The Computer Journal Advance Access published April 20, 2007
© The Author 2007. Published by Oxford University Press on behalf of The British Computer Society. All rights reserved.
For Permissions, please email: journals.permissions@oxfordjournals.org
doi:10.1093/comjnl/bxm008

Example-Based Regularization Deployed to Super-Resolution Reconstruction of a Single Image

MICHAEL ELAD* AND DMITRY DATSENKO
Department of Computer Science, The Technion—Israel Institute of Technology, Haifa 32000, Israel
**Corresponding author: elad@cs.technion.ac.il*

In super-resolution (SR) reconstruction of images, regularization becomes crucial when insufficient number of measured low-resolution images is supplied. Beyond making the problem algebraically well posed, a properly chosen regularization can direct the solution toward a better quality outcome. Even the extreme case—a SR reconstruction from a single measured image—can be made successful with a well-chosen regularization. Much of the progress made in the past two decades on inverse problems in image processing can be attributed to the advances in forming or choosing the way to practice the regularization. A Bayesian point of view interpret this as a way of including the prior distribution of images, which sheds some light on the complications involved. This paper reviews an emerging powerful family of regularization techniques that is drawing attention in recent years—the *example-based* approach. We describe how examples can and have been used effectively for regularization of inverse problems, reviewing the main contributions along these lines in the literature, and organizing this information into major trends and directions. A description of the state-of-the-art in this field, along with supporting simulation results on the image scale-up problem are given. This paper concludes with an outline of the outstanding challenges this field faces today.

Keywords: Regularization/example-based/nearest neighbor/Bayesian reconstruction/MMSE/MAP/PCA/clustering/K-D tree/scale-up/super-resolution

Received 11 May 2006; revised 28 October 2006

1. INTRODUCTION

The conventional super-resolution (SR) process uses a multitude of measured low-quality images to produce the super-resolved outcome. It is well known that such SR process may lead to higher optical (i.e. true) resolution. The higher-frequencies in the resulting image, which represent the newly-introduced details, are in fact available in the measurements in an aliased form. The SR process recovers these high frequencies by exploiting the various given images, each exhibiting a different aliasing effect. This explains why such resolution improvement is possible in the first place. However, for such a process to succeed, sufficient number of low-resolution images are needed, so as to enable the recovery of the aliased frequencies uniquely [1–3].

Based on the above reasoning, one might be led to the natural conclusion that SR based on a single measured image is impossible. Is it indeed so? The answer depends on the available information the reconstruction process has access to. Clearly, one type of information that is made available to the reconstruction process is the measured image(s). Those alone could suffice if enough of them are available, as described above. If only one image is given, an alternative source of information is necessary, so as to compensate for the lack of data. An *a priori* knowledge about the objects in the image could be proposed as such source of information. This leads naturally to the concept of regularization [4, 5].

Regularization plays a vital role in inverse problems, and especially in ill-posed ones, where insufficient data are available. One way to interpret the regularization is a way of gaining an algebraic stability in the reconstruction process. However, regularization is much more than a mere stabilization technique. A Bayesian point of view interprets such addition to inverse problems as a way of exploiting the probability density function (PDF) of images—the prior. This way, a properly chosen regularization can direct the solution toward a better quality outcome, by bringing into account the proper behavior of the desired image. Indeed, in the extreme case, SR from a single measured image—the image scale-up

problem—can be made possible and successful due to such well-chosen prior.

Much of the progress made in the past two decades on inverse problems in image processing can be attributed to the advances in forming or choosing the way to practice the regularization. The simplest regularization practiced was based on Tikhonov's idea, enforcing spatial smoothness uniformly on the output image [6]. This option leads to the well-known Wiener filter for image restoration, and is known to over-smooth image edges [4, 5]. Introduction of spatially adaptive smoothness priors was shown to lead to better results, leaning first on a weighted least-squares scheme, and later on robust statistics techniques [7]. In fact, much of the activity that brought partial differential equations (PDE) to the realm of image processing has to do with ways of defining edge-preserving regularization terms [8]. In parallel to those techniques, sparsity of transform coefficients (e.g. wavelet) has also been used as a way of forming regularization in inverse problems [9].

Common to all the above regularization methods is the use of closed-form simplistic mathematical expressions in defining the PDF of images. One must ask: can the wealth of image content be grasped by such simple expressions? Judging by the quality of results obtained in challenging inverse problems (e.g. deblurring and SR) that employ these regularization methods, the answer is unfortunately negative. While such methods perform much better than previously practiced reconstruction algorithms, the quality of the results is typically far from being satisfactory. Realizing this, in recent years, there has been a trend of seeking better and more complex priors of various sorts.

One fascinating and promising such direction is the use of examples, basically suggesting that instead of arbitrarily and intuitively defining the PDF, let image examples help in defining it. This paper focuses on this *example-based* approach, describing how examples can and have been used effectively for regularization, reviewing the main contributions along these lines in the literature. As it turns out, there are three main effective ways to exploit examples in inverse problems—use them to fine-tune the parameters of previously defined regularization expressions [10–17], use them directly for the reconstruction procedure [18–23] or fuse the above two techniques somehow [20, 21, 24, 25]. These options are presented in detail in this paper.

The use of examples becomes much more effective when handling narrow family of images, such as scanned documents or face images. Beyond the offered review on the use of examples in inverse problems, this paper also presents a description of our recent efforts in developing effective algorithms for image scale-up, focusing on the above two families of images. We show how effective pruning of examples can lead to better results, both visually and in mean squared error (MSE). Along side to the description of the state-of-the-art in this arena, we outline the outstanding challenges of this fiefld.

This paper is organized as follows: Section 2 gives the necessary background, describing the maximum-likelihood estimator (MLE) that solves inverse problems based on measurements alone, the Bayesian approach that introduces the image prior and the evolution of regularization expressions in the past decades. In Section 3, we describe how examples can be used, surveying the various contributions in the literature along these lines and putting some order to these techniques, based on their rationale. In Section 4, we describe our recent work on scanned documents and face images, concentrating on a holistic way of defining the prior based on examples and how those examples can be pruned to fine-tune the results. This section includes also supporting simulation results. Section 5 concludes this paper with an attempt to clearly define the grand challenges this field faces today.

2. BACKGROUND ON REGULARIZATION

2.1. The maximum-likelihood estimator (MLE)

A fundamental signal processing problem is the recovery of a signal x ϵ $\mathbb{R}^N$ from a measurement vector, $\boldsymbol{y} \epsilon \mathbb{R}^M$, related to it through

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{v}. \tag{1}$$

In this equation, the matrix $\boldsymbol{H} \epsilon \mathbb{R}^{M \times N}$ represents some linear degradation operation (it may include blur, decimation, geometrical warp and more), and $\boldsymbol{v} \epsilon \mathbb{R}^M$ stands for an additive noise, assumed to be a zero-mean and white (with a standard deviation σ) Gaussian random vector with probability

$$\boldsymbol{p}(\boldsymbol{v}) = \frac{1}{(2\pi)^{M/2}\sigma^M} \cdot \exp\left\{-\frac{\boldsymbol{v}^{\mathrm{T}}\boldsymbol{v}}{2\sigma^2}\right\}. \tag{2}$$

The discussion brought in this and the next section on how the above-described problem is addressed belongs now to the classics of signal and image processing. For more information, the reader is referred to [4, 5].

The MLE suggests to choose $\boldsymbol{x}$ that leads $\boldsymbol{p}(\boldsymbol{y}|\boldsymbol{x})$, known as the likelihood function, to maximum. This means that we choose the signal that makes the measurements the most likely to take place, and thus the name of this method. Clearly, such method exploits the measurements alone in forming the estimated result.

Considering the model in Equation (1), based on the Gaussianity of the noise and the fact that $\boldsymbol{x}$ is assumed to be known, the measurement vector is also a Gaussian random vector with a shifted mean. Thus, the likelihood function becomes

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{(2\pi)^{M/2}\sigma^M} \cdot \exp\left\{-\frac{1}{2\sigma^2} \cdot \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2\right\}. \tag{3}$$

Therefore, the MLE result is given by

$$\hat{\boldsymbol{x}}_{\mathrm{ML}} = \mathrm{Arg}\max_{x} \boldsymbol{p}(\boldsymbol{y}|\boldsymbol{x}) = \mathrm{Arg}\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2. \quad (4)$$

In cases where the Gram matrix $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}$ is positive definite, the problem is considered well posed, and there is a unique solution to the above minimization, being

$$\hat{\boldsymbol{x}}_{\mathrm{ML}} = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{y}. \quad (5)$$

In cases where $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}$ is singular, there are infinitely many possible solutions, caused by the null-space of the matrix $\boldsymbol{H}$. In such a case, the problem is considered ill-posed, and more information is necessary to tune the reconstruction toward a unique solution. This leads naturally to the notion of regularization. From a pure algebraic point of view, regularization of the MLE is done by turning the penalty function into a strictly convex one, thus guaranteeing a unique solution. A simple way of achieving this goal is via Tikhonov's approach,

$$\begin{aligned}\hat{\boldsymbol{x}}_{\mathrm{RML}} &= \mathrm{Arg}\min_{\boldsymbol{x}}\left\{\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{S}\boldsymbol{x}\|_2^2\right\} \\ &= (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H} + \lambda\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{y}, \end{aligned} \quad (6)$$

where $\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S}$ is assumed to be positive definite. The new solution corresponds to a regularized ML (RML) approach. Notice that an arbitrary quadratic term $\|\boldsymbol{S}\boldsymbol{X}\|_2^2$ has been added here, and while it removes the ill-posedness of the original problem, it is unclear at all whether it helps in getting a proper result.

The above discussion might lead to the wrong impression that regularization is necessary only if the problem is ill-posed. Considering the signal denoising problem, where $\boldsymbol{H} = \boldsymbol{I}$, the matrix $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}$ is positive definite and thus, the problem is well-posed. Nevertheless, the MLE result due to Equation (5) is $\hat{\boldsymbol{x}}_{\mathrm{ML}} = \boldsymbol{y}$, which unveils the weakness of the MLE.

2.2. The Bayesian point of view and regularization

The Bayesian approach starts with the replacement of the likelihood function with the posterior probability $\boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y})$. With this seemingly minor change comes a revolutionary perception of the problem at hand, because now $\boldsymbol{x}$ is assumed to be random as well. The Bayes rule ties the above two conditional probabilities by

$$\boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y}) = \frac{\boldsymbol{p}(\boldsymbol{y}|\boldsymbol{x})\boldsymbol{p}(\boldsymbol{x})}{\boldsymbol{p}(\boldsymbol{y})}. \quad (7)$$

Generally speaking, there are two ways to practice the Bayesian approach and lead to a constructive point[1] estimate of $\boldsymbol{x}$—the maximum *a posteriori* probability (MAP) and the minimum mean-squared error (MMSE) methods. The simpler method is the MAP method, choosing the $\boldsymbol{x}$ that maximizes $\boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y})$. Using Equation (7), this reads

$$\hat{\boldsymbol{x}}_{\mathrm{MAP}} = \mathrm{Arg}\max_{\boldsymbol{x}} \boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y}) = \mathrm{Arg}\max_{\boldsymbol{x}} \boldsymbol{p}(\boldsymbol{y}|\boldsymbol{x})\boldsymbol{p}(\boldsymbol{x}). \quad (8)$$

Observe that the denominator $\boldsymbol{p}(\boldsymbol{y})$ has been removed from consideration, since it is considered as constant with respect to the optimization task. For reasons to be made clear shortly, a convenient way to describe the PDF of $\boldsymbol{x}$ is the Gibbs distribution, which represents $\boldsymbol{p}(\boldsymbol{x})$ in an exponential form

$$\boldsymbol{p}(\boldsymbol{x}) = \mathrm{Const} \cdot \exp\{-\alpha A(\boldsymbol{x})\}. \quad (9)$$

Such a description loses no generality, as every non-negative function can be written in such a format. The constant in front of the exponential is a normalization factor, guaranteeing that the integral over all $\boldsymbol{x}$ is 1. The term $A(\boldsymbol{x})$ is a non-negative energy function, supposed to be low for highly probable signals and high otherwise. Using this, and the expression we already have for the likelihood function in Equation (3), we obtain

$$\begin{aligned}\hat{\boldsymbol{x}}_{\mathrm{MAP}} &= \mathrm{Arg}\max_{\boldsymbol{x}} \boldsymbol{p}(\boldsymbol{y}|\boldsymbol{x})\boldsymbol{p}(\boldsymbol{x}) \\ &= \mathrm{Arg}\min_{\boldsymbol{x}}\{\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2 + 2\sigma^2\alpha \cdot \boldsymbol{A}(\boldsymbol{x})\}. \end{aligned} \quad (10)$$

We see that the MAP method leads naturally to the concept of regularization as already described in Equation (6), only this time giving a probabilistic meaning to the additional expression $A(x)$, rather than settling with the gained algebraic stability.

A second, more involved, way to practice the Bayesian approach is the MMSE estimator. This option chooses the expected value of $\boldsymbol{x}$ based on its conditional density $\boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y})$, i.e.

$$\hat{\boldsymbol{x}}_{\mathrm{MMSE}} = E\{\boldsymbol{x}|\boldsymbol{y}\} = \int_{x1}\int_{x2}\cdots\int_{x_N} \boldsymbol{x} \cdot \boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y})\mathrm{d}\boldsymbol{x}. \quad (11)$$

It is easily seen that this solution leads to the minimizer of the expression

$$E\{\|\boldsymbol{x} - \boldsymbol{x}\|_2^2|\boldsymbol{y}\} = \int_{x1}\int_{x2}\cdots\int_{x_N} \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2^2 \cdot \boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y})\mathrm{d}\boldsymbol{x}, \quad (12)$$

which is the MSE measure. This explains the name chosen for this estimator. Since the integral is n-dimensional (as the dimension of $\boldsymbol{x}$), such an approach is typically prohibitive for non-scalar cases.

Whichever method chosen, MMSE or MAP, the estimation of $\boldsymbol{x}$ using the Bayesian approach requires a clear definition of the energy function $\boldsymbol{A}(\boldsymbol{x})$. When dealing with images, this energy function is essentially describing how natural images behave. In the next section, we describe the main choices

[1]The methods we describe here provide a point estimate, as opposed to techniques that estimate the entire posterior distribution or sample from it, such as Markov Chain Monte–Carlo methods [26].

made for $A(\boldsymbol{x})$ in the past 2–3 decades, showing the evolution of ideas on this matter.

2.3. Evolution of image priors

Assuming the Gibbs distribution for images, as in Equation (9), what should $A(\boldsymbol{x})$ be so as to reflect the distribution of natural images? This question poses one of the most fundamental problems in image processing. This enigma has drawn a considerable research attention in the past 2–3 decades, and is still considered an open question. In this section, we briefly describe the main milestones in this arena, showing how priors are getting smarter and more complex, all in the attempt to better describe image content.

The Tikhonov regularization presented earlier was among the first to be practiced [35]. Choosing $A(\boldsymbol{x}) = \|\boldsymbol{L}\boldsymbol{x}\|_2^2$ with $\boldsymbol{L}$ chosen as the Laplacian operator, this prior promotes spatial smoothness across the image in a uniform way. The uniformity is a key feature of this choice, as it leads to numerical convenience in the classic deblurring problem, where $\boldsymbol{H}$ is a linear space-invariant blurring operation. In such a case, the matrix inversion required in Equation (6) can be easily done in the frequency domain, since $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H} + \lambda\boldsymbol{L}^{\mathrm{T}}\boldsymbol{L}$ is block-circulant (or could be approximated as such, after proper boundary treatment). This becomes the well-known Wiener filter algorithm, which for many years was the leading approach in image deblurring [4, 5].

The choice of the Laplacian operation for measuring smoothness, both here and in later priors proposed, is not the only possibility, and similar regularization expressions can be practiced with other derivatives. For example, the choice $A(\boldsymbol{x}) = \|\boldsymbol{D}_{\mathrm{h}}\boldsymbol{x}\|_2^2 + \|\boldsymbol{D}_{\mathrm{v}}\boldsymbol{x}\|_2^2$, with $\boldsymbol{D}_{\mathrm{h}}$ and $\boldsymbol{D}_{\mathrm{v}}$ being horizontal and vertical derivatives, respectively, can also be used. The difference is in the kind of smoothness that is expected from the outcome—while first-order derivatives promote constant values, second derivatives such as the Laplacian allow tilted planes (and saddle points) as well.

By the late 1980s and early 1990s, it became clear that the Wiener filter is not producing good enough results, and better ones are within reach when avoiding the enforced spatial uniformity. This basic idea of forcing smoothness adaptively across the image found many manifestations in various proposed image priors. One of the simplest ways is the weighted least-squares expression—$A(\boldsymbol{x}) = (\boldsymbol{L}\boldsymbol{x})^{\mathrm{T}}\boldsymbol{W}(\boldsymbol{L}\boldsymbol{x})$. The matrix $\boldsymbol{W}$ is a diagonal one with positive entries along the main diagonal being 1 for smooth regions, and close to 0 for edge or texture zones. This matrix can be built based directly on the measurements $\boldsymbol{y}$, assumed to contain enough information to yield such a segmentation. One positive feature of this choice is the fact that the MAP estimator remains linear, although frequency domain solutions are no longer possible. Thus, iterative restoration techniques came to be prevalent and unavoidable [5].

Since the reconstruction process is iterative, one could update the weight matrix $\boldsymbol{W}$ based on the current solution (assumed to be better than the measurements), and this way direct the solution toward a better result. As it turns out, this option is effectively obtained when exploiting concepts from robust-statistics. The field of robust statistics focuses on estimation in the presence of outliers. In the regularization expressions we have seen above, edges appear as outliers in the associated prior. While most of the regions in an image provide low energy after the Laplacian operation (due to their smoothness), edges cause very high and exceptionally different values. Penalizing those using the ℓ^2-norm leads to a very strong penalty, which is avoided by smoothing out the edges, as indeed happens. The alternative is to use robust measures such as $\ell 1$-norm [9], the Huber–Markov or the Cauchy functions, etc. [5]. Then the choice of $A(\boldsymbol{x})$ becomes $A(\boldsymbol{x}) = \boldsymbol{1}^{\mathrm{T}}\ \rho(\boldsymbol{L}\boldsymbol{x})$, where $\rho(\cdot)$ is a scalar robust (having a sub-linear derivative) function. When operating on a vector it is applied entry-wise, and thus, the above is simply the sum over all the entries. Clearly, with this choice of prior, the overall reconstruction algorithm becomes non-linear. If $A(\boldsymbol{x})$ is convex then a unique solution is guaranteed, and can be found via an iterative procedure. This leads to systematic ways of designing non-linear filtering techniques, as indeed required in images due to their non-homogeneity [7].

A vast amount of activity in image processing, which seems to be independent of the above discussion, is the introduction of PDEs filtering techniques into image processing. As it so happens, contributions such as the total variation (TV) by Rudin *et al.* [27], the Beltrami flow due to Sochen *et al.* [28, 29], the directional filter due to Weickert [8] and many more, are directly coupled with the robust-statistics techniques, although formulated in the continuum. For example, the TV suggests the energy function $A(\boldsymbol{x}) = \||\nabla\boldsymbol{x}|\|_1$, which clearly uses a derivative and a robust integration measure. To this date, the TV and its variants are considered among the best regularization techniques available, and are often used in image processing.

In parallel to the impressive progress made on the use of PDEs in image processing in defining regularization expressions, the field of approximation theory contributed its own techniques for this purpose, and in particular via the use of the wavelet transform. Empirical observations suggested that after a wavelet transform, the coefficients of signals tend to sparsity, i.e. many of them are zero or near zero. This led to a proposed regularization expression of the form $A(\boldsymbol{x}) = \|\boldsymbol{T}\boldsymbol{x}\|_p^p$, where $\boldsymbol{T}$ the wavelet transform operator in matrix form, and the ℓ^p-norm (with $p \leq 1$) comes to sum over these coefficients in a way that promotes sparsity [9, 30].

While substantially different from the previous options discussed, this regularization also applies some sort of derivatives, followed by a robust measure. The wavelet transform performs an inner product of the signal with zero-mean vectors that can be interpreted as multi-scale derivatives.

While the TV and the-like use a fixed scale and shift-invariant derivative (i.e. a derivative that applies uniformly across the signal), the wavelet option suggests a multi-scale set of derivatives, but without the shift-invariance property. More recent works deploy redundant transforms, leading to longer sequence of coefficients, so as to obtain both scale and shift invariance. Such constructions are considered among the best available methods today.

In summary, the quest for better regularization expressions for images is very much active today, with many new contributions that extend the above list of options and improve on them. Using the above rationale in forming regularization, one must question the fundamental ability of a simple analytical expression $A(\boldsymbol{x})$ to grasp the complexity and wealth of general image content. This brings us to the main part of this paper, presenting a new way of forming the regularization, based on image examples.

3. USING EXAMPLES—SURVEY OF TECHNIQUES

An emerging powerful regularization methodology that has been drawing research attention in recent years is the use of examples. Rather than guessing the image PDF and forcing a simple expression to be used to describe it, we let image examples guide us in the construction of the prior. Examples can be used in a variety of ways, and the various proposed methods can be roughly divided into three categories:

(1) *Learning prior parameters*: If we are generally pleased with the above-described analytical priors, those can be further improved by learning their parameters.
(2) *Learning the posterior directly*: Rather than learn the image prior and then plug it in a MAP/MMSE reconstruction penalty term, one can use the examples to directly learn the posterior PDF, and then use it for the reconstruction.
(3) *Building a regularization expression with examples*: This is a fusion of the above two techniques, where examples are found as part of the reconstruction process, and then plugged directly into an explicit regularization expression.

In the following sections, we expand on each of those families of methods, and describe related work found in the literature.

3.1. Learning prior parameters via examples

Considering the vast progress made on the formation of regularization expressions, as described above, the most natural way to introduce examples into inverse problems is to keep the use of those expressions, and exploit examples to tune some parameters that control these priors. Thus, the regularization expression is $A(\boldsymbol{x}, \theta)$ where θ are the parameters to be found.

A pioneering work by Zhu and Mumford [11] considered this approach, where a Markov random field prior is trained from the examples. The energy function considered is

$$A(\boldsymbol{x}) = \sum_{i=1}^{n} \lambda_i \rho_i(\boldsymbol{L}_i \boldsymbol{x}). \tag{13}$$

This function leans on a weighted average of robust measures of smoothness, using different robust functions $\rho_i(\cdot)$, analysing filters $\boldsymbol{L}_i$, and weights λ_i. All these can be learned in principle using a large body of high-quality image examples, $\{\boldsymbol{x}_k\}_{k=1}^{K}$.

There can be many ways to tune the prior parameters, each considering a different objective. The work reported in [11] suggests to learn the parameters such that the marginals of the prior fit empirical observations, while maximizing the entropy of the PDF, so as to consider a worst case scenario [39]. A different method of similar flavor has been proposed recently by Roth and Black [14], addressing the same energy function. Their approach, termed *fields of experts*, aims to minimize the Kulback–Leibler distance between the empirical distribution of the example set and the prior trained.

Still using a database of high-quality images, the work by Buccigrossi and Simoncelli [31] propose a prior learning for natural images, based on the statistics of such images in the wavelet domain. While the classic use of wavelets promotes sparsity and assumes an independence between the coefficients, their work considered learning of the joint probability of neighboring wavelet coefficients (space, orientation, or scale-neighborhoods). Their algorithm is far simpler than the ones in [11, 14], owing to the simplicity gained by the wavelet transform that allows for simple marginals to describe the required addition.

An entirely different approach for learning prior parameters is the one reported by Haber and Tenorio [13]. Whereas the previous methods learned the prior and based this on a set of K high-quality images, the work in [13] uses K pairs of images, $\{\boldsymbol{x}_i, \boldsymbol{y}_k\}_{k=1}^{K}$, representing the high-quality image and its degraded version, using the same degradation (and noise) to be overcome in the inverse problem at hand. Thus, the images $\boldsymbol{y}_k$ are generated by simulating the degradation effects,

$$\boldsymbol{y}_k = \boldsymbol{H}\boldsymbol{x}_k + \boldsymbol{v}_k. \tag{14}$$

Since the role of regularization is primarily to get better reconstruction, Haber's way of finding the parameters is to minimize the reconstruction error on the above training set. Considering a MAP reconstruction formula

$$\hat{\boldsymbol{x}}_{\mathrm{MAP}}^{k}(\theta) = \operatorname{Arg}\min_{\boldsymbol{x}} \{\|\boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}\|_2^2 + \lambda \cdot A(\boldsymbol{x}, \theta)\}, \tag{15}$$

the results are functions of the parameters θ. By minimizing

$$\sum_{k=1}^{K} \|\hat{\boldsymbol{x}}_{\text{MAP}}^{k}(\theta) - \boldsymbol{x}_i\|_2^2 \tag{16}$$

with respect to θ, we tune the parameters to lead to the minimal MSE in an empirical sense. Such interesting mixture of MMSE and MAP methods is very effective, and breaks the pure Bayesian interpretation of the energy function $A(\boldsymbol{x})$, since now it is related to the degradation operation. Just as before, here one also faces a complicated optimization task, which can be handled only in simple parametric forms.

One last family of techniques that falls under the same regularization learning methodology is the one that targets the quest for a dictionary that yields sparse representations [10, 12, 15–17]. These methods are based on the assumption that the signal in mind, $\boldsymbol{x}$, could be created as a sparse linear combination of columns from the dictionary $\boldsymbol{D}$, namely, $\boldsymbol{x} = \boldsymbol{D}\alpha$. The matrix $\boldsymbol{D}$ is full-rank, having more columns than rows. This implies that there are infinitely many ways to construct $\boldsymbol{x}$ as linear combination of columns from $\boldsymbol{D}$. Among all these possibilities, we consider the sparsest—the one that fuses the smallest number of columns in such construction. Thus, handling the general inverse problem posed in Section 2, the MAP approach in this case leads to [7],

$$\hat{\boldsymbol{x}}_{\text{MAP}} = \boldsymbol{D} \cdot \text{Arg}\min_{\alpha}\{\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{D}\alpha\|_2^2 + \lambda \cdot \|\alpha\|\}. \tag{17}$$

In this penalty function, we describe the desired signal as $\boldsymbol{x} = \boldsymbol{D}\alpha$ and force its representation vector α to be sparse. All the work reported in [10, 12, 15–17] considers the question of training the dictionary $\boldsymbol{D}$ to perform best in such inverse problems, with differences in the numerical methods proposed, or the way to fuse local and global relationships in the spatial domain.

Common to all the above methods is the fact that a parametric energy function is used and its parameters are tuned by the examples. Also, all these methods call for an involved optimization procedure, but one that should be done off-line. Once the regularization expression is ready, it can be deployed for use in the inverse problem in mind. Haber's work restricts the parameters to the same inverse problem trained on, while the other techniques are more general, allowing the use of the prior found in every inverse problem.

3.2. Learning the posterior directly via examples

The above-described approach uses examples indirectly, by training the regularization parameters. An entirely different way of exploiting examples is to use them directly within the reconstruction process. In such an approach, the examples are gathered to a database and used explicitly in the on-line reconstruction algorithm. These gathered examples may be regarded as samples from the posterior $\boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y})$, and as such they offer a direct way of performing reconstruction.

The example database organization is similar to the way described in Haber's method described above—gathering a set of high-quality images and a corresponding set of degraded versions thereof, obtained by applying $\boldsymbol{H}$ on each and adding noise. This gives us the set of image pairs $\{\boldsymbol{x}_k, \boldsymbol{y}_k\}_{k=1}^{K}$. Thus, the obtained database is tightly coupled to the type of degradation $\boldsymbol{H}$ that characterizes the inverse problem to be solved.

The basic idea behind the direct use of examples is one of *pattern matching*, i.e. given a low-quality image $\boldsymbol{y}$, seek in the database similar low-quality examples. Taking their corresponding high-quality pairs, those could be used for the reconstruction as they provide the high-quality content that fits the measurements. Clearly, such a process cannot be operated on large size images, since this implies an impossibly large database, so as to guarantee that every possible content encountered can be found. Therefore, the above method is applicable for small patches of images, with typical size of the low-quality images ranging between 5×5 and 25×25 pixels. This also implies that the above process is operated locally, or even on a pixel-by-pixel basis. Thus, given the image pairs described above, we sweep through the low-quality image set, and extract all image patches of size $n \times n$ (possibly with overlaps). This gives a very large set of example patches, denoted as $\mathcal{Y} = \{\boldsymbol{y}_k\}_{k=1}^{K_s}$ (with $K_s \gg K$—a typical database should contain at least $K_s = 10^6$ examples, and often much more). Per patch $\boldsymbol{y}_k \in \mathcal{Y}$, there is a corresponding patch of size $m \times m$ in the high-quality images. We denote the corresponding patches as $\mathcal{X} = \{\boldsymbol{x}_k\}_{k=1}^{K_s}$.

The choice of n (and hence m) is not trivial—choosing too small n means that the low-resolution patch is too small, and thus many irrelevant examples join the reconstruction process and divert it. Too large n may lead to no adequate examples in the database, and thus, to failure again. As to the choice of m, it depends on n, and on the degradation operation. A critical value of m is the one that contains all the pixels in $\boldsymbol{x}_k$ that are involved in constructing the measurement $\boldsymbol{y}_k$. For example, for $n = 5$, and a degradation that includes a 3×3 blur followed by 2:1 decimation in each axis, we get $m = 11$ (see a figure that illustrates this in [25]). Choosing a smaller value for m wastes an information within the corresponding measurements, and choosing a larger value for m implies that the high-resolution patch relies on the spatial context, rather than the measurements alone, and as such, it may be misleading. Interestingly, the various works reported in [5, 18–21, 23] all assume much smaller m. In [25] it has been shown that using the critical value of m leads to best results.

Once the database $\{\mathcal{X},\mathcal{Y}\} = \{\boldsymbol{x}_k, \boldsymbol{y}_k\}_{k=1}^{K_s}$ is ready, it can be used directly in the reconstruction algorithm. Note that this data set may contain the original gray-scale images, as described in [25], or high-pass (and possibly multi-scale)

versions as has been commonly used in [20–22, 24]. This choice of representation is tightly coupled with the type of images we target.

Turning to the reconstruction process, consider a given low-quality image $\boldsymbol{y}$, known to be damaged by $\boldsymbol{H}$ and by additive white Gaussian noise of strength σ^2. Per every location $[i, j]$ in the image, we extract a patch of size $n \times n$, denoted as $\boldsymbol{y}_{[i, j]}$. At the heart of the reconstruction process lies the need to find the nearest neighbors of $\boldsymbol{y}_{[i, j]}$ from $\mathcal{Y}$. We consider all the candidate all examples $\boldsymbol{y}_k$ in $\mathcal{Y}$ satisfying

$$\|\boldsymbol{y}_{[i,j]} - \boldsymbol{y}_k\|_2^2 \leq T \tag{18}$$

as possible matches (or the single nearest among them, if only one is desired). The threshold T depends on the patch size and the noise variance (e.g. $T = 4n^2\sigma^2$). Having found this subset of examples, $\mathcal{Y}_{[i, j]} = \{\boldsymbol{y}_k^{[i, j]}\}_{k\epsilon\ \Omega[i, j]}$, their corresponding pairs $\mathcal{X}_{[i, j]} = \{\boldsymbol{x}_k^{[i, j]}\}_{k\in\ \Omega[i, j]}$ are the candidate patches to be used for the reconstruction.

Given the reference vector $\boldsymbol{y}_{[i, j]}$ of length n^2 and the database $\mathcal{Y}$ that contains K_s examples, the above-described search should be done efficiently, as it is part of the on-line algorithm. One way to accelerate this search is by the K–D tree algorithm [32], which organizes the database off-line to enable a fast search, by defining an optimal binary tree of thresholds on the input coordinates. This pre-organization requires $\mathcal{O}\{n^2 \cdot K_s \log K_s\}$ in computations and $\mathcal{O}\{K_s\}$ in memory. The thresholds in this algorithm are chosen optimally so as to expedite the search, and indeed, the K–D tree algorithm leads to an $\mathcal{O}\{\log K_s\}$ expected number of distance evaluations in the quest for *any predetermined* number of the closest neighbors. By choosing a large number of neighbors, we guarantee to find all the relevant ones, satisfying (18). Alternative methods that have been considered in the literature for speeding-up the search include clustering techniques, principal component analysis and other fast nearest-neighbor methods [33].

The above process is performed for every location $[i, j]$ in $\boldsymbol{y}$, or with jumps to reduce computational complexity. Assuming a full-overlap approach, for every location there is a set of candidate high-resolution $m \times m$ patches $\mathcal{X}_{[i, j]} = \{\boldsymbol{x}_k^{[i, j]}\}_{k\epsilon\ \Omega[i, j]}$. There are several ways one can use these results. Defining an output canvas $\hat{\boldsymbol{x}}$ as expanding the low-resolution image, we need to fill-in the pixel values. Every example found, $\boldsymbol{x}_k^{[i, j]}$, has a known footprint on this canvas, and thus there are several intuitive ways to proceed:

(1) *Scalar MMSE Estimate:* Considering the pixel $[I, J]$ in the output canvas $\hat{\boldsymbol{x}}$, it has many contributions, coming from all patches in $\mathcal{X}_{[i, j]}$ that overlap it. By simply averaging these values we essentially perform an approximate MMSE estimate. This is because these values can be considered as samplings from the posterior $\boldsymbol{p}(\boldsymbol{x}|\boldsymbol{y})$. By creating a histogram of these values, we get a 1D approximate description of this posterior, and the expected value can be computed by a simple mean of the samples.

(2) *Scalar MAP estimate:* The above procedure is susceptible to outliers. Using the very same histogram of those values, one can seek it's peak, and this will be the MAP estimation for the desired output. From a practical point of view, it is likely that this histogram is too poor to work with because of insufficient data, and curve fitting or smoothing will be needed.

(3) *A special case—non-overlap and 1-NN:* If this algorithm extracts only the nearest neighbor, and if the patches used are taken with no overlap, we get only one value per location $[I, J]$, and then the above two methods coincide, suggesting that the output at this location is simply the candidate value.

All these are pixel-based reconstructions, and as such, they are easy to implement. However, their simplicity comes with a price—the examples found contain many outliers, and those may divert the desired result. As we shall see in the next section, in some cases, the number of outliers may exceed the number of proper ones, and in those cases, even the MAP method may deteriorate.

The works reported in [20–22] employ the non-overlapping option with 1-NN. Freeman *et al.* [20, 21] also considered some (not full) overlaps, in the spirit of the MMSE approach described above. Other algorithms that lean on similar rational for texture synthesis, denoising or inpainting (filling in holes) are found in [18, 19, 23, 34]. This set of works is also markedly different in the origin of the examples—rather than taking them from a separate set of images, the examples are drawn from the given image itself. Another very related recent example-based work of extreme importance is the one reported in [35, 36]. These papers present an example-based image denoising algorithm, using examples from the corrupted image itself, averaged via weighting to obtain denoising.

3.3. Building the regularization expression with examples

The last family of techniques is one that fuses the above two approaches, and thus improves on both. On one hand, a regularization expression that considers the entire unknown image as a whole is better than a local treatment, and as such, should be preferred. Furthermore, when joined to the likelihood term, the influence of the measurements and the regularization can be merged in a clear way to define the objective of the reconstruction procedure. On the other hand, a local treatment enables parallelization and simplification of the algorithms, and providing a direct way to use the examples in the reconstruction, rather than leaning on a guessed expression.

One can enjoy both worlds when merging the two techniques. First, operate locally as described above, and find per every pixel $y_{[i,j]}$ its relevant nearest-neighbor patches $\mathcal{X}_{[i,j]}$. However, instead of a simple operation such as voting or averaging, as proposed above, plug these examples into an especially tailored regularization expression. Such expression would represent a tighter description of the forces the unknown image is supposed to satisfy, and in a holistic way that considers all the image. This idea has been practiced successfully in several recent works [20, 21, 24, 25, 37]. Beyond the expected improvement caused by handling the reconstruction process globally, such an approach is able to better handle outliers in the found examples.

Interestingly, the regularization obtained in the above-fused technique deviates from the classic Bayesian point of view. The above-proposed regularization cannot be considered as a general image prior, because it is a much narrower point of view of the image in mind. Furthermore, this expression is heavily dependent on the measurements, from which we have obtained the high-resolution nearest neighbors, and as such, this expression 'sees' much more than just the ideal signal behavior. One could consider this prior term as an attempt to model the true image prior in the vicinity of its true values, and as such being local in the signal space.

The pioneering work by Baker and Kanade [24] was the one to fully practice the above set of ideas. In handling the SR problem, Baker and Kanade formed an explicit regularization expression that requires proximity between the spatial derivatives of the unknown image to those of the found examples. The examples in their work are found by a pyramidal derivative set of features, which means that rather than using the raw data directly and an ℓ^2 measure of distance, a weighted ℓ^2 is effectively used. Every location obtains one example, being the nearest-neighbor, and all these forces are merged into one global expression. A similar and simpler method appears in [37], where direct gray-values are used, as we shall consider in the next section.

Freeman *et al.* [20, 21] also considered a similar approach, but with some important differences. Rather than forming an explicit regularization expression, their MAP method adopts a Bayesian network point of view. Their algorithm defines local probabilities that take into account the proximity between the low-resolution measurements and the database patches (these parallel the likelihood term), and the agreement between high resolution neighboring patches between themselves (which parallel the regularization). The proposed algorithm remains local, as it does not consider the unknown image as a whole. Indeed, rather than concentrating on the true unknown $\boldsymbol{x}$, the focus is on the network interpretation of the data, discovering the nearest–neighbors that survive a Bayesian belief propagation (BPP) algorithm, using those in the formation of the solution. A similar technique with BBP is also described in [38], although leaning on examples from the image built.

Our recent work reported in [25] was inspired by the above algorithms, and considered a simplified MAP method that targets scanned document images. Similar to [24], an explicit regularization expression is formed, although using the raw data directly, instead of complicated features. It was shown that for the specific images handled (scanned documents), this approach leads to better results. Also, instead of using a single example per location, the work in [25] uses a multitude of them, and then pruning those based on the very same MAP formulation. More on this and new results in face images are given in the next section.

3.4. Using examples in inverse problems: a summary

In this section, we have seen many ways to practice the use of examples in forming a regularization for inverse problems. The major questions one faces when designing such algorithms are:

(i) Which examples to use? The examples can be taken from the corrupted image(s) itself or from other images. Also, one could work with pairs of low and high-quality images, or only high-quality images.
(ii) Which estimator to use? We have seen the MMSE and MAP being used, and a related question is whether to work globally or locally.
(iii) How to use the examples? We have seen them used indirectly by training a regularization parameters, directly in constructing the reconstruction result or plugged into a tailored regularization expression.
(iv) How to represent examples? Beyond the natural use of raw data, one can extract features, as high frequencies, multi-scale derivatives and more.
(v) How to organize the examples? In algorithms that employ on-line searches for the nearest–neighbor a pre-organization is mandatory for a fast search. We have mentioned the K–D tree, clustering methods and PCA brought to use.
(vi) What is specific to the inverse problem being considered? The overall algorithm depends on the type of inverse problem addressed.

We now turn to describe our own recent efforts in using examples for the image scale-up problem.

4. RECENT RESULTS

In this section, we briefly summarize the method described in [25] for scanned documents, and then show how it is extendable with some modifications to facial images.

4.1. General algorithm

Figure 1 describes a low-resolution (the degradation details are as those described at the beginning of Section 4.2) patch of size 5×5 taken from a text image. The figure also presents the original high-resolution corresponding patch. Searching in a database with 197,000 examples, taken from a similarly scanned printed page, Figure 1 shows the closest 50 examples. All are well within the required distance to assure a proper proximity (in the low-resolution domain). However, when computing the root-mean-squared error (RMSE) between the chosen high-resolution patches and the original content, we see that most of the chosen examples are outliers with irrelevant content.

The remedy to the above-described outliers problem is to exploit the coherence we expect to have between adjacent patches. However, to exploit this potential, we have to abandon the pixel-based methods. As mentioned above, using a global penalty function that ties the examples to each other may help in addressing this problem. The method proposed uses the found examples to define a global image regularization. This by itself is not sufficient for robustness against outliers. Thus, we use the emerging MAP penalty function to choose the problematic patches and prune them out. As opposed to the work described in [20, 21] (where high-pass filtered images are used), gray-scale values are used directly; this simplifies the overall algorithm.

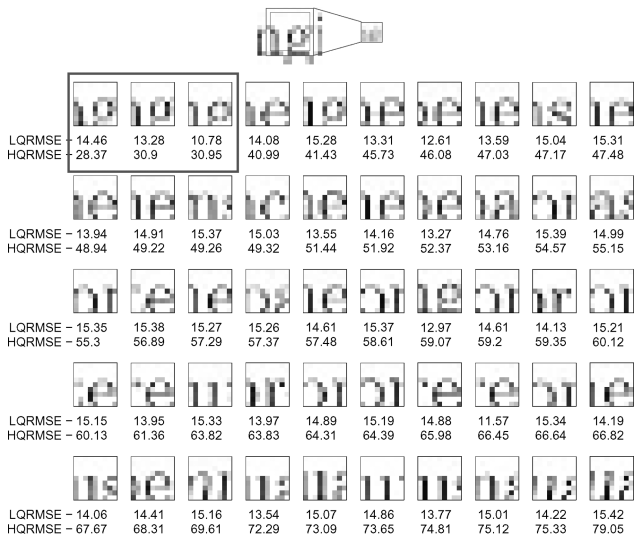


FIGURE 1. Outliers in searching nearest-neighbor. Top: The high quality image (left), and the corresponding measurements (right). Both 11×11 and 9×9 blocks are marked. Bottom: The 50 nearest-neighbors found, their RMSE in the low-resolution and the high-resolution (9×9) domains. As can be seen, while all examples are close in the low-resolution, many of them are in fact outliers.

Given the chosen examples, we can propose the following MAP penalty functional:

$$\epsilon(\hat{\boldsymbol{x}}) = \|\boldsymbol{H}\hat{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 + \lambda \sum_{i,j} \sum_{k \in \Omega[i,j]} \|\boldsymbol{R}_{[i,j]}\hat{\boldsymbol{x}} - \boldsymbol{x}_k^{[i,j]}\|_2^2. \quad (19)$$

In this functional, the first term stands for the log-likelihood, with the assumption that the noise is white and Gaussian. The second is the regularization term, and it is defined via the use of the examples found per location $[i, j]$. The operator $\boldsymbol{R}_{[i,\,j]}$ extracts a block of size $m \times m$ pixels from the image $\hat{\boldsymbol{x}}$ that matches the footprint of the corresponding examples. The inner summation is done over all found nearest–neighbors, their indices taken from the set $\Omega[i, j]$. The outer summation runs through all pixels in the high-resolution image, using the indices (i, j). Thus, this expression suggests that the reconstructed image should agree with every found example and in every location. A similar concept appears in [24], where multi-scale derivatives are matched, rather than direct gray-values, as done here.

The above-proposed penalty functional in Equation (19) uses the local examples to define a global regularization for the unknown image. However, unfortunately this is not enough. To get an intuition for this expression, when $\lambda \to \infty$, its minimization leads to the simple pixel-based averaging algorithm described earlier. Furthermore, for a general value of λ and when considering the denoising problem (where $\boldsymbol{H} = \boldsymbol{I}$), the minimizing result is also a simple averaging, including the measurement at this pixel. While it is an improvement over the MMSE algorithm we had before, we have clearly failed to force spatial coherence between the patches, as desired. In fact, this also implies that the algorithm described in [24] has no robustness to outliers as well.

Some degree of outlier-resistance can be achieved by replacing the ℓ^2 norm in the prior terms with an ℓ^1 one. However, considering the denoising problem again, such change replaces the mean by a median, and for too many outliers as often happens, this method still fails. Furthermore, rather than discarding complete patches, upon discovering that they are misleading, the outliers will be handled on a pixel-by-pixel basis, which loses much of the existing potential.

The solution we propose is to assign a weight to every example, so that those examples 'living in harmony' with their surroundings are weighted high, while others are down-weighted. Thus, the alternative MAP penalty becomes

$$\epsilon(\hat{\boldsymbol{x}}) = \|\boldsymbol{H}\hat{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 + \lambda \sum_{[i,j]} \sum_{k \in \Omega[i,j]} w_k^{[i,j]} \|\boldsymbol{R}_{[i,j]}\hat{\boldsymbol{x}} - \boldsymbol{x}_k^{[i,j]}\|_2^2. \quad (20)$$

There are many ways to estimate or choose these weights. Indeed, the work in [20, 21] offers a BPP as an attempt to prune the various examples. The work presented in [25] concentrates on a simplified and yet very effective case, where the

weights are binary: '0' for a bad example and '1' for a good one. One has to make sure, however, that not all the examples in a specific location get a zero weight, because then we may get a hole in our reconstruction. As has been shown, the MAP functional itself serves in evaluating these weights.

The proposed algorithm starts with the assignment $w_k^{[i,j]} = 1$ for all i, j and k. In a sequential process, these examples are pruned one patch at a time. For the current choice of weights, the minimizer of (20) is computed, and the value of $\epsilon(\hat{x})$ at the minimum, $\epsilon(\hat{x}_{\text{MAP}})$ serves as a reference value. Per patch $x_k^{[i,j]}$ with $w_k^{[i,j]} = 1$ (i.e. still active), we compute the optimal output image minimizing the modified MAP function

$$\tilde{\epsilon}(\hat{x}, i, j, k) = \epsilon(\hat{x}) - \lambda \|R_{[i,j]}\hat{x} - x_k^{[i,j]}\|_2^2. \tag{21}$$

Clearly, the value of this penalty term is necessarily smaller than the reference one. Among all these examined patches, we prune the one that gives the largest difference between the reference penalty value and the modified MAP penalty value. We denote those differences as $\Delta_k^{[i,j]}$. The patch discarded is considered to be the least compatible with the remaining patches.

While the above description implies a computationally heavy algorithm, several ways to speed it up dramatically can be proposed. First, in assessing $\Delta_k^{[i,j]}$ per patch, rather than recompute the minimizer of the modified penalty term, it can be updated only locally, in the vicinity of the removed patch. This local processing is based on the assumption that the effect of a removed patch is local, and exponentially decreasing outside its support, as empirically verified. Secondly, the update of the minimizer can be obtained by applying 2–5 conjugate gradient iterations only on such reduced support, using the previous image as initialization. Since the optimal solution changes slightly, such simple algorithm is sufficient. Finally, the same update of the solution is applicable for updating the optimal output image after the removal of an outlier patch.

A side benefit of this process is that we obtain a sequence of output images, one after each pruning step. Thus, beyond the first step that computes the optimal output image globally, all remaining steps are local and of low-complexity. As the punning process proceeds, the value of the MAP penalty in (20) is consistently decreasing. An efficient stopping rule for this process is the dynamic range found in the set $\Delta_k^{[i,j]}$—we consider the ratio between the maximal value of $\Delta_k^{[i,j]}$ to its median, and compare this to a fixed threshold. When this ratio gets below C (chosen as 0.25 the initial value in our experiments), all remaining patches are considered as positive contributors, and the algorithm is stopped. Alternatively, the removal of patches can be stopped when per location $[i, j]$ we have one example remaining. Since the algorithm prunes sequentially patches from the found set, and since their number is finite, the proposed process necessarily stops at some point.

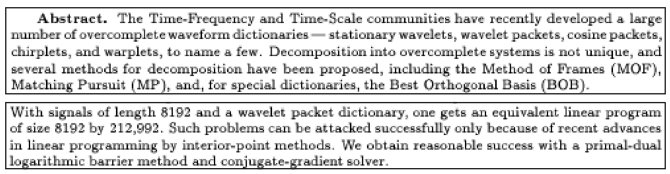


FIGURE 2. Experiment #1 – a text image: Patches of size 11×11 taken from these two images form the example database for Experiment #1 of text image reconstruction. The patch pairs are obtained by creating a degraded image using a separable 3-tap blur with the kernel [0.25, 0.5, 0.25], a scale factor of 2, and using patches of size 5×5 in the low-resolution image. Overall, there are $Ks =$ 47,000 examples in the database.

4.2. Examples on scanned documents

The above algorithm has been tested on scanned documents, where the use of raw gray values in representing the examples seems to perform the best [25]. Here, we provide two new examples to illustrate the behavior of the algorithm, one on a text image and the other on a drawing.

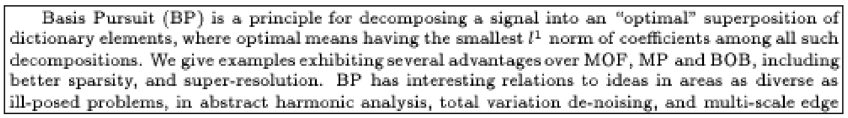
The first experiment involves a text image. The images shown in Figure 2 were used for extracting nearest-neighbor examples. In this and later experiments, we used $m = 11$ (high-quality patch size) and $n = 5$ (low-quality patch size). The degradation operator used is a 2D low-pass separable 3-tap blur [0.25, 0.5, 0.25], a scale factor of 2, and an additive white Gaussian noise with $\sigma = 8$.

Figure 3 presents the original image, its degraded version, and its reconstruction results using bi-cubic interpolation, MMSE estimator (i.e. averaging the examples per pixel as described in Section 3.2) and the result after pruning. In this and later experiment, we have used one example per location with full overlaps, implying that per every pixel we have 25 candidate values. We fixed $\lambda = 1.6e - 2$, and performed 1090 pruning stages out of the overall 6860 initial examples.[2]

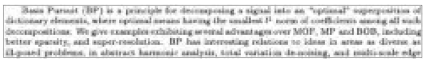
Figures 4 and 5 show similar training information and reconstruction results for the second experiment that considers a drawn cartoon. In this experiment, the noise power was chosen to be $\sigma = 2$, and thus $\lambda = 1e - 3$, as λ is supposed to be proportional to σ^2. The algorithm performed 288 pruning stages out of the initial 1650 examples.

As can be seen in these two experiments, examples can lead to surprisingly good results, even with a simple averaging of the nearest examples. Nevertheless, we also see that when pruning is performed as proposed above, a further improvement is obtained.

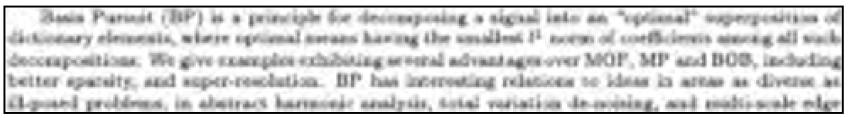
[2]Here and elsewhere, the number of pruning stages is governed by a stopping rule as described in [25].



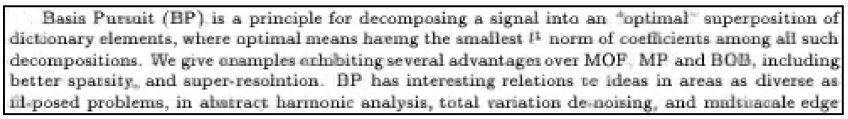
(a) Original image of size 65×499



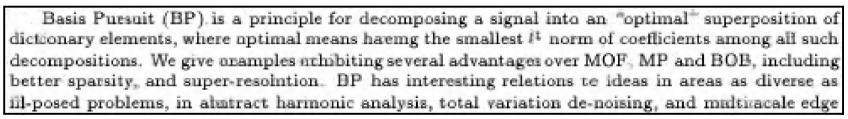
(b) Degraded image



(c) Bi-cubic interpolation result (MSE = 2694.6)



(d) MMSE reconstruction (MSE = 434.5)



(e) Reconstruction after 1090 pruning iterations (MSE = 378.8)

FIGURE 3. Experiment #1—a text image.

4.3. Treatment of facial images

When turning to handle facial images, a change is needed in the way the examples are fitted. Using the raw gray-values directly leads to no feasible neighbors in many cases, as the diversity of the image patches is large. Following the idea of using high-pass operators such as spatial derivatives as promoted in [20, 21, 24], we consider the image patches with their mean removed. This means that in the database construction stage, the pairs $\{\boldsymbol{y}_k, \boldsymbol{x}_k\}_{k=1}^{K_s}$ are gathered without their mean, i.e. $\boldsymbol{1}^{\mathrm{T}}\boldsymbol{y}_k = 0$ and $\boldsymbol{1}^{\mathrm{T}}\boldsymbol{x}_k = 0$ for all k. Given the measured low-quality patch $\boldsymbol{y}_{[i,j]}$, its mean

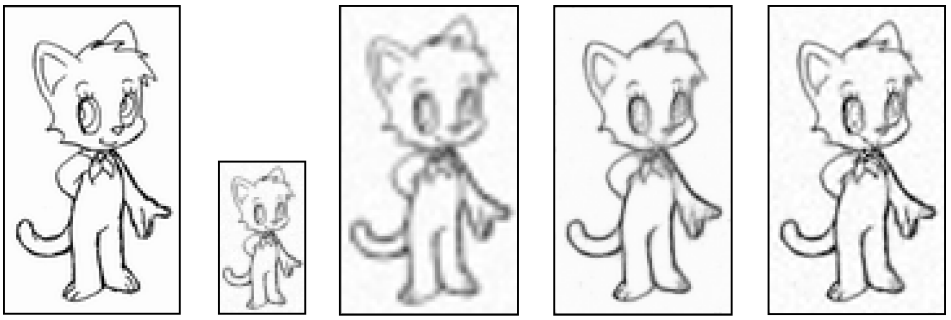
$$d_{[i,j]} = \frac{1}{n^2}\boldsymbol{1}^{\mathrm{T}}\boldsymbol{y}_{[i,j]} \tag{22}$$

is also removed and kept aside for later use. The nearest neighbors are found as before with the meanless patches, and we obtain the sub-set of high-quality patches $\mathcal{X}_{[i,j]}$ as before. Those are used in our formulation in Equation (20), with a constant $d_{[i,j]}$ added to them.

The above method resembles the example-search technique proposed in [20, 21], but there are a few differences. Their method suggests an application of a low-pass filter on the measured image $\boldsymbol{y}$ to obtain the low-frequencies of the reconstruction $\boldsymbol{y}_{\mathrm{LPF}}$. This image is up-sampled by a plain (e.g. bi-cubic) interpolation to the high-resolution canvas, and serves as the low-frequencies in the destination image. The reconstruction process is applied on the residual, by interpolating also $\boldsymbol{y} - \boldsymbol{y}_{\mathrm{LPF}}$ to the higher grid, and fitting examples to patches in this image. Thus, their method requires more computations, as it works on a wider image and larger patch sizes.

We tested the above-described algorithm on the ORL face image database, which contains 400 images of 40 people, as shown in Figure 6.

FIGURE 4. Experiment #2—a drawing image. Patches of size 11×11 taken form these four images and rotated versions of them in 5° increments (around a full circle) form the example database for Experiment #2 of drawn cartoon image reconstruction. The image pairs are obtained by creating a degraded image using a separable 3-tap blur with the kernel [0.25, 0.5, 0.25], a scale factor of 2, and using patches of size 5×5 in the low-resolution image. Overall, there are $K_s \approx 3e + 6$ examples in the database.



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

FIGURE 5. Experiment #2—a drawing image.(a) Original image of size 119 × 69.(b) Degraded image.(c) Bi-cubic interpolation result (MSE = 1118.9).(d) MMSE reconstruction (MSE = 463.2).(e) Reconstruction after 288 pruning iterations (MSE = 366.0).

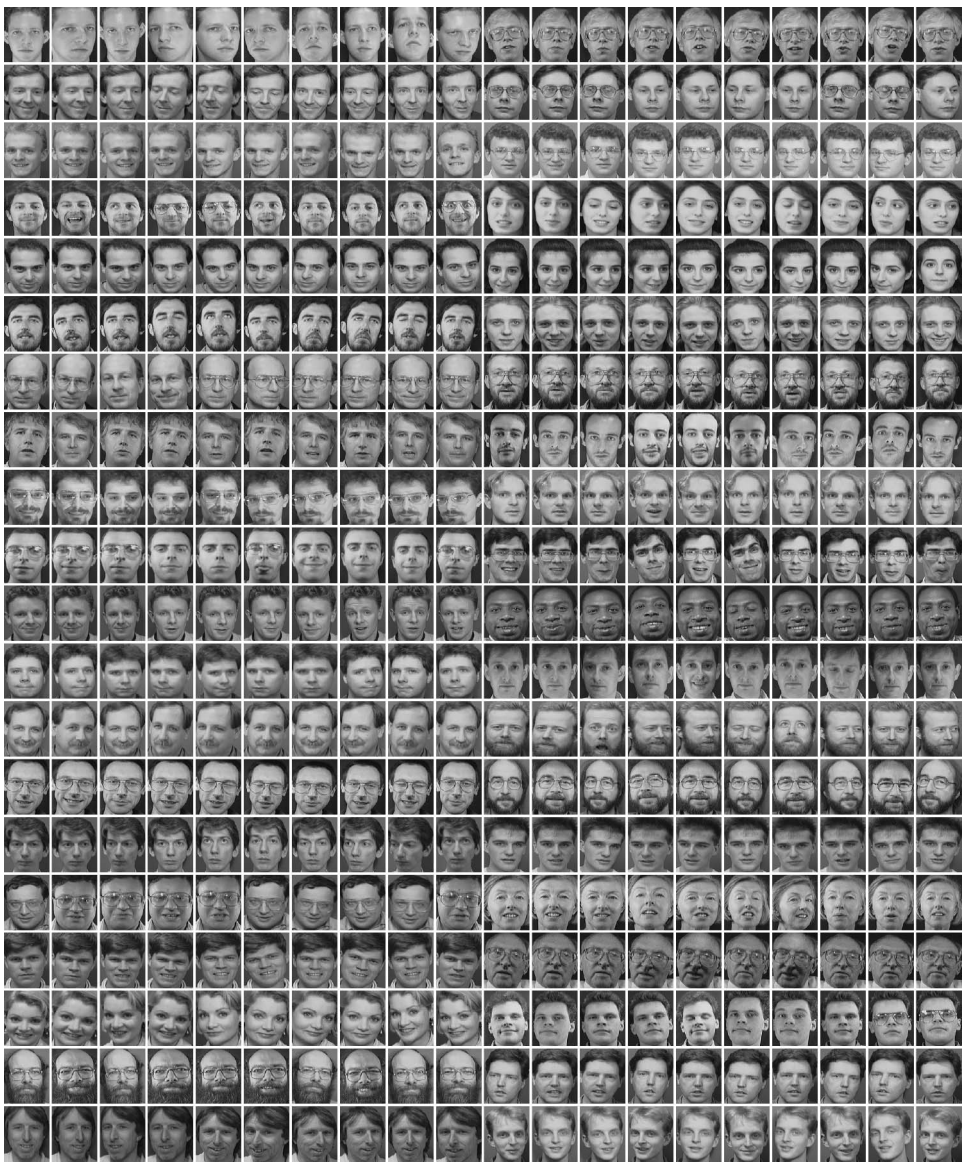


FIGURE 6. The ORL face database, containing 40 different people, each with 10 images of size 112 × 92 pixels.

As before, the reconstruction tests include a bi-cubic interpolation, the MMSE (averaging the gray-values of the found examples per location), and the pruned result. The tests reported here assume a Gaussian blur of width 5×5 pixels with $\sigma = 1.5$, a down-scaling factor of 3:1 and an additive noise with $\sigma = 2$ gray values. The reconstruction is based on 10 nearest-neighbor patches per pixel, using $\lambda = 2e - 4$, and patch sizes $n = 4$ and $m = 12$. In the example-based reconstruction tests, we tested three options: (i) using all 399 remaining faces; (ii) using the 9 images of the same person; and (iii) using 390 images of all other people in the database, excluding the same person. The results for two different people (chosen in random) in the database are shown in Figures 7 and 8, and support the need for pruning as proposed here.

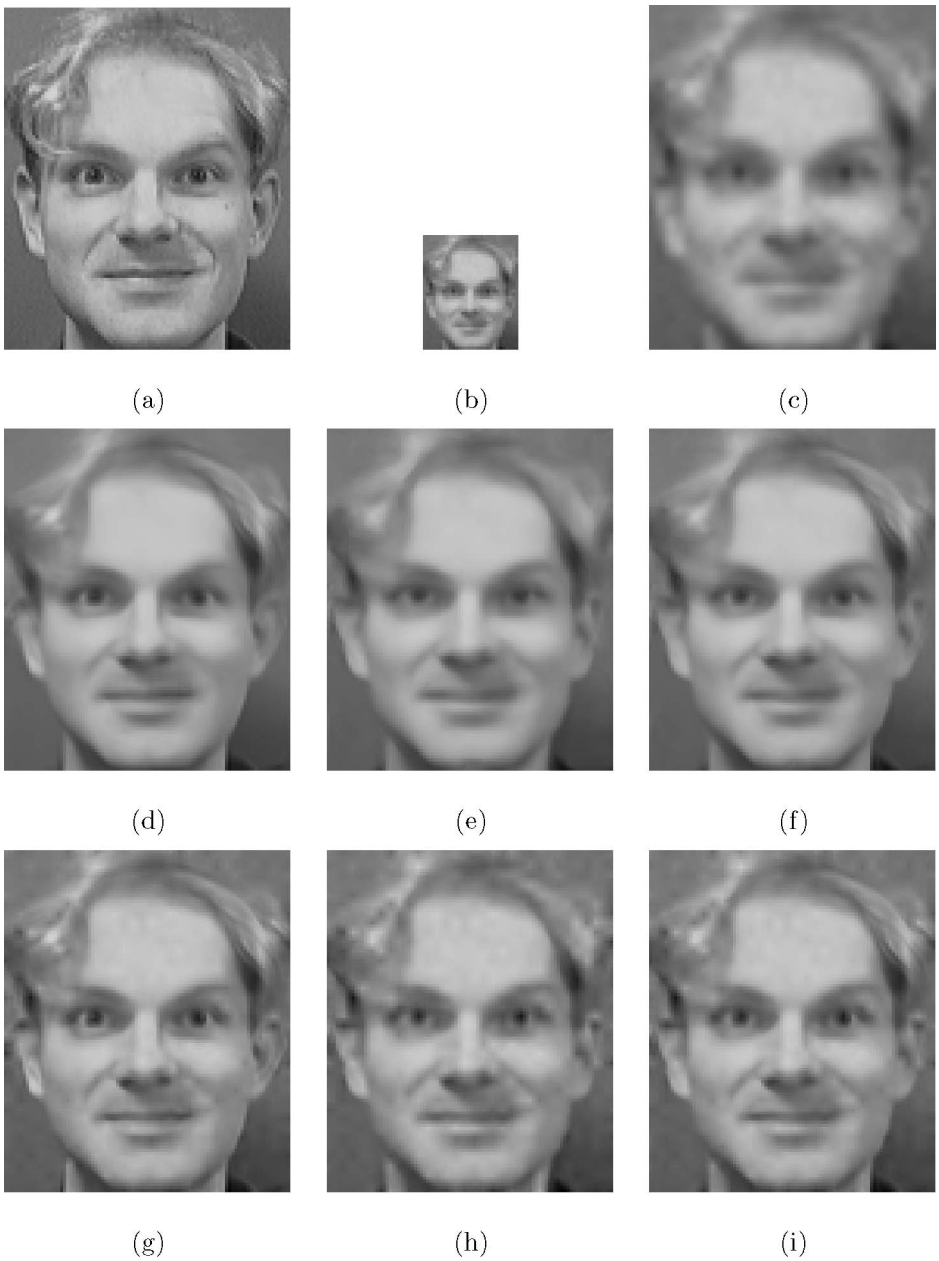


FIGURE 7. Face 1.(a) Original image.(b) Degraded image.(c) Bi-cubic interpolation result (MSE = 149.64).(d) MMSE reconstruction (MSE = 75.4) based on nine images.(e) MMSE reconstruction (MSE = 97.1) based on 390 images.(f) MMSE reconstruction (MSE = 79.9) based on 399 images.(g) Pruned (1970 steps) reconstruction (MSE = 62.38) based on nine images.(h) Pruned (3060 steps) reconstruction (MSE = 88.01) based on 390 images.(i) Pruned (2030 steps) reconstruction (MSE = 72.71) based on 399 images.

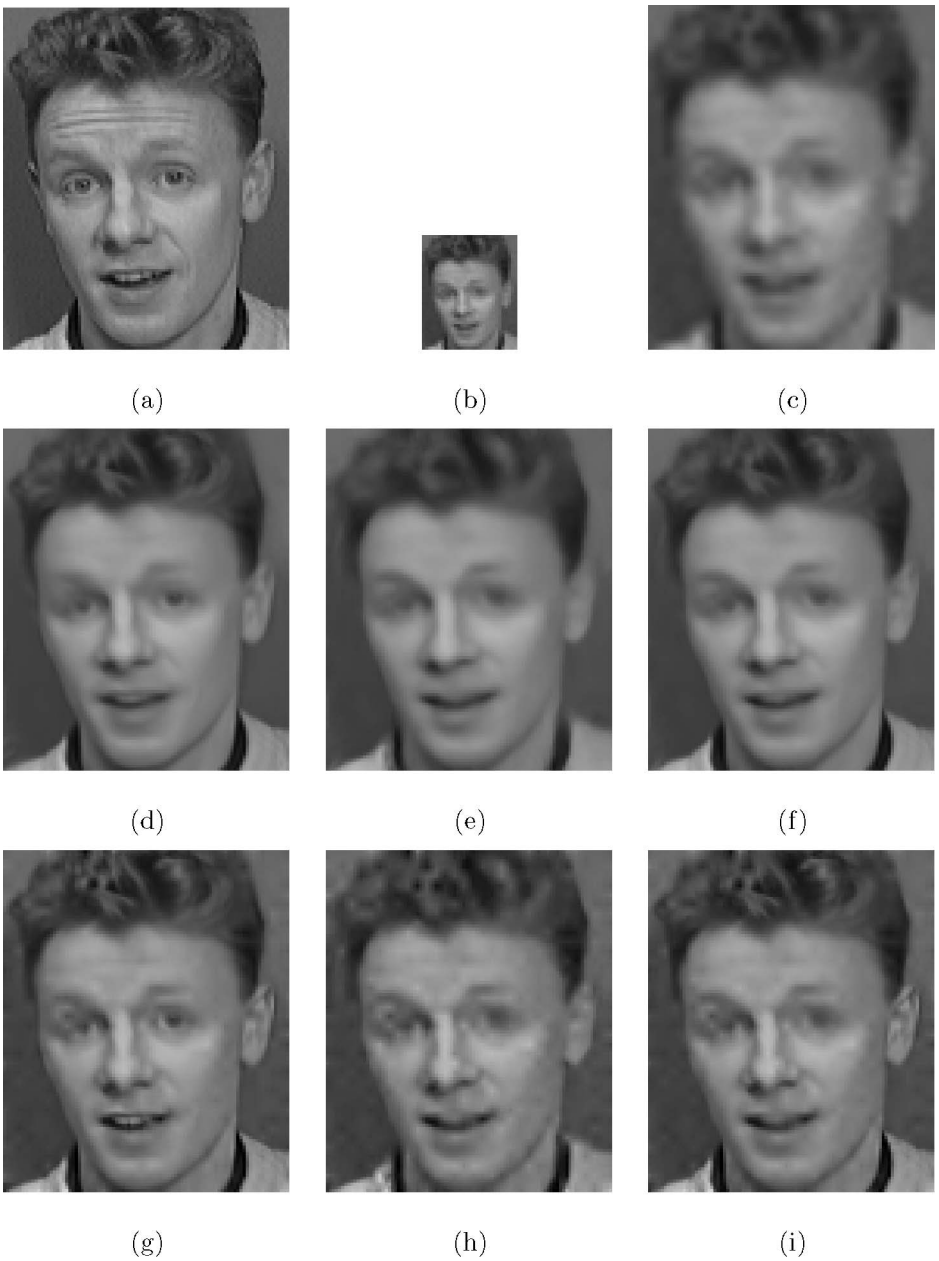


FIGURE 8. Face 2.(a) Original image.(b) Degraded image.(c) Bi-cubic interpolation result (MSE = 159.25).(d) MMSE reconstruction (MSE = 65.24) based on nine images.(e) MMSE reconstruction (MSE = 102.97) based on 390 images.(f) MMSE reconstruction (MSE = 68.90) based on 399 images.(g) Pruned (2280 steps) reconstruction (MSE = 60.08) based on nine images.(h) Pruned (5320 steps) reconstruction (MSE = 92.66) based on 390 images.(i) Pruned (3780 steps) reconstruction (MSE = 65.61) based on 399 images.

5. CONCLUSIONS

Examples can be used for obtaining an effective regularization in inverse problems involving images. This is especially true for specific type of images, such as faces or scanned documents, as demonstrated in this paper. The use of examples provides a step forward in reconstruction quality, compared to the classic priors or regularizations that have been proposed in the past decade. There are several ways to exploit examples in inverse problems: three techniques have been described in this paper—learning parameters of the regularization expression, a direct use of the examples in forming the posterior, or in merging these techniques. This paper describes a specific method that belongs to the later family of algorithms, focusing on the need to prune outlier examples to fine-tune the outcome.

In deploying the above set of ideas to inverse problems, there are many open questions that are yet to be addressed. Here we list few of those:

(i) Multi-scale treatment? It seems natural to consider a multi-scale method that considers image patches of varying sizes in seeking fitting examples. One way to implement this idea is by choosing the maximal size that gives sufficient number of examples. However, a fast nearest-neighbor algorithm that can cope with varying patch sizes should be devised.

(ii) Theoretical foundations? While all the above discussion makes a lot of sense and seems intuitive, using examples should be strengthen by a supporting theoretical study. No such study has been proposed as of yet.

(iii) How big should the database be? This question is tightly coupled with the previous one. We are sampling a specific distribution, and we need sufficient number of examples so as to claim reasonable proximity to every instance that can be encountered. Thus, the richness (or entropy) of the image distribution should be taken into account in gathering the database.

(iv) Regularization or prior? We have seen examples used both as a way to drive a prior (i.e. practice a pure Bayesian approach), or for forming a measurement-dependent regularization expression. It is unclear at all which of the two techniques is better, and how these two methods could be compared.

(v) What about general content images? When the inverse problem deals with a general content image, the amount of examples should grow dramatically, and perhaps leading to the point of requiring an impractical algorithm. Are example-based techniques at all fitted for handling general images?

These and many more related questions will be probably studied by researchers in the coming years. The potential of examples in handling inverse problems better is unquestionable, and as such, the interest in this field is expected to grow.

REFERENCES

[1] Elad., M. and Feuer, A. (1997) Restoration of single super-resolution image from several blurred, noisy and down-sampled measured images. *IEEE Trans. Image Process.*, **6**(12), 1646–1658.

[2] Farsiu, S., Robinson, D., Elad, M. and Milanfar, P. (2004) Fast and robust multi-frame super-resolution. *IEEE Trans. Image Process.*, **13**(10), 1327–1344.

[3] Farsiu, S., Robinson, D., Elad, M. and Milanfar, P. (2004) Advanced and challenges in super-resolution. *Int. J. Imag. Syst. Technol.*, **14**(2), 47–57.

[4] Jain, A.K. (1989) *Fundamentals of Digital Image Processing*. Prentice-Hall, Englewood Cliffs, NJ.

[5] Lagendijk, R.L. and Biemond, J. (1991) *Iterative Identification and Restoration of Images*. Kluwer Academic Publishers, Boston.

[6] Tikhonov, A.N. and Arsenin, V.A. (1977) *Solution of Ill-posed Problems*. Winston & Sons, Washington.

[7] Elad, M. (2002) On the bilateral filter and ways to improve it. *IEEE Trans. Image Process.*, **11**(10), 1141–1151.

[8] Weickert, J. (1998) *Anisotropic Diffusion in Image Processing*. ECMI Series, Teubner, Stuttgart.

[9] Donoho, D.L. and Johnstone, I.M. (1994) Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, **81**(3), 425–455.

[10] Olshausen, B.A. and Field, D.J. (1997) Sparse coding with an overcomplete basis set: a strategy employed by V1? *Vis. Res.*, **37**, 311–325.

[11] Zhu, S.C. and Mumford, D. (1997) Prior learning and Gibbs reaction-diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, **19**(11), 1236–1250.

[12] Engan, K., Aase, S.O. and Hakon-Husoy, J.H. (1999) Method of optimal directions for frame design. *IEEE Int. Conf. Acoustics, Speech, and Signal Processing* (*ICASSP*), Phoenix, Arizona. March 15–19, vol. **5**, 2443–2446.

[13] Haber, E. and Tenorio, L. (2003) Learning regularization functionals. *Inverse Probl.*, **19**, 611–626.

[14] Roth, S. and Black, M.J. (2005) Fields of experts: a framework for learning image priors. *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, San-Diego, California, June 20–25, vol. **2**, pp. 860–867.

[15] Aharon, M., Elad, M. and Bruckstein, A.M. (2006) The K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Proc.*, **54**(11), 4311–4322.

[16] Aharon, M., Elad, M. and Bruckstein, A.M. (2006) On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *J. Linear Algebr. Appl.*, **416**(11), 48–67.

[17] Elad, M. and Aharon, M. (2006) Image denoising via sparse and redundant representations over Learned dictionaries. *IEEE Trans. Image Process.*, **15**(12), 3736–3745.

[18] Efros, A.A. and Leung, T.K. (1999) Texture synthesis by non-parametric sampling. *IEEE Int. Conf. Computer Vision (ICCV)*, Corfu, Greece, September 20–25, pp. 1033–1038.

[19] Wei, L.-Y. and Levoy, M. (2000) Fast texture synthesis using tree-structured vector quantization. *Proc. of SIGGRAPH*, New Oleans, Louisiana, pp. 479–488.

[20] Freeman, W.T., Pasztor, E.C. and Carmichael, O.T. (2000) Learning low-level vision. *Int. J. Comput. Vis.*, **40**(1), 25–47.

[21] Freeman, W.T., Jones, T.R. and Pasztor, E.C. (2002) Example-based super-resolution. *IEEE Comput. Graphi. Appl.*, **22**(2), 56–65.

[22] Nakagaki, R. and Katsaggelos, A.K. (2003) VQ-based blind image restoration algorithm. *IEEE Trans. Image Process.*, **12**(9), 1044–1053.

[23] Criminisi, A., Perez, P. and Toyama, K. (2004) Region filling and object removal by exemplar-based image inpainting, *IEEE Trans. Image Process.*, **13**(9), 1200–1212.

[24] Baker, S. and Kanade, T. (2002) Limits on super-resolution and how to break them. *IEEE Trans. Pattern Anal. Mach. Intell.*, **24**(9), 1167–1183.

[25] Datsenko, D. and Elad, M. (2007) Example-based single document image super-resolution: a global MAP approach with outlier rejection. *J. Math. Signal Process.*, to appear.

[26] Geman, S. and Geman, D. (1984) Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, **6**, 721–741.

[27] Rudin, L., Osher, S. and Fatemi, E. (1992) Nonlinear total variation based noise removal algorithms. *Physica D*, **60**, 259–268.

[28] Sochen, N., Kimmel, R. and Malladi, M. (1998) A geometrical framework for low level vision. *IEEE Trans. Image Process.*, **7**(3), 310–318.

[29] Sochen, N., Kimmel, R. and Bruckstein, A.M. (2001) Diffusions and confusions in signal and image processing. *J. Math. Imag. Vis.*, **14**(3), 195–209.

[30] Chen, S.S., Donoho, D.L. and Saunders, M.A. (2001) Atomic decomposition by basis pursuit. *SIAM Rev.*, **43**(1), 129–159.

[31] Buccigrossi, R.W. and Simoncelli, E.P. (1999) Image compression via joint statistical characterization in the wavelet domain. *IEEE Trans. Image Process*. **8**(12), 1688–1701.

[32] Friedman, J.H., Bentley, J.L. and Finkel, R.A. (1977) An algorithm for finding best matches in logarithmic expected time. *ACM Trans. Math. Softw*, **3**(3), 209–226.

[33] Nene, S.A. and Nayar, S.K. (1997) A simple algorithm for nearest–neighbor search in high dimensions. *IEEE Trans. Pattern Anal. Mach. Intell.*, **19**(9), 989–1003.

[34] Weissman, T., Ordentlich, E., Seroussi, G., Verdu, S. and Weinberger, M.J. (2005) Universal discrete denoising: known channel. *IEEE Trans. Inf. Theory*, **51**(1), 5–28.

[35] Buades, A., Coll, B. and Morel, J.M. (2005) A non-local algorithm for image denoising. *IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, San-Diego, California, June 20–25, **2**, pp. 60–65.

[36] Buades, A, Coll, B. and Morel, J.M. (2005) A review of image denoising algorithms, with a new one. *SIAM J. Multiscale Model. Sim. (MMS)*, **4**(2), 490–530.

[37] Pickup, L.C., Roberts, S.J. and Zisserman, A. (2003) A sampled texture prior for image super-resolution. *Adv. Neural Inf. Process. Syst.*, vol. **16** (NIPS 2003), 1587–1594.

[38] Storkey, A. (2003) Dynamic structure super-resolution. *Adv. Neural Inf. Process. Syst.*, **15**, 1295–1302.

[39] Jaynes, E.T. (1982) On the rationale of maximum-entropy methods. *IEEE Proc.*, **70**(9), 939–952.