Filling In the Black-outs of Various Types of Signals

Final Report

Doru-Cristian Balcan
Dept. of Computer Science
Carnegie Mellon University
dbalcan@cs.cmu.edu

Abstract

The problem that we address in this paper is that of filling in missing data ("black-outs") in uni- or multidimensional signals, by determining and using the intrinsic structure of the signals. A typical example would be that, given many co-evolving time sequences, some of which may temporarily black-out, we estimate the black-out values.

Various particular cases of the more general problem have been solved in different frameworks using several different approaches and assumptions. For instance, the black-out filling could be regarded as a problem of prediction, smoothing, or filtering, each of these corresponding to a particular case of using Kalman filters. One problem is that in general, the difficulty of treating a particular signal increases with the noise level (also regarded as the "distance from smoothness"). In order to overcome this obstacle, additional information about the underlying signal structure is very useful. Representing the signal structure is well recognized as one of the main applications of wavelets in signal processing. As an alternative method, wavelets make use of a predict-correct mechanism, and in addition they could bring in the advantages of multiresolution/multiscale signal analysis. Another relatively new and versatile method suited for learning signal structure is Independent Component Analysis. The advantage of using this method over wavelets is that instead of representing the signal by a set of fixed basis functions (i.e. "sparse coding"), we simply learn the best code *directly* from the data. Although all these approaches may seem distinct and even unrelated to each other, we can regard every one of them as corresponding to a particular case of Gaussian process model. In this respect, the original problem we try to solve, can be explicitly treated within this larger, general framework of Gaussian processes, as a nonlinear regression problem.

The application we chose for illustrating our study involve the treatment of single images, although the same basic principles could be applied to any number of signals, of the same (arbitrary) dimension. We compared the results obtained by different methodologies and propose further directions of investigation.

DRAFT

1 Introduction

In this paper, we are concerned with the problem of filling the "black-outs" in a variate range of signals. In the following, we will introduce a formal definition of the problem. Moreover, we will point out four particular cases, which we think are not only theoretically interesting in themselves, but also very likely to be encountered in real world applications.

1.1 Problem definition

Let $d, n \in N^*$ be two positive integers, d being called "the dimension". Let S be a subset of $\mathbb{R}^d$, and y be some function $y : \mathbb{R}^d \to \mathbb{R}^n$ such that for every $t \in S$, we know the exact value $y(t)$. Our goal is to interpolate y, over the set D, where $S \subseteq D \subseteq \mathbb{R}^d$. That is, we want to infer the values of the function y in every element of D, which is a superset of S, using only the values of y on S.

Throughout the paper, we will use the term "interpolation" to cover both the problem of "regression" (fitting a curve through noisy data) and the task of fitting an interpolant function that passes exactly through the given data points. The appropriate sense shall be derived from the context.

For instance, here are four particular cases which we may encounter.

1. If $d = n = 1$, then we can regard our points as moments in time, for instance. The problem in this case would first require to identify D, the set of the moments of interest, that is the set of points in time we wish to specify a value for y. The next step is to infer the values of y on D, using mainly the values of signal y on S.

   Depending, on the task, set D could be for instance an interval bounded by two points in S, or a union of such intervals, or the set of all future moments, or even the entire complement of set S. If the moments of interest are "in the future" we will use the term "prediction", although the conventional name for this is "extrapolation". If the points ("moments") where y is not known are interleaved with the points in S, then this would correspond to "filling the black-outs" in our time sequence D (e.g., see [19]).

2. If $d = 1 < n$, then we can regard our points as samples of n co-evolving time sequences. In this case, the problem would be essentially the same as the previous case, namely to predict or to fill in the "black-outs" in multidimensional case.

3. A particular sub-case of case 2 would be to infer the n-dimensional vector $y(t)$, where $t \in D$, when we know $n_1 < n$ of its entries. This is just another way of saying that we use the n_1 known values of co-evolving sequences at moment t in order to predict the unknown values of the other sequences.

DRAFT

4. If $d = 2$ and $n = 1$, then it is possible to regard our data as points in a plane, and to consider their associated values as pixel values. Then our problem could translate as "filling in the missing pixels" of an image.

1.2 Applications

The possible applications of this problem are very diverse. We mention a few of them for each of the four cases of the problem listed above.

1. Speech processing, financial data prediction - the black-out means interruptions in the data flow, possibly due to perturbance of the signal (in the case of speech), or it refers to the future values of a time series, for instance the total sales figure in every week (in the other case).

2. Currency values prediction, music CD repair/restore - dealing with multiple time sequences, out of which all could be blacked-out in the same time, for instance the left and right audio channels of a scratched CD can be considered as interrupted in the portions physically corresponding to the medium deterioration.

3. Weather forecasting, autonomous navigation - due to various causes, multiple time sequences could be interrupted , but not necessarily all at once; for instance, not all temperatures of all the meteorologic stations in a city-wide area are necessary to predict the weather forecast for the next day (some devices may be broken, or).

4. Image restoration, fax data transmission - due to the noise of the sensors, due to the transmission or storing medium, images may appear as having some of their pixels missing; this is sometimes the case of fax images, but it could also happen in image acquisition.

2 Survey

The problem we address has been around for a long time. The available solutions differ by approach, speed, and generality.

Time series forecasting is a subject of major interest to many research fields. In statistics and signal processing, valuable tools for forecasting and time series processing were developed. One of the "classic" methodologies for forecasting is Box-Jenkins's Auto-Regressive Integrated Moving Average (ARIMA) [3]. ARIMA belongs to the class of linear time-series forecasting, i.e., it postulates a linear dependency of the future value on the past values. Non-linear forecasting methods constitute an open research area [19].

A famous solution to the discrete-data linear filtering problem was given in 1960 by R. Kalman [6], [21]. This is a recursive, computationally efficient technique used for prediction, filtering and smoothing of time sequences, based on the least-squares method. Its prevalence over Wiener filtering is due to its incremental nature. More specifically, in order to realize a value estimation it does not require the ("batch") processing of all the data, but only the value of the "one-step before" estimate. It became one of the most prominent tools of control theory, being used intensively in autonomous navigation and Geographic Information Systems [21]. Still, this method is much more suited for types of signals with relatively reduced noise level, as compared to the variance of the signal itself.

MUSCLES ("MUlti-SequenCe LEast Squares"), a fast method for analyzing co-evolving time sequences, is presented in [22]. This method allows estimation and forecasting of missing/delayed/future values, and is also suited for quantitative data mining and for outlier detection.

A remarkable tool in the way time series/signal processing is carried out, is represented by spectral methods. Thus, instead of ignoring frequency content of a signal, we can use this to gain information about the behavior of the signal.

Spectral methods have been increasingly applied to pattern matching and data mining [4], [17], [18], [20]. Wavelets have offered a great help in signal processing, especially due to their multi-scale/multiresolution characterization of signals. Fast encoding schemes have added to their efficient signal encoding properties and practically they are now the standard method in Signal Processing. The application of wavelet techniques in database-related research are also very successful ([10], [12]).

A relatively new approach in signal representation, processing and coding is Independent Component Analysis. The advantage ICA has over wavelets is given by the "adaptive" coding of signals, as opposed to using "fixed"/off-the-shelf (wavelet) basis functions. Moreover, because of this feature, with ICA it is possible to learn efficient codes of the large class of natural signals, in a very similar way to how the brain does it [11].

The process through which the ICA basis is derived ("learned") involves the optimization of both the independence of the coefficients, and of the coding efficiency, within the limits of the linear model. Thus, various studies have reflected that the ICA code is the one having the lowest entropy within the family of linear model codes [9], [7].

3 Proposed Method

From a data mining point of view, a particularly interesting solution to the problem of "filling the black-outs" can be obtained with probabilistic tools, and more specifically, with Bayesian modeling. In the following, we present shortly the Bayesian methodology and the solution it offers to our problem [9].

Consider a digital signal y, as an array whose entries are either known or unknown. The latter represent the actual "missing values". Since we can process only one part of the whole signal at a time, we shall proceed by dividing it into fixed size windows. Naturally, each such window has its own known

DRAFT

and unknown values.

We shall denote such a signal window by x. Although our ultimate goal was to fill in the black-outs in y, let's observe that we can reach our goal by choosing a set of windows which cover all the unknown spots in y, and by filling the black-outs in each window. In other words, we solve the original problem by dividing it into smaller, identical problems.

A simple, yet powerful model for signal representation could be provided by the linear model. Say we have a vector ("signal window") x, regarded as a sample from the total set of signal windows. We want to represent x as a linear superposition of the elements of a spanning independent set (a "basis"). More precisely, given a matrix A whose columns form a basis of the vector space, we would like to find a vector s of coefficients such that $x = As$. [1]

If we choose smooth basis functions, then any (weighted) sum of them will also be smooth. One problem that might appear is that our signal may in fact be very noisy (that is, it may present sudden, strong variations, and thus it may look very "bumpy"). In this case, it would be neither possible, nor reasonable to write x as a sum of smooth functions.

Instead, we'll settle for a more reasonable form of representation, which could be more computationally demanding to determine, though. The linear *noisy* model can be simply written as

$$x = As + \epsilon,$$

where ϵ is a vector called "noise", "perturbance", or "error". One question that can be asked is: how could we infer the right representation from our data, even if it contains a significant amount of noise, and what are the assumptions, that permit us to actually solve this problem?

Finding the exact values of the coefficients s may be very tricky (in fact it is an ill-posed problem), due to the fact that the noise ϵ is unknown. Still, we could infer s from x, by maximizing the conditional probability distribution of the coefficients s, given the signal x and the basis A. From Bayes rule, we can express this as:

$$P\left(s|x, A\right) \propto P\left(x|A, s\right) P\left(s\right).$$

The first term in the right-hand side is the likelihood of the signal under the model, for a given state of the coefficients. Assuming $\epsilon \sim N\left(0, \sigma^2 I_d\right)$ (which seems to be a reasonable and widespread assumption), we have: $P\left(x|A, s\right) = C_\sigma e^{-\frac{||x-As||^2}{2\sigma^2}}$.

The second term is the prior probability distribution of the coefficients. In order to specify a "good" prior for the coefficients, we need some information about the type of signal we are treating.

[1] Further in the paper, we will always consider x as a vector, since we can bring a signal window of any dimension, and the corresponding basis functions, to a vector form; this will obviously not interfere with the linear model we describe. Moreover, we shall refer to x as to "the signal", since we obviously can describe the original signal, if we know how to describe all windows of a fixed size.

DRAFT

For instance, by examining the spectrum of signals belonging to the class of natural signals, it has been shown that the coefficients corresponding to the usually follow a generalized Gaussian distribution. More specifically, for natural images, this is very close to the Laplacian distribution [1], [2], [16].

We shall search for ŝ, the MAP (maximum a posteriori) estimation of s, that is we want to find the value of s which maximizes the posterior distribution $P(s|x, A)$.

What gives the strength of this approach is its versatility. That is, for any kind of basis A (be it DCT, wavelet, ICA, PCA, etc.), and for any kind of signal (one-, or multi-dimensional), the proper choice on the prior distribution of the coefficients, that is, the representation of x under the model, reduces to essentially the same optimization problem.

Let's see now why and how this method can be applied to our problem, that is what is the connection between Bayesian denoising and black-out filling. The basic idea is that we can consider the missing values as being extreme with respect to the range of all possible values of the signal in any location (for instance, if we deal with 8-bit grayscale images, we assign 255 to any location of a missing pixel). Of course, in this way there will appear *big* discontinuities (large variations) in the signal. In fact, we will consider that the noise level in these particular positions is arbitrary large (or, virtually infinite), while for all the other positions we have a finite (estimable) amount of noise. Because of this, when computing the likelihood, we will include in the sum of squares, only those components of the vector x that correspond to *known* signal values. By constraining the coefficients to fit the signal with the basis functions in the known positions, we will implicitly find some values for the "black-out" positions [9], [15], [14].

In order to show how well Bayesian denoising is able to perform the interpolation task, we focused our investigation on what we think there are the best suited methodologies in Signal Processing, for describing signal structure. The reasons for which we selected these, are explained in the following.

3.0.1 Wavelets

A hot topic in the past 25 years, Wavelet signal analysis has extended its influence into almost any field that deals with signals or series [18], [4], [17], [20]. The advantage of having a better representation than just the "simple"/raw signal has lead to efficient compression schemes, to a superior control over the "underlying" structure of observed processes or, in our case, to a better capability of "discovering" missing values.

We start off with a basis of wavelet functions, for instance a Daubechies or Haar basis. We apply the Discrete Wavelet Transform (DWT) induced by this basis on a signal window, in order to find the corresponding coefficients of the basis functions. Of course, our image being corrupted by noise and by the missing pixels, the coefficients we obtain will need "readjustment". The criterion we have for optimization is MAP, as described before. For the optimal value of s, we apply the inverse transform

DRAFT

and get the corresponding image. This wavelet-based method is also known as Bayesian Wavelet Coring [14].

A big advantage of the Wavelet transform, over other transforms from the family of spectral transforms is the ability of representing multiscale/multiresolution versions of a signal. Thus, the underlying structure could be much more efficiently and more reliably recovered.

One reserve towards the fidelity of representation (or, coding efficiency) obtained by using wavelets, is generated by the fact that the basis functions are always fixed. Thus, for all signals of the same dimension, and for the same wavelet family, the description relies only on the wavelet coefficients. From an Information Theory point of view, this type of coding would not give a code with the lowest possible entropy. But considering that in many applications their performance with respect to time (namely the linear complexity of the coding) and space (the compression ratio) is sufficiently good for practical purposes, this downside is barely remembered.

One other method to compare against ours, was suggested by [12]. There, after applying the DWT, Spiros Papadimitriou uses ARIMA in order to predict the future values for the coefficients. In fact, any other prediction method could be used for this purpose, but for performance reasons (speed) this one was preferred. By applying the *inverse* DWT to all the new wavelet coefficient vectors, he obtains a vector containing the desired predicted value.

By using the Bayesian Denoising method, we do not intend to fill in only single points at a time, but we aim for multi-point fill, and we want our method to work, even if the black-out zone is much vaster than the working window. Thus, in order for the method to be feasible, we shall impose that every signal window we treat for black-out fill, contains enough known (or already "filled in") values to allow an appropriate fitting. For safety reasons, we choose the windows such that the minimum ratio between the number of known and unknown value points in a signal window should be 35%. However, for high-dimensional signals, the constraints/connections between neighboring positions are multi-directional, and so they are much more powerful than for time series.

If the black-out entries are more or less uniformly distributed across the original signal, then practically for each signal window we choose, the ratio of known vs. unknown points, is about the same. It is pretty easy to observe that the problem is more difficult to solve if more data are missing, and also

3.0.2 Independent Component Analysis

As for the wavelet approach, we used a basis for better signal representation. The difference from the wavelet basis is that the ICA basis functions are derived ("learned") directly from the signal, instead of being fixed functions. The advantage is that in this way, we have chances of finding a (provable [9]) better code for the signal (that is, a code with a smaller entropy) than a code obtained with a fixed function basis. (Moreover, it has been shown that ICA basis functions for natural images and sounds

DRAFT

have biological correspondence in the impulse responses of the neurons in the human cortex [11].)

One apparent disadvantage of the ICA approach as compared to the wavelet approach might be that learning the basis usually takes time, unless a fast-converging algorithm is used. This might prohibit online data processing (or in our case, filling in the missing values "as soon as they appear"). This problem can be overcome partially by learning the basis in advance for the particular type of signal, and using this basis as is for any signal in that class.

We enriched this ICA framework by embedding multiresolution into the model. That is, obtained basis functions for different resolution versions of the original signal. This way, the ICA ("learned") basis would "cancel" one advantage that wavelets had until now. This idea has not been published before, and the results we obtained until now are quite promising [8]. After obtaining the basis as we described above, we used it in a similar way as the wavelet basis. To give an example, we show hereby a picture of a multiresolution ICA basis allowing three different resolution levels.

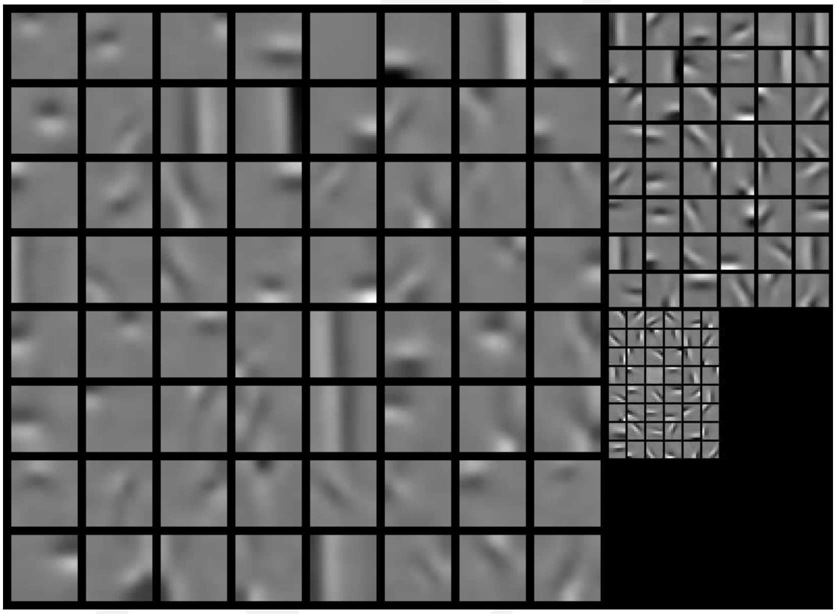

Figure 1: Multiresolution ICA basis with 3 levels.

4 Experiments

In the Bayesian Denoising approach, we needed to obtain the basis functions matrix corresponding to each chosen linear transform. We employed for this purpose Eero Simoncelli's "matlabPyrTools" library ([13], [14]). In the case of wavelets, we obtained the chosen basis by building the corresponding wavelet pyramid and then use its data structure to derive the basis. As for the ("multiresolution") ICA case, we wished to develop a division of the signal spectrum, and get a version of the signal that contains the signal's frequency content in each subband. In order to learn the ICA basis, we used a variant of the FastICA algorithm [5]. We performed the optimization step of the algorithm (that is, the maximization of the likelihood of the noisy linear model), that is for the maximization of the model (log-)likelihood, we used the unconstrained optimization functions in Matlab's Optimization Toolbox.

For evaluation, we performed experiments in several of the application areas mentioned in the Introduction. We considered that the same approaches could just as successfully be employed for all the other applications.

4.1 Image Restoration

We will look at the problem of filling in missing pixels in a digital image. For example, we can imagine a digital camera that has some damaged sensors so that there is missing data in the resulting image. The most appropriate measure to use here in order to evaluate how our method performs is by computing the signal-to-noise ratio SNR.

We used a set of 12 images, representing relatively different types of natural images: landscapes, portraits, medical images. From every image, we removed a percentage p of the pixels, uniformly randomly, and we kept around a "black-out bitmap", that is a matrix of the same dimensions as the image, having its entries either 0 (in the positions corresponding to the missing pixels) or 1 (everywhere else). The next step is to divide the images into blocks, each of them having the size of a basis function. For instance, we could use 8×8, 16×16 or 32×32 basis functions. Then, writing each patch as a superposition of the basis functions, and maximizing the likelihood of the model, measured on the *known* entries in the patch, we find a set of coefficients. These coefficients, together with the basis functions, will *implicitly* determine certain values for the black-outs of the patch. The result is a "filled" patch.

If we perform the procedure described above on every patch, and then we place the filled patches in their original positions in the image, then we get the reconstructed image.

In the following we display several examples of the performance of the method as applied on natural images. We show the results obtained on subimages of various sizes extracted from our image set. We display the typical results we got for, four types of basis: DCT, Haar, (classic, "plain") ICA and multiresolution ICA, with 1 multiresolution level. Also, we varied the number of pixels drawn out of the images, in a range varying from 15% to 50%. Every final result includes the overall SNR value.

The performance of the method is pretty remarkable, since for the human subject is almost impossible to figure out the contents of an image with about 20% (uniformly distributed) missing pixels. For instance, for the "clock" image, not only that the denoising manage to get the right information, but even see the digits on the clock appear to be reconstructed pretty fairly.

DRAFT

Figure 2: The image set used for Bayesian Denoising.

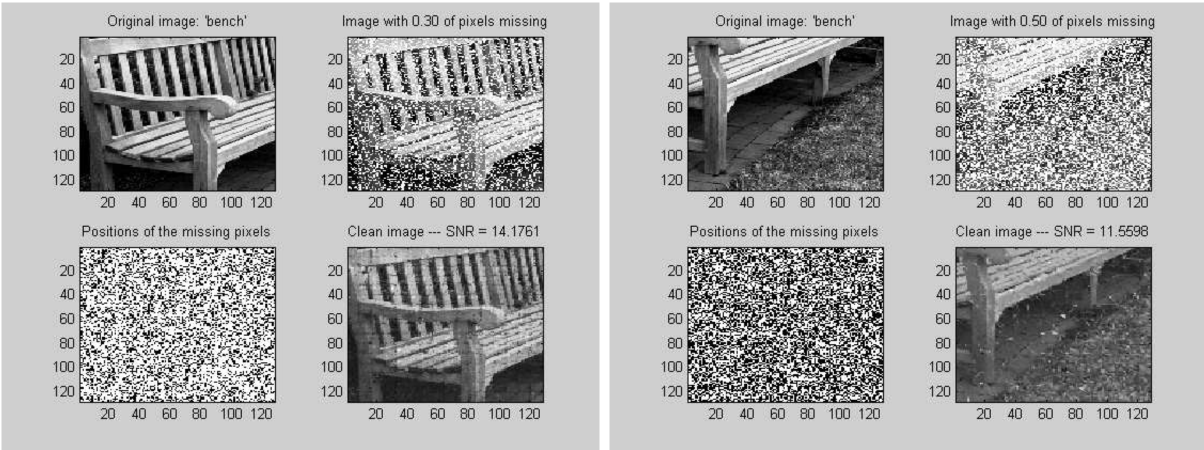

Figure 3: Example of Image restoration. Left: 128×128 subimage from "bench", 8 size DCT basis, 30% pixels missing; Right:128×128 subimage from "bench", 8 size ICA basis, 50% pixels missing

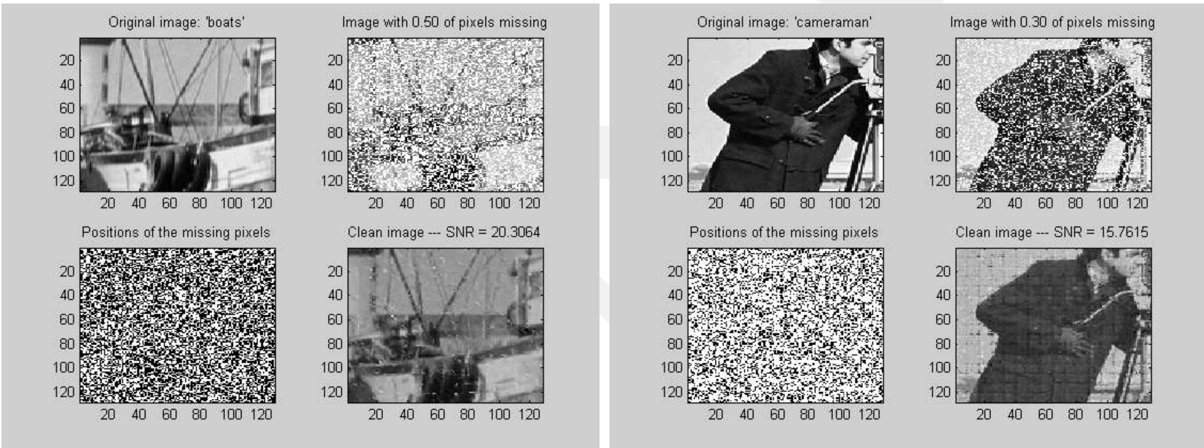

Figure 4: Example of Image restoration. Left: 128×128 subimage from "boats", 8 size ICA basis, 50% pixels missing ; Right:128×128 subimage from "cameraman", 8 size DCT basis, 30% pixels missing

DRAFT

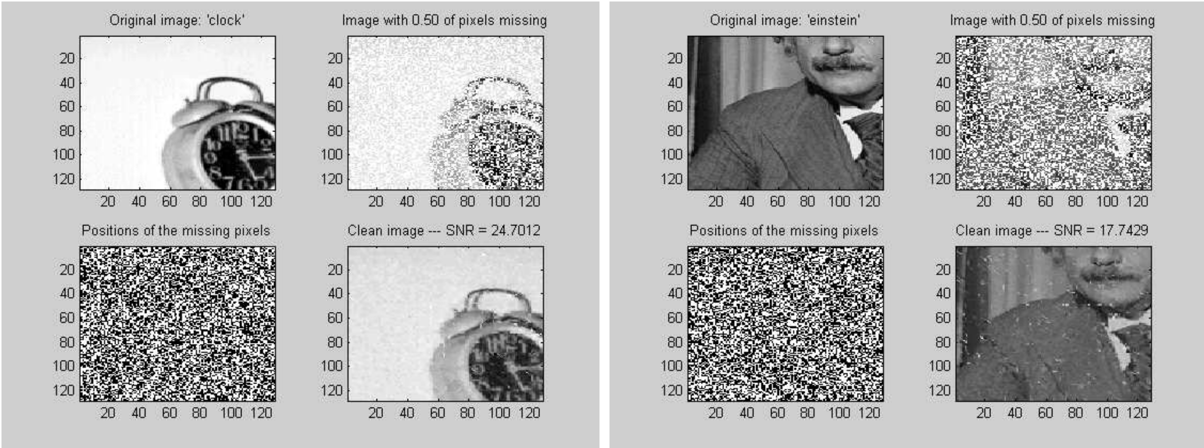

Figure 5: Example of Image restoration. Left: 128×128 subimage from "clock", 8 size ICA basis, 50% pixels missing; Right:128×128 subimage from "einstein", 8 size ICA basis, 50% pixels missing

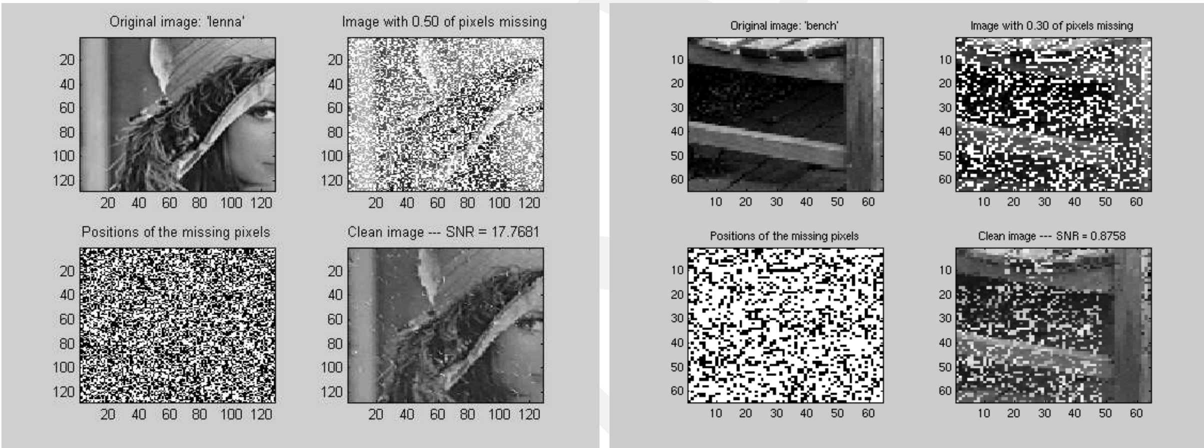

Figure 6: Example of Image restoration. Left: 128×128 subimage from "lenna", 8 size ICA basis, 50% pixels missing ; Right:64×64 subimage from "bench", 16 size Haar basis, 30% pixels missing

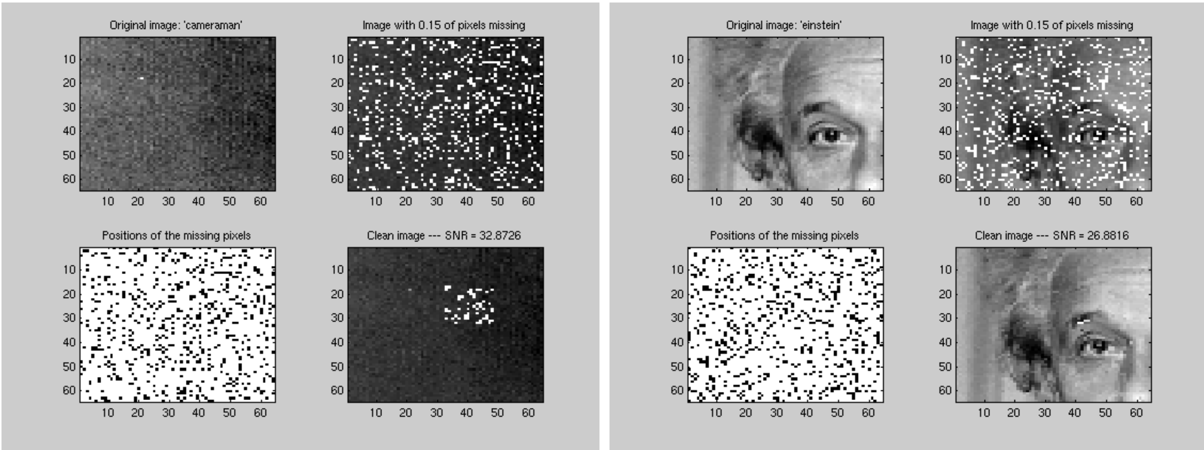

Figure 7: Example of Image restoration. Left: 64×64 subimage from "cameraman", 16 size Haar basis, 30% pixels missing ; Right:64×64 subimage from "einstein", 16 size Haar basis, 30% pixels missing

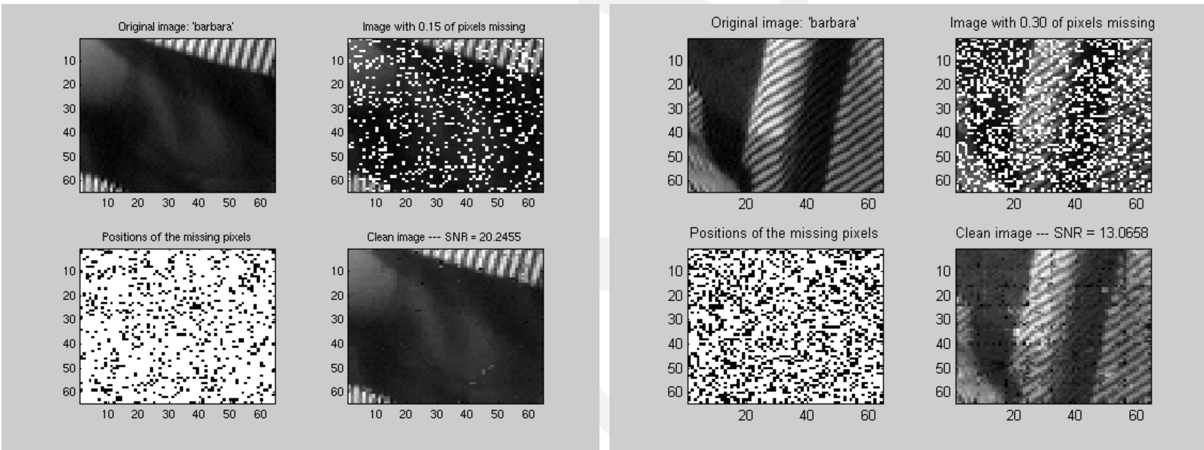

Figure 8: Example of Image restoration. Left: 64×64 subimage from "barbara", 8 size Haar basis, 15% pixels missing ; Right:64×64 subimage from "barbara", 8 size DCT basis, 30% pixels missing

5 Conclusions

In this paper, we described several approaches to the problem of filling in the black-outs in various types of signals. The approach we used was based on Bayesian denoising. For this, we used two of the best methodologies known in signal representation, namely wavelets and Independent Component Analysis.

The Bayesian Denoising method performs remarkably well, for various basis size, image size, missing pixel percentage, or type of basis. To more objectively confirm the performance, the SNR values are usually high. The best results we obtained when using either plain ICA or multiresolution ICA. Although, there appear blocking artifacts for the plain ICA, they are much less evident than for the DCT case.

DRAFT

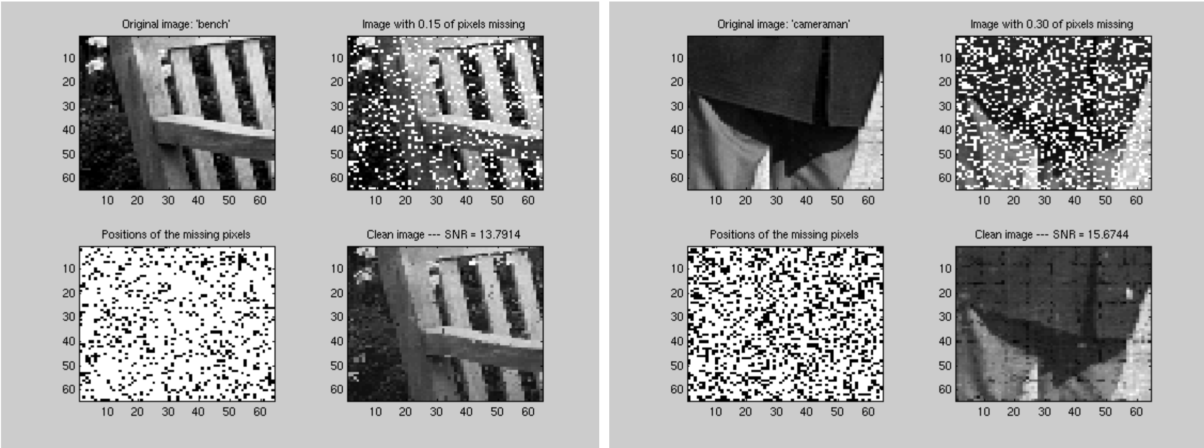

Figure 9: Example of Image restoration. Left: 64×64 subimage from "bench", 8 size Haar basis, 15% pixels missing ; Right:64×64 subimage from "cameraman", 8 size DCT basis, 30% pixels missing

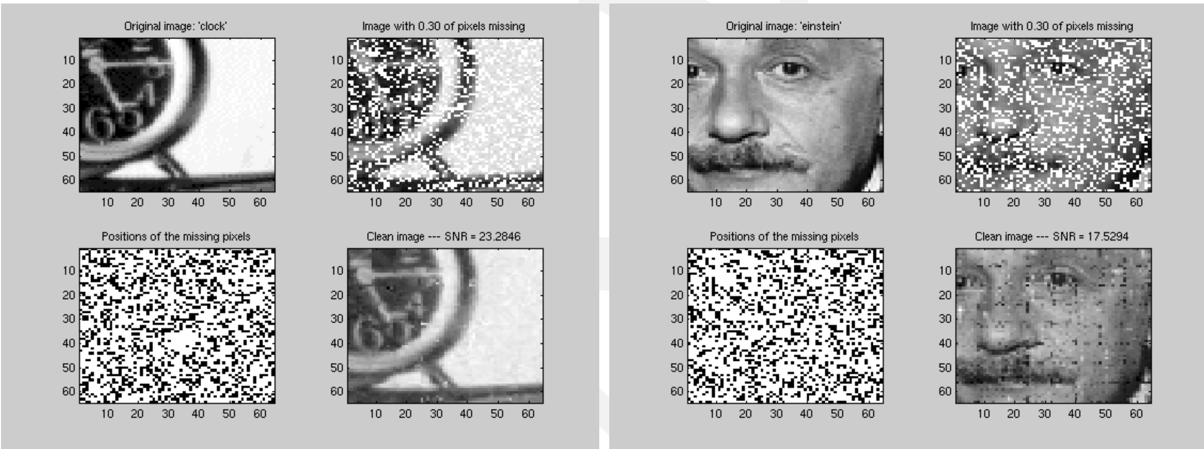

Figure 10: Example of Image restoration. Left: 64×64 subimage from "clock", 8 size ICA basis, 30% pixels missing ; Right:64×64 subimage from "einstein", 8 size DCT basis, 30% pixels missing

The novel approach was the use of multiresolution ICA. The signal representation is more accurate, and so is the image It looks like this is a very promising methodology to pursuit an due to its great promise it announces as a great competition for the wavelets in the processing of natural signals. The downside is the dependence of a good (i.e. *accurate* **and** *fast*) learning algorithm, although according to [5] the polynomial computational complexity of FastICA insures very good computational performances.

References

[1] J.H. van Hateren A. van der Schaaf. Modelling the power spectra of natural images: statistics and information. *Vision Research*, 36:2759–2770, 1996.

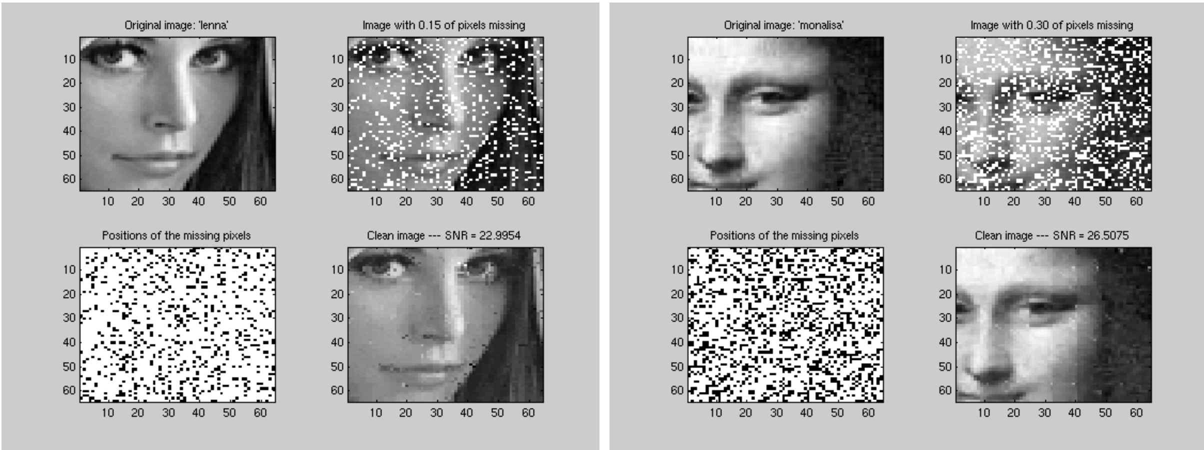

Figure 11: Example of Image restoration. Left: 64×64 subimage from “lenna”, 8 size Haar basis, 15% pixels missing ; Right:64×64 subimage from “monalisa”, 8 size ICA basis, 30% pixels missing

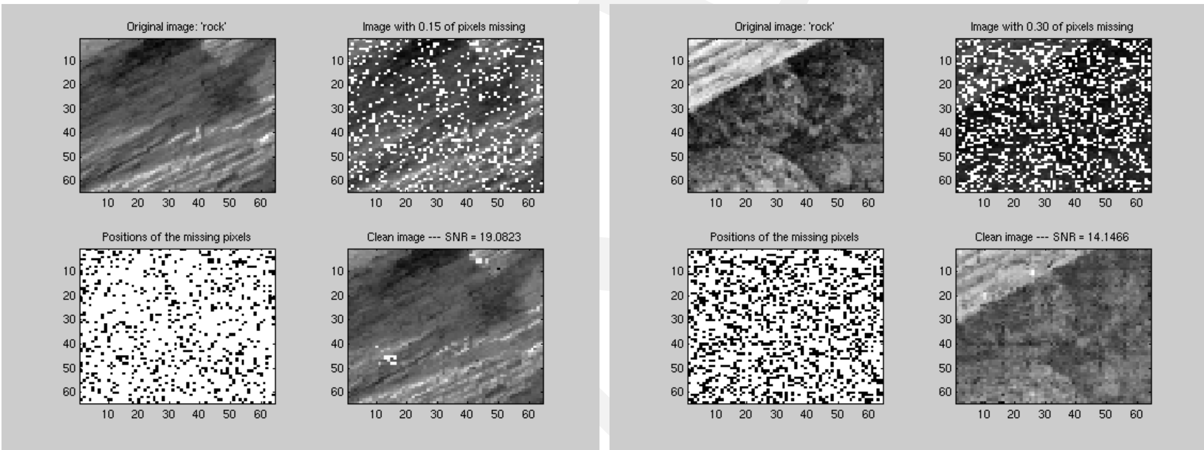

Figure 12: Example of Image restoration. Left: 64×64 subimage from “rock”, 8 size Haar basis, 15% pixels missing ; Right:64×64 subimage from “rock”, 8 size DCT basis, 30% pixels missing

[2] A. J. Bell and T. J. Sejnowski. The ‘independent components’ of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

[3] G. Box, G. Jenkins, and G. Reinsel. *Time Series Analysis: Forecasting and Control*. Prentice Hall, Englewood Cliffs, NJ, 3rd edition, 1994.

[4] Amara Graps. An Introduction to Wavelets. *http://www.amara.com/IEEEwave/IEEEwavelet.html*, 2003.

[5] A. Hyvarinen and E. Oja J. Karhunen. *Independent Component Analysis*. John Wiley and Sons, 2001.

DRAFT

[6] R. E. Kalman. A New Approach To Linear Filtering and Prediction Problems. *ASME - Journal of Basic Engineering*, 82(6):35–45, June 1960.

[7] Y. Karklin and M.S. Lewicki. Higher-order Structure of Natural Images. *Advances in Neural Information Processing Systems*, 2002.

[8] M.S. Lewicki and D. Balcan. Multiresolution ICA. *To be published*, 2003.

[9] M.S Lewicki and B. Olshausen. Probabilistic Framework for the Adaptation and Comparison of Image Codes. *J. Opt. Soc. Am. A*, 16(7):1587–1601, July 1999.

[10] Y. Matias, J. S. Vitter, and M. Wang. Wavelet-based Histograms for Selectivity Estimation. pages 448–459, 1998.

[11] B.A. Olshausen and D.J. Field. Emergence of simple-cell receptive-field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[12] Spiros Papadimitriou. Personal communication. 2003.

[13] E.P. Simoncelli and E.H. Adelson. Subband transforms. In John W Woods, editor, *Subband Image Coding*, chapter 4, pages 143–192. Kluwer Academic Publishers, Norwell, MA, 1990.

[14] E.P. Simoncelli and E.H. Adelson. Noise Removal via Bayesian Wavelet Coring. *Proc. 3rd Int. Conf. Image Processing, Lausanne, Switzerland*, page 379383, 1996.

[15] E.P. Simoncelli. Bayesian denoising of visual images in the wavelet domain. In *Lecture Notes in Statistics. Bayesian Inference in Wavelet-Based Models*, chapter 18, page 291308. Springer-Verlag, New York, 1999.

[16] E. Simoncelli and B. Olshausen. Natural image statistics and neural representation. *Annual Review of Neuroscience*, 24:1193–1216, May 2001.

[17] G. Strang and T. Nguyen. *Wavelets and Filter Banks.* Cambridge Press, 1997.

[18] W. Sweldens and P. Schröder. Building your own wavelets at home. In *Wavelets in Computer Graphics*, pages 15–87. ACM SIGGRAPH Course notes, 1996.

[19] V. Tresp and R. Hofmann. Missing and noisy data in nonlinear time-series prediction. *Neural Networks for Signal Processing 5*, pages 1–10, 1995.

[20] M. Vetterli and J. Kovacevic. *Wavelets and Subband Coding.* Prentice Hall, Englewood Cliffs, NJ, 1995.

DRAFT

[21] Greg Welsh and Gary Bishop. An introduction to the Kalman filter. *UNC-Chapel Hill, Technical Report 95-041*, pages 1–16, March 2002.

[22] B.K. Yi, N.D. Sidiropoulos, T. Johnson, H.V. Jagadish, C. Faloutsos, and A. Biliris. Online Data Mining for Co-Evolving Time Sequences. *Proceedings of the IEEE Sixteenth ICDE*, pages 13–22, 2000.

DRAFT

Contents

DRAFT

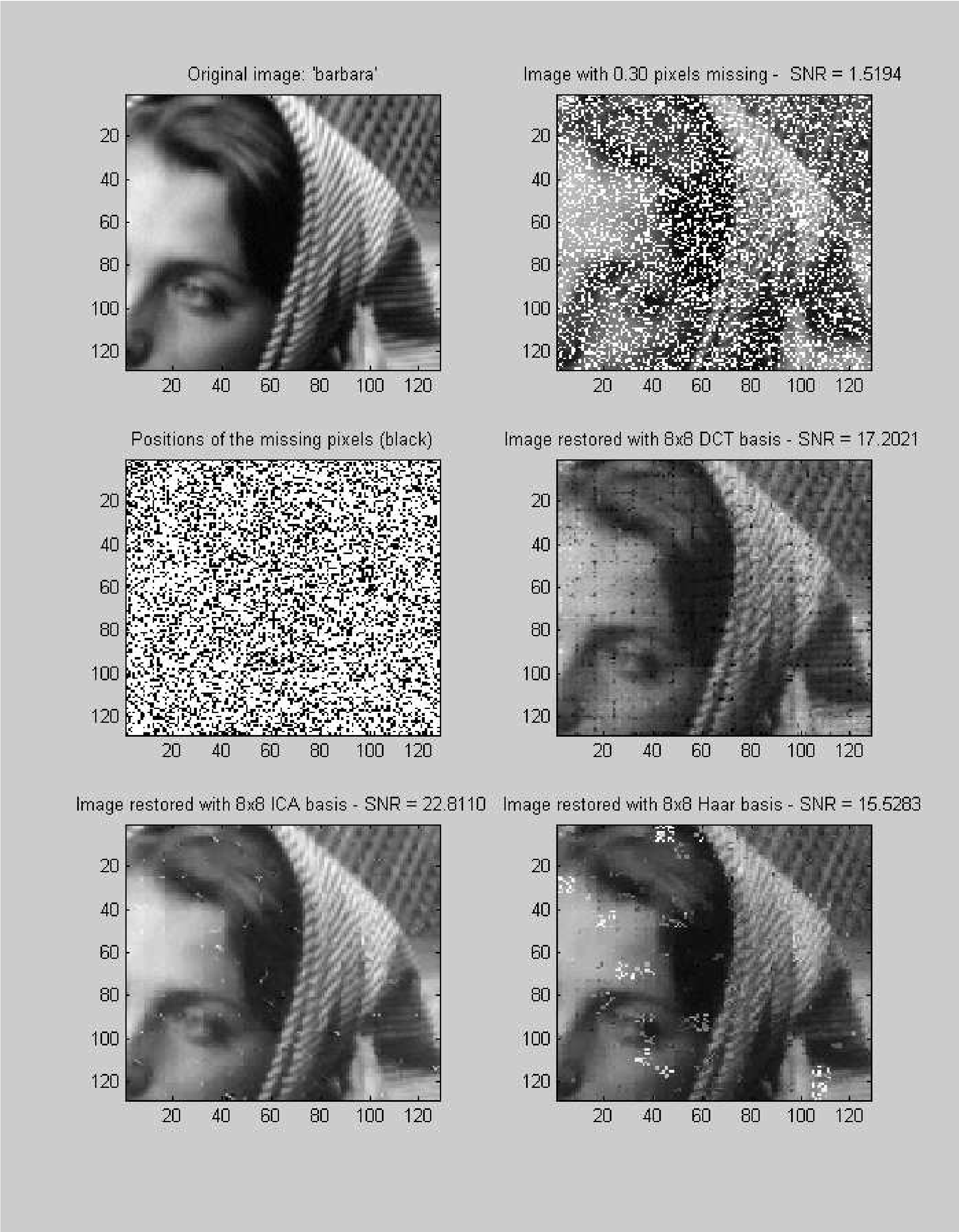

Figure 13: Comparative Example of Image restoration. 128×128 subimage from "barbara", 30% pixels missing