A STATISTICAL EXTENSION OF NORMALIZED CONVOLUTION AND ITS USAGE FOR IMAGE INTERPOLATION AND FILTERING

Matthias Mühlich and Rudolf Mester

Institute for Applied Physics, J.W.Goethe-University Frankfurt
Robert-Mayer-Str. 2–4, 60054 Frankfurt am Main, Germany
email: [muehlich|mester]@iap.uni-frankfurt.de
web: www.uni-frankfurt.de/fb13/iap/cvg/

ABSTRACT

The natural characteristics of image signals and the statistics of measurement noise are decisive for designing optimal filter sets and optimal estimation methods in signal processing. Astonishingly, this principle has so far only partially found its way into the field of image sequence processing. We show how a Wiener-type MMSE optimization criterion for the resulting image signal, based on a simple covariance model of images or image sequences provides direct and intelligible solution for various, apparently different problems, such as error concealment, or adaption of filters to signal and noise statistics.

1. INTRODUCTION

Wiener filtering [1] is often considered to be computationally expensive, somehow tied to signal processing in the Fourier or DFT domain, and more or less exhausted with proliferating lowpass filters that suppress noise in image on the cost of obtaining unsharp edges. In contrast to these (possibly slightly exaggerated) prejudices, Norbert Wiener's theory of optimum linear filters provides an overwhelming richness of useful and moreover elegant solutions to many current problems of image and video sequence processing. Almost everything to be found in this paper is a direct consequence of Wiener's theory developed in the 1940ies; as far as the discrete case is concerned, everything goes back even further, since it can all be deduced from the Gauss-Markov theorem. Nevertheless, we claim that the approaches presented here give a fresh and quite useful view on many image processing problems.

We will discuss here the case of linear finite impulse response (FIR) filters in one, two or more dimensions. The concept presented here allows to exploit knowledge on both noise characteristics (in terms covariance matrices) and statistics of the underlying signal (in terms of the autocorrelation function). We emphasize that optimal filter design cannot be done without considering the signal structure and noise structure. Our filter approach called *signal and noise adaptive filter* or *SNA-filter* allows to combine both sources of knowledge to a combined optimal filter.

2. SIGNAL MODEL

2.1 Filter model for an ideal error-free signal $\mathbf{s}$

Let us first assume that we have access to a certain block of noise-free image data, for instance a rectangular block from a still image, or a three-dimension space-time volume from an image sequence. In any of these situations, by scanning a neighborhood of a given pixel in an arbitrary but fixed order, we may convert the given data into a vector $\mathbf{s}$ which then serves as input for our filtering task. The desired filter coefficients can be stacked analogously to form a vector $\mathbf{h}$. The filter equation that describes the output g which we would like to obtain is given by

$$g = \mathbf{h}^T \mathbf{s} \,. \tag{1}$$

In contrast to this ideal situation, in practise there will always be errors in the image data. It might even be that some of the pixel values are not observable, e.g. due to sensor dropouts. Can we still use the same filter coefficient vector $\mathbf{h}$ or would it be appropriate to adjust these coefficients?

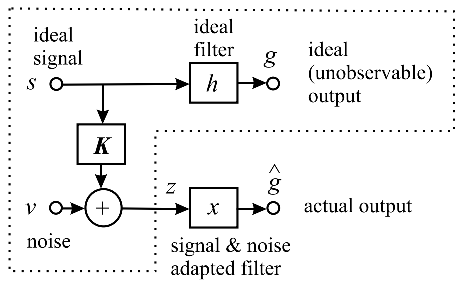

Figure 1: Block diagram of the filter design scenario (unobservable entities in the dashed frame)

In order to answer this question, we must model the signal as well the noise, and the final solution will be a special case of the theory of optimum filters. It is not surprising that the second order moments of the signal and the noise play a decisive role in this process.

2.2 General model of the observed signal

To put this all in mathematical terms, let $\mathbf{s} \in \mathbb{R}^M$ be the uncorrupted signal vector with M components. What we can observe is not necessarily M-dimensional as well, for instance if some data values are missing. For representing this, we define an observation matrix $\mathbf{K} \in \mathbb{R}^{N \times M}$. In the case of missing data, $\mathbf{K}$ consists of 0 and 1 values only, but the theory would work also for arbitrary matrix elements $k_{ij} \in \mathbb{R}$. Without noise, we would obtain the vector $\mathbf{z} = \mathbf{K}\mathbf{s} \in \mathbb{R}^N$ (i.e. a linear transformation of the pixels in the considered neighborhood) as the observable entity, but in real-world applications, there is an additive noise component $\mathbf{v}$ in the measurable pixel vector: $\mathbf{z} = \mathbf{K}\mathbf{s} + \mathbf{v}$. Filtering this 'vectorized' pixel set $\mathbf{z}$ can thus be written as scalar product $\hat{g} = \mathbf{x}^T \mathbf{z}$ using a filter coefficient vector $\mathbf{x} \in \mathbb{R}^N$. The block diagram in fig. 1 provides a graphical representation of our model, and the corresponding equation for the actual filter output $\hat{g}$ reads:

$$\hat{g} = \mathbf{x}^T \mathbf{z} = \mathbf{x}^T (\mathbf{K}\mathbf{s} + \mathbf{v}) = \mathbf{x}^T \mathbf{K}\mathbf{s} + \mathbf{x}^T \mathbf{v} \,. \tag{2}$$

Our task is to choose $\mathbf{x}^T$ in such a way that the filtered output $\hat{g}$ approximates, on an average, the desired output g (eq. 1) of the error-free case as closely as possible.

2.3 Statistical joint moments of signal and noise

The next step is to define the statistical properties of the signal and the noise processes, respectively. Let the noise vector $\mathbf{v} \in \mathbb{R}^N$ be a zero-mean random vector with covariance matrix $\mathbf{C}_v$ (which is in this case equal to its correlation matrix $\mathbf{R}_v$):

$$\mathsf{E}[\mathbf{v}] = \mathbf{0} \quad \text{and} \quad \mathsf{E}\left[\mathbf{v}\mathbf{v}^T\right] = \mathbf{R}_v\,.$$

Furthermore, we assume that the process that generated the signal $\mathbf{s} \in \mathbb{R}^N$ can be described by the expectation $\mathbf{m}_s$ of the signal vector, and an autocorrelation matrix $\mathbf{R}_s$.

$$\mathsf{E}[\mathbf{s}] = \mathbf{m}_s \quad \text{and} \quad \mathsf{E}\left[\mathbf{s}\mathbf{s}^T\right] = \mathbf{R}_s\,.$$

All these statistical moments can be measured from actual image data, although it must be taken care that the correlation matrices $\mathbf{R}_s$ and $\mathbf{R}_v$ should be positive definite[1].

Our last assumption is that noise and signal are uncorrelated: $\mathsf{E}\left[\mathbf{s}\mathbf{v}^T\right] = \mathbf{0}$. Knowing these first and second order statistical moments for both the noise as well as the signal allows the derivation of the optimum filter $\mathbf{x}$.

2.4 Designing the Optimal SNA-Filter

Applying the signal model $(\mathbf{m}_s, \mathbf{R}_s)$ and the error model $(\mathbf{0}, \mathbf{R}_v)$ on equations (1) and (2), we obtain

$$\mathsf{E}[g] = \mathbf{h}^T\mathbf{m}_s \quad \text{and} \quad \mathsf{E}[\hat{g}] = \mathbf{x}^T\mathbf{K}\mathbf{m}_s$$

for the expectation values (first order moments), and

$$\begin{aligned}
\mathsf{E}\left[g^2\right] &= \mathbf{h}^T\mathbf{R}_s\mathbf{h} \quad \text{and} && (3)\\
\mathsf{E}\left[\hat{g}^2\right] &= \mathbf{x}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)\mathbf{x} \quad \text{and} && (4)\\
\mathsf{E}[g\hat{g}] &= \mathbf{x}^T\mathbf{K}\mathbf{R}_s\mathbf{h} && (5)
\end{aligned}$$

for the second order statistical moments[2]. Next, we define the approximation error e between the ideal output g and the actual output $\hat{g}$: $e = \hat{g} - g$.

In general, the approximation error e is not a zero-mean random variable: $\mathsf{E}[e] = \mathsf{E}[\hat{g}] - \mathsf{E}[g] = (\mathbf{h}^T - \mathbf{x}^T\mathbf{K})\mathbf{m}_s$. The expected squared error Q as a function of the vector $\mathbf{x}$ can be computed from equations (3) to (5):

$$\begin{aligned}
Q(\mathbf{x}) &= \mathsf{E}\left[e^2\right] = \mathsf{E}\left[\hat{g}^2\right] - 2\mathsf{E}[g\hat{g}] + \mathsf{E}\left[g^2\right]\\
&= \mathbf{x}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)\mathbf{x} - 2\mathbf{x}^T\mathbf{K}\mathbf{R}_s\mathbf{h} + \mathbf{h}^T\mathbf{R}_s\mathbf{h}
\end{aligned}$$

We see that a minimum mean squared error (MMSE) estimator can now be designed. We set the derivative $\partial Q(\mathbf{x})/\partial\mathbf{x}$ to $\mathbf{0}$ and obtain

$$\mathbf{x} = (\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)^{-1}\mathbf{K}\mathbf{R}_s\mathbf{h}\,. \qquad (6)$$

Applying this filter to the observable input $\mathbf{z} = \mathbf{K}\mathbf{s} + \mathbf{v}$ (exploiting the symmetry of $\mathbf{R}_s$ and $\mathbf{R}_v$), we obtain

$$\begin{aligned}
\hat{g} &= \mathbf{x}^T\mathbf{z} = \mathbf{h}^T\mathbf{R}_s\mathbf{K}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)^{-1}\mathbf{z} && (7)\\
&= \mathbf{h}^T\mathbf{R}_s\mathbf{K}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)^{-1}\mathbf{K}\mathbf{s} + \\
&\quad\; \mathbf{h}^T\mathbf{R}_s\mathbf{K}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)^{-1}\mathbf{v} = \mathbf{h}^T\mathbf{J}\mathbf{s} + f(\mathbf{v}) && (8)
\end{aligned}$$

with some noise- (and not signal-) dependent second summand $f(\mathbf{v})$ which vanishes in expectation and some complicated weight matrix $\mathbf{J} = \mathbf{R}_s\mathbf{K}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)^{-1}\mathbf{K}$ between $\mathbf{h}^T$ and the input vector $\mathbf{s}$. Comparing with equation $g = \mathbf{h}^T\mathbf{s}$ for the desired ideal output g, we see that the essential difference is this weight matrix $\mathbf{J}$. The closer $\mathbf{J}$ gets to identity matrix, the better the $\hat{g}$ approximates g. But both unobservable data ($N < M$) and the existence of noise ($\mathbf{R} \neq \mathbf{0}$) cause deviations of $\mathbf{J}$ from $\mathbf{I}$ and make a perfect approximation impossible.

[1] The special case of some pixels being known exactly, which leads to a positive semidefinite noise covariance matrix, is handled later in section 2.5.

[2] Note: these second order moments are *no* variances because neither g nor $\hat{g}$ are zero-mean random vectors!

2.5 Some special cases

Special case 1: Square and non-singular $\mathbf{K}$ (all pixels observable, but arbitrarily reweighted and combined in linear form). In other words: we observe a 'blurred' version of the original data. The noise $\mathbf{v}$ has the same dimensionality as the signal $\mathbf{s}$; therefore, we can introduce a new error vector $\mathbf{v}' = \mathbf{K}^{-1}\mathbf{v}$. In the block diagram, adding $\mathbf{v}'$ *before* the transformation $\mathbf{K}$ is equivalent to adding $\mathbf{v}$ afterwards. The covariance/correlation matrix transforms according to

$$\mathbf{R}_v = \mathbf{K}\mathbf{R}'_v\mathbf{K}^T$$

Equation (7) then can be simplified to

$$\hat{g} = \mathbf{h}^T\mathbf{R}_s(\mathbf{R}_s + \mathbf{R}'_v)^{-1}\mathbf{K}^{-1}\mathbf{z}$$

This equation states that each coordinate in the canonical coordinate frame has to be reweighted according to its signal-to-noise ratio.

Under the additional assumption of no noise ($\mathbf{R}_v = \mathbf{0}$, $\mathbf{K}^{-1}\mathbf{z} = \mathbf{s}$), we furthermore see that

$$\hat{g} = \mathbf{h}^T\mathbf{R}_s\mathbf{R}_s^{-1}\mathbf{s} = \mathbf{h}^T\mathbf{s} = g$$

holds, i.e. we can achieve a perfect approximation for arbitrary 'mixing' of signal components in $\mathbf{K}$ as long as $\mathbf{K}$ is non-singular.

Special case 2: If some elements of the projected signal vector $\mathbf{K}\mathbf{s}$ are known exactly, it is appropriate to extend our signal model to include this case in a numerically advantageous way. In order to obtain the corresponding signal and error model, we partition

$$\mathbf{K}\mathbf{s} = \begin{pmatrix} \mathbf{K}_1 \\ \mathbf{K}_2 \end{pmatrix}\mathbf{s} \quad \text{and} \quad \mathbf{x} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix}$$

in an error-free part (subscript 1, N_1-dimensional) and an erroneous part (subscript 2, N_2-dimensional; $N_1 + N_2 = N$). Using basically the same reasoning as before, we arrive at

$$\begin{aligned}
\mathbf{x}_1 &= (\mathbf{K}_1\mathbf{R}_s{\mathbf{K}_1}^T)^{-1}\mathbf{K}_1\mathbf{R}_s\mathbf{h} && (9)\\
\mathbf{x}_2 &= (\mathbf{K}_2\mathbf{R}_s{\mathbf{K}_2}^T + \mathbf{R}_v)^{-1}\mathbf{K}_2\mathbf{R}_s\mathbf{h}\,. && (10)
\end{aligned}$$

for the computation of the optimum filter $\mathbf{x}$. Note that the single equation (6) could *in principle* also serve for the computation of the whole vector $\mathbf{x}^T = (\mathbf{x}_1^T, \mathbf{x}_2^T)$ if there are error-free observations. The restriction of $\mathbf{R}_v$ to positive definite matrices was not necessary. Semi-definiteness of $\mathbf{R}_v$ is no problem as long as $\mathbf{K}\mathbf{R}_s\mathbf{K}^T$ is positive definite and, hence, invertible. However, the enhanced numerical stability of the scheme and the reduced size of the matrices that have to be multiplied and inverted call for the partitioned equations (9) and (10) in this case.

2.6 A different derivation: Gauss-Markov theorem

Equation (7) can be expressed as

$$\hat{g} = \mathbf{h}^T(\mathbf{S}\mathbf{z}) = \mathbf{h}^T\hat{\mathbf{s}}$$

This means that the operator

$$\mathbf{S} = \mathbf{R}_s\mathbf{K}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T + \mathbf{R}_v)^{-1} \qquad (11)$$

could be interpreted as a *general signal restoration operator* that transforms a signal (or two-dimensional image or higher-dimensional spatio-temporal volume) $\mathbf{z}$ to an equally sized signal (image, volume) $\hat{\mathbf{s}}$. In fact, it is the Wiener filter for this special case. For specifying this operator, we only need the signal and noise correlation matrices; it is *independent of the sought filter* $\mathbf{h}$.

For increased simplicity, we now assume $\mathbf{K} = \mathbf{I}$. Then

$$\mathbf{S} = \mathbf{R}_s(\mathbf{R}_s + \mathbf{R}_v)^{-1}$$

A similar derivation of $\mathbf{S}$ can be obtained by minimizing the mean squared error of the approximated signal $\hat{\mathbf{s}}$ directly. Using the Gauss-Markov-Theorem [2, page 296], this leads to

$$\mathbf{S} = (\mathbf{R}_s^{-1} + \mathbf{R}_v^{-1})^{-1}\mathbf{R}_v^{-1}$$

Although both representations look rather different, they are in fact identical. Proof: Let $\mathbf{A}$ and $\mathbf{B}$ be two equally sized non-singular matrices. Then

$$\begin{aligned}\mathbf{A}(\mathbf{A}+\mathbf{B})^{-1} &= \big((\mathbf{A}+\mathbf{B})\mathbf{A}^{-1}\big)^{-1} = \big(\mathbf{I}+\mathbf{B}\mathbf{A}^{-1}\big)^{-1}\\ &= \big(\mathbf{B}(\mathbf{B}^{-1}+\mathbf{A}^{-1})\big)^{-1} = (\mathbf{B}^{-1}+\mathbf{A}^{-1})^{-1}\mathbf{B}^{-1}\,.\end{aligned}$$

2.7 Relation to normalized convolution

The SNA-filter approach presented here closely resembles to the concept of *normalized convolution* [3, 4]. Written in the notation used here, this approach tries to *approximate* the observed signal $\mathbf{z}$ (implicitly assumed to have the same dimensionality as the underlying signal $\mathbf{s}$ – no special error model is provided) with a properly weighted sum of 'base functions'. The resulting vector[3] $\hat{\mathbf{z}}$ is given by:

$$\hat{\mathbf{z}} = \mathbf{B}\underbrace{(\mathbf{B}^T\mathbf{W}_a\mathbf{W}_c\mathbf{B})^{-1}\mathbf{B}^T\mathbf{W}_a\mathbf{W}_c\mathbf{z}}_{\mathbf{r}} \qquad (12)$$

where the columns of $\mathbf{B}$ are the so-called 'base functions' (which should rather be called 'base vectors' or 'sampled base functions'). $\mathbf{W}_a$ $\mathbf{W}_c$ are diagonal matrices called 'applicability' and 'certainty'. The (shift-invariant) applicability indicates the (user-defined) general 'importance' of each point in the neighborhood and the certainty gives the relative weight for each actual pixel. The resulting weight is a product of both values.

In spite of the similarity between $\mathbf{B}$ in (12) and the observation matrix $\mathbf{K}$ in our equations, it is something fundamentally different. $\mathbf{K}$ maps the signal $\mathbf{s}$ to the observable vector space in which $\mathbf{z} = \mathbf{K}\mathbf{s}+\mathbf{v}$ exist, whereas $\mathbf{B}$ is the signal model we want to impose and one is often only interested in the weight for each different base vector, i.e. in the coefficients of $\mathbf{r}$. For instance, if gradients in 3×3 patches are sought, two base function would be the vectorized versions of

$$\begin{bmatrix} -1 & -1 & -1\\ 0 & 0 & 0\\ 1 & 1 & 1\end{bmatrix}\cdot\mathbf{G} \quad\text{and}\quad \begin{bmatrix} -1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{bmatrix}\cdot\mathbf{G}$$

where $\cdot$ denoted pointwise multiplication and $\mathbf{G}$ is some averaging (Gaussian, binomial, or other). A third base function would have to be an all-one column vector which captures all DC offset (and thus makes the other two components insensitive to dc offsets). The result of normalized convolution would be a decomposition of the given image in four parts: x-gradient + y-gradient + offset (these three parts sum up to the respective central values of $\hat{\mathbf{z}}$) + *'part which does not fit in this simple signal model'* (difference $\mathbf{z}-\hat{\mathbf{z}}$). The general concept of normalized convolution is '*reduction of a given signal to a simplified signal model'*.

Our approach, on the contrary, serves for optimally computing filter sets for sought ideal outputs.[4] One has to define the desired output for the ideal noise free signal by setting $\mathbf{h}$ accordingly – without considering errors or missing data. Given the signal-to-noise ratio, it is then possible to *derive* the SNA-filter coefficients. For instance, for de-noising natural images with their typical low-pass autocorrelation, our scheme will in general compute filter coefficients which *roughly look like* Gaussians – just because the structure of the signal dictates it.

In order to illustrate the fundamental difference between the two concepts, we decided to deal with a problem which was also handled in [4]: restoration of an image with missing pixels. We assume noise-free data, thus setting aside the special strength of our algorithm for handling different signal-to-noise ratios properly.

By applying normalized convolution, we would set $\mathbf{B}$ to one single column vector of 1s, i.e. the signal model is: uniform gray value in the considered signal patch. What we effectively do is filtering the image with a modified applicability (e.g. Gaussian) where all coefficients corresponding to missing data are set to zero (and normalized afterwards). The important point is that weighted averaging will appear *everywhere* – even if we know that some pixels are known without error. A highly peaked applicability will reduce this effect, but known pixels *will* change.

Later on, we will show that the very same task leads to a different result with SNA filtering: an image without noise and only missing data will remain unaltered at the available pixels, whereas missing pixels will be interpolated from their neighbors.

3. APPLICATION EXAMPLES

3.1 Reconstruction of missing data

The most simple case is the reconstruction of a signal from a vector $\mathbf{s}$ representing a noisy spatio-temporal data volume. The model operator $\mathbf{h}$ is trivial to construct; it is 1 for the central value and 0 else, i.e. $\mathbf{h}$ is $\mathbf{0}$ except for a single 1 at the central element.

Equation (10) gives the optimum solution for any arbitrary configuration of noise and signal covariances. Nevertheless, we will focus on an especially interesting and illustrative case here: reconstruction from missing data. Let us assume that we have no noise at all. Furthermore, we assume that only N out of M pixels are observable. The observation matrix $\mathbf{K}\in\mathbb{R}^{N\times M}$ is the $N\times N$ identity matrix with $M-N$ additional $\mathbf{0}$ column vectors inserted at the positions corresponding to missing data.

The matrix $\mathbf{K}^T\mathbf{K}$ then is a $M\times M$ identity matrix where all diagonal elements corresponding to missing data have been set to 0. This means that $\mathbf{K}^T\mathbf{K}\mathbf{h} = \mathbf{0}$ if the central pixel is missing and $\mathbf{K}^T\mathbf{K}\mathbf{h} = \mathbf{h}$ if it is available.

For the latter case, we get

$$\begin{aligned}\hat{g} &= \mathbf{s}^T\mathbf{K}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T)^{-1}\mathbf{K}\mathbf{R}_s\mathbf{h}\\ &= \mathbf{s}^T\mathbf{K}^T(\mathbf{K}\mathbf{R}_s\mathbf{K}^T)^{-1}(\mathbf{K}\mathbf{R}_s\mathbf{K}^T)\mathbf{K}\mathbf{h}\\ &= \mathbf{s}^T\mathbf{K}\mathbf{K}^T\mathbf{h} = \mathbf{s}^T\mathbf{h} = g\,.\end{aligned}$$

i.e. we leave the central pixel unchanged and the resulting approximation error will be 0. An image with some missing pixels will remain unaltered at the given points. If, however, the central pixel is missing, the same approach yields

$$\mathbf{x} = (\mathbf{K}\mathbf{R}_s\mathbf{K}^T)^{-1}\mathbf{K}\mathbf{R}_s\mathbf{h}$$

to interpolate the central pixel from its neighborhood. Clearly, $\mathbf{x}$ is an coefficient vector that depends on the individual pattern of missing pixels. The knowledge of the statistical correlation between the (missing) central pixel and available neighbors automatically provides an individual, properly weighted interpolation kernel. The very same approach allows also to interpolate from noisy data.

We will present some experiments for interpolating a low resolution baboon image (64×64 pixels) using 5×5 FIR filters. The low resolution makes this task a little bit more difficult because important image structures like eyes only consist of a few pixels and are hard to reconstruct if they are missing. Figure 2 shows a possible incomplete baboon image. We eliminated 25% of the pixels here. All missing points are set to white. For our experiments, we

[3] Although all image examples in [4, chapter3] implicitly take the central pixel for composing the output images, the mathematical core is the approximation of a M-dimensional signal vector by a weighted sum of M-dimensional base vectors.

[4] If several filters are to be applied, e.g. gradients in x and y direction, it is possible to form a matrix $\mathbf{H}^T = (\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_p)$ to produce an output $\hat{\mathbf{g}} = (\hat{g}_1, \hat{g}_1, \ldots, \hat{g}_p)$. Everything between $\mathbf{h}_i^T$ and $\mathbf{z}$ (this matrix-valued operation could be denoted as a *general signal restoration operator*) is independent of $\mathbf{h}_i^T$ and has to be computed only once.

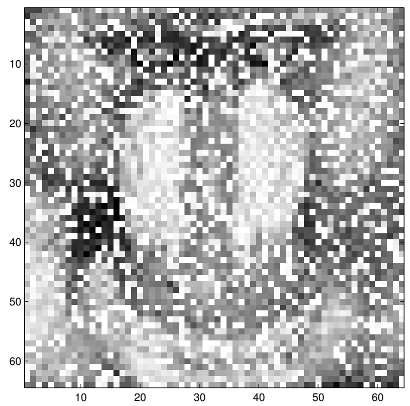

Figure 2: The well known baboon image with 25% of the pixels (1024 out of 4096) missing. Computing gradients from images like this is obviously a difficult problem...

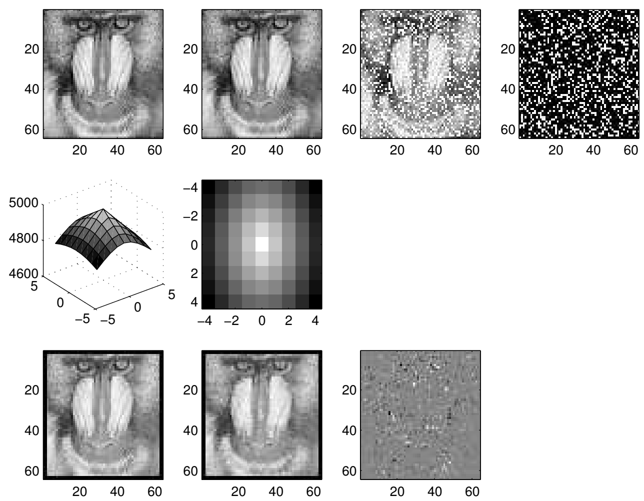

Figure 3: Reconstruction from an image with missing data

took the original image, added noise, and then marked some randomly chosen pixels as missing. These steps correspond to the first three images of the image array in fig. 3. The last picture in the first row is a binary picture and shows the missing points.

The second row illustrates the image model. We assumed direction dependent exponential decay of the image autocorrelation function (acf). It can be seen that the chosen image has an anisotropic acf.

The reconstructed image is shown at the second position in the bottom row. For comparison, the result of ideal filter (here: central pixel only) applied to the true and complete image is shown to the left of the reconstructed image. The last image in the third row is the (scaled) difference images between the ideal filter output and SNA filter output.

We want to point out that the theory presented here is applicable to any arbitrary desired ideal filters $\mathbf{h}$. Therefore, we changed the task from reconstruction to computing a horizontal derivative, i.e. the second example gives the SNA-filter output if $\mathbf{h}$ is chosen to be the vectorized form of a 5×5-matrix with $(.5, -1, 0, 1, -.5)$ as central row vector and 0 elsewhere. Figure 4 shows the approximated difference filter output. It is ordered in the same way as fig. 3. Comparing the SNA-filter output with the ideal one, it is visible that all horizontal edges were captured.

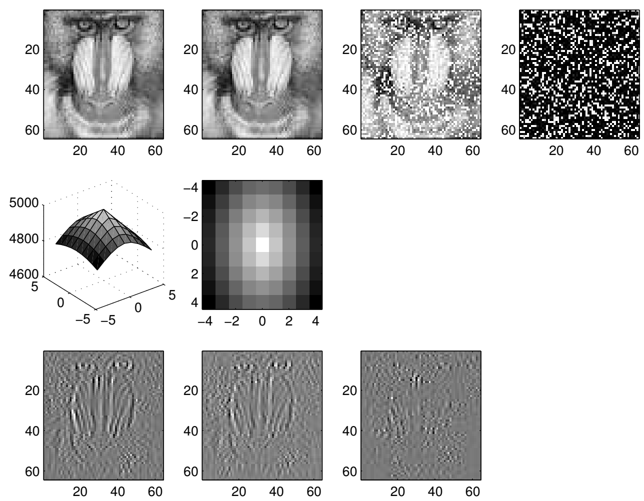

Figure 4: Computing the derivative in one spatial direction from an image with missing data

3.2 Global prefiltering

In general, the SNA-filter $\mathbf{x}$ has to be computed for each pixel individually. However, if the error structure is not pixel dependent, it is possible to compute a common SNA-filter operator for all pixels. Except for some highly artificial situations this means that we have to restrict $\mathbf{K}$ to the identity matrix.

The model of independent and identically distributed Gaussian noise in all pixels, however, can be handled very well with a global SNA prefilter for noise removal.

The usual method of applying Gaussian or binomial filters for de-noising can now be replaced with a statistically justified method of averaging. The new method takes the signal structure into account and the width of the filters is no free parameter anymore, but determined by the autocorrelation of the image (or, more precisely: the multidimensional equivalent of the signal-to-noise ratio).

4. CONCLUSIONS

We have shown that apparently different tasks such as image interpolation (error concealment), and the design of filter operators can be unified in a common framework based on the MMSE error criterion. Within this framework, design parameters that are often difficult to optimize manually are replaced by principles that are based on measurable characteristics of images, such as the autocorrelation function. The discrete and finite formulation of this theory lends itself especially well to image and sequence processing.

REFERENCES

[1] Norbert Wiener, *The Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications*, J. Wiley, 1949.

[2] Henry Stark and John W. Woods, *Probability, Random Processes, and Estimation Theory for Engineers*, Prentice Hall, 2nd edition, 1994.

[3] H. Knutsson and C.-F. Westin, "Normalized and differential convolution: Methods for interpolation and filtering of incomplete and uncertain data," in *Proc. CVPR*, New York City, USA, June 1993, pp. 515–523, IEEE Computer Society Press.

[4] Gunnar Farnebäck, *Polynomial Expansion for Orientation and Motion Estimation*, Ph.D. thesis, Linköping University, Sweden, SE-581 83 Linköping, Sweden, 2002, Dissertation No 790, ISBN 91-7373-475-6.