Multidim Syst Sign Process (2007) 18:103–121
DOI 10.1007/s11045-007-0018-z

Example-based single document image super-resolution: a global MAP approach with outlier rejection

Dmitry Datsenko · Michael Elad

Received: 16 December 2005 / Revised: 4 April 2006 /
Accepted: 6 June 2006 / Published online: 13 April 2007

Abstract Regularization plays a vital role in inverse problems, and especially in ill-posed ones. Along with classical regularization techniques based on smoothness, entropy, and sparsity, an emerging powerful regularization is one that leans on image examples. In this paper, we propose an efficient scheme for using image examples as driving a powerful regularization, applied to the image scale-up (super-resolution) problem. In this work, we target specifically scanned documents containing written text, graphics, and equations. Our algorithm starts by assigning per each location in the degraded image several candidate high-quality patches. Those are found as the nearest-neighbors (NN) in an image-database that contains pairs of corresponding low- and high-quality image patches. The found examples are used for the definition of an image prior expression, merged into a global MAP penalty function. We use this penalty function both for rejecting some of the irrelevant outlier examples, and then for reconstructing the desired image. We demonstrate our algorithm on several scanned documents with promising results.

Keywords Regularization · Example-based · Nearest-neighbor · Bayesian reconstruction · MMSE · MAP · Outliers · K-D tree

1 Introduction

Regularization plays a vital role in inverse problems, and especially in ill-posed ones. Such is the case in recovering a single image f that has gone through a sequence of degradations, including a blur, down-sampling, and additive noise — *the image*

D. Datsenko (✉) · M. Elad
Department of Computer Science, The Technion - Israel Institute of Technology, Haifa 32000, Israel
e-mail: datsenko@cs.technion.ac.il

M. Elad
e-mail: elad@cs.technion.ac.il

scale-up or super-resolution problem. In this case we model the formation of the measurements $\underline{g}$ from the ideal image $\underline{f}$ by

$$\underline{g} = \mathbf{D}\underline{f} + \underline{v}. \tag{1}$$

This is the problem we target in this paper, where we assume that the degradation operator $\mathbf{D}$ and the noise characteristics are known. For simplicity we will assume throughout this work that the noise is Gaussian, white, zero mean, and iid, with variance σ^2.

In a broader context, regularization is more than just a way to stabilize such inverse problems. Rather, it is a systematic method for adding more information to the reconstruction system. The Bayesian point of view suggests that such a regularization is tightly coupled with the desired-signal probability density function (PDF), often referred to as the signal prior. Thus, the art of choosing a proper prior for an inverse problem is really an educated guess as to the PDF of the signals in mind, $P(\underline{f})$. Once chosen, the prior can be used for the formation of the posterior probability $P(\underline{f}|\underline{g})$. Choosing the image $\underline{f}$ that maximizes this probability leads to the MAP estimate, and choosing the expected value leads to the MMSE estimate — both considered as Bayesian estimation techniques.

During the past 30 years, many attempts were made to describe image priors as simple analytical expressions. The classical priors are attempts to claim how "good-looking" images should behave. For example, the TV-expression, $\||\nabla\underline{f}|\|_1$, is built on a smoothness assumption, suggesting that the image should tend to be piece-wise constant (Rudin, Osher, & Fatemi, 1992; Sochen, Kimmel, & Bruckstein, 2001). Similarly, an expression of the form $\underline{f}^T \log(\underline{f})$ assumes that the scalar-entropy in the desired signal should be brought to an extreme (Jaynes, 1982). A prior of the form $\|\mathbf{W}\underline{f}\|_1$ ($\mathbf{W}$ being the wavelet transform) implies that we expect a sparsity of the wavelet coefficients for the desired signal (Chen, Donoho, & Saunders, 2001; Donoho & Johnstone, 1994).

Along with these classical smoothness-, entropy-, and sparsity-based priors, an emerging powerful regularization that is drawing research attention in recent years is one that leans on examples. Rather than guessing the image PDF and forcing a simple expression 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 two 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. For example, in a Markov random field (MRF) prior that leans on a robust measure of smoothness, the robust function and the derivative filters employed can both be learned from image examples (Buccigrossi & Simoncelli, 1999; Haber & Tenorio, 2003; Roth & Black, 2005; Zhu & Mumford, 1997). Similarly, in a sparsity based prior, the dictionary can be trained (Aharon, Elad, & Bruckstein, 2005a,b; Engan, Aase, & Hakon-Husoy, 1999; Olshausen & Field, 1997).
2. Learning the posterior: Rather than learn the image prior $P(\underline{f})$ and then plug it in a MAP/MMSE reconstruction penalty term, one can use the examples to directly learn the posterior probability density function $P(\underline{f}|\underline{g})$. Due to dimensionality problems, this would typically be done by forming a 1D function (histogram) for each pixel in $\underline{f}$ given the measurements in its vicinity. Alternatively, the examples can be considered as drawn samples from $P(\underline{f}|\underline{g})$, and used as such.

(Criminisi, Perez, & Toyama, 2004; Efros & Leung, 1999; Freeman, Jones, & Pasztor, 2002; Freeman, Pasztor, & Carmichael, 2000; Nakagaki & Katsaggelos, 2003; Wei & Levoy, 2000; Weissman, Ordentlich, Seroussi, Verdu, & Weinberger, 2005)

In this paper, we follow the above line of works, and propose an efficient algorithm for using image examples as driving a powerful regularization. We demonstrate the proposed scheme on the above-described image scale-up problem, targeting specifically scanned documents containing written text, graphics, equations, etc.. The approach we take falls in between the above two categories, and in that respect, our algorithm share some similarities with the methods presented in Baker and Kanade (2002) and Freeman et al. (2000, 2002). More on these ties will be described throughout the paper.

Our algorithm starts by assigning per each location in the degraded image several candidate high-quality patches. Those are found as the nearest–neighbors (NN) in an image-database that contains pairs of corresponding low- and high-quality image patches. This part of the algorithm is close in spirit to the methods that learn the posterior, where NN examples are drawn and used. Similar to these works, a way must be found to expedite the NN search – in this work we have used the K-D tree algorithm (Friedman, Bentley, & Finkel, 1977).

Instead of using the examples to compute the reconstructed image directly (as methods in this class do), the examples found are used for the formation of an image prior. As such, the constructed prior in our method is not a classic Bayesian prior, as it is based on the measurements. This prior is then used within a global MAP penalty function. In this respect, at this stage our method resembles the methods that employ learning prior parameters, and especially the work reported in Baker and Kanade (2002). However, rather than working on features (e.g., derivatives) and pyramidal structure, as done in Baker and Kanade (2002), in our algorithm plain gray-scale examples are used as is, which leads to a much simpler algorithm.

As the NN examples found include many outliers, our algorithm proposes a way of detecting and pruning them. Outliers cannot be avoided in the NN-search, because of the non-negligible null-space of the degradation operator. Such a null-space implies that many irrelevant patches are disguised as near-perfect matches. In our algorithm we use the very same MAP penalty function both for rejecting some of the irrelevant examples, and then for reconstructing the desired image. By forcing coherence between overlapping areas, both the rejection and the reconstruction stages enjoy a global view of the outcome, in-spite of the inherent locality of the patches used.

In the next section we describe the core scheme, and several intuitive reconstruction algorithms that emerge from it. In Sect. 3, we discuss how spatial coherence between the chosen examples is taken into account, and how pruning of outliers can be done within the MAP scope. Section 4 then demonstrates our algorithm on several text, graphics, and equation images, showing promising results.

2 Proposed scheme : The basics

In this section, we discuss the basics of how image examples can and have been used for the image scale-up problem, concentrating first on the database, and efficient searches in it. Given the found examples, we show how a preliminary and intuitive

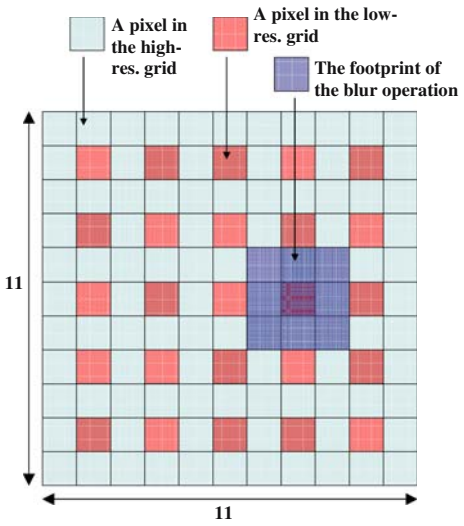


Fig. 1 For a 5×5 patch in the low-resolution grid, and assuming a degradation that includes a 3×3 footprint blur and a $2:1$ decimation in each axis, the corresponding maximal patch size in the high-resolution grid is of size 11×11 pixels

reconstruction is possible is several ways, starting with simple voting per pixel, and to the formation of the 1D probability density function that describe the posterior.

2.1 The image database

Given a set of high quality images, we can produce form them a corresponding set of degraded images, by applying the operator **D** on each. Given those image pairs, we sweep through the low-quality image set, and extract all image patches of size $n \times n$ (including overlaps). This gives us a very large set of examples, denoted as $\mathcal{Y} = \{\underline{y}_k\}_{k=1}^{K}$.

Per each patch $\underline{y}_k \in \mathcal{Y}$, there is a corresponding patch of size $m \times m$ in the high-quality images. We denote the corresponding patches as $\mathcal{X} = \{\underline{x}_k\}_{k=1}^{K}$. As for their size, we consider first all the pixels in $\underline{x}_k$ that are involved in the computation of $\underline{y}_k$. For example, if $n = 5$, and the degradation includes a 3×3 blur, followed by $2:1$ decimation in each axis, then $m = 11$, as Fig. 1 shows. This means that every pixel in the low-resolution $\underline{y}_k$ can be computed as a linear combination of a subset of the pixels in the corresponding patch $\underline{x}_k$.

Since the border pixels in this window are of weaker reliability, being related to fewer measurements, m could be chosen as 9, or 11 with down-weighting of the borders.[1] Choosing a smaller value for m wastes an information within the corresponding measurements. Choosing a larger 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 Efros & Leung (1999),

[1] A weight mask can be created by accumulating the blur kernels in their proper locations. We disregard this option for simplicity of the discussion.

Wei & Levoy (2000), Freeman et al. (2000, 2002), Nakagaki & Katsaggelos (2003), and Criminisi et al. (2004) all assume much smaller m. We will return to this point in the experimental section, showing that indeed, choosing $m = 9$ leads to better performance for the case described here.

We have now a large set of pairs $\{\mathcal{X}, \mathcal{Y}\} = \{\underline{x}_k, \underline{y}_k\}_{k=1}^{K}$, and this is the data that will be used directly for the reconstruction. This data-set contains simple gray-scale images, as opposed to features that have been commonly used in previous works (Baker and Kanade, 2002; Freeman et al., 2000, 2002; Nakagaki & Katsaggelos, 2003). This choice of features is tightly coupled with the type of images we target in this work – document images that are obtained via scanning. Such images are not subject to degrading illumination effects, their content is typically with high (and spatially fixed) contrast, and self-similarity across different contrast and brightness levels is highly unlikely.

2.2 Nearest neighbor search and K-D tree

Consider a given low quality image $\underline{g}$, known to be damaged by **D** and by an additive white Gaussian noise of strength σ^2. We are interested in recovering it using the above-constructed database. Per every location $[i, j]$ in the image we extract a patch of size $n \times n$, denoted[2] as $\underline{g}_{[i,j]}$. At the heart of the reconstruction process lies the need to find the nearest neighbors of $\underline{g}_{[i,j]}$ from $\mathcal{Y}$. We consider *all* the candidate examples in $\mathcal{Y}$ satisfying

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

as possible matches. The threshold T depends on the patch size and the noise variance (e.g., $T = 4n^2\sigma^2$). Define $\Omega[i, j]$ as the set of indices of the found NN. Having found this subset of examples, $\mathcal{Y}_{[i,j]} = \{\underline{y}_k^{[i,j]}\}_{k\in\Omega[i,j]}$, their corresponding pairs $\mathcal{X}_{[i,j]} = \{\underline{x}_k^{[i,j]}\}_{k\in\Omega[i,j]}$ are the candidate patches to be used for the reconstruction.

Given the reference vector $\underline{g}_{[i,j]}$ of length n^2 and the database $\mathcal{Y}$ that contains K examples, the above-described search is done in this work using the K-D tree algorithm (Freeman et al., 1977). This algorithm organizes the database off-line to enable a fast search, by defining a binary tree of thresholds on the input coordinates. This pre-organization requires an $\mathcal{O}\{n^2 \cdot K \log K\}$ in computations and $\mathcal{O}\{K\}$ 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 $\mathcal{O}\{\log K\}$ expected number of distance evaluations in the quest for *any pre-determined* number of the closest neighbors. By choosing a large number of neighbors, we guarantee to find all the relevant ones, satisfying (2).

2.3 Pixel-based reconstructions

Assume that the entire image $\underline{g}$ has been scanned. This implies that for every location $[i, j]$ in $\underline{g}$, a patch of size $n \times n$ has been extracted, and a set of candidate high-resolution $m \times m$ patches $\mathcal{X}_{[i,j]} = \{\underline{x}_k^{[i,j]}\}_{k\in\Omega[i,j]}$ has been found. There are several ways one can use these results. Defining an output canvas $\underline{\hat{f}}$ as expanding the low-resolution image as

[2] This patch is a lexicographic ordered column-vector of length n^2.

described in Fig. 1, we need to fill-in the pixel values. Every example found, $\underline{x}_k^{[i,j]}$, 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 $\hat{f}$, it has many contributions, coming from all pathes in $\mathcal{X}_{[i,j]}$ that overlap it. For example, if every low-resolution patch gets 100 NN, and we consider the case described in Fig. 1, then there are $25n^2$ values per each location $[I,J]$ in the high-resolution canvas (disregarding image boundaries). 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 $P(f_{[I,J]}|\underline{g})$. 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 each 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 next, 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 Freeman et al. (2000, 2002) and Nakagaki & Katsaggelos (2003) employ both a non-overlapping option with 1-NN, and with overlaps, in the spirit of the MMSE approach described above. We should also note that the work in Freeman et al. (2000, 2002) considered a pruning of the found examples in order to reject outliers. More on this would be mentioned in the next section.

To conclude this section, we discuss briefly the parameters involved. Our experience shows that a typical database should contain at least $K = 10^5$ examples, and this can grow to $K = 10^8$ and beyond when handling more complex content. The choice of n dictates the complexity and the accuracy of the reconstruction algorithm, as well as the required size of the database, K. As n grows, both the the reconstruction quality and the complexity grow. This, however, is true with the assumption that the database is rich enough. For too large values of n, the NN search may fail to provide neighbors altogether, and then more examples are needed. Our experience, as will be demonstrated in Sect. 4, shows that n in the range 3–10 is reasonable. We should note that adopting a multi-scale approach, where n varies from one pixel to another, based on content, availability of examples, and more, could lead to substantial improvement, but we have not pursued this option in this work.

3 Exploiting spatial coherence

Figure 2 describes a low-resolution (the degradation details are as those described at the beginning of Sect. 4) patch of size 5×5 taken from a text image. The Figure also

Fig. 2 *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

presents the original high-resolution corresponding patch. Searching in a database with 197,000 examples, taken from a similarly scanned printed page, Fig. 2 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, in order to exploit this potential, we have to abandon the pixel-based methods. Previous work handled this task in several ways. Freeman et al. (2000, 2002) suggested to model the inter-relations between adjacent/overlapping patches by a random Markov network, and use of the Bayesian-Belief-Propagation (BBP) for choosing the proper examples in the recovery. A second approach used in their simulations is far simpler, using a raster-scan sequential reconstruction, and requiring a compatibility between the already recovered areas, and the newly considered and overlapping patches.

Addressing the same problem, we are suggesting a different and more intuitive solution. Following (Baker and Kanade, 2002), we use the found examples to define a global image prior. 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 Freeman et al. (2000, 2002),

we use gray-scale images directly, which simplifies the overall algorithm. Also, our approach forces no causality in the image space, and the entire image is recovered as a whole.

3.1 A global MAP penalty

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

$$\varepsilon(\underline{\hat{f}}) = \|\mathbf{D}\underline{\hat{f}} - \underline{g}\|_2^2 + \lambda \sum_{i,j} \sum_{k \in \Omega[i,j]} \left\| \mathbf{R}_{[i,j]}\underline{\hat{f}} - \underline{x}_k^{[i,j]} \right\|_2^2. \tag{3}$$

In this functional the first term stands for the log-likelihood, with the assumption that the noise is white and Gaussian. The second term is the prior, and it is defined via the use of the examples found. The operator $\mathbf{R}_{[i,j]}$ extracts a block of $m \times m$ from the image $\underline{\hat{f}}$ that matches the footprint of the corresponding examples. The inner summation is done over all found NN, 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 Baker and Kanade (2002), were a multi-scale derivatives are matched, rather than direct gray-values, as done here.

A note about the deviation from the classic Bayesian point of view is in order here. One cannot claim that the above-proposed prior is indeed a prior for general images. Rather, it is a much narrower prior that seeks the recovery of the specific image in mind. Furthermore, this expression is heavily dependent on the measurements, from which we have obtained the high-resolution NN, 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.

Interestingly, the work reported in Freeman et al. (2000, 2002) also uses a MAP point of view in defining the reconstruction objective. Both works define expressions with similar forces that take into account the proximity between the low-resolution measurements and the database patches, and the agreement between high resolution neighboring patches between themselves. However, there are several important differences between the MAP expressions proposed that make the two methods distinct and substantially different. The expression posed in Eq. (3) is defined as a global one, considering the unknown image as a whole. No such treatment is shown explicitly in Freeman et al. (2000, 2002). Furthermore, the algorithm in Freeman et al. (2000, 2002) is targeting a search for the states of a network of probabilities that are assigned per each candidate example. This way, rather than concentrating on the true unknown $\underline{f}$, the work in Freeman et al. (2000, 2002) focuses on the network interpretation of the data. Discovery of the NN that survive a BBP algorithm leads to the formation of their solution. Our work, on the other hand, defines the unknown $\underline{f}$ and defines direct forces that should apply on it. Finally, we should note that while our expression confronts the chosen examples against the unknown image $\underline{f}$ directly, the work in Freeman et al. (2000, 2002) considers the inter-relations between pairs of overlapping patch examples.

The above-proposed penalty functional in Eq. (3) is using the local examples in order to define a global prior for the unknown image. However, unfortunately this

is not enough. In order 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 $\mathbf{D} = \mathbf{I}$), 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 Baker and Kanade (2002) 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

$$\varepsilon(\hat{\underline{f}}) = \|\mathbf{D}\hat{\underline{f}} - \underline{g}\|_2^2 + \lambda \sum_{[i,j]} \sum_{k \in \Omega[i,j]} w_k^{[i,j]} \left\| \mathbf{R}_{[i,j]}\hat{\underline{f}} - \underline{x}_k^{[i,j]} \right\|_2^2. \tag{4}$$

There are many ways to estimate/choose these weights. Indeed, the work in Freeman et al. (2000, 2002) offers a BBP as an attempt to weight the various examples. In this work we concentrate on a simplified and yet very effective case, where the weights are binary: '0' for a bad example and '1' for a good one. We have 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 we shall show next, the MAP functional itself will serve us in evaluating these weights.

3.2 Pruning irrelevant examples

Our algorithm starts with the assignment $w_k^{[i,j]} = 1$ for all i, j, and k. In a sequential process, we will prune one patch at a time, based on the following basic procedure:

1. For the current choice of weights, the minimizer of (4) is computed. We refer to the value of $\varepsilon(\hat{\underline{f}})$ at the minimum as a reference value.
2. Per each patch $\underline{x}_k^{[i,j]}$ with $w_k^{[i,j]} = 1$ (that is still active), we compute the optimal output image minimizing the modified MAP function
$$\varepsilon(\hat{\underline{f}}) - \lambda \left\| \mathbf{R}_{[i,j]}\hat{\underline{f}} - \underline{x}_k^{[i,j]} \right\|_2^2$$
(i.e., the original MAP with the omission of this patch). Clearly, the value of this penalty term is necessarily smaller than the reference one obtained in Step 1.
3. Among all these examined patches, we prune the one (by assigning $w_k^{[i,j]} = 0$) 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 each patch,

rather than re-compute 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. Second, 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 a 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 (4) 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 each 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 as some point.

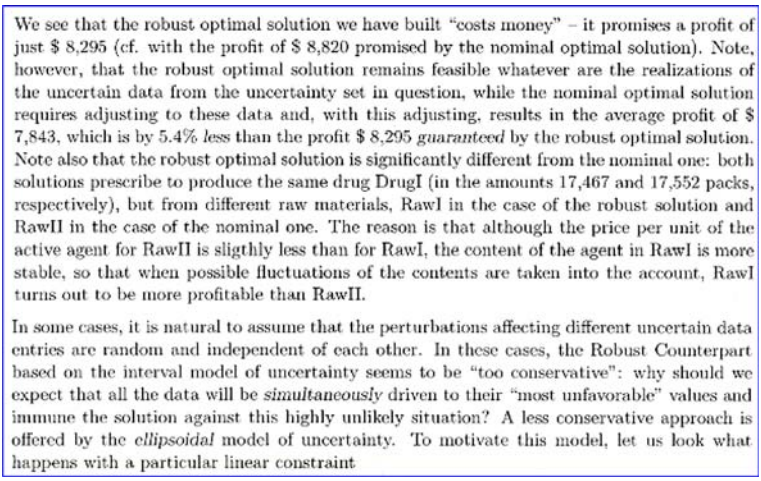
4 Results

We demonstrate the various reconstruction methods discussed above by showing the results on four experiments involving scanned documents with different content types — text, equations, and graphics. On the first example we intend to demonstrate various aspects of the algorithm, including the parameters chosen and the choice of features. On the remaining examples we show the reconstruction results and the stopping rule adopted.

Experiment #1 shows the reconstruction obtained for a text image. A database of scanned patches, containing $K = 197,000$ examples has been used, based on the image shown in Fig. 3. The degradation in this case includes a separable 3-tap blur with the kernel [0.25, 0.5, 0.25], a decimation by factor 2, and an additive Gaussian noise with $\sigma = 8$. In the various reconstructions demonstrated we have used $n = 5$ and $m = 9$. The NN are defined by the threshold $T = 6400$ in Eq. (2).

Figure 4 shows a test done on a portion of a text image. This figure show the original high-resolution image, the degraded one, and several reconstruction results. As can be seen, the pixel-based MMSE and MAP results are reasonably good, with some advantage to the MAP result due to its ability to handle outliers better. When turning to the proposed global scheme, the initial result should be similar to the pixel-based MMSE one, but due to the introduction of $\lambda = 1e-3$ (chosen manually and fixed throughout the experiments), it is somewhat better. As pruning takes place, the result improves substantially, due to the removal of 940 outlier examples out of the original 15,000 patches.

The stopping rule used here is the one described above (testing the dynamic range of $\Delta_k^{[i,j]}$). Figure 5 presents the reconstruction RMSE as a function of the pruning


We see that the robust optimal solution we have built "costs money" – it promises a profit of just $ 8,295 (cf. with the profit of $ 8,820 promised by the nominal optimal solution). Note, however, that the robust optimal solution remains feasible whatever are the realizations of the uncertain data from the uncertainty set in question, while the nominal optimal solution requires adjusting to these data and, with this adjusting, results in the average profit of $ 7,843, which is by 5.4% *less* than the profit $ 8,295 *guaranteed* by the robust optimal solution. Note also that the robust optimal solution is significantly different from the nominal one: both solutions prescribe to produce the same drug DrugI (in the amounts 17,467 and 17,552 packs, respectively), but from different raw materials, RawI in the case of the robust solution and RawII in the case of the nominal one. The reason is that although the price per unit of the active agent for RawII is sligthly less than for RawI, the content of the agent in RawI is more stable, so that when possible fluctuations of the contents are taken into the account, RawI turns out to be more profitable than RawII.

In some cases, it is natural to assume that the perturbations affecting different uncertain data entries are random and independent of each other. In these cases, the Robust Counterpart based on the interval model of uncertainty seems to be "too conservative": why should we expect that all the data will be *simultaneously* driven to their "most unfavorable" values and immune the solution against this highly unlikely situation? A less conservative approach is offered by the *ellipsoidal* model of uncertainty. To motivate this model, let us look what happens with a particular linear constraint

Fig. 3 Experiment #1 — a text image: patches of size 9×9 taken form this image form the example database for Experiment 1 of text 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 $K = 197,000$ examples in the DB

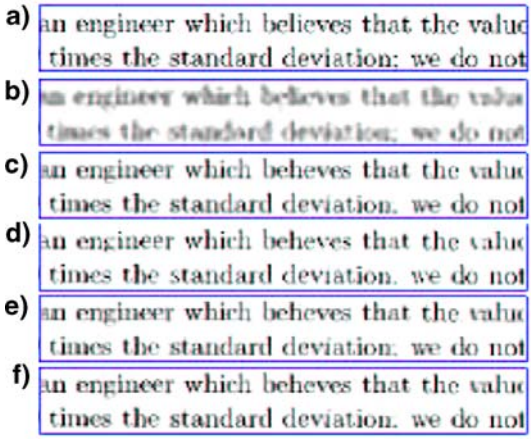


Fig. 4 Experiment #1 — a text image: (**a**) Original image; (**b**) degraded image - bi-cubic interpolation; (**c**) MMSE reconstruction (MSE = 314); (**d**) MAP reconstruction (MSE = 300); (**e**) initial proposed reconstruction, m = 9, 15,000 candidate patches (MSE = 313); (**f**) reconstruction after 940 pruning iterations (MSE = 258)

stages. Overlayed on this graph is the stopping rule curve, showing the dynamic range of $\Delta_k^{[i,j]}$ as a function of the pruning stages, and the threshold it meets in the stopping rule. As can be seen, while not perfect, the proposed stopping rule does succeed in stopping the algorithm in the vicinity of the best MSE.

Figure 6 shows how the results are affected by the change of the patch sizes in the low-resolution (n) and the high-resolution (m). Per each choice of (n, m) the basic algorithm (forming the initialization) is run and the resulting MSE is recorded. Instead of presenting the performance as a function of n and m, it is described as several curves parameterized by n, and as a function of the relative value of m. Each value of n implies a maximal patch size in the high-resolution, which contains all the high-resolution pixels influencing the low-res. patch. This size is defined as *patchSize*.

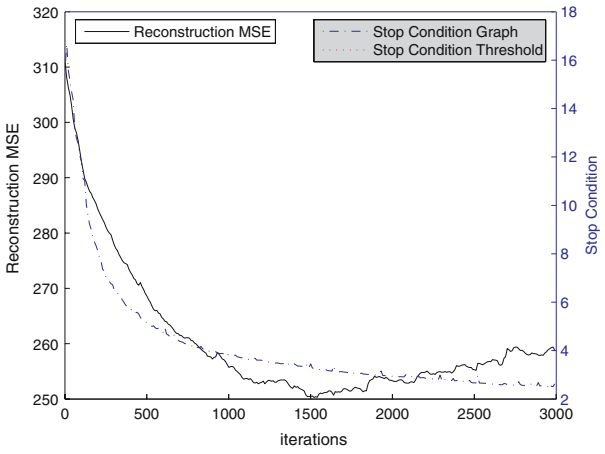


Fig. 5 Experiment #1 — a text image: the pruning effect on the reconstruction MSE

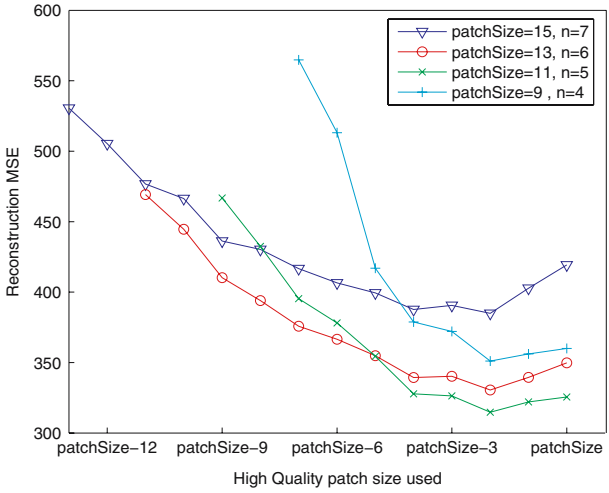


Fig. 6 Experiment #1 — a text image: performance comparison for different high- and low-resolution patch sizes. The value of n defines the low-resolution image patch size, *patchSize* stands for the maximal high quality patch size with respect to n. The various graphs show the influence of the number of pixels used from the high quality patch on the overall MSE

As anticipated, for the case $n = 5$ used in the above experiments, the choice $m = 9$ is the best option.

The next experiment relies on the same database, and also deals with text content, handling a different portion of the test image. Figures 7 and 8 show the reconstruction results and the stopping rule behavior for this different portion of a test image. As can be seen, the results and the conclusions from these figures are generally the same as above.

Figures 9–11 show the results of the third experiment, involving an image containing a formula portion. Figure 9 gives the image from which examples have been drawn. In this experiment there are $K = 82,000$ such examples. The degradation assumed in this experiment includes the same blur, followed by a scale-down factor of 3. The reconstruction results are shown in Fig. 10. In this experiment the pixel-based MMSE estimate is substantially better than the MAP one, which may be explained by a lack of string outliers. As before, the pruning algorithm leads to a better MSE

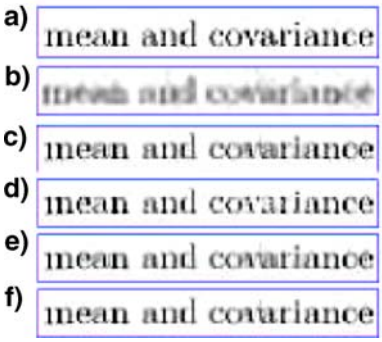


Fig. 7 Experiment #2 — (**a**) text image: (**a**) original image; b Degraded image — bi-cubic interpolation; c MMSE reconstruction (MSE = 420); (**d**) MAP reconstruction (MSE = 407); (**e**) initial proposed reconstruction, m = 9, 2,948 candidate patches (MSE = 419); (**f**) reconstruction after 300 pruning iterations (MSE = 339)

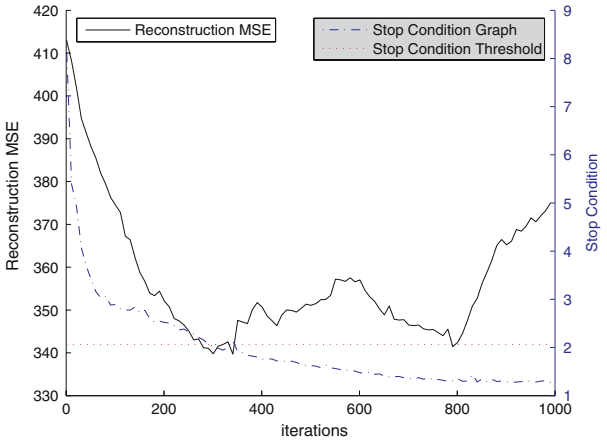


Fig. 8 Experiment #2 — a text image: the pruning effect on the reconstruction MSE

$$P(x \mid y) = \frac{1}{Z}\prod_{(ij)}\psi_{ij}(x_i, x_j)\prod_{i}\phi_i(x_i, y_i)$$

$$\psi_{ij}(x_i, x_j) = \exp\left(-\frac{d_{ij}(x_i, x_j)}{2\sigma^2}\right)$$

$$m_{ij}(x_j) = \sum_{x_i}\phi_{ij}(x_i, x_j)\prod_{k\neq j} m_{ki}(x_i)\phi_i(x_i, y_i)$$

Fig. 9 Experiment #3 — an image with formula: $K = 82,000$ patches taken form this image form the examples database for formula image reconstruction. The scale factor used is 3, and the patch sizes are $n = 5,\ m = 15$

result. Figure 9 presents the stopping rule, again successfully leading to near optimal MSE performance.

The fourth experiment shows reconstruction results related with an image containing a portion of a graph. Fig. 12 shows the training image, Fig. 13 shows the reconstruction results in this case, and Fig. 14 demonstrate the stopping rule behavior. The results are generally the same as in previous experiments, showing the strength of the pruning technique proposed.

We conclude the results section with an experiment that returns to the text content images, and the database as defined for experiments #1 and #2. Our aim is to show that for the content dealt with here, the best approach is the use of the raw information

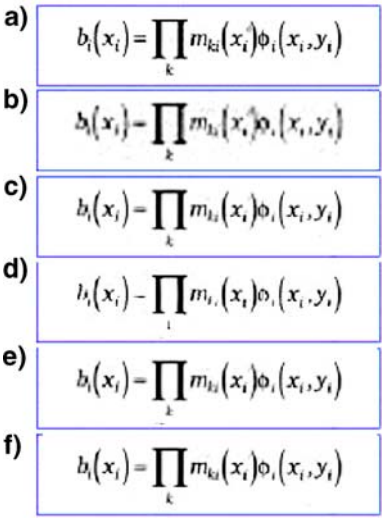


Fig. 10 Experiment #3 — an image with formula: (**a**) Original image; (**b**) Degraded image - bi-cubic interpolation; (**c**) MMSE reconstruction (MSE = 435); (**d**) MAP reconstruction (MSE = 587); (**e**) Initial proposed reconstruction, 16,000 candidate patches (MSE = 433); (**f**) Reconstruction after 4,710 pruning iterations (MSE = 319)

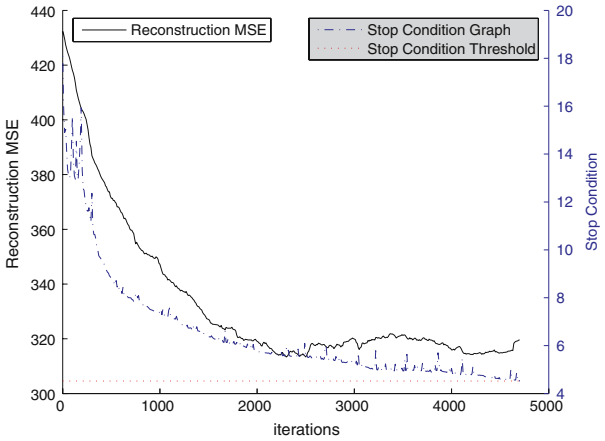


Fig. 11 Experiment #3 — an image with formula: the pruning effect on the reconstruction MSE

and not high/medium frequencies, as often deployed on natural images. Figure 15 shows how the training is done, when aiming to use a pre-process of removing the low-frequencies. The low-pass filter we have used here is a simple 5×5 Gaussian filter of width $\sigma = 1.5$. Figure 16 presents the way this database is deployed to the actual reconstruction. Figure 17 shows the reconstruction results with a low-pass filter and without it (our method), for a noisy case and noiseless one. As can be seen, while all results are good, better performance is obtained using the plain raw data, as we recommend.

5 Conclusions

The concept of using image examples in reconstruction problems is appealing because of its ability to bypass the need for guessing the image prior. There are many ways to

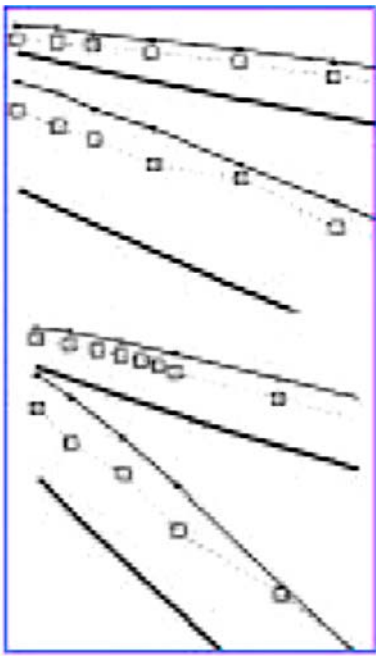


Fig. 12 Experiment #4 — a portion of a graph: patches taken form this image form the example database for the graph image reconstruction,with a scale factor of 3, patch sizes $n = 5$, $m = 15$, and with $K = 110,000$ examples

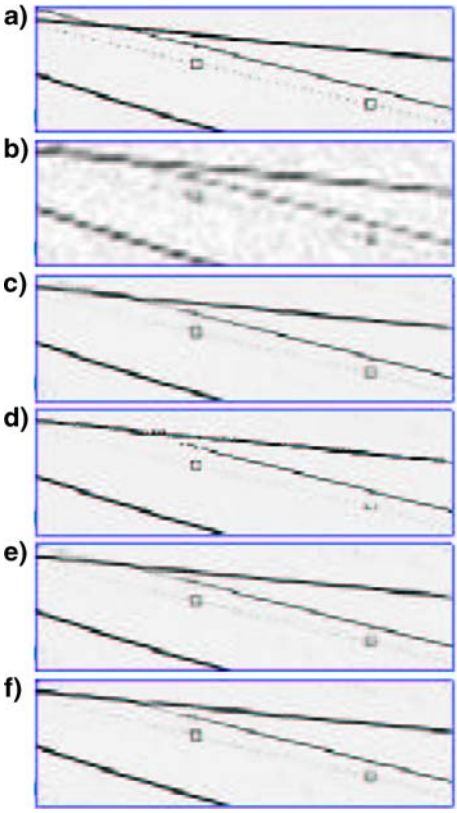


Fig. 13 Experiment #4 – a portion of a graph: (**a**) Original image; (**b**) Degraded image - bi-cubic interpolation; (**c**) MMSE reconstruction (MSE = 312); (**d**) MAP reconstruction (MSE = 392); (**e**) Initial proposed reconstruction, with 22,000 candidate patches (MSE = 305); (**f**) Reconstruction after 1,100 pruning iterations (MSE = 268)

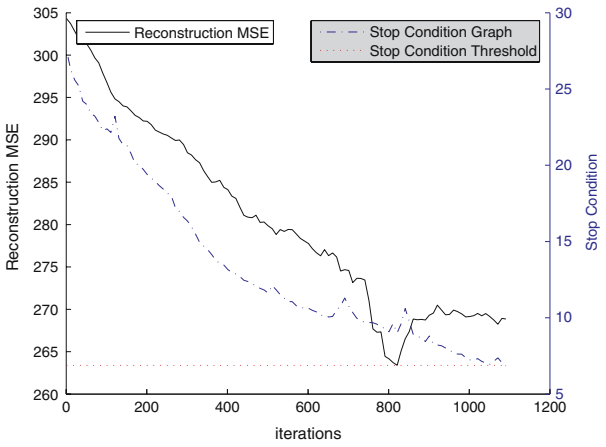


Fig. 14 Experiment #4 — a portion of a graph: the pruning effect on the reconstruction MSE

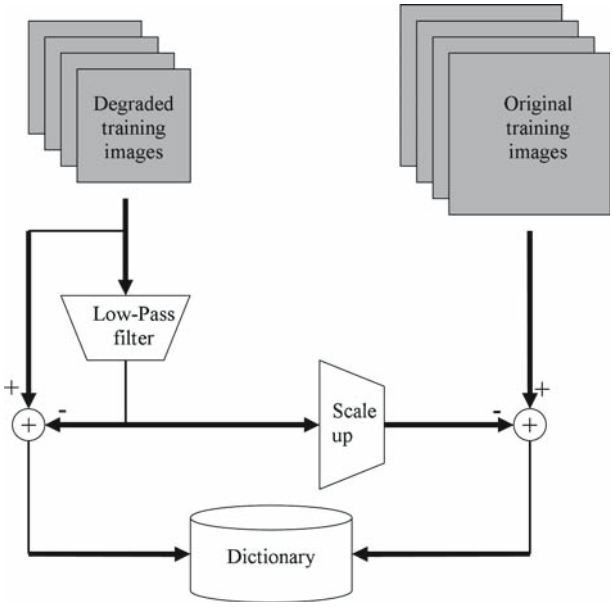


Fig. 15 A block diagram of the training algorithm when using the higher frequencies as features

incorporate examples in such inverse problems. In this work, we concentrated on a simple yet effective algorithm that uses plain gray-scale image patches. This algorithm sweeps through the low quality image, finding per each location several high-resolution patches that may fit it. Those are pruned by a global MAP penalty functional, that is used also for the reconstruction. We have described a computationally effective pruning method that consistently improves the outcome, and we have demonstrated this algorithm on scanned document images successfully.

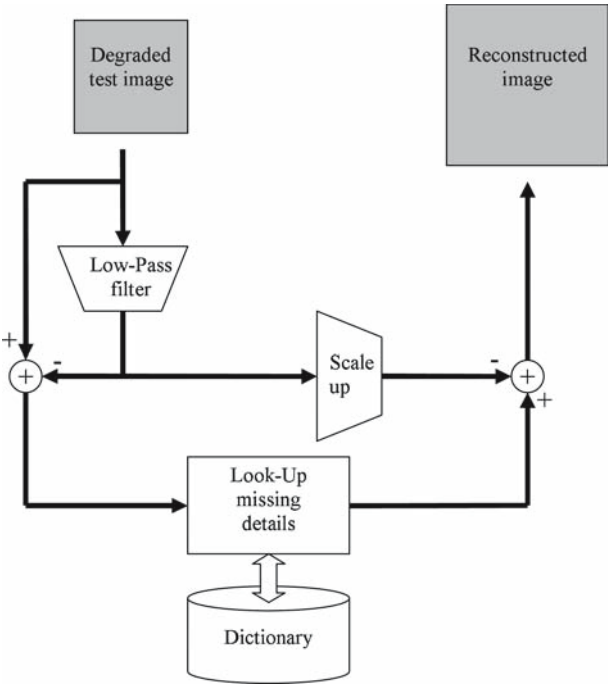


Fig. 16 A block diagram of the reconstruction algorithm using the high-frequencies features

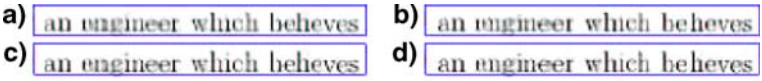


Fig. 17 Comparison between an algorithm that operates on raw data and one that uses high frequencies only: (**a**) Scale factor is 2, noise power is $\sigma = 8$, using high frequencies (MSE = 406); (**b**) Scale factor is 2, noise power is $\sigma = 8$, using raw data (MSE = 367); (**c**) Scale factor is 2, no noise, using high frequencies (MSE = 354); (**d**) Scale factor is 2, no noise, using raw data (MSE = 322)

References

Aharon, M., Elad, M., & Bruckstein, A. M. (2006a). The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing, 54*(11), 4311–4322.

Aharon, M., Elad, M., & Bruckstein, A. M. (2006b). On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *Journal of Linear Algebra and Applications*, *416*(7), 48–67.

Baker, S., & Kanade, T. (2002). Limits on super-resolution and how to break them. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(9), 1167–1183.

Buccigrossi, R. W. & Simoncelli, E. P. (1999). Image compression via joint statistical characterization in the wavelet domain. *IEEE Transactions on Image Processing, 8*(12), 1688–1701.

Chen, S. S., Donoho, D. L., & Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM Review, 43*(1), 129–159.

Criminisi, A., Perez, P., & Toyama, K. (2004). Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing, 13*(9), 1200–1212.

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

Efros, A. A., & Leung, T. K. (1999). Texture synthesis by non-parametric sampling. *In Proceedings of the IEEE International conference on computer vision* (ICCV'99), Corfu, Greece.
Engan, K., Aase, S. O., & Hakon-Husoy, J. H. (1999). Method of optimal directions for frame design. *IEEE Internationall Conference on Acoustics, Speech, and Signal Processing, 5*, 2443–2446.
Freeman, W. T., Jones, T. R., & Pasztor E. C. (2002). Example-based super-resolution. *IEEE Computer Graphics And Applications, 22*(2), 56–65.
Freeman, W. T., Pasztor, E. C., & Carmichael, O. T. (2000). Learning low-level vision. *International Journal of computer Vision, 40*(1), 25–47.
Friedman, J. H., Bentley, J. L., & Finkel, R. A. (1977). An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software, 3*(3), 209–226.
Haber, E., & Tenorio, L. (2003). Learning regularization functionals. *Inverse Problems, 19*, 611–626.
Jaynes, E. T. (1982). On the rationale of maximum-entropy methods. *IEEE Proceedings, 70*(9), 939–952.
Nakagaki, R., & Katsaggelos, A. K. (2003). VQ-based blind image restoration algorithm. *IEEE Transcations on image Processing, 12*(9), 1044–1053.
Olshausen, B. A., & Field. D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research, 37*, 311–325.
Roth, S., & Black, M. J. (2005). Fields of experts: A framework for learning image priors. *IEEE Conference on Computer Vision and Pattern Recognition, 2*, 860–867.
Rudin, L., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, 60, 259–268.
Sochen, N., Kimmel, R., & Bruckstein, A. M. (2001). Diffusions and confusions in signal and image processing. *Journal of Mathematical Imaging and Vision, 14*(3), 195–209.
Wei, L.-Y., & Levoy, M. (2000). Fast texture synthesis using tree-structured vector quantization. *Proceeding in SIGGRAPH 2000 (New Orleans, Louisiana, July 23–28, 2000). In Computer Graphics Proceedings, Annual Conference Series, 2000*. ACM SIGGRAPH, pp. 479–488.
Weissman, T., Ordentlich, E., Seroussi, G., Verdu, S., & Weinberger, M. J. (2005). Universal discrete denoising: Known channel. *IEEE Transactions on Information Theory, 51*(1), 5–28.
Zhu, S. C., & Mumford, D. (1997). Prior learning and Gibbs reaction-diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligences, 19*(11), 1236–1250.

Biographical sketches

Dmitry Datsenko received the B.Sc. and M.Sc. degrees from the Department of Computer Science in 2001 and 2006, respectively. During his studies towards B.Sc. degree he worked at Intel. From 2001 to 2004, he served in the Israeli Air Force. Currently Dmitry works at Mediguide, specializing in the fields of image processing and computer vision for medical imaging.

Michael Elad received his B.Sc, M.Sc. and D.Sc. from the department of Electrical engineering at the Technion, Israel, in 1986, 1988 and 1997 respectively. From 1988 to 1993 he served in the Israeli Air Force. From 1997 to 2000 he worked at Hewlett-Packard laboratories as an R&D engineer. From 2000 to 2001 he headed the research division at Jigami corporation, Israel. During the years 2001 to 2003 Michael was a research associate with the computer science department at Stanford university (SCCM program). Starting on September 2003, Michael is with the department of Computer science, the Technion, Israel Institute of Technology (IIT) as an assistant professor.

Michael Elad works in the field of signal and image processing, specializing in particular on inverse problems, sparse representations and over-complete transforms. Michael received the Technion's best lecturer award four times (1999, 2000, 2004, and on 2005). Michael is also the recipient of the Guttwirth and the Wolf fellowships. He is currently serving as an associate editor for IEEE Trans. on image processing, and EURASIP signal processing journals.