

Some of the aforementioned methods can be adopted for texture preservation. In [9], a potential function has been incorporated into the diffusion equation which effectively prevents smoothing of specific texture details in an image. In sparseness-based approaches (not discussed in this work; for a review see [10]), a separate dictionary is used for handling textures.

Textures, in general, can be divided into two main types: Regular, or structured, and stochastic [11, 12]. One can define the former as spatially-replicated instances of a single or several repetitive patterns. An example of a regular texture is a brick wall. To compare with, stochastic textures do not contain a specific pattern. Instead, they are considered to be realizations of random processes. This type of textures cannot be modelled in a similar manner to regular textures. It is important to note that this division is by no means a dichotomy, as natural textures depict the entire range between regular and stochastic. The texture spectrum, as defined in [11], contains textures in varying complexity and regularity.

As the two types of textures are visually and conceptually different, different techniques are used in order to enhance them. Most of the effort in texture enhancement has been devoted, even inadvertently, to the regular textures. Inasmuch as regular textures contain replicated versions of a single or a few basic patterns, in varying amount of distortion, one can use enhancement methods based on a search for similar shapes, in higher resolution, within the same image or in other images bearing some similarity to the target image [1].

Numerous methods, not necessarily applied to textures but more so to on images containing cartoon-type skeletons as well, are based on a machine learning approach. These methods attempt to build up and exploit a database of natural images and/or textures. This database is then used for training with sets of low and high resolution image patches, and the final image is then obtained by predicting the high resolution patch according to the training database [13]. It is important to note that regular textures still obey, in general, the model of bounded variation, and can therefore be enhanced using known approaches.

1.2 Stochastic textures

Unlike regular textures, stochastic textures are not characterized by repetitive patterns. They are, instead, defined by their statistical properties. This type of textures exhibits statistical properties such as non-local [14], long-range dependencies and self-similarity, as their pixel distribution remains the same across scales, up to a scaling parameter [15–18]. Enhancement of such textures can hardly be achieved by using example-based methods that have been demonstrated to be successful on regular textures, as the stochastic textures do not contain basic patterns but are rather governed by a latent random process.

Sparseness-based or GMM-based approaches perform enhancements locally (on patches) [19], and the resulting dictionary usually appears similar to Fourier, DCT or overcomplete Wavelet bases, with discontinuities to match edges in images as well. However, these models assume that an image can be modelled locally, whereas in stochastic textures, there is an advantage to a long-range or global model, exploiting correlations between as many pixels as possible.

1.3 The Gaussian assumption

Extensive work has been done deriving a model for natural images [20, 21]. These studies have substantiated the notion that the distribution of natural images is highly kurtotic and non-Gaussian. Wainwright et al. have proposed using Gaussian Scale Mixture (GSM) as a model for wavelet coefficients of natural images. Indeed, when one inspects the histogram of coefficients in various scales and orientations, this highly kurtotic model shows promising results. The same phenomenon is expressed also in the 2D histograms of adjacent scales and orientations.

However, for stochastic textures, the Gaussian assumption is in fact relevant. This behaviour that might be overlooked when considering an ensemble of a natural images, where the least of which are distinctly stochastic textures. An example of this phenomenon is depicted in Fig. 1. While a common natural image (first column) is characterized by non-Gaussian behaviour, in the case of a stochastic texture (third column), the statistics match a Gaussian distribution. This is the case for a large class of textures.

1.4 Processing of stochastic textures

While regular textures can be enhanced by using methods of edge enhancement (considering, for example, the enhancement of the edges of a brick in a brick wall texture), in the stochastic case such edges do not exist. Attempting to apply edge enhancement to such a texture, may, in some cases, create a staircasing effect, while smoothing out the fine details in the neighborhood of the newly-created edge.

A different approach for regular and stochastic texture enhancement is the texture synthesis, in which a sample-patch is used in order to create a newly formed image of larger size and the same visual appearance as the original [11,12,22]. While such methods show successful results in visual resemblance to the original, they are less effective in deconvolution problems such as superresolution, in which a high resolution estimate has to represent the input low resolution image. Additionally, such synthesis, based on local dependencies, may fail to capture the global statistical structure of the texture, in case of stochastic textures.

In this study, we present a model for stochastic textures. This model is based on fractional Brownian motion (fBm); a Gaussian random process which exhibits properties that characterize stochastic textures [16]. This process is used to regenerate a high frequency estimation from a given degraded image. Realizations of the model are displayed, and an optimization scheme is derived, to perform single-image superresolution.

The basic model is suitable for isotropic textures, as will be discussed later. In order to be suitable for a broader class of textures, PDE-based regularization is introduced. This is based on tensor diffusion - a known PDE-based image enhancement method [23]. The tensor diffusion has yielded successful results in performing deblurring tasks, due to the anisotropic structure, which is obtained by introducing tensor diffusivity, $D(\nabla X)$, in Eq. (1.2):

$$X_t = \tilde{h} * (X * h - Y) - \lambda \nabla (D(\nabla X) \nabla X)). \tag{1.3}$$

We present a modified diffusion scheme which recovers missing texture details and preserves them under the diffusion.

We have previously reported [24] of an algorithm which performs fBm-based deblurring. In this work, we use the principles of the fBm to yield an image model and a true superresolution algorithm.

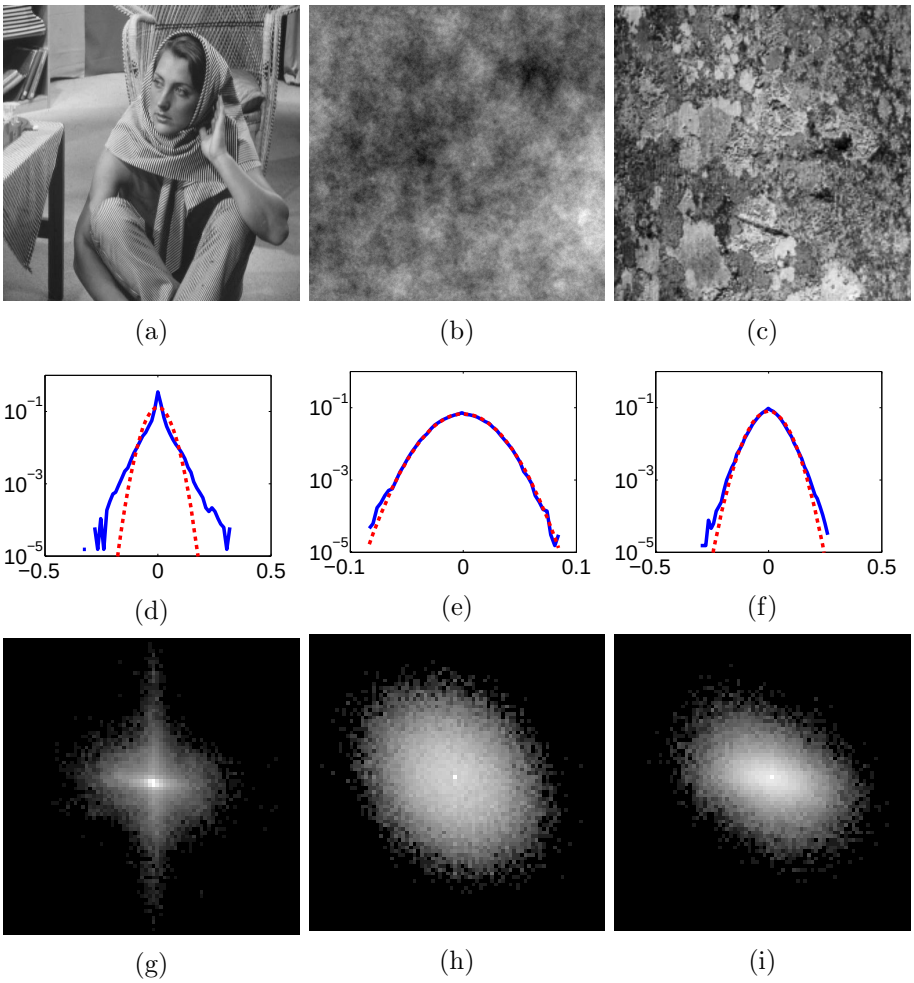


Figure 1: Empirical first and second order distribution for wavelet coefficients of natural images. Three images are presented: (a) A common natural image (Barbara), (b) a 2D fractional Brownian motion realization (discussed on section 3) and (c) a stochastic texture. First order empirical distributions are displayed on the second row: (d) Distribution of Barbara (blue), which deviates significantly from the Gaussian ML fit (red); (e) Distribution of the 2D fBm; (f) Distribution of a stochastic texture, which almost perfectly fits the Gaussian distribution. The second order log-distributions are displayed in the third row. While non-Gaussianity is observed in the natural image, for the stochastic texture, Gaussianity is still valid. This holds even in the case of the displayed texture, which is not strictly stochastic but tends towards the near-regular.

2 Problem statement

The following form of the superresolution problem is considered: A high-resolution (HR) image is degraded by a blurring filter, representing, for example, the PSF of an optical sensor. It is subsequently subsampled. Noise is then additively mixed with the blurred and subsampled image to create the available low-resolution (LR) image. Let $X(\eta_1, \eta_2)$ and $Y(\eta_1, \eta_2)$ denote the original (HR) image and observed (LR) noisy image, respectively. The imaging model can be represented as follows:

$$Y(\eta_1, \eta_2) = \mathcal{D}\left((Y * b)(\eta_1, \eta_2)\right) + N(\eta_1, \eta_2), \tag{2.1}$$

where $\mathcal{D}$ is the subsampling operator, $b(\eta_1, \eta_2)$ is a noninvertible blur kernel and $N(\eta_1, \eta_2)$ is an independent additive white Gaussian noise. In the case of SR, we assume that the noise has low variance, unlike denoising problems where the noise is substantial. The blur kernel is assumed to have limited spatial support, and therefore the decimation operator introduces aliasing.

The superresolution (SR) problem is severely ill-posed due to the decimation operator. For this reason, the effect of decimation is often ignored in SR studies. The single-frame SR problem is a special case of the classical multi-frame SR problem [25–27]. In the latter, several degraded images are available, each containing unique details of the single original image, acquired by sub-pixel shifts. A high-resolution reconstruction is obtained in this case by recovering complementary information from all the measurements.

In contrast, in single-frame SR, known in the literature also as upscaling, only a single measurement is available. The single-frame SR problem can be formally stated as follows:

$$\hat{X}(\eta_1, \eta_2) = \arg\min_{X \in \mathcal{X}} \|Y(\eta_1, \eta_2) - \mathcal{D}\left((X * b)(\eta_1, \eta_2)\right)\|_2. \tag{2.2}$$

The SR image, $\hat{X}(\eta_1, \eta_2)$, thus obtained is the best one in that it yields the smallest $\mathcal{L}_2$ error relative to the original image (ground truth). The SR problem appears, at first look, to be equivalent to a deconvolution problem, in which a SR image, $\hat{X}(\eta_1, \eta_2)$, is to be recovered from the blurred and noisy image, $Y(\eta_1, \eta_2)$. It is important to emphasize that this is by no means the case. In the case of SR, the decimation operator introduces aliasing artifacts due to the loss of details in the subsampling.

Deblurring algorithms cannot be applied directly in the case of SR of textures (Fig. 2). There are studies that first solve the deblurring problem, and then apply interpolation to "inverse" the effect of the decimation operator. This approach is applicable only when the blur filter acts as an anti-aliasing filter. Blur filters considered in the present study, and in SR problems in general, have, however, a small spatial support, and cannot be considered as anti-aliasing filters. Therefore, our emphasis is on the *model* for images, based on which, the missing data can be reconstructed, to yield a high fidelity estimate of the original image.

(a) (b) (c) (d) (e)

Figure 2: Applying deblurring algorithms on blurred and decimated images. While BM3D-deblurring [28] performs deblurring successfully on blurred images, when decimation is introduced the results deteriorate. This is the case in natural images (first row), and it is more pronounced in the case of textures (bottom row). (a) The original image. (b) and (c) Blurred and BM3D-deblurrred image respectively. (d) and (e) Blurred and subsampled image and BM3D-deblurrred image respectively.

3 Fractional Brownian motion

The fractional Brownian motion (fBm) is a Gaussian random process, which was introduced by Mandelbrot and Van Ness as a model suitable for natural images [16]. This process generalizes the well-known Brownian motion in that the increments are stationary but not independent. It is defined, in one dimension, as a Gaussian process with zero mean and the following autocorrelation function:

$$E\left[B_H(t)B_H(s)\right] = \frac{\sigma^2}{2}\left(|t|^{2H} + |s|^{2H} - |t-s|^{2H}\right), \tag{3.1}$$

where

$$\sigma^2 = \frac{\sigma_W^2}{2}\frac{\cos(\pi H)}{\pi H}\Gamma(1-2H), \tag{3.2}$$

σ_W^2 is a known variance, and the Hurst parameter, $H \in (0,1)$, controls the regularity of the process. This is a non-stationary process with stationary increments - a property to be exploited later in efficient synthesis. The first sample is usually set to zero, $B(0,0) = 0$, further indicating that it cannot be stationary (unless it is zero everywhere). For Hurst parameter values of $H \in \left(0, \frac{1}{2}\right)$, this process exhibits negative correlation between samples (anti-persistence), and for $H \in \left(\frac{1}{2}, 1\right)$, it exhibits positive correlation and long range dependencies. A special case is $H = \frac{1}{2}$, for which this process becomes the well-known Brownian motion, or Wiener

process. This process exhibits two important properties, known to characterize images, in the context of Mandelbrot's work on fractals [16]. The first one is long-range dependencies between samples, where for $H > 0.5$ the sum of the correlation of the increments diverges. The second property of the fBm, which is exploited in this study, is the statistical self-similarity, defined as:

$$B_H(at) \overset{d}{=} |a|^H B_H(t), \tag{3.3}$$

for a positive number a, where the superscript d stands for equality in distribution. This equality indicates that the sample distribution across different scales is varied only by a constant depending on the scale, a, and the Hurst parameter. These two properties highlight the relevance of this process to natural textures, as the latter often exhibit phenomena with such similarities between adjacent as well as distant pixels [18].

3.1 Synthesis in two dimensions

Since the fBm process is a Gaussian process with known covariance function, one can explicitly synthesize a realization in the discrete domain [29]. Recall the following covariance property for multivariate random variables:

$$\text{cov}(LZ) = L\Sigma_Z L^T, \tag{3.4}$$

where Z is a random vector with autocorrelation matrix Σ_Z, and L is a matrix. In order to synthesize the fBm in this manner, one needs to first build the autocorrelation matrix according to Eq. (3.1). Then, the Cholesky decomposition is used to obtain a matrix, L, such that $M = LL^T$, where M is the fBm autocorrelation matrix. This decomposition is possible since it is positive-definite. Then, upon multiplying the lower-triangular matrix L by a multivariate random vector Z with a unity covariance matrix, the covariance of the resulting vector $B = LZ$ fits fBm.

This simple algorithm can be extended to two dimensions, by representing (stacking) a 2D image as a vector and constructing the autocorrelation matrix with the respective 2D dependencies. At this stage it is also worth to note that this process is isotropic in the statistical sense. Let $p = (x_1, y_1)^T$ and $q = (x_2, y_2)^T$ be two points. Then, the autocorrelation of the fBm in 2D is defined as follows:

$$E\left[B_H(p)B_H(q)\right] = \frac{\sigma^2}{2}\left(\|p\|^{2H} + \|q\|^{2H} - \|p-q\|^{2H}\right). \tag{3.5}$$

Let M be a rotation matrix, indicating $|M| = 1$ and $M^{-1} = M^T$. It is straightforward to check that for an arbitrary point $l = (x_0, y_0)^T$, $\|Ml\| = \|l\|$ and therefore $E[B_H(Mp)B_H(Mq)] = E[B_H(p)B_H(q)]$. This indicates the autocorrelation is invariant under rotations and the process is isotropic.

While the aforementioned method can produce the exact 2D fBm process, it is very inefficient. This is due to the dimensional requirement of the covariance matrix,

which dictates a size of $N^2 \times N^2$ for an image size of $N \times N$ and time complexity of $O(N^6)$ due to the Cholesky decomposition. The space and time complexity render this method to become impractical even in application to moderately-sized images.

It is possible, however, to use more efficient methods of synthesizing nearly-exact realizations of the 2D fBm. We adopt the method proposed by Kaplan and Kuo [30], which implements Fourier synthesis, and yields accurate realizations with time complexity of $O(N^2 log_2(N))$ and space complexity of $O(N^2)$.

This method utilizes the stationary increments of the fBm and builds the realization by first calculating the autocorrelations of the increments in the $\underline{x}$, $\underline{y}$, and $(\underline{x}, \underline{y})$ directions, synthesizing the increments in the frequency domain, and then summing them to produce the final result. Two realizations of the 2D fBm process, for two typical values of H, are displayed in Fig. 3.

3.2 Synthesis of non-stationary fields with stationary increments

The fBm can be considered as a special case of a family of non-stationary processes with stationary increments. As such, it is a simple case, as it is statistically isotropic and is being governed by a single parameter, H. In [31], the 2D fBm synthesis algorithm of Kaplan and Kuo is generalized for any such field. The synthesized fields are derived with reference to an initial white noise image, $W(\eta_1, \eta_2)$, and a structure function, $\phi(\eta_1, \eta_2)$, which defines the autocorrelation of the fBm, $F(\eta_1, \eta_2)$, by the following equation:

$$\begin{aligned} E[F(\eta_1, \eta_2)F(\eta_1', \eta_2')] = -E[F(0,0)^2] + \\ + \frac{1}{2}(\phi(\eta_1, \eta_2) + \phi(\eta_1', \eta_2') - \phi(\eta_1 - \eta_1', \eta_2 - \eta_2')), \end{aligned} \tag{3.6}$$

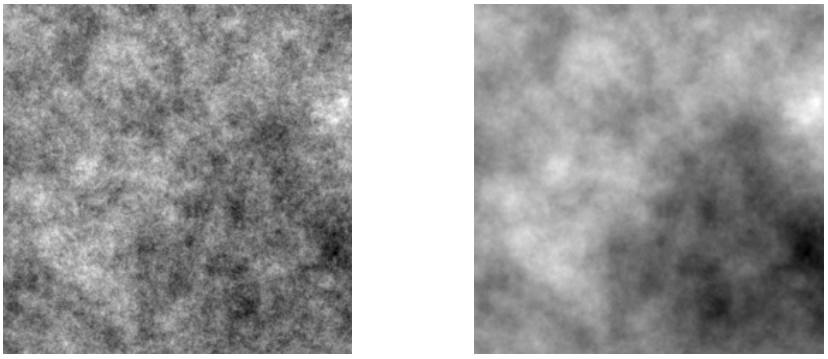
where $F(0,0)$ is set to zero for the synthesis process. This structure function defines the autocorrelation of the increments of the desired field. For the fBm case, we obtain:

$$\phi(x, y) = C(x^2 + y^2)^H, \quad H \in (0, 1), \tag{3.7}$$

for a suitable normalizing constant, C, and a proper Hurst parameter, H. In this manner, anisotropic fBm fields can be synthesized by choosing a different structure function - one that depends on more than a single Hurst parameter. For the exact synthesis algorithm, as well as other limitations of the structure function, see [31]. We take it one step further and propose an adaptive structure function to be used in image enhancement of textured images.

3.3 Remarks

We assume that the details missing in degraded textures had, originally, dependencies similar to those characteristic of an fBm. Therefore, using a single realization



(a) (b)

Figure 3: 2D fBm realization for two values of H: (a) A value of $H = 0.1$, depicting negative correlation between adjacent pixels. (b) A value of $H = 0.6$, depicting high correlation between adjacent pixels. A low value of H is better fitted to high-frequency content relevant to stochastic textures.

of a proper fBm as an initial image, we should be able to restore missing high-resolution details by fitting such a realization onto a degraded image. Since we are interested in high frequencies, the common values for the Hurst parameter should be low in general, and usually $H \leq 0.2$.

4 Phase of the frequency response

The importance of phase, and of "local phase", in signal and image processing is well-established [32, 33]. It has been shown that for natural images, the important information of the image is, in fact, stored in the phase rather than in the magnitude of the frequency domain representation of the image [33,34]. The magnitude contains information about the frequencies present in the image, which are common to a large class of natural images. The phase contains information about the spatial relationship of these frequencies in a specific image. Let $X(\eta_1, \eta_2)$ and $Y(\eta_1, \eta_2)$ be two images with respective Fourier transforms $\tilde{X}(\tilde{\eta_1}, \tilde{\eta_2})$ and $\tilde{Y}(\tilde{\eta_1}, \tilde{\eta_2})$. The Fourier transform, $\tilde{M}(\tilde{\eta_1}, \tilde{\eta_2})$, of a signal, $M(\eta_1, \eta_2)$, can be decomposed into magnitude and phase components:

$$\tilde{M}(\tilde{\eta_1}, \tilde{\eta_2}) = |\tilde{M}(\tilde{\eta_1}, \tilde{\eta_2})| \exp\Big(j \cdot \angle \tilde{M}(\tilde{\eta_1}, \tilde{\eta_2})\Big). \tag{4.1}$$

In this manner, one can define a frequency representation of an image, $\hat{\tilde{Y}}(\tilde{\eta_1}, \tilde{\eta_2})$, as follows:

$$\hat{\tilde{Y}}(\tilde{\eta_1}, \tilde{\eta_2}) = |\tilde{Y}(\tilde{\eta_1}, \tilde{\eta_2})| \exp\Big(j \cdot \angle \tilde{X}(\tilde{\eta_1}, \tilde{\eta_2})\Big), \tag{4.2}$$

thereby obtaining the magnitude of the original image, $Y(\eta_1, \eta_2)$, and the phase from the other, $X(\eta_1, \eta_2)$. The resulting image, $\hat{Y}(\eta_1, \eta_2)$, obtained by the inverse Fourier

transform of $\hat{\tilde{Y}}(\tilde{\eta_1}, \tilde{\eta_2})$, will contain the frequency components present in $Y(\eta_1, \eta_2)$, but the phase, and therefore the spatial shifts, will be according to $X(\eta_1, \eta_2)$.

This simple property enables us to exploit a synthetic 2D fBm image for our purposes. The synthetic fBm is inherently random, as it is generated by an iid Multivariate normal matrix. However, by using the general notion expressed by Eq. (4.2), one can benefit from the frequency spectrum characteristic of a 2D fBm image, while fitting the phase relationships of a specific image. We thereby effectively exploit the fBm correlations while imposing minor dependence on the initial white noise.

5 A model for self-similar textures

We now present a model for self-similar textures. This model is based on the high frequencies of the 2D fBm process and fits a variety of images corresponding to the fBm statistics. The model can be used to generate such textures as well as be used for image enhancement tasks, such as superresolution or denoising (although denoising is not covered by this study).

The model is defined as follows. Let $X_{LP}(\eta_1, \eta_2)$ be a low frequency image. This image can be generated using fBm models, ARMA models, or be based upon a natural image. Let $X_{HP}(\eta_1, \eta_2)$ be a high-frequency image, obtained as follows:

$$X_{HP}(\eta_1, \eta_2) = \mathcal{P}_{H,W(\eta_1,\eta_2)}\left(X_{LP}(\,\eta_1, \eta_2)\right), \tag{5.1}$$

where $\mathcal{P}_{H,W(\eta_1,\eta_2)}$ is an operator performing phase matching, as described earlier, for a 2D-fBm image generated according to the Hurst parameter, H, from the white noise image, $W(\eta_1, \eta_2)$. The value of H will typically be low, $H \leq 0.2$, to generate high frequencies. The texture image, $X(\eta_1, \eta_2)$, is then constructed from the superposition of both images as follows:

$$X(\eta_1, \eta_2) = X_{LP}(\eta_1, \eta_2) + (X_{HP} * h_{HP})(\eta_1, \eta_2) + V(\eta_1, \eta_2), \tag{5.2}$$

where $h_{HP}(\eta_1, \eta_2)$ is a high-pass filter and $V(\eta_1, \eta_2)$ is residual noise compensating for model inaccuracies.

We note that an fBm image can be derived as a special case of this model, since $X_{LP}(\eta_1, \eta_2)$ can be the low-pass version of the fBm generated by $\mathcal{P}_{H,W(\eta_1,\eta_2)}$. Obviously, natural image textures cannot be represented by this model without error. It is straightforward to check that given an image, $I(\eta_1, \eta_2)$, the energy content of the error, $V(\eta_1, \eta_2)$, will reside in the high-frequency range. Therefore, in order to assess the model in natural images, we have to consider the error in the high-frequency range.

While a naïve measure norm, such as $\mathcal{L}_2$ norm, may be suggested for the evaluation of this error, we do not use it. This is due to the well-known shortcomings associated with $\mathcal{L}_2$-based comparisons [35], which are especially emphasized in the

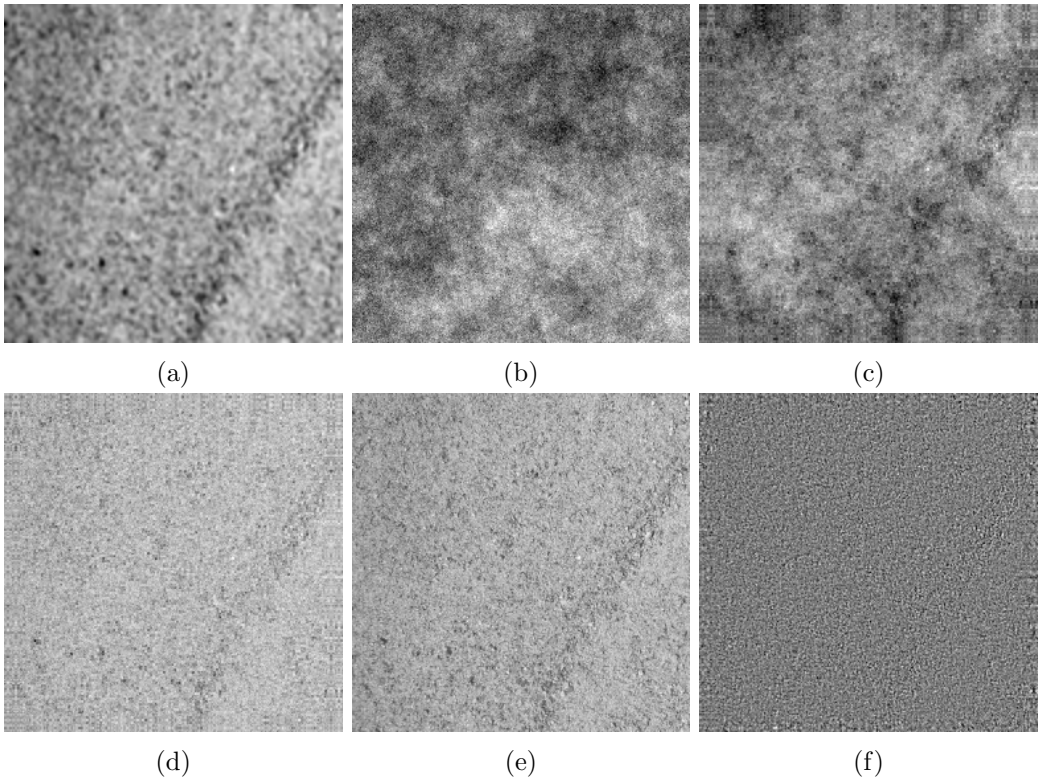


Figure 4: Representation of a texture according to the model outlined in section 5, with a natural texture image as $X_{LP}(\eta_1, \eta_2)$: (a) Low-pass image, $X_{LP}(\eta_1, \eta_2)$, generated by blur and subsampling of a real image. (b) A 2D fBm realization with $H = 0.1$. (c) $X_{HP}(\eta_1, \eta_2)$, a 2D fBm image obtained after phase matching for $X_{LP}(\eta_1, \eta_2)$. (d) The artificial texture image, $X(\eta_1, \eta_2)$. (e) The original image used for generating $X_{LP}(\eta_1, \eta_2)$, for comparison. (f) The difference between the original image and $X(\eta_1, \eta_2)$. Note the similarity in details between the original and the model representation, expressed also in the difference image.

case of stochastic textures. Instead, and with a lack of a better method, the results will be inspected both visually and by means of image statistics such as the histogram.

An example of the model is depicted in Fig. 4. $X_{LP}(\eta_1, \eta_2)$ (Fig. 4a) is generated by blur and subsampling of a real image texture. While the resulting image (Fig. 4d) is not identical to the original image (Fig. 4e), the high frequencies are visually similar. This encourages us to use this model in our texture superresolution algorithm.

Fig. 5 depicts two typical images, one suitable for the proposed model and the other one is not. In the first row we have a stochastic texture. Two high-passed versions are shown; Fig. 5b is the high-pass of the original texture ($I_{HP}(\eta_1, \eta_2)$), and

Fig. 5d is the high-pass of the texture according to the proposed model, $X_I(\eta_1, \eta_2)$. These two high-pass images are then compared by their histograms (Fig. 5e). We observe that the high-pass versions are both visually similar, and have similar histograms. Additionally, we observe that the histogram is of a Gaussian shape. As will be shown in subsection 5.2, a necessary condition for an fBm image is to have a Gaussian-shaped 1D histogram, in the limit, as $H \to 0$ and $N \to \infty$. Despite this being only a necessary condition, we are encouraged to seek further indications of the relevance of this model to textured images. We further substantiate in the next subsection in the form of 2D histogram.

In contrast with the texture of 5a, Fig. 5f depicts a regular (structured) texture, and its high-pass versions is shown in Fig. 5g. In this case, the two histograms (Fig. 5j) are distinctly different. We conclude that the model can not faithfully represent regular textures. This is due to the fact that the fBm cannot represent such images.

5.1 2D histograms

The 1D histogram provides an empirical estimate for image first order distribution. Consequently, it constitutes many-to-one mapping that is not sufficient for a suitable assessment of the model. We therefore present a second indicator, in the form of 2D histograms of adjacent pixels in an image, in the $\underline{x}$, $\underline{y}$ and diagonal orientations. The latter provides an estimate of the image second order distribution. The 2D histogram of an image $X(\eta_1, \eta_2)$ with 2D bins $\{S_{l_1,l_2}\}_{l_1,l_2}$ is defined as follows:

$$H(i,j) = \frac{1}{N} \sum_{(p_1,p_2)\in\mathcal{Q}} 1_{(X(p_1),X(p_2))\in\mathcal{S}_{i,j}}, \tag{5.3}$$

where the set $\mathcal{Q}$ contains the locations of all adjacent pixels in $X(\eta_1, \eta_2)$, $p_i = (\eta_1^i, \eta_2^i)$ is a pixel location, and $N = |\mathcal{Q}|$. This is a discrete surface with two coordinates, representing the two gray levels of adjacent pixels, denoted by l_1 and l_2.

For smooth images, it is expected that $|l_1 - l_2| < \epsilon$ for a small ϵ. Most of the energy of the histogram resides close to the line $l_1 = l_2$. This is the case in the presence of edges as well, due to the relatively low number of edge pixel pairs with respect to smooth pixel pairs in natural images.

In the case of stochastic textures, suitable for the fBm model, adjacent pixels exhibit negative correlation. It is therefore expected that $|l_1 - l_2|$ will be large. For isotropic stochastic textures, the histogram shape is expected to approximate a 2D normal distribution, whereas for anisotropic stochastic textures, negative correlation is still exhibited, but the histogram shape deviates from the normal distribution.

Fig. 6 depicts three examples of characteristic images concerned with the proposed model, along with their 2D histograms and cross-cuts along the main and secondary diagonals. The first image (Fig. 6a) is clearly a non-stochastic texture. In this and other structured natural images, there is significant correlation between

adjacent pixels as they are mostly smooth. Therefore, the energy of the 2D histogram is concentrated near the diagonal, $l_1 = l_2$. The cross-cut graph shows a non-Gaussian behaviour.

The second image (Fig. 6b) is of a stochastic texture, well fitting the proposed fBm model. In this case, we distinctly see a 2D normal distribution. The third image (Fig. 6c) is of a stochastic texture, which does not fit the model in its current form. For this type of anisotropic textures we propose a regularization that takes care of the anisotropy in the context of SR.

Anisotropic stochastic textures, while being similar to isotropic in large scales, are composed of small oriented shapes, which are anisotropic and do not exhibit self-similarity in arbitrarily small scales. The characteristic 2D histogram of such textures deviates from a 2D multivariate normal, but it is still sufficiently close enough to the Gaussian.

In Wavelet-based analysis of coefficients in adjacent scales or orientations, similar to what was done in [20], the same properties can be observed, indicating that stochastic textures indeed obey a Gaussian distribution. This was previously shown in Fig. 1.

5.2 Histogram of an fBm image

In this subsection it will be shown that images which are fBm realizations have a histogram shaped as a Gaussian density function. While the latter is not a sufficient condition to fit the model, it nevertheless provides a justification for the usage of this in the classification algorithm as a feature. The theorem will be proven for the 1D case.

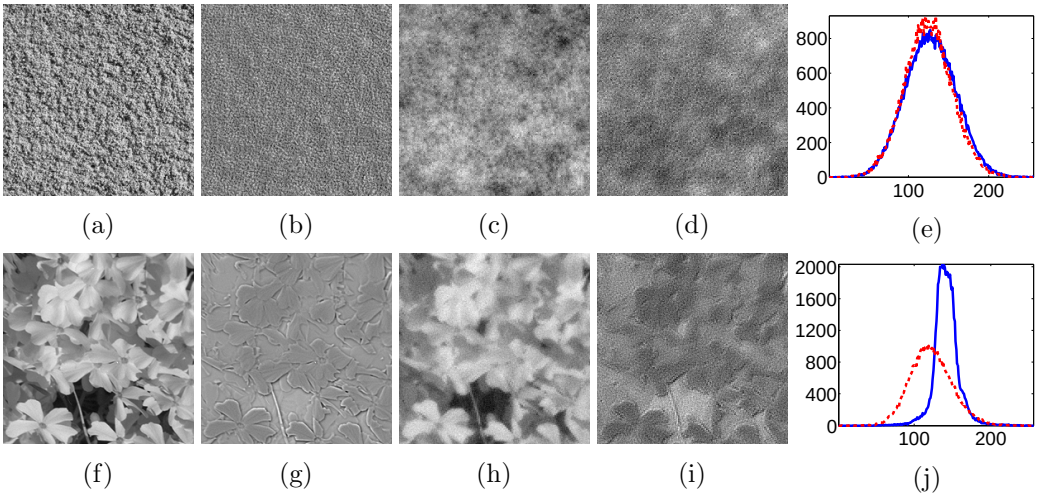
Theorem 1. *Let $\{B_H(n)\}_{n=0}^{N}$ be a 1D fBm process, sampled on a discrete grid, and let $h_b(n)$ be its histogram. Then, in the limit of $H \to 0$ and $N \to \infty$, the histogram is of Gaussian shape; the ith bin, in the range $(t_i, t_i + 1]$, is defined as follows:*

$$h_b(i) = N\left(\Phi\left(t_{i+1}\right) - \Phi\left(t_i\right)\right), \tag{5.4}$$

where $\Phi(x)$ is the cumulative distribution function of a standard Normal variable.

Proof. Let $\{B_H(n)\}_{n=0}^{N}$ be a 1D fBm process, sampled on a discrete grid. Let $\hat{h}_b(n)$ be a histogram, where b is the number of bins. Let $\{t_i\}_{i=0}^{b}$ be a set of points, so that $t_0 < t_1 < ... < t_b$, and let $\mathcal{S}_i = (t_i, t_{i+1}]$. The ith histogram bin is defined as follows:

$$\hat{h}_b(i) \;=\; \sum_{k=0}^{N} 1_{B_H(k) \in \mathcal{S}_i}. \tag{5.5}$$



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 5: Comparison of two natural textures, one suitable and one unsuitable for the model described in section 5: (a) and (f) Original images of natural textures. (c) and (h) 2D fBm versions of the respective images. (b) and (g) High-pass versions of the original images respectively. (d) and (i) High-pass versions of the 2D fBm images, respectively. (e) and (j) Histograms of the two high passed versions respectively. The high pass version histogram of the original images is highlighted in red-dashed, and the histogram of the high pass versions of the 2D fBm is highlighted in blue. In (e), the histograms are similar, indicating the model is suitable for the image. In (j), however, the two histograms are distinctly different, indicating a failure in the attempt to represent the texture and the model.

Since $B_H(n)$ is a random process, we define its histogram bins as follows:

$$\begin{aligned} h_b(i) &= E\left[\hat{h}_b(i)\right] = E\left[\sum_{k=0}^{N} 1_{B_H(k)\in\mathcal{S}_i}\right] \\ &= \sum_{k=0}^{N} E\left[1_{B_H(k)\in\mathcal{S}_i}\right] \\ &= \sum_{k=0}^{N} P\left(B_H(k)\in\mathcal{S}_i\right). \end{aligned} \tag{5.6}$$

$B_H(k)$ is a multivariate Normal process with a positive definite autocorrelation matrix. It therefore has a suitable density, $f_B(\alpha_1,...,\alpha_N)$. The inner term in Eq. (5.6) can be calculated as follows:

$$P\left(B_H(k)\in\mathcal{S}_i\right) = \int_{\alpha\in\Omega_i} f_B(\alpha)d\alpha_1\cdots d\alpha_N, \tag{5.7}$$

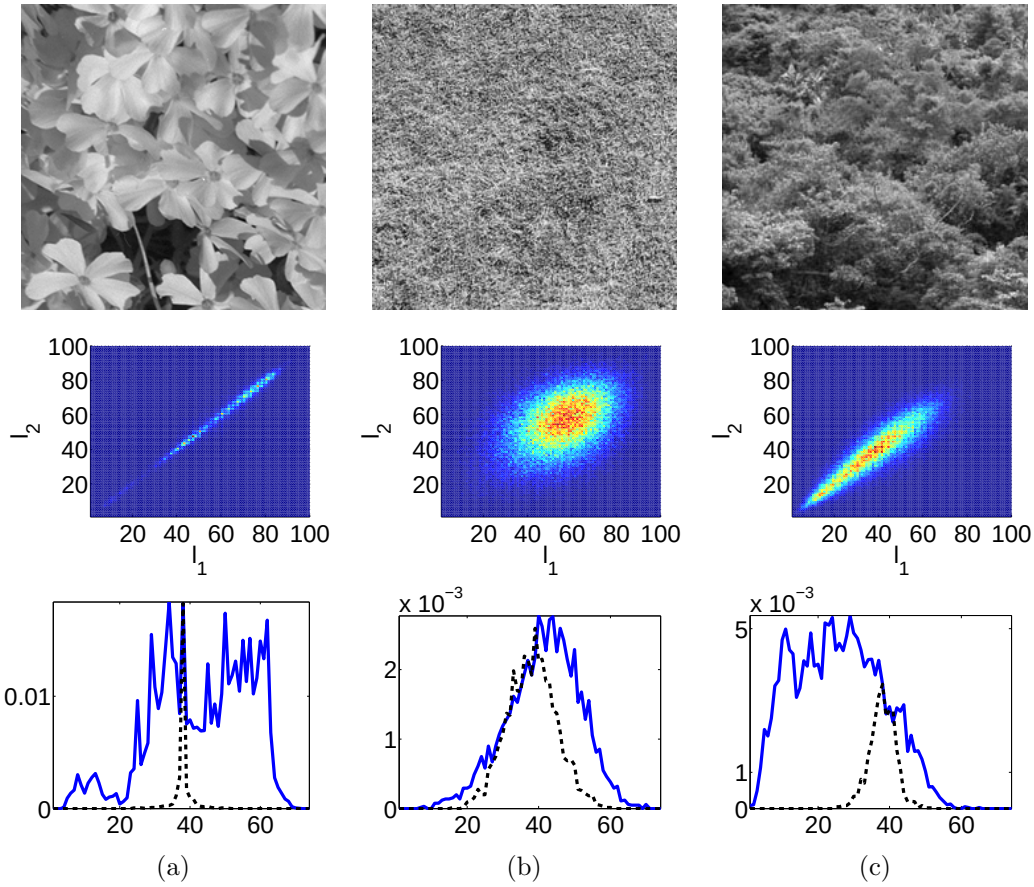


Figure 6: 2D histogram of three characteristic textures: Original images (first row), 2D histograms (second row) and cross-cuts of the main and secondary diagonals of the histograms (third row), depicted in blue and black-dashed, respectively. (a) Non-stochastic texture: The 2D histogram energy data are clustered co-linearly along the line $l_1 = l_2$ due to the high correlation between adjacent pixels, and the cross-cuts show non-Gaussian behaviour. (b) Stochastic texture, suitable for the fBm model: The 2D histogram depicts a shape of a normal distribution, also apparent in the cross-cuts. (c) Anisotropic stochastic texture, which does not fit the current model: In this case, one obtains a non-Gaussian distribution. In both the histogram and the cross-cuts it is apparent that the shape is close to a Gaussian. These types of images will be considered for the anisotropic model.

where this is an N-dimensional integral, and the domain, Ω_i, is $\mathbb{R}$ for any $\alpha_j, i \neq j$, and $\mathcal{S}_i$ for α_i.

The autocorrelation of $B_H(n)$ is discrete and given by:

$$\begin{aligned} R_B(t,s) &= E[B(t)B(s)] \\ &= \frac{\sigma^2}{2}\left(t^{2H}+s^{2H}-(t-s)^{2H}\right). \end{aligned}$$

In the limit of $H \to 0$, we obtain:

$$\begin{aligned} \lim_{H\to 0^+}\frac{2R_B(t,s)}{\sigma^2} = \\ \lim_{H\to 0^+}\begin{cases} t^{2H}+s^{2H}, & t=s \\ t^{2H}+s^{2H}-(t-s)^{2H}, & t\neq s \end{cases} \\ = \begin{cases} 2, & t=s \\ 1, & t\neq s. \end{cases} \end{aligned} \tag{5.8}$$

Therefore, in this case, $B_H(n)$ is a stationary process, and for any sample point of the process, k, the following equation holds:

$$P(B_H(k)\in\mathcal{S}_i) = P(B_H(1)\in\mathcal{S}_i).$$

Eq. (5.7) now simplifies to the following equation:

$$P(B_H(k)\in\mathcal{S}_i) = \int_{\alpha_1\in\mathcal{S}_i}\int_{\alpha_2\in\mathbb{R}}\cdots\int_{\alpha_N\in\mathbb{R}} f_B(\alpha)d\alpha, \tag{5.9}$$

and the histogram can be represented as follows:

$$h_b(n) = N\cdot P(B_H(1)\in\mathcal{S}_i). \tag{5.10}$$

Note that if $B_H(k)$ was an independent vector, then we would obtain that:

$$\begin{aligned} P(B_H(k)\in\mathcal{S}_i) &= \int_{\alpha\in\mathcal{S}_i} f_{B_1}(\alpha_1)d\alpha\cdot \\ &\quad \cdot\int_{\alpha\in\Omega} f_{B'}(\alpha')d\alpha' \\ &= \int_{\alpha\in\mathcal{S}_i} f_{B_1}(\alpha_1)d\alpha, \end{aligned} \tag{5.11}$$

where $B'(n)$ is the process without the first sample, and $\alpha' = (\alpha_2,\ldots,\alpha_N)$. However, they are dependent, but stationary, with the following covariance matrix:

$$\Lambda = \begin{pmatrix} 2\alpha & \alpha & \cdots & \alpha \\ \alpha & 2\alpha & \ddots & \vdots \\ \alpha & \ddots & \ddots & \alpha \\ \alpha & \alpha & \alpha & 2\alpha \end{pmatrix}, \tag{5.12}$$

for a suitable α, which depends on the variance, σ^2. Without loss of generality, we henceforth assume that $\alpha = 1$. Using the lemma shown in appendix A, we obtain the inverse of the covariance matrix:

$$\Lambda^{-1} \triangleq \Sigma = I - \frac{1}{N(N+1)}\mathbf{1}\mathbf{1}^T, \tag{5.13}$$

where $\mathbf{1} \in \mathbb{R}^{N\times 1}$ is a vector with 1 in every entry. In the limit of $H \to 0$, the following can be observed:

$$\begin{aligned} d_\Sigma(N) &= \|\Sigma - I\| = \left\| \frac{1}{N(N+1)}\mathbf{1}\mathbf{1}^T \right\| \\ &= N \cdot \left\| \frac{1}{N(N+1)} \right\| = \frac{1}{N+1}. \end{aligned} \tag{5.14}$$

For a size of small images, for instance $N = 128^2 = 2^{14}$, this difference is $d_\Sigma(N) \cong 6 \cdot 10^{-5}$. Recalling the probability density of $B_H(t)$, $f_B(\alpha_1, ..., \alpha_N)$, we obtain the following:

$$\begin{aligned} f_B(\alpha) &= \frac{1}{(2\pi|\Lambda|)^{N/2}} \exp\left(-\frac{1}{2}\alpha^T\Lambda^{-1}\alpha\right) \\ &= \frac{1}{(2\pi|\Lambda|)^{N/2}} \exp\left(-\frac{1}{2}\alpha^T\Sigma\alpha\right) \\ &= \frac{1}{(2\pi|\Lambda|)^{N/2}} \exp\left(-\frac{1}{2}\alpha^T\left(I - \frac{1}{N(N+1)}\mathbf{1}\mathbf{1}^T\right)\alpha\right) \\ &= C\exp\left(-\frac{1}{2}\alpha^T I\alpha + \frac{1}{2}\frac{1}{N(N+1)}\alpha^T\mathbf{1}\mathbf{1}^T\alpha\right), \end{aligned}$$

where $C \triangleq \frac{1}{(2\pi|\Lambda|)^{N/2}}$. Denote α' as $(\alpha_2, \ldots, \alpha_N)$, $\alpha^\dagger = (\alpha')^T$. Therefore:

$$\begin{aligned} f_B(\alpha) &= C\exp\left(-\frac{1}{2}\alpha_1^2 + \frac{\alpha_1^2 + 2\alpha_1\sum_{j=2}^N \alpha_j}{2N(N+1)}\right) \cdot \\ &\quad \cdot \exp\left(-\frac{1}{2}\alpha^\dagger I\alpha' + \frac{1}{2}\frac{1}{N(N+1)}\alpha^\dagger\mathbf{1}\mathbf{1}^T\alpha'\right). \end{aligned} \tag{5.15}$$

Denoting the second term as $g(\alpha')$, we obtain:

$$g(\alpha') = \exp\left(-\frac{1}{2}\alpha^\dagger\Sigma_2\alpha'\right),$$

where $\Sigma_2 = I - \frac{1}{N(N+1)}\mathbf{1}\mathbf{1}^T$ and $I \in \mathbb{R}^{N-1\times N-1}$. This matrix is equal to Σ for a vector dimension of $N-1$. Therefore, $g(\alpha')$ constitutes a multivariate Normal

density (up to a scaling factor). Turning to the argument of the first term in Eq. (5.15), we obtain:

$$\begin{aligned} & -\frac{1}{2}\alpha_1^2 + \frac{1}{2N\left(N+1\right)}\left(\alpha_1^2 + 2\alpha_1 \sum_{j=2}^{N} \alpha_j\right) \\ = \quad & -\frac{1}{2}\alpha_1^2 + \frac{\alpha_1^2}{2N\left(N+1\right)} + \frac{\alpha_1}{N\left(N+1\right)} \sum_{j=2}^{N} \alpha_j. \end{aligned} \tag{5.16}$$

The last term in the equation above can be bounded as follows:

$$\frac{\alpha_1 \cdot \alpha_{\min}}{N+1} \leq \frac{\alpha_1}{N\left(N+1\right)} \sum_{j=2}^{N} \alpha_j \leq \frac{\alpha_1 \cdot \alpha_{\max}}{N+1}. \tag{5.17}$$

Therefore, as $N \to \infty$, this term is negligible. The same applies for the second term as well, and in the limit of $N \to \infty$ we, therefore, obtain:

$$f_B\left(\alpha\right) \quad = \quad \mathcal{N}\left(\alpha_1; 0, 1\right) \cdot \mathcal{N}\left(\alpha'; 0, \Sigma_2^{-1}\right),$$

where $\mathcal{N}\left(x; \mu, S\right)$ is a multivariate Normal density function, for the vector x, with mean vector μ and covariance matrix S.

The probability in Eq. (5.9) can, therefore, be calculated as follows:

$$\begin{aligned} P\left(B_H\left(k\right) \in \mathcal{S}_i\right) \quad &= \quad \int\limits_{\alpha_1 \in \mathcal{S}_i} \mathcal{N}\left(\alpha_1; 0, 1\right) \cdot \\ & \qquad \cdot \int\limits_{\mathbb{R}} \cdots \int\limits_{\mathbb{R}} \mathcal{N}\left(\alpha'; 0, \Sigma_2^{-1}\right) d\alpha_2 \cdots d\alpha_N \\ &= \quad \int\limits_{\alpha_1 \in \mathcal{S}_i} \mathcal{N}\left(\alpha_1; 0, 1\right) d\alpha_1 \\ &= \quad \Phi\left(t_{i+1}\right) - \Phi\left(t_i\right). \end{aligned} \tag{5.18}$$

The i'th bin in the histogram of $B_H\left(t\right)$ will therefore have a value of $N \cdot (\Phi\left(t_{i+1}\right) - \Phi\left(t_i\right))$, implying a shape of a Gaussian distribution. □

5.3 Hurst parameter under blur

Under the 2D fBm model, the effects of blur and decimation are of importance.

1. Estimation of the original Hurst parameter may not be achieved directly from the degraded image, as the degraded image may not fit the fBm model, and therefore may not generally possess such a parameter.

2. The Hurst parameter controls the irregularity of the image, which translates to high-frequency details, lost by degradation. Obtaining the original H of an image can be used as a stopping criteria for the SR process.

3. Analysis of the imaging model effects can be used to properly identify the details lost by degradation.

At this stage, the analysis is presented for 1D discrete fBm. Let $x \sim \mathcal{N}(0, \Sigma_X)$ be a multivariate normal variable with covariance matrix Σ_X of a discrete fBm with a given H parameter. Let $n \sim \mathcal{N}(0, \sigma_n^2 I)$ be white Gaussian noise. Let $y = DBx + n \triangleq Wx + n$ be the degraded version of x, where D and B are decimation and blur matrices respectively. We further assume that the noise is negligible (for the sake of simplicity) and there is no decimation. Future studies will include the effects of decimation as well. The blur matrix performs Gaussian blur with given variance σ_B^2, and its support is approximately limited by N_B pixels.

The autocorrelation of the fBm (Eq. (3.1)) can be divided to two parts:

$$R_{X,1}(t,s) = \frac{\sigma^2}{2}(|t|^{2H} + |s|^{2H}) \tag{5.19}$$

$$R_{X,2}(t,s) = \frac{\sigma^2}{2}(-|t-s|^{2H}), \tag{5.20}$$

where $R_X(t,s) = R_{X,1} + R_{X,2}$. The first part, $R_{X,1}(t,s)$, is non-stationary, and the second part is stationary. The covariance of x shall be the discrete, sampled, version of this autocorrelation. Under the imaging model, the covariance of y shall be the following:

$$\Sigma_Y = B\Sigma_X B^T = B\Sigma_{X_1} B^T + B\Sigma_{X_2} B^T. \tag{5.21}$$

Lemma 1. *Let $b \in \mathbb{R}^{N_f}$ be a blur filter and $a \in \mathbb{R}^N$ be a vector with values $a_i = i^\alpha$, where $\alpha \in (0,1)$. Let S_m be the following sum:*

$$S_m = (b * a)(m) = \sum_{i=0}^{N_f} b_i a_{m-i}, \tag{5.22}$$

Then, $|S_m - a_m| < \epsilon$ for a small positive number ϵ.

Proof. The error for any m is defined as follows:

$$e_m = \left| m^\alpha - \sum_{i=0}^{N_f} f_i \cdot (m-i)^\alpha \right|$$

In the case of $m > N_f$, $m > i$. Using the generalized binomial theorem for $(m-i)^\alpha$

and since $\alpha < 1$,

$$\begin{aligned}\binom{\alpha}{k} &= \frac{\alpha\left(\alpha-1\right)\left(\alpha-2\right)\cdots\left(\alpha-k+1\right)}{k!} \\ &= \left(-1\right)^{k-1}\frac{\alpha\left(1-\alpha\right)\left(2-\alpha\right)\cdots\left(k-1-\alpha\right)}{k!} \\ &\triangleq \left(-1\right)^{k-1}A\left(\alpha,k\right),\end{aligned}$$

where $A\left(\alpha,k\right) > 0$. Therefore,

$$\begin{aligned}e_m &= \left|m^\alpha - m^\alpha\sum_{i=0}^{N_f}\left(f_i\cdot\sum_{k=0}^{\infty}\binom{\alpha}{k}\left(-1\right)^k\left(\frac{i}{m}\right)^k\right)\right| \\ &= \left|m^\alpha - m^\alpha\sum_{i=0}^{N_f}f_i + \sum_{i=0}^{N_f}\left(f_i\cdot\sum_{k=1}^{\infty}\binom{\alpha}{k}\left(-1\right)^k\left(\frac{i}{m}\right)^k\right)\right| \\ &= \left|\sum_{i=0}^{N_f}\left(f_i\cdot\sum_{k=1}^{\infty}\binom{\alpha}{k}\left(-1\right)^k\left(\frac{i}{m}\right)^k\right)\right| \\ &= \left|-\sum_{i=0}^{N_f}\left(f_i\cdot\sum_{k=1}^{\infty}A\left(\alpha,k\right)\left(\frac{i}{m}\right)^k\right)\right| \\ &= \sum_{i=0}^{N_f}\left(f_i\cdot\sum_{k=1}^{\infty}A\left(\alpha,k\right)\left(\frac{i}{m}\right)^k\right).\end{aligned}$$

Since $A\left(\alpha,k\right) = \frac{\alpha(1-\alpha)(2-\alpha)\cdots(k-1-\alpha)}{k!} < \frac{\alpha(k-1)!}{k!} = \frac{\alpha}{k}$,

$$\begin{aligned}e_m &< \alpha\sum_{i=0}^{N_f}\left(f_i\cdot\sum_{k=1}^{\infty}\frac{1}{k}\left(\frac{i}{m}\right)^k\right) \\ &= \alpha\sum_{i=0}^{N_f}\left(f_i\cdot-\left(\log\left(1-\frac{i}{m}\right)\right)\right) = \alpha\log\left(\prod_{i=0}^{N_f}\left(\frac{m}{m-i}\right)^{f_i}\right) \triangleq \epsilon.\end{aligned}$$

Since $i < N_f$, $\frac{m}{m-i}$ is close to 1, and the resulting term is negligible.

□

Corollary 1. *A blur operator has no more than negligible effect on the non-stationary part of the fBm autocorrelations.*

Proof. Inspecting the right-hand side of Eq. (5.21), the non-stationary part after blur is $B\Sigma_{X_1}B^T$. This is composed of two terms, one varying in the horizontal direction and one varying in the vertical direction. Let us inspect the effects of

applying B on the left-hand side of Σ_{X_1}. Denoting $A = B\Sigma_{X_1}$, each $[A]_{(i,j)}$ is the inner product of b_i and v, where b_i is a 1D blur filter with support N_f and centered at pixel i, and v is a vector with values $v_i = |i|^\alpha$. Using lemma 1 with $\alpha = 2H$, we therefore obtain that

$$\left|[A]_{(i,j)} - [\Sigma_{X_1}]_{(i,j)}\right| < \alpha \log \left(\prod_{i=0}^{N_f} \left(\frac{m}{m-i} \right)^{f_i} \right) \tag{5.23}$$

□

To demonstrate the derived bound, we can use two common low-pass filters: An averaging filter with $N_f = 5$ and a Gaussian filter, with $\sigma = 1.5$ and effective support of $N_f = 5$. Direct calculation with $H = 0.2$ yields a bound of 0.028 for both filters.

Regarding the stationary part in Eq. (5.21), $B\Sigma_{X_2}B^T$, the resulting term is obtained as 1D blurring of each column of the original matrix, Σ_{X_2}. From the same considerations as the above, this causes negligible change in points far from the main diagonal. As for the main diagonal, let us observe a cross-section, given by Eq. (5.20). In continuous setting, this is a unimodal, symmetric function with a non-differentiable peak at $t = s$. Applying a blur filter on this function causes the peak to be "smeared", or truncated, while relatively unaffecting the rest of the function.

We conclude this discussion by the following observation:

$$\Sigma_Y \overset{\sim}{=} \Sigma_X - \sigma_b I_b, \tag{5.24}$$

where Σ_Y is the blurred signal covariance, Σ_X is the original covariance, σ_b is a parameter and I_b is a blurred identity matrix. This allows us to obtain an estimate of the covariance matrix of the original signal, which can be exploited to obtain the original value of H. It is important to note that this equation can assist in restoring the covariance (or other statistical properties) of the original signal, but it is insufficient for restoring the signal itself.

5.4 Simulation and H estimation

The phenomenon described by Eq. (5.24) can be observed via simulation, as well as exploited to create an estimator for the Hurst parameter of the original signal under blur. Depicted in Figs. 7 and 8 are the effects of two common blur filters (averaging and Gaussian) applied on an fBm, on the autocorrelation of the signal. We see (Figs. 7h and 8h) that the principle effect of the blur filter on $R_{X,2}$ is equivalent to a subtraction of an approximated scalar matrix. While blurring causes distortions in $R_{X,1}$ as well (Figs. 7f and 8f), it is less pronounced in the resulting image, since these correlations are of a distant sample with the first few samples.

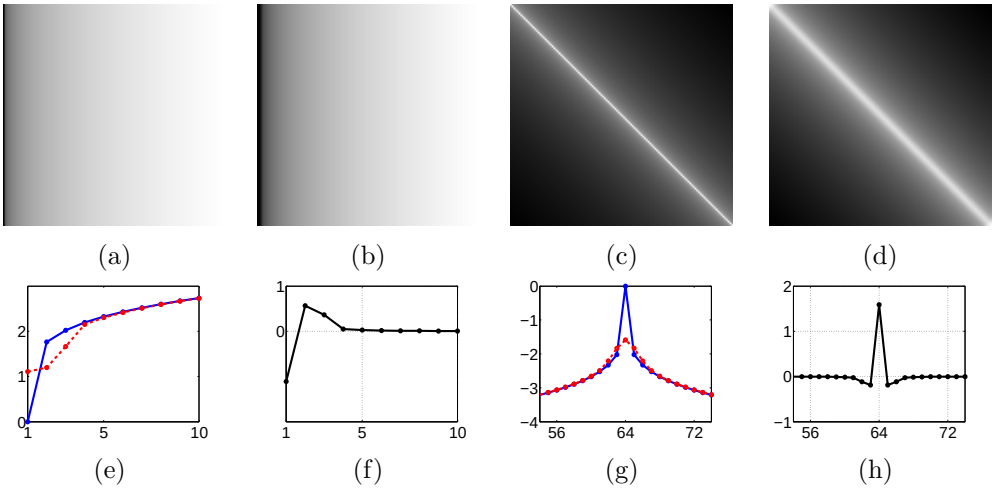


Figure 7: Effects of an averaging blur (kernel size 5 pixels) on an fBm ($N = 2^7$) with $H = 0.1$. (a) and (b) The horizontal term from $R_{X,1}$ and its blurred counterpart respectively (Eq. (5.20)). (c) and (d) $R_{X,2}$ and its blurred counterpart respectively. (e) A comparison of two cross-cuts from Figs. (a) and (b). (f) The difference between the two cross-cuts. (g) A comparison of two cross-cuts from Figs. (c) and (d). (h) The difference between the two cross-cuts. The main effect of the blur filter is depicted in the latter; one can restore the original autocorrelation matrix from a blurred (discrete) fBm signal via compensating for the blurred diagonal matrix lost by the blur filter. In both this case and the Gaussian blur case (Fig. 8), the diagonal matrix is approximately an identity matrix.

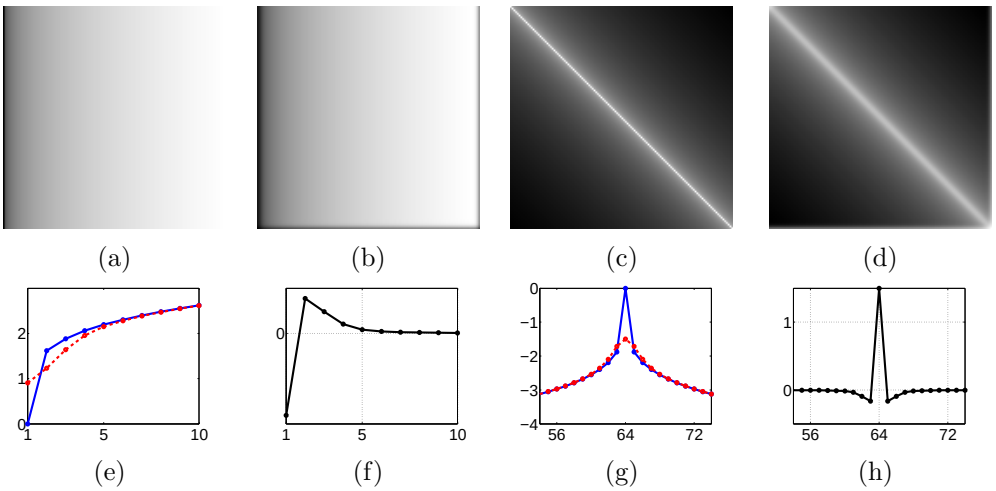


Figure 8: Effects of a Gaussian blur ($\sigma = 1.5$) on an fBm ($N = 2^7$) with $H = 0.1$. For details see Fig. 7.

Estimation source	Mean	Standard deviation	Error
Original image ($H = 0.1$)	$\hat{H}_{orig} = 0.103$	$\sigma_{orig} = 5 \cdot 10^{-3}$	3.5%
Blurred image	$\hat{H}_{blur} = 0.84$	$\sigma_{blur} = 15 \cdot 10^{-3}$	$> 100\%$
Y_r	$\hat{H}_r = 0.095$	$\sigma_r = 6 \cdot 10^{-3}$	5%

Figure 9: Estimating H from a corresponding image Y_r, created by superposition of a blurred image with a random image with autocorrelation compensating for the degradation (for details see text). To test the validity of our estimator, we first use it on the ground truth images ($T = 50$ images were used in the experiments). The corresponding image yields a relatively accurate estimate of the Hurst parameter.

Given a blurred fBm image, we propose the following estimation scheme for the original Hurst parameter: Given the blur filter properties, it is possible to calculate the difference between the original autocorrelation matrix of a discrete fBm and the autocorrelation of the blurred one. Then, using Eq. (5.24) (with the required adaptations for a 2D image), we create a new image, Y_r, which we denote the corresponding image, as follows:

$$Y_r(\eta_1, \eta_2) = Y(\eta_1, \eta_2) + Y_h(\eta_1, \eta_2), \tag{5.25}$$

where $Y_h(\eta_1, \eta_2)$ is a Gaussian random image with autocorrelation matrix $\sigma_b I_b$. $Y_r(\eta_1, \eta_2)$ will therefore posses the autocorrelation of the original image, $X(\eta_1, \eta_2)$. Then, we apply Hurst parameter estimation algorithms to estimate H.

This scheme was simulated as well. For the estimation of H we used a regression scheme on the logarithm of the variance of the image increments. This method yields both the parameter H and the variance of the fBm (which is dependent on H). For $T = 50$ fBm images of size $2^8 \times 2^8$ with $H = 0.1$, we first estimated the value of H from the original images, and then estimated H from the blurred and corresponding images. The results are summarized in Table 9.

The results show that estimation of the Hurst parameter from the blurred image does not yield good results. This high value of H ($\hat{H}_{blur} = 0.84$) can be partially justified by the resemblance of a blurred fBm to an fBm with a higher H (where the higher correlation between pixels yields a smoother image), but this is not the case; as shown previously, the autocorrelation of a blurred fBm image is no longer an fBm. On the other hand, with an error of 5%, the estimation via the corresponding image yields relatively accurate results, encouraging us to further explore this method in order to estimate the original H from a given image.

6 Superresolution algorithm

6.1 Anisotropic diffusion

A brief review of the anisotropic diffusion that will suffice for our application is provided. For a comprehensive exposition see, for example, [23, 36]. Using PDE-based methods allows for adaptive filtering of an image, with low computational complexity. The following PDE equation suitable for image processing was introduced in this context by Perona and Malik [37]:

$$I_t = \nabla \cdot (g(\nabla I)\nabla I), \tag{6.1}$$

with the initial condition $I|_{t=0} = I_0$, the diffusivity $g(s)$ being a decreasing function with the following properties: $g(0) = 1, g(s) \geq 0$ and $g(s) \to 0$ as $s \to \infty$. The diffusivity function is, in general, designed for the adaptive processing of images to allow high diffusion in low gradient areas, assumed to be noisy, and low diffusion in high gradient areas, indicating the presence of edges that should not be compromised.

Many choices have been suggested for this function. Perona and Malik (PM) proposed $g(s) = e^{-(s/K)^2}$ and $g(s) = \frac{1}{1+\left(\frac{s}{K}\right)^2}$, which are commonly used with a suitable constant, K. Also useful is the TV-based diffusivity, using the $\mathcal{L}_1$ norm: $g(s) = \frac{1}{|s|}$, or the regularized version, $g(s) = \frac{1}{\sqrt{\epsilon^2+s^2}}$. Gilboa et al. have even extended it to the negative regime [36].

This diffusion, although commonly referred to *anisotropic*, is in fact non-linear but isotropic. This has been noted by Weickert, who introduced a truly anisotropic diffusion process, commonly referred to as tensor diffusion:

$$I_t = \nabla \cdot (D(\nabla I)\nabla I), \tag{6.2}$$

where $D \in \mathbb{R}^{2\times 2}$ is a tensor that is represented, using an eigenvalue decomposition, as follows:

$$D = (\omega_1, \omega_2) \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \end{pmatrix}, \tag{6.3}$$

where ω_1 and ω_2 are eigenvectors which satisfy:

$$\omega_1 \parallel \nabla I, \quad \omega_2 \perp \nabla I, \tag{6.4}$$

and λ_1 and λ_2 are the corresponding eigenvalues. This formulation allows for different types of diffusion to be performed in different orientations within the image. In edge enhancing diffusion, for instance, only the diffusion coefficient perpendicular to the edge orientation will assume a significant value. This method further emphasizes edges while smoothing noisy image areas. Instead of a single diffusivity function, $g(x)$, two functions are used - one for each eigenvalue.

6.2 Texture-based tensor diffusion

The tensor, D, is a function of the gradient of the image, ∇I. Due to the fact that textures contain small oriented elements, the tensor diffusion is a natural choice for their enhancement. However, commonly used tensor diffusion schemes do not yield a successful enhancement of textures; while the edges are emphasized, the high frequency components of textural details are lost. This is due to their spectral resemblance of noise. Further, when attempting to enhance a blurred image, much of the high frequency details are in effect non-existing to begin with.

This has encouraged us to consider a different function, instead of ∇I, for the calculation of the tensor. This function is required to represent the desired properties of the texture, while still resembling the shape of the texture itself.

One cannot expect to represent a natural texture using a single parameter, H. As discussed in section 3.2, it is possible to consider a structure function, $\phi(\eta_1, \eta_2)$, to create a non-stationary field which better represents the desired image. Instead of using a general function, we use a structure function generated from the degraded image itself. This yields an image which contains the details of the degraded image, along with correlations introduced according to the specific structure of the non-stationary field. We refer to the structure function derived from the degraded image as the empirical structure function (ESF).

The method to recover the ESF from a given, degraded, image is based on an inverse procedure to the method of obtaining the image from the structure function, devised in [31]. Let $Y(\eta_1, \eta_2)$ be a degraded image. The increments in the $\underline{x} = \eta_1$ and $\underline{y} = \eta_2$ orientations are defined as:

$$\begin{aligned} Y_{\Delta\eta_1}(\eta_1, \eta_2) &= Y(\eta_1, \eta_2) - Y(\eta_1 - \Delta_{\eta_1}, \eta_2), \\ Y_{\Delta\eta_2}(\eta_1, \eta_2) &= Y(\eta_1, \eta_2) - Y(\eta_1, \eta_2 - \Delta_{\eta_2}), \end{aligned} \tag{6.5}$$

respectively, where Δ_{η_1} and Δ_{η_2} are small increments set to 1 in discrete schemes. The increments in the $(\underline{x}, \underline{y}) = (\eta_1, \eta_2)$ coordinates are defined as:

$$\begin{aligned} Y_{\Delta\eta_1,\Delta\eta_2}(\eta_1, \eta_2) = Y(\eta_1, \eta_2) - Y(\eta_1 - \Delta_{\eta_1}, \eta_2) \\ -Y(\eta_1, \eta_2 - \Delta_{\eta_1}) + Y(\eta_1 - \Delta_{\eta_1}, \eta_2 - \Delta_{\eta_2}). \end{aligned} \tag{6.6}$$

Let $R_{\eta_1}(\eta_1, \eta_2)$, $R_{\eta_2}(\eta_1, \eta_2)$ and $R_{\eta_1,\eta_2}(\eta_1, \eta_2)$ be the autocorrelation functions of the increments $Y_{\Delta\eta_1}$, $Y_{\Delta\eta_2}$ and $Y_{\Delta\eta_1,\Delta\eta_2}$, respectively. The autocorrelation functions for the 1D increments, $R_{\eta_1}(\eta_1, \eta_2)$ and $R_{\eta_2}(\eta_1, \eta_2)$, are derived from the structure function, $\phi(\eta_1, \eta_2)$, as follows:

$$\begin{aligned} R_{\eta_1}(\eta_1, \eta_2) =& \frac{1}{2}(\phi(\eta_1 + \Delta_{\eta_1}, \eta_2) + \phi(\eta_1 - \Delta_{\eta_1}, \eta_2) - 2\phi(\eta_1, \eta_2)), \\ R_{\eta_2}(\eta_1, \eta_2) =& \frac{1}{2}(\phi(\eta_1, \eta_2 + \Delta_{\eta_2}) + \phi(\eta_1, \eta_2 - \Delta_{\eta_2}) - 2\phi(\eta_1, \eta_2)), \end{aligned} \tag{6.7}$$

and the autocorrelation for the 2D increments, $R_{\eta_1,\eta_2}(\eta_1,\eta_2)$, is accordingly derived as:

$$R_{\eta_1,\eta_2}(\eta_1,\eta_2) = 2R_{\eta_1}(\eta_1,\eta_2) + 2R_{\eta_2}(\eta_1,\eta_2) \tag{6.8}$$

$$- \frac{1}{2}(\phi(\eta_1+\Delta_{\eta_1},\eta_2+\Delta_{\eta_2}) + \phi(\eta_1-\Delta_{\eta_1},\eta_2+\Delta_{\eta_2}) + \phi(\eta_1+\Delta_{\eta_1},\eta_2-\Delta_{\eta_2}) + \phi(\eta_1-\Delta_{\eta_1},\eta_2-\Delta_{\eta_2})). \tag{6.9}$$

To obtain the empirical structure function, it is therefore required to invert the equations, and produce $\phi(\eta_1,\eta_2)$, given the increment autocorrelation functions of $Y(\eta_1,\eta_2)$. Substituting $\Delta_{\eta_1} = \Delta_{\eta_2} = 1$ in Eq. 6.7, it follows that the 1D autocorrelation functions can be represented using convolution equations with derivative filters:

$$\begin{aligned} R_{\eta_1}(\eta_1,\eta_2) &= (\phi * f_d)(\eta_1,\eta_2), \\ R_{\eta_2}(\eta_1,\eta_2) &= (\phi * f_d^T)(\eta_1,\eta_2), \end{aligned} \tag{6.10}$$

where $f_d = \frac{1}{2}(1,-2,1)$. The 2D autocorrelation can be represented in a similar manner, using the following equation:

$$R_{\eta_1,\eta_2}(\eta_1,\eta_2) = (\phi * f_{d2})(\eta_1,\eta_2), \tag{6.11}$$

where

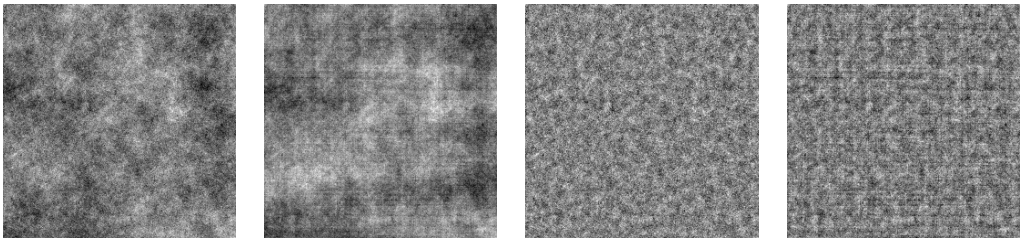
$$f_{d2} = \frac{1}{2}\begin{pmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{pmatrix}. \tag{6.12}$$

Obtaining the ESF from the degraded image, is therefore reduced to solving Eqs. (6.10) and (6.11). This can be formulated as the following least-squares (LS) problem (Appendix B):

$$\phi = \arg\min_x \|D_f x - r\|_2^2, \tag{6.13}$$

where ϕ is the column-stack representation of the ESF, $\phi(\eta_1,\eta_2)$, D_f in a suitable matrix representation of f_d and f_{d2}, and r is a suitable column-stack representation of $R_x(\eta_1,\eta_2)$, $R_y(\eta_1,\eta_2)$ and $R_{x,y}(\eta_1,\eta_2)$. This is an ill-posed problem, due to rank deficiency of the derivative matrix, D_f. It is similar to problems encountered in gradient domain processing. This poses a challenge to a least-squares procedure and we currently do not employ regularization techniques, although these may be relevant in further studies. Additionally, unlike problems in which the vector, ϕ, needs to be recovered exactly, in this case only the derivatives (in the manner of the derivative filters, f_d and f_{d2}) of the ESF are required. The derivation of these matrices and vectors is addressed in further details in appendix B.

An example is presented in Fig. 10, where two sets of images are shown. The first (Fig. 10a and Fig. 10b) contains a 2D fBm image, with $H = 0.1$, and the



(a) (b) (c) (d)

Figure 10: Comparison of 2D fBm using the empirical structure function (ESF). (a): Original 2D fBm image. (b) Restored 2D fBm image, using ESF derived from Fig. (a). High pass versions of (a) and (b) are shown in Fig. (c) and (d), respectively.

restored image derived by extracting the ESF from the 2D fBm image by the process described earlier, and reconstructing the field by the algorithm presented in [31]. Due to the ill-posedness of the LS problem, the two images do not look alike. The second set (Fig. 10c and Fig. 10d) depicts the respective images obtained from the first set, after a high-pass filtering was performed by subtracting the result from a Gaussian low-pass filter with $\sigma = 15$. Since only the high frequency range is lost by degradation, this range is of more importance. Indeed, the high pass versions of Figs. (10a) and (10b), depicted in Figs. (10c) and (10d) appear to be visually similar.

Using the ESF, its a possible to obtain an image, $Y_\phi(\eta_1, \eta_2)$, from the degraded image, $Y(\eta_1, \eta_2)$, by calculating the autocorrelation of the first- and second-order increments, solving the LS problem in Eq. (6.13) to obtain a structure function $\phi(\eta_1, \eta_2)$, and using the synthesis algorithm in [31]. The resulting image is referred to as the empirical image.

6.3 Tensor diffusion

We now consider the modifications required to enable the tensor diffusion to perform superresolution on natural textures. The tensor, $D(\nabla I)$, introduced earlier, is set instead to be $D(\nabla(I_t + \alpha Y_\phi(\eta_1, \eta_2)))$, where $Y_\phi(\eta_1, \eta_2)$ is the empirical image, and α is a weight parameter. This allows for the introduction of missing texture details, while still emphasizing the edges of a degraded texture image.

The superresolution algorithm is presented by considering the following energy functional, in column-stacked image representation:

$$E(X) = \int_\Omega (BX - Y)^2 + (\hat{X}_{HP} - H_{HP}X)^2 + \\ + \beta\Psi(|\nabla X + \alpha\nabla Y_\phi|^2)dxdy, \tag{6.14}$$

where X is the SR image, Y is the degraded image, and B is a matrix performing blur and decimation. The second term penalizes deviations of the solution, X, from

the fBm model of the image (discussed in section 5), given by $\hat{X}$. This is derived as follows:

$$\begin{aligned}\|X - \hat{X}\|^2 &= \|X_{LP} + X_{HP} - \hat{X}_{LP} - \hat{X}_{HP}\|^2 \\ &\cong \|X_{HP} - \hat{X}_{HP}\|^2 \\ &= \|H_{HP}X - \hat{X}_{HP}\|^2, \end{aligned} \tag{6.15}$$

where H_{HP} is a Gaussian high-pass filter described earlier, and $\hat{X}_{HP}$ is the high-frequency component, given by $\hat{X}_{HP} = \mathcal{P}_{H,W}(\hat{X}_{LP})$. X_{LP} and $\hat{X}_{LP}$ are assumed to be sufficiently close so that their difference is approximately zero. This is a valid assumption, as the imaging model does not degrade the low frequencies.

The solution for this equation satisfies the following Euler-Lagrange equation:

$$\begin{aligned}B^T(BX - Y) - \beta\nabla \cdot (\Psi'(|\nabla(X + \alpha Y_\phi)|^2)\nabla X) \\ -2\alpha\beta\nabla \cdot (\Psi'(|\nabla(X + \alpha Y_\phi)|^2)\nabla Y_\phi) = 0. \end{aligned} \tag{6.16}$$

This derivation is further explained in appendix C. A gradient descent scheme for the minimization of the energy functional, and the introduction of the tensor diffusion, $D(\nabla(X+Y_\phi))$, instead of the scalar diffusivity, $\Psi(|\nabla(X+Y_\phi)|^2)$, yields the following diffusion-reaction process:

$$\begin{aligned}X_t &= 2B^T(BX - Y) - 2H_{HP}^T(\hat{X}_{HP} - H_{HP}X) \\ &\quad -\beta\nabla \cdot (D(\nabla(X + \alpha Y_\phi))\nabla X) \\ &\quad -2\alpha\beta\nabla \cdot (D(\nabla(X + \alpha Y_\phi))\nabla Y_\phi). \end{aligned} \tag{6.17}$$

This equation differs from the usual deblurring tensor diffusion with regard to the following:

1. As previously discussed, the tensor, D, is a function of two images. Based on this formulation, an extra facet emerges in the optimization.

2. The estimated high-frequency image, X_{HP}, is added, in order to recover the missing details.

The eigenvalue parallel to the gradient, λ_1, was set to a regularized $\mathcal{L}_1$ scalar diffusivity function:

$$\lambda_1(s) = \frac{1}{\sqrt{\epsilon^2 + s^2}}, \tag{6.18}$$

and the perpendicular eigenvalue, λ_2, was set to 0.01.

Using PDE-based regularization inflicts high diffusion on fine details, characterized by low gradients. Therefore, even if details are restored, the diffusion may smooth them unless posed at high gradient areas. The two terms therefore work in tandem; while recovered detail emerges from the texture model term, the modified

tensor term preserves it rather than smooth it. Further, the modified tensor term recovers lost image structure, according to the desired statistics. An example of this is demonstrated in Fig. 11: The three figures on the first row depict the original (Fig. 11a), degraded (Fig. 11b) and naive PDE-based deblurring (with bicubic interpolation) (11c) result. The second row shows different configurations of the modified diffusion. The result, using only the texture model term, is displayed in 11d; in comparison with Fig. 11c, it is possible to see new details, where these have not been smoothed - in high gradient areas. The result, using only the modified tensor term, is displayed in 11e; While no fine details are restored, the image structure is closer to the original, using the empirical image. Finally, in Fig. 11f, we see the combined effect of both terms. The new details are both restored and preserved.

6.4 Remarks

1. The empirical image, $Y_\phi(\eta_1, \eta_2)$, is initially derived from the degraded image, $Y(\eta_1, \eta_2)$. However, as the diffusion advances and the image is refined, it is beneficial to update $Y_\phi(\eta_1, \eta_2)$ as well. Due to the time consuming LS it entails, this is performed periodically after several iterations of the diffusion process.

2. The parameters of this algorithm are H, α, β and the number of diffusion iterations. Although H can be estimated from the degraded image itself, by methods of fractal dimension estimation, we currently set it to be a constant, $H = 0.1$. The other parameters have fixed values, invariant of the image in question. In an ongoing work, we investigate the effects of blur and decimation on the Hurst parameter and on an fBm in general, in order to estimate these parameters adequately from a degraded image.

3. Given a statistical model, one can use MAP or MMSE estimation, with the fBm as a prior. However, neither MAP nor MMSE produce good results in the case of SR, due to the severe loss of details. This is a further indication that an $\mathcal{L}_2$-based error criteria is not suitable for textured images. In the case of denoising, however, substantial improvement was noted, which renders this type of optimization useful.

7 Results

The proposed algorithm is implemented and used on stochastic textures. The decimation operator in Eq. (2.1) performs $\downarrow 2$ decimation in both dimensions, the blur kernel is a Gaussian with $\sigma = 1.5$ and effective support of 5×5 pixels. A small noise is added so that the BSNR is $40dB$. The contrast in all the examples is normalized

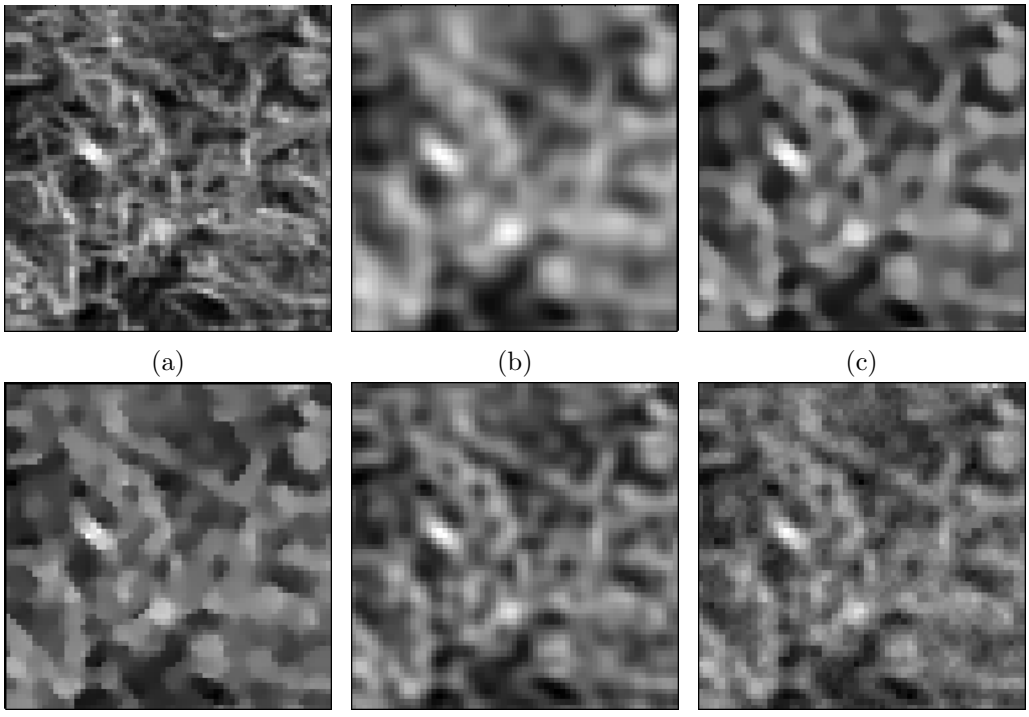


Figure 11: Visualizing the texture-based diffusion equation. (a) Original texture image. (b) Degraded image. (c) Result using standard anisotropic diffusion: The edges are emphasized and the image is less blurry with respect to the degraded image, but texture details are not recovered. (d) Result using the fBm-based texture model, without modified tensor: While the result is better than the previous one, recovered fine details are not pronounced due to the diffusivity tensor. (e) Result using the modified tensor, and without the fBm-based texture model: Contours are better recovered with respect to the standard diffusion, but fine details are still missing. (f) Result using the complete scheme: In this case, new details are recovered via the fBm-based model, and are preserved by the modified diffusion tensor.

after blurring[1].

We present several types of textures from the texture "spectrum" (Figs. 12, 14,16,18 and zoomed Figs. 13, 15,17,19 respectively): isotropic stochastic (first image), near-regular (second image) and anisotropic stochastic (third and fourth images). We find the latter the most challenging. The degraded versions of these images suffer from loss of high-frequency textural details, rather than edge and contour degradation. The results are compared with bicubic interpolation, example-based SR [38] and sparseness-based SR [39]. In all examples, the visual structure of the image, enhanced by the proposed algorithm, resembles that of the original (ground-truth). While the high frequency spectrum is not identical to the original, the visually-appearing texture structure is restored.

We do not rely on PSNR or on other $\mathcal{L}_2$-based comparisons for the assessment of the algorithm's performance. A high PSNR value, in the range of $25dB$ and above, indicates it is also a valid comparison method. This is the case in general images, containing sharp edges, but limited texture data. In the case of fully-textured images, the PSNR values are, however, significantly lower after degradation, and are confined to the range of $5 - 20dB$. While other comparison methods, such as the structural similarity index (SSIM) [40], have been suggested, there do not exist widely-accepted criteria for performance evaluation, other than visual assessment (geometrically-based natural metrics, potentially-suitable for performance evaluation are under investigation) [35].

It is also important to note that in textured images, such as the ones considered here, one needs to look for the highest fidelity in comparison with the original image. This may be in contradiction to desired properties in other images. Sharp edges, for example, are desired in many applications, but in this case, a sharp edge in the SR image may in fact be undesired if the original image edges are not sharp.

The example-based SR results shown here depict the characteristic behaviour of the example-based SR methods; while emphasizing edges, improving on the result of the bicubic interpolation, it does not successfully restore missing fine details. This is apparent in all types of textures presented.

[1]Additional examples can be found at `http://vision.technion.ac.il/demos/texture-sr/pde-based`

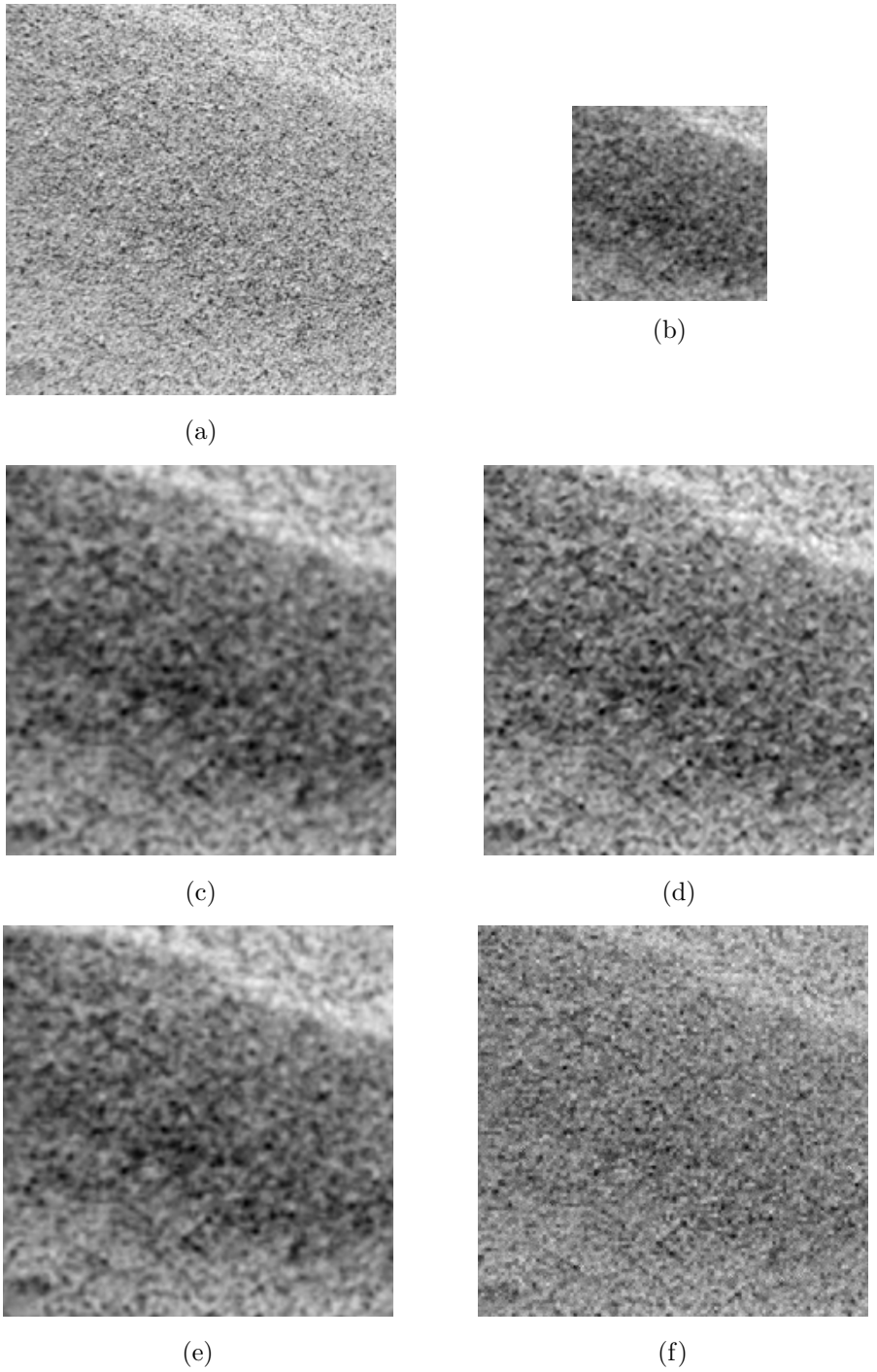


Figure 12: Superresolution of textured images. The degraded images are obtained by $2 \downarrow 1$ subsampling in both axes after a Gaussian blur with $\sigma = 1.5$. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result. While the example-based SR performs edge enhancement relative to the cibubic interpolation result, it fails to recover missing textural details. The proposed algorithm result bears more resemblance to the original image in terms of both contours and textured details.

8 Discussion

The theoretical framework and algorithms presented in this study are concerned with superresolution of fully textured images, wherein the texture incorporates both stochastic and structured elements. The superresolution paradigm considered here is the so-called single-image superresolution, where only one image is available as an input. Considering first the more challenging aspect of the granularity and non-stationarity of structures often encountered in natural textures, a stochastic texture model has been developed, based on fBm. PDE-based regularization has been introduced in order to capture anisotropic texture details and a diffusion-based single-image superresolution scheme was derived.

As is the case in similar underdetermined problems, the emphasis is on side information, inherent in the underlying image model that captures (represents) the essence of natural stochastic texture. The results obtained in our study, encourage the use of global fBm-based model (rather than patch-based) for natural textured images, as a method for reconstruction of degraded textures.

The proposed model and concomitant algorithm are based on the empirical observation that stochastic textures are characterized by the property of self-similarity. An appropriate random process is estimated with reference to the existing low-resolution image. The initial restoration of missing details is based on an arbitrary realization of an fBm image. One may, therefore, expect that the results will be different for different evaluations. However, due to the phase matching and optimization, results for different random seeds yield almost identical results. In our on-going study, we attempt to remove the formal dependency on an initial arbitrary image, and obtain a model which depends on the fBm statistics.

Our observation, depicted in Fig. 1, is that stochastic textures obey a Gaussian distribution. This is in contast to general natural images, where the distribution has been shown to be non-Gaussian and kurtotic. This further emphasizes the validity of the fBm model, which is the only self-similar Gaussian process (in 1D). This observation will be futher substantiated in future studies, looking into a broad class of natural textures.

A comparison with state-of-the-art example-based, single-image, superresolution algorithms highlights the main advantage inherent in the proposed algorithm: It reconstructs high frequency textural details that are otherwise missing, while the example-based algorithms emphasize, for the most part, edges but do not restore other, textural, missing details.

Whereas the fBm has been widely used as a model of image structure, it is in fact most suitable for modelling natural textures, as this study indicates, but it is not congruous with image skeletal structures comprised of edges and contours. Further research is nonetheless called for in an attempt to expand the model to better model anisotropic textures as well, and to minimize thereby the need for regularization. Such a model may yield other enhancement algorithms suitable for a broader class

of stochastic textures.

Despite of the above goal, yet to be accomplished, the proposed PDE-based regularization presents new terms which need to be investigated. The empirical structure function is obtained via an ill-posed scheme, and better solutions for this problem may result in better understanding of textures and yield thereby better enhancement results.

The proposed model has been exploited for solving the SR problem. It can also be used for other image enhancement problems, such as denoising or in-painting. This is a challenge in the case of textures, due to the overlap in the frequency range with that of the noise, and due to the lack of local, small-scale, smoothness. It should be emphasized that existing denoising algorithms usually succeed in restoring edges and smooth segments, but not in the recovery of fine details. Preliminary results show that the fBm, used as a prior in MAP estimation, can effectively act as a regularizer which performs denoising on fBm-based images.

9 Acknowledgements

This research program has been supported by the Technion Ollendorff Minerva Center for Vision and Image Sciences, and by a special grant awarded to Y.Y.Z by the Minerva Foundation. Ido Zachevsky has been an Ollendorff Fellow.

10 Appendices

A Proof of the lemma

Lemma 2. *Let $\Lambda \in \mathbb{R}^{N\times N}$ be a matrix, whose cells are defined as follows:*

$$(\Lambda)_{i,j} = \begin{cases} 2, & i = j \\ 1, & i \neq j, \end{cases} \tag{A.1}$$

then $\Lambda^{-1} \triangleq \Sigma$ is defined as follows:

$$\Sigma = I - \frac{1}{N(N+1)}\mathbf{1}\mathbf{1}^T, \tag{A.2}$$

where $I \in \mathbb{R}^{N\times N}$ is the identity matrix and $\mathbf{1} \in \mathbb{R}^N$ is a column vector, in which every entry is equal to 1.

Proof. Λ is diagonalizable, as a real and symmetric. Let $\lambda_1, \ldots, \lambda_N$ denote the eigenvalues and $\{v_i\}_{i=1}^N$ denote the respective eigenvectors of Λ. Since we have that the sum of every column j, $s(j)$, is the following:

$$s(j) = \sum_{i=1}^{N} (\Lambda)_{i,j} = 2 + (N-1) = N+1, \tag{A.3}$$

we conclude that $\lambda_1 = (N+1)$ and $v_1 = (1,\ldots,1)^T$. It then follows from symmetry that the rest of the eigenvalues are equal, $\lambda_i \triangleq \lambda \forall i \in \{2,\ldots,N\}$. The rest of the eigenvalues calculated as follows:

$$\begin{aligned} \sum_{i=1}^{N} \lambda_i &= trace\,(\Lambda)\,, \\ \lambda_1 + (N-1)\,\lambda &= N \cdot 2, \\ (N-1)\,\lambda &= -\,(N+1) + 2N, \\ \lambda &= 1. \end{aligned} \tag{A.4}$$

The suitable orthonormal eigenvectors can be derived, and form the eigenvalue matrix, V, which satisfies $V^{-1} = V^T$ due to the orthonormality. The matrix, Λ, can therefore be decomposed as $\Lambda = VDV^{-1}$, where D is the eigenvalue diagonal matrix. The inverse, $\Sigma \triangleq \Lambda^{-1}$, is calculated as follows:

$$\Sigma = VD^{-1}V^{-1}. \tag{A.5}$$

Since D^{-1} is a diagonal matrix, it can be decomposed as follows:

$$D^{-1} = I + I_C, \tag{A.6}$$

where $I_C = D^{-1} - I$, and since $N-1$ of the eigenvalues of Λ are equal to 1, this matrix contains a single non-zero entry, $(I_C)_{N,N} = \lambda_N^{-1} - 1 \triangleq \gamma$. The inverse matrix can, therefore, be represented as follows:

$$\begin{aligned} \Sigma &= V\,(I + I_C)\,V^{-1} \\ &= VV^{-1} + VI_CV^{-1} \\ &= I + VI_CV^T, \end{aligned} \tag{A.7}$$

where the second transition follows from the orthogonal eigenvector matrix V. Due to the structure of I_C, the above equation is simplified further:

$$\Sigma = I + V_CV^T, \tag{A.8}$$

where $V_C \triangleq VI_C$ is a matrix which has non-zero entries only in the last column:

$$V_C = VI_C = \begin{pmatrix} | & \cdots & | \\ v_1 & \cdots & v_n \\ | & \cdots & | \end{pmatrix} \begin{pmatrix} 0 & \cdots & 0 & 0 \\ \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \gamma \end{pmatrix}.$$

Recalling that the last eigenvector, v_n, is a constant, we denote $v_n^i = \beta$ and therefore:

$$V_C = \begin{pmatrix} 0 & \cdots & 0 & \gamma\beta \\ \vdots & \ddots & 0 & \gamma\beta \\ 0 & 0 & 0 & \gamma\beta \\ 0 & 0 & 0 & \gamma\beta \end{pmatrix} = \gamma \begin{pmatrix} 0 & \cdots & 0 & | \\ \vdots & \ddots & 0 & v_n \\ 0 & \cdots & 0 & | \end{pmatrix}.$$

Next, multiplied by V^T, we obtain:

$$\begin{aligned} V_C V^T &= \gamma \begin{pmatrix} 0 & \cdots & 0 & | \\ \vdots & \ddots & 0 & v_n \\ 0 & \cdots & 0 & | \end{pmatrix} \begin{pmatrix} - & v_1 & - \\ \vdots & \vdots & \vdots \\ - & v_n & - \end{pmatrix} \\ &= \gamma\beta^2 \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{pmatrix}, \end{aligned} \tag{A.9}$$

and the inverted matrix, Σ, is therefore:

$$\Sigma = I + \gamma\beta^2 \mathbf{1}\mathbf{1}^T, \tag{A.10}$$

where $\mathbf{1} \in \mathbb{R}^{N\times 1}$ is a vector with 1 in every entry. The value of β can be derived from the orthonormality of V. Since v_n is constant, we thus obtain $\beta = \frac{1}{N}$. Substituting β and γ, we obtain the following matrix:

$$\begin{aligned} \Sigma &= I + \left(\frac{1}{N+1} - 1\right) \frac{1}{N^2} \mathbf{1}\mathbf{1}^T, \\ \Sigma &= I - \frac{1}{N\,(N+1)} \mathbf{1}\mathbf{1}^T. \end{aligned} \tag{A.11}$$

□

B The ESF LS problem

The LS problem, Eq. (6.13), can be considered as follows: Let the filters f_d and f_{d2}, and the autocorrelation functions $R_{\eta_1}(\eta_1, \eta_2)$, $R_{\eta_2}(\eta_1, \eta_2)$ and $R_{\eta_1,\eta_2}(\eta_1, \eta_2)$ be as presented in subsection 6.2. The matrix, D_f, in Eq. (6.13) is derived by three vertically-stacked matrices, as follows:

$$D_f = \begin{pmatrix} D_{f,\eta_1} \\ D_{f,\eta_2} \\ D_{f2} \end{pmatrix}, \tag{B.1}$$

where D_{f,η_1} and D_{f,η_2} are the matrix representation of the filters f_d and f_d^T respectively, and D_{f2} is the matrix representation of the filter f_{d2}. The matrices perform convolution with symmetric boundaries.

In a respective manner, the vector r in Eq. (6.13) is formed as follows:

$$r = \begin{pmatrix} r_{\eta_1} \\ r_{\eta_2} \\ r_{\eta_1,\eta_2} \end{pmatrix}, \tag{B.2}$$

where r_{η_1}, r_{η_2} and r_{η_1,η_2} are the column-stacked versions of the autocorrelation matrices $R_{\eta_1}(\eta_1,\eta_2)$, $R_{\eta_2}(\eta_1,\eta_2)$ and $R_{\eta_1,\eta_2}(\eta_1,\eta_2)$. The solution of the LS problem yields ϕ, a column-stacked vector, which is then converted back to matrix form in the size of the image.

C Derivation of the diffusion equation

Let $E(X,\nabla X) = E(X,X')$ be defined as the energy functional presented in Eq. 6.14. The Euler-Lagrange equation for this functional is given by

$$\frac{\partial E(X,X')}{\partial X} - \frac{d}{dt}\frac{\partial E(X,X')}{\partial X'} = 0. \tag{C.1}$$

Due to the time invariance, we obtain:

$$\frac{\partial E(X,X')}{\partial X} = 0. \tag{C.2}$$

Let $E_r(X)$ denote the reaction, and $E_d(X)$ denote the diffusion term in $E(X)$, defined as follows:

$$\begin{aligned} E(X) &= \int_\Omega E_r(X) + E_d(X)dxdy, \\ E_r(X) &= (BX-Y)^2 + (\hat{X}_{HP} - H_{HP}X)^2, \\ E_d(X) &= \beta\Psi(|\nabla X + \alpha\nabla Y_\phi|^2). \end{aligned} \tag{C.3}$$

$\frac{\partial E_r(X)}{\partial X}$ is derived as follows:

$$\begin{aligned} \frac{\partial E_r(X)}{\partial X} &= 2B^T(BX-Y) - 2H_{HP}^T(\hat{X}_{HP} - H_{HP}X) \\ &= 2B^T(BX-Y) - 2H_{HP}^T\hat{X}_{HP} + 2H_{HP}^T H_{HP}X. \end{aligned} \tag{C.4}$$

Since a Gaussian filter is used, $H_{HP}^T = H_{HP}$.

$\frac{\partial E_d(X)}{\partial X}$ is derived as follows:

$$\frac{\partial E_d(X)}{\partial X} = \beta\Psi'(|\nabla X + \alpha\nabla Y_\phi|^2)\cdot f'(X), \tag{C.5}$$

where $f(X) = |\nabla X + \alpha\nabla Y_\phi|^2$, is derived as follows:

$$\begin{aligned} f(X) &= (\nabla X + \alpha\nabla Y_\phi)^T(\nabla X + \alpha\nabla Y_\phi) \\ &= (\nabla X)^T(\nabla X + \alpha\nabla Y_\phi) + \alpha(\nabla Y_\phi)^T(\nabla X + \alpha\nabla Y_\phi) \\ f'(X) &= \nabla\cdot(\nabla X) + \alpha\nabla\cdot\nabla Y_\phi + \alpha\nabla\cdot(\nabla Y_\phi)^T \\ &= \nabla\cdot(\nabla X) + 2\alpha\nabla\cdot(\nabla Y_\phi). \end{aligned} \tag{C.6}$$

Plugging $f'(X)$ back in the previous equation, we obtain:

$$\begin{aligned}\frac{\partial E_d(X)}{\partial X} =&\beta\nabla\cdot(\Psi'(|\nabla X+\alpha\nabla Y_\phi|^2)\nabla X)+\\ &+2\alpha\beta\nabla\cdot(\Psi'(|\nabla X+\alpha\nabla Y_\phi|^2)\nabla Y_\phi).\end{aligned} \tag{C.7}$$

Finally, substituting the derivatives of $E_r(X)$ and $E_d(X)$ we obtain the diffusion-reaction equation in Eq. (6.16).

References

[1] Daniel Glasner, Shai Bagon, and Michal Irani, "Super-resolution from a single image," *2009 IEEE 12th International Conference on Computer Vision*, pp. 349–356, Sept. 2009.

[2] KwangIn Kim and Younghee Kwon, "Example-based learning for single-image super-resolution," in *Pattern Recognition*, Gerhard Rigoll, Ed., vol. 5096 of *Lecture Notes in Computer Science*, pp. 456–465. Springer Berlin Heidelberg, 2008.

[3] L.C. Pickup, S.J. Roberts, and A. Zisserman, "A sampled texture prior for image super-resolution," *Advances in neural information processing systems*, vol. 16, 2003.

[4] Dmitry Datsenko and Michael Elad, "Example-based single document image super-resolution: a global map approach with outlier rejection," *Multidimensional Systems and Signal Processing*, vol. 18, pp. 103–121, 2007.

[5] Jianchao Yang, J. Wright, T.S. Huang, and Yi Ma, "Image super-resolution via sparse representation," *Image Processing, IEEE Transactions on*, vol. 19, no. 11, pp. 2861 –2873, nov. 2010.

[6] Kwang In Kim and Younghee Kwon, "Single-image super-resolution using sparse regression and natural image prior," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 6, pp. 1127 –1133, june 2010.

[7] Yann Gousseau and Jean-Michel Morel, "Are Natural Images of Bounded Variation?," *SIAM Journal on Mathematical Analysis*, vol. 33, no. 3, pp. 634–648, Jan. 2001.

[8] Alfred S Carasso, "Singular integrals, image smoothness, and the recovery of texture in image deblurring," *SIAM Journal on Applied Mathematics*, vol. 64, no. 5, pp. 1749–1774, 2004.

[9] Ori Honigman and Yehoshua Y. Zeevi, "Enhancement of Textured Images Using Complex Diffusion Incorporating Schrodinger's Potential," in *ICASSP*, 2006, pp. 633–636.

[10] Michael Elad and Mario a T Figueiredo, "On the Role of Sparse and Redundant Representations in Image Processing," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, June 2010.

[11] Wen-Chieh Lin, James Hays, Chenyu Wu, Vivek Kwatra, and Yanxi Liu, *A comparison study of four texture synthesis algorithms on regular and near-regular textures*, Number January. Citeseer, 2004.

[12] Alexei A Efros and Thomas K Leung, "Texture synthesis by non-parametric sampling," *Proceedings of the 7th IEEE International Conference on Computer Vision*, vol. 2, pp. 1033–1038, 1999.

[13] William T Freeman, Thouis R Jones, and Egon C Pasztor, "Example-Based Super-Resolution," *Computer Graphics and Applications, IEEE*, vol. 22, no. 2, pp. 56–65, 2002.

[14] Guy Gilboa and Stanley Osher, "Nonlocal operators with applications to image processing," *Multiscale Modeling & Simulation*, vol. 7, no. 3, pp. 1005–1028, 2008.

[15] H.O. Peitgen, D. Saupe, M.F. Barnsley, Y. Fisher, and M. McGuire, *The science of fractal images*, Springer New York etc., 1988.

[16] B.B. Mandelbrot and J.W. Van Ness, "Fractional brownian motions, fractional noises and applications," *SIAM review*, vol. 10, no. 4, pp. 422–437, 1968.

[17] B. Pesquet-Popescu and J.L. Vehel, "Stochastic fractal models for image processing," *Signal Processing Magazine, IEEE*, vol. 19, no. 5, pp. 48 – 62, sep 2002.

[18] J.M. Keller, S. Chen, and R.M. Crownover, "Texture description and segmentation through fractal geometry," *Computer Vision, Graphics, and Image Processing*, vol. 45, no. 2, pp. 150–166, 1989.

[19] Daniel Zoran and Yair Weiss, "From learning models of natural image patches to whole image restoration," *2011 International Conference on Computer Vision*, pp. 479–486, Nov. 2011.

[20] Martin J Wainwright and Eero P Simoncelli, "Scale Mixtures of Gaussians and the Statistics of Natural Images," in *NIPS*, 1999, pp. 855–861.

[21] Javier Portilla, Vasily Strela, Martin J Wainwright, and Eero P Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain.," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 12, no. 11, pp. 1338–51, Jan. 2003.

[22] David J Heeger and James R Bergen, "Pyramid-based texture analysis/synthesis," in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. 1995, vol. 3, pp. 229–238, ACM.

[23] Joachim Weickert, *Anisotropic Diffusion in Image Processing*, Teubner Stuttgart, 1998.

[24] Ido Zachevsky and Yehoshua Y. Zeevi, "Single-image superresolution of self-similar textures," in *IEEE International Conference on Image Processing (accepted)*, 2013.

[25] Michal Irani and Shmuel Peleg, "Super resolution from image sequences," in *Pattern Recognition, 1990. Proceedings., 10th International Conference on*. 1990, vol. 2, pp. 115–120, IEEE Comput. Soc. Press.

[26] Michael Elad and Arie Feuer, "Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images," *Image Processing, IEEE Transactions on*, vol. 6, no. 12, pp. 1646–1658, Jan. 1997.

[27] Jianchao Yang and Thomas Huang, "Image super-resolution: Historical overview and future challenges," *Super-resolution imaging*, 2010.

[28] Aram Danielyan, Vladimir Katkovnik, and Karen Egiazarian, "BM3D frames and variational image deblurring," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1715–1728, Apr. 2012.

[29] Stephan Hoefer, H Hannachi, Madhukar Pandit, and Ramdas Kumaresan, "Isotropic Two-Dimensional Fractional Brownian Motion and its Application in Ultrasonic Analysis," in *Engineering in Medicine and Biology Society, 1992 14th Annual International Conference of the IEEE*, 1992, pp. 1267–1269.

[30] Lance M Kaplan and C-CJ Kuo, "An improved method for 2-D self-similar image synthesis.," *Image Processing, IEEE Transactions on*, vol. 5, no. 5, pp. 754–761, Jan. 1996.

[31] Béatrice Pesquet-Popescu and Pascal Larzabal, "Synthesis of nonstationary fields with stationary increments," *Image Processing and Its Applications*, vol. 1, pp. 303–307, 1997.

[32] Alan v. Oppenheim and Jae S. Lim, "Importance of Phase in Signals," *Proceedings of the IEEE*, vol. 69, no. 5, pp. 529–541, 1981.

[33] Jacques Behar, Moshe Porat, and Yehoshua Y Zeevi, "Image Reconstruction from Localized Phase," *IEEE Transactions on Signal Processing*, vol. 40, no. 4, pp. 736–743, 1992.

[34] Nikolay Skarbnik, Chen Sagiv, and Yehoshua Y Zeevi, "Edge detection and skeletonization using quantized localized phase," in *European Signal Processing Conference*, 2009, number Eusipco, pp. 1542–1546.

[35] Zhou Wang and Alan C Bovik, "Mean squared error: Love It or Leave It?," *IEEE Signal Processing Magazine*, , no. January, pp. 98–117, 2009.

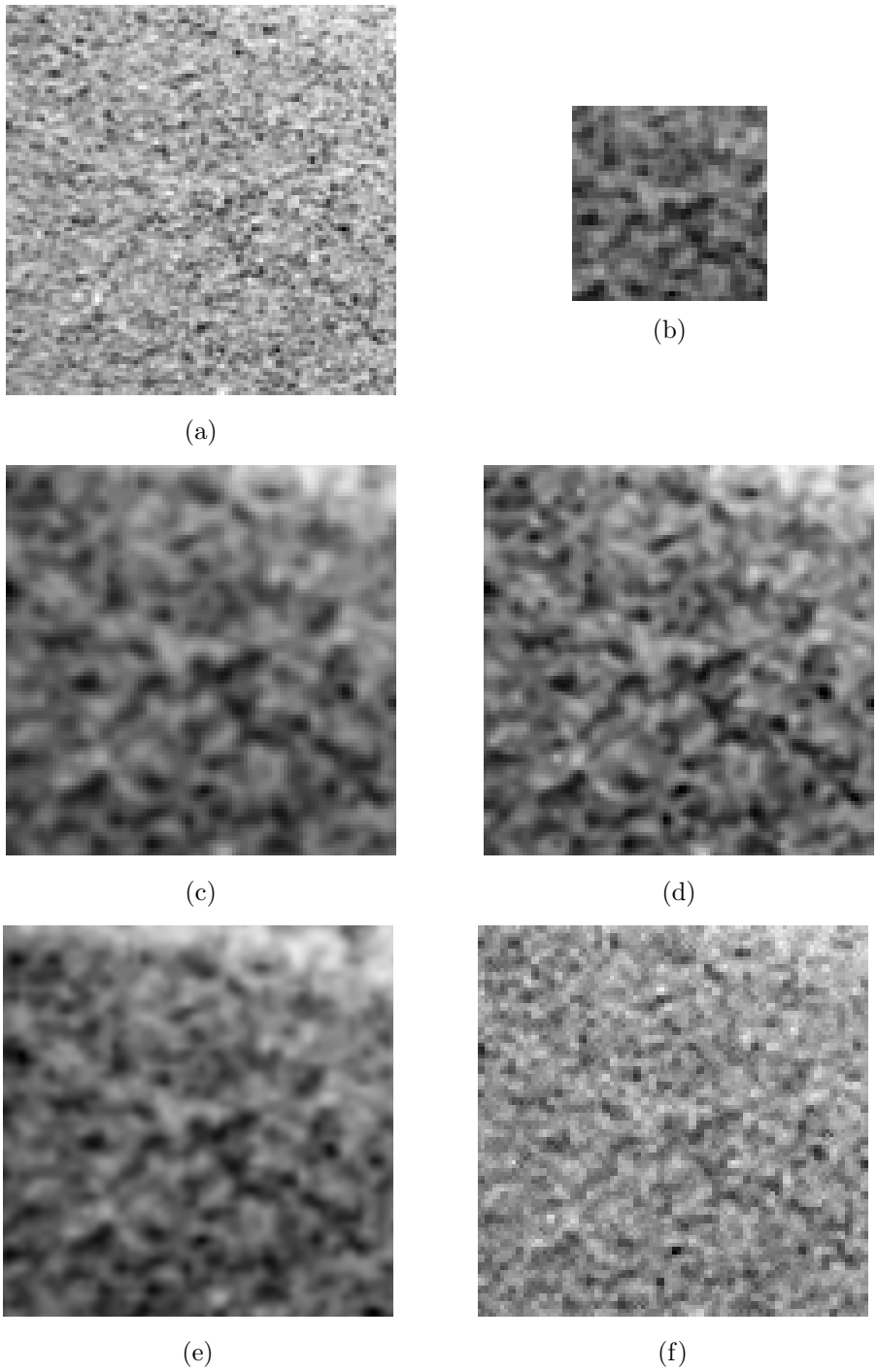
[36] Guy Gilboa, Nir Sochen, and Yehoshua Y Zeevi, "Forward-and-backward diffusion processes for adaptive image enhancement and denoising.," *IEEE Transactions on Image Processing*, vol. 11, no. 7, pp. 689–703, Jan. 2002.

[37] Pietro Perona and Jitendra Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, July 1990.

[38] Kwang In Kim and Younghee Kwon, "Example-Based Learning for Single-Image Super-Resolution," in *Pattern Recognition*, pp. 456–465. Springer, 2008.

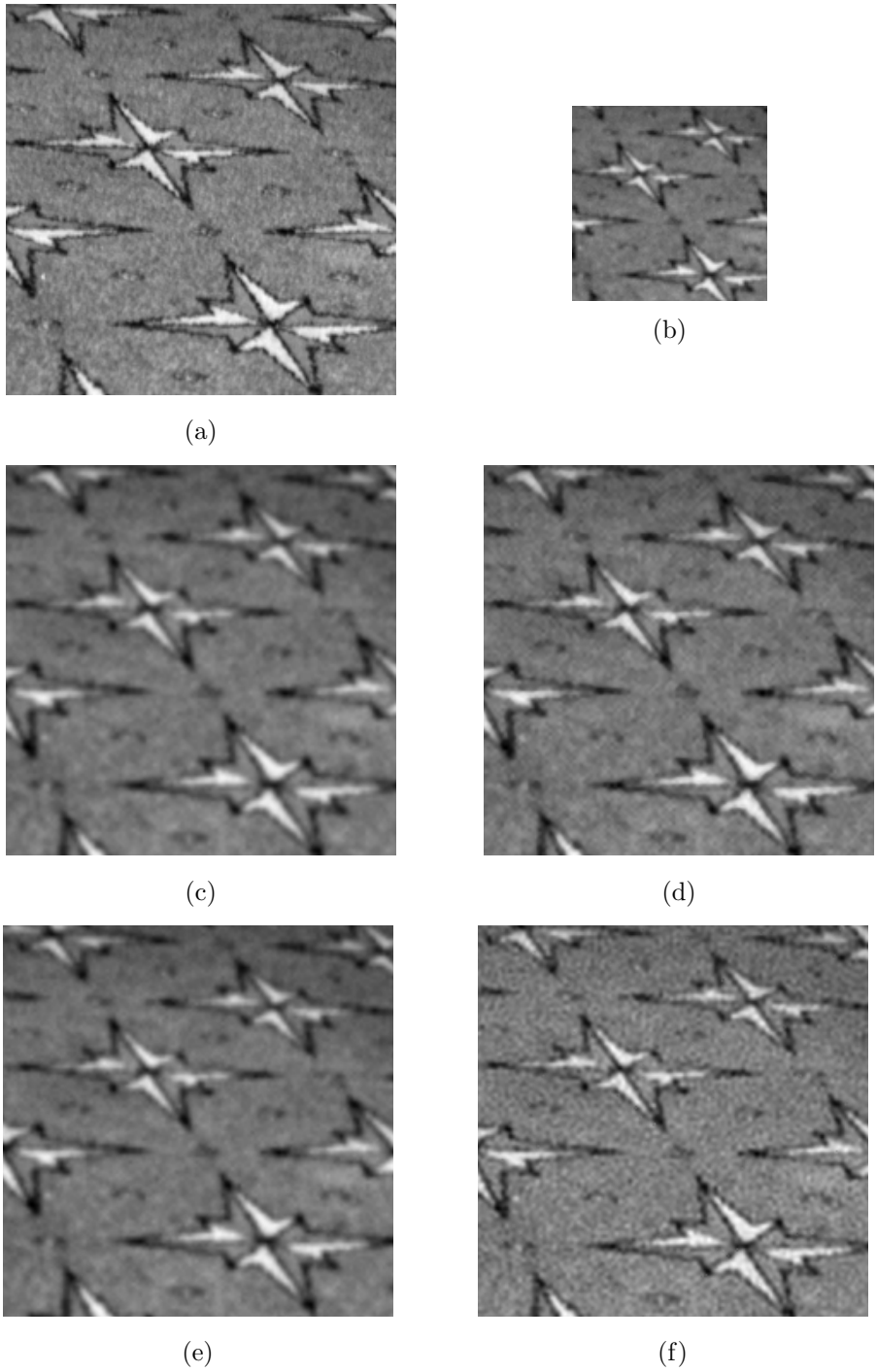
[39] Jianchao Yang, John Wright, Thomas Huang, and Yi Ma, "Image Super-Resolution via Sparse Representation.," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 19, no. 11, pp. 2861–2873, May 2010.

[40] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, Student Member, Eero P Simoncelli, and Senior Member, "Image Quality Assessment : From Error Visibility to Structural Similarity," *IEEE transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.



(a) (b) (c) (d) (e) (f)

Figure 13: Zoomed version of Fig. 12. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result.



(a) (b) (c) (d) (e) (f)

Figure 14: Superresolution of textured images. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result. For more details see Fig. 16

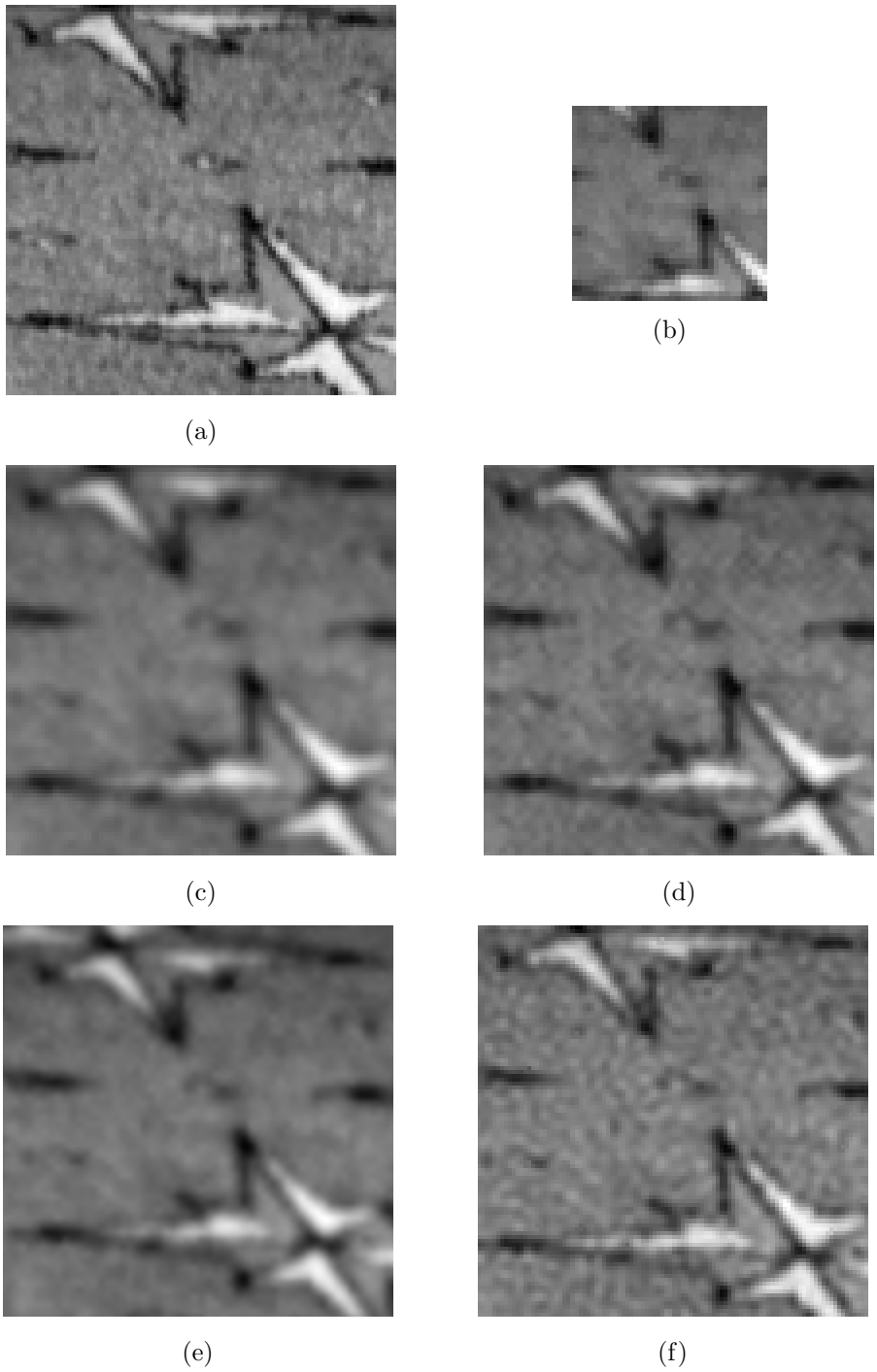
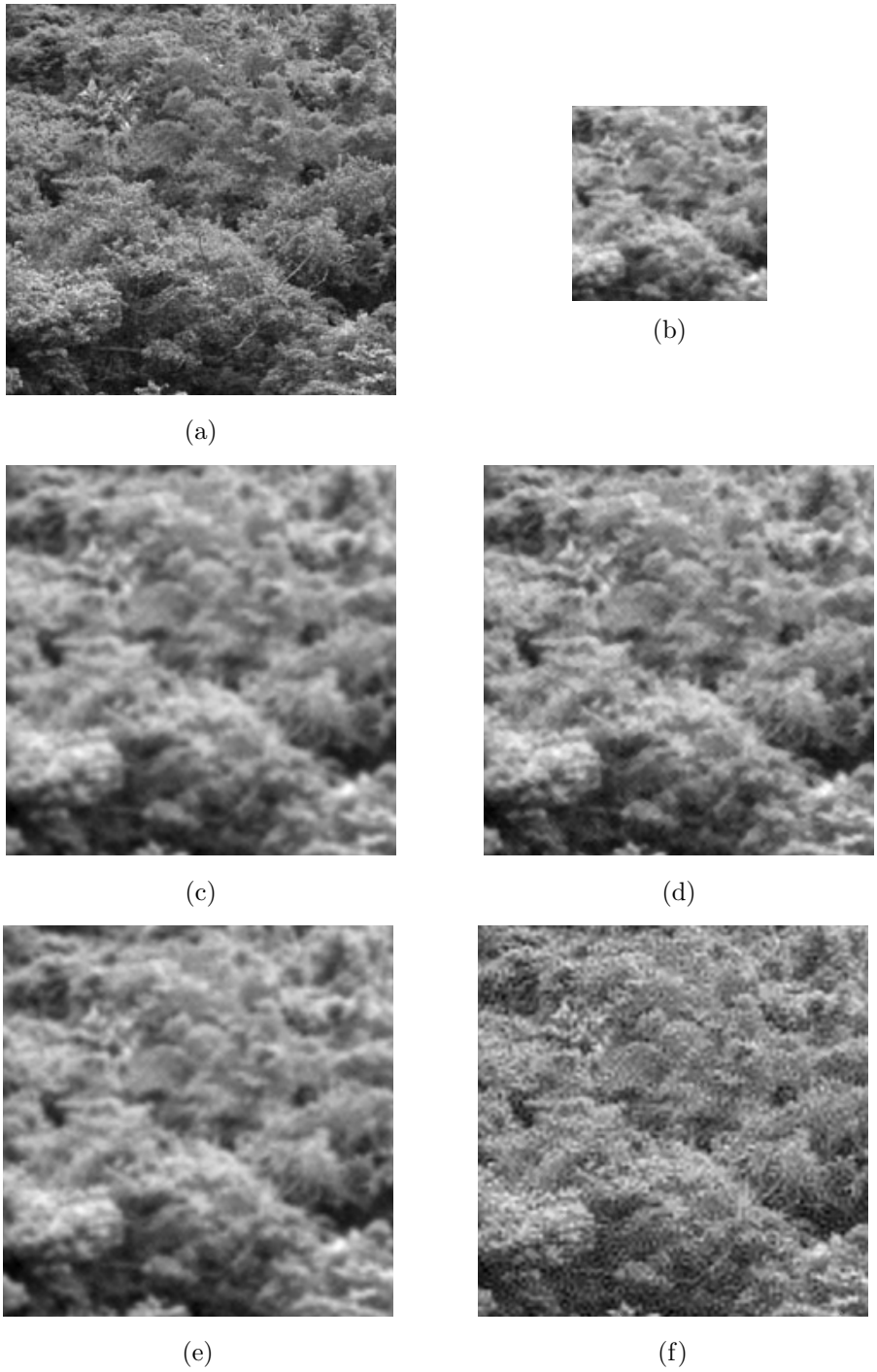


Figure 15: Zoomed version of Fig. 14. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result.



(a) (b) (c) (d) (e) (f)

Figure 16: Superresolution of textured images. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result. For more details see Fig. 16

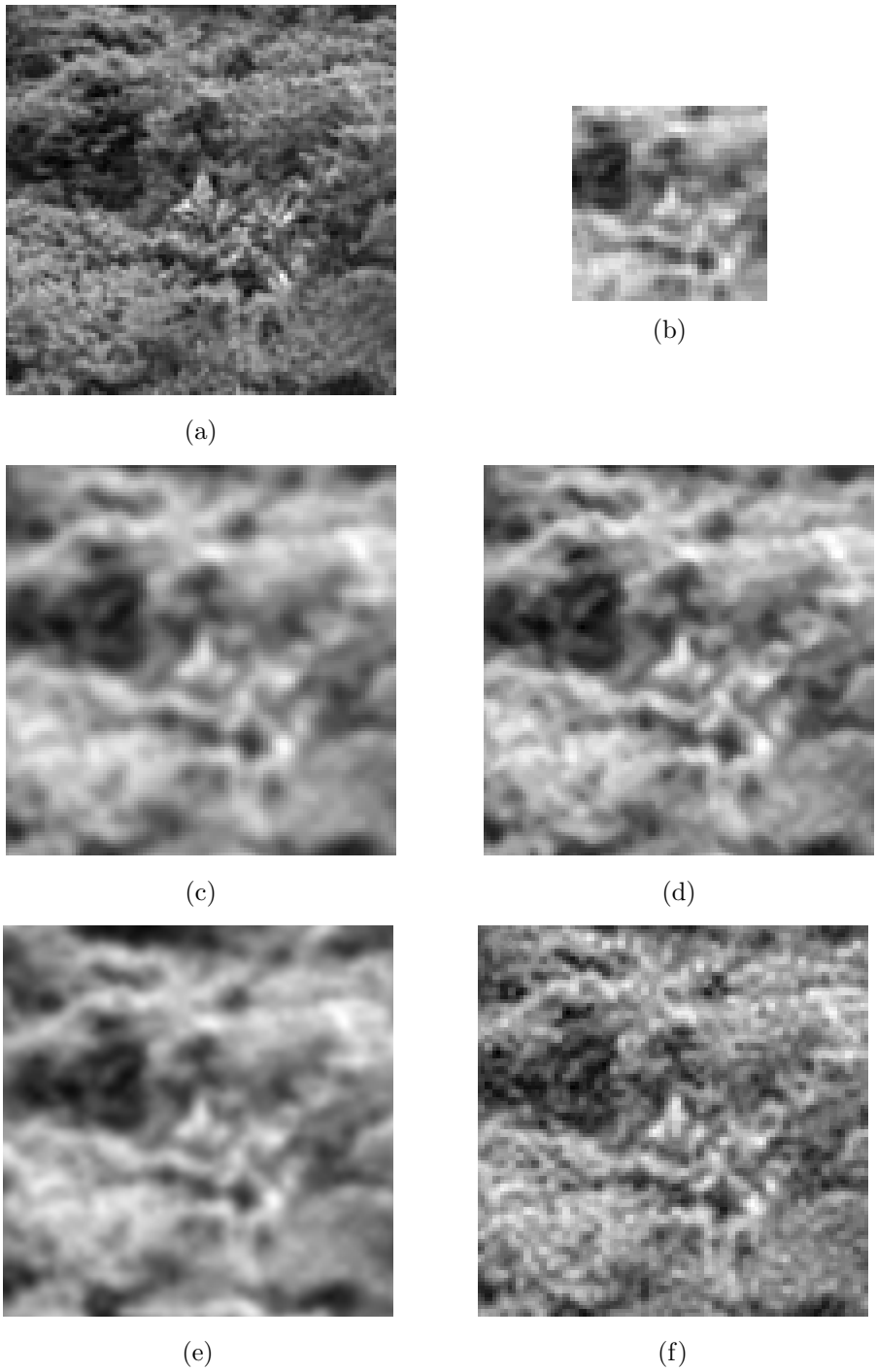


Figure 17: Zoomed version of Fig. 16. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result.

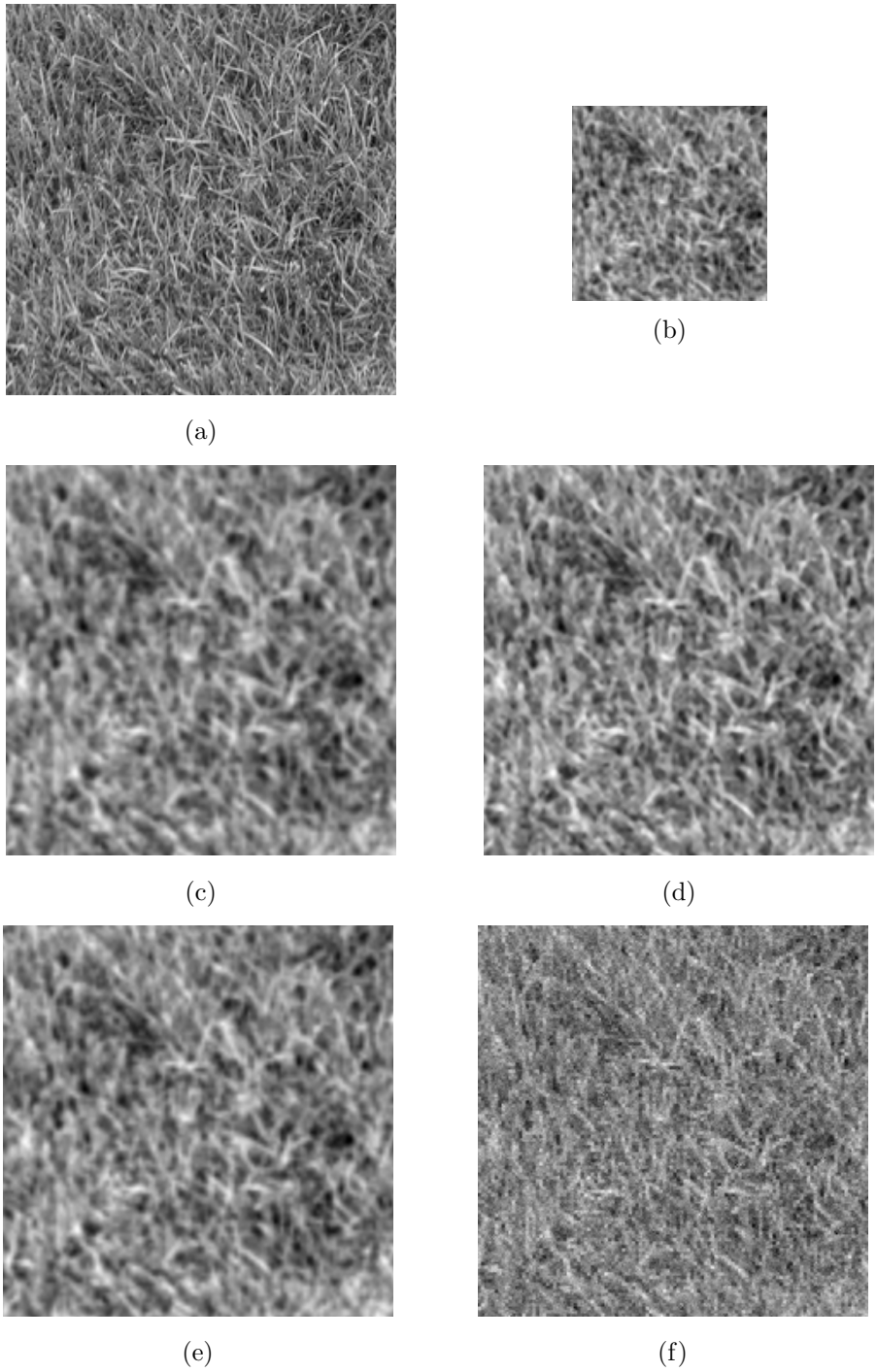


Figure 18: Superresolution of textured images. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result. For more details see Fig. 16

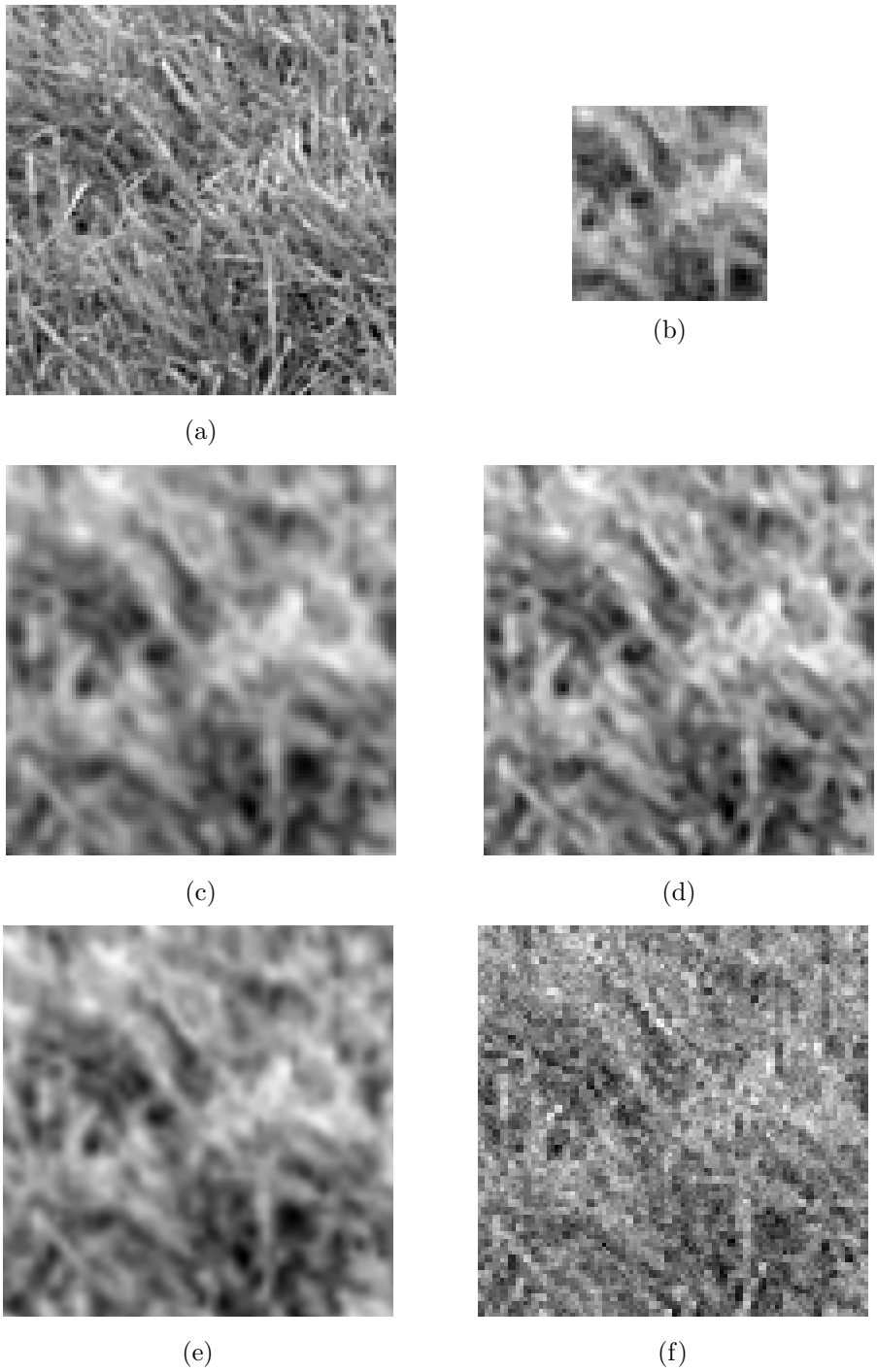


Figure 19: Zoomed version of Fig. 18. (a) Original (ground-truth) images. (b) Low-resolution image. (c) Bicubic interpolation. (d) Sparseness-based SR result [39]. (e) Example-based SR result [38]. (f) Proposed algorithm SR result.