Weierstraß-Institut
für Angewandte Analysis und Stochastik

im Forschungsverbund Berlin e.V.

Preprint ISSN 0946 – 8633

Adaptive Smoothing of Digital Images: the R Package adimpro

Jörg Polzehl[1] and Karsten Tabelow[1 2]

submitted: 27th October 2006

[1] Weierstrass Institute
for Applied Analysis and Stochastics,
Mohrenstr. 39, 10117
Berlin, Germany
E-Mail: polzehl@wias-berlin.de

No. 1177

Berlin 2006

[2]Supported by the DFG Research Center MATHEON "Mathematics for key technologies" in Berlin
2000 *Mathematics Subject Classification.* 62G05.

Key words and phrases. adaptive weights; local structure; propagation; separation; image processing.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Mohrenstraße 39
10117 Berlin
Germany

Fax: + 49 30 2044975
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract

Digital imaging has become omnipresent in the past years with a bulk of applications ranging from medical imaging to photography. When pushing the limits of resolution and sensitivity noise has ever been a major issue. However, commonly used non-adaptive filters can do noise reduction at the cost of a reduced effective spatial resolution only.

Here we present a new package **adimpro** for R, which implements the Propagation-Separation approach by Polzehl and Spokoiny (2006) for smoothing digital images. This method naturally adapts to different structures of different size in the image and thus avoids oversmoothing edges and fine structures. We extend the method for imaging data with spatial correlation. Furthermore we show how the estimation of the dependence between variance and mean value can be included. We illustrate the use of the package through some examples.

1 Introduction

Digital imaging has seen a huge progress over the last decade through the broad availability of digital cameras. An image in its digital form can easily undergo image processing Gonzalez and Woods (2002) in order to improve its quality, to enhance certain details, or to simply achieve some artistic effect. In the search for ever increased resolution and sensitivity denoising has been a major issue since the sources of noise are inherently connected with the physical properties of the technology. Many of the cameras therefore include smoothing algorithms in their imaging process in order to achieve a maximum of image quality for any lightening conditions.

However, denoising algorithms have to balance the trade-off between denoising and preservation of structure. Non-adaptive Gaussian filter typically tend to oversmooth edges and thus reduce noise at the cost of sharpness of the image. In order to overcome such a drawback a new spatially adaptive smoothing algorithm has been proposed in a series of papers (see e.g. Polzehl and Spokoiny (2000, 2006)). In contrast to classical non-adaptive filters the smoothing method based on the Propagation-Separation (PS) approach naturally adapts to different structures of different size in the image. This is realized through local adaptive weighting schemes which ensure two main properties of the algorithm from which it draws its name. The propagation property means, that the smoothing algorithm behaves like a non-adaptive filter in homogeneous regions. On the other hand the separation property characterizes the restriction of the weighting scheme to homogeneous regions with respect to a specified model for the image data. The procedure has been very successfully applied to the analysis of functional magnetic resonance imaging Polzehl and Spokoiny (2001); Tabelow et al. (2006). Furthermore it has been suggested to use this adaptive smoothing method in the context of image denoising Polzehl and Spokoiny (2000, 2007).

In this paper we present an implementation of an algorithm based on PS for denoising of grayscale and color images. The package **adimpro** is available for R R Development Core Team (2005).

Let $i = (i_h, i_v) \in I\!R^2$, $1 \leq i_h \leq n_h$, $1 \leq i_v \leq n_v$ denote the $n = n_h \times n_v$ pixel coordinates of a $2D$-image and $Y_i \in \mathcal{Y} \subseteq I\!R^q$ an observed gray value $(q = 1)$ or a vector of color values $(q \in \{3, 4\})$ at coordinates i. We assume that the distribution of each Y_i is determined by a parameter $\theta_i = \theta(i)$ which may depend on the coordinates i.

Example 1.1 [Gray scale images] Every Y_i follows a Gaussian distribution with mean θ_i and unknown variance σ^2 that may depend on θ_i.

Example 1.2 [Color images] In color images Y_i denotes a vector of values in a 3 dimensional color space at pixel coordinates i. A 4th component may code transparency information. The observed vectors Y_i can often be modeled as multivariate Gaussian, i.e. $Y_i \sim N_q(\theta_i, \varSigma_i)$ with some unknown covariance $\varSigma$ that again may depend on θ_i.

Depending on the image acquisition process noise in digital images may have an other distribution than a simple Gaussian Gonzalez and Woods (2002). In all examples the situation may be aggravated by spatial correlation due to physical reasons or preprocessing steps.

The paper is organized as follows. Color images require the definition of some appropriate space to represent color. Section 2 therefore reviews the definition of some important color spaces. Section 3 provides the smoothing algorithm based on the Propagation-Separation approach. Spatial correlation and models for heteroscedasticity can be incorporated into the approach. We provide estimates for both spatial correlation and heteroskedastic variances. Examples are given in Section 4.

2 Color Spaces

The concept of color spaces is closely related to the perception of color by the human eye. Sunlight when passing a glass prism does show up a whole spectrum of pure colors ranging from red to violet which blend smoothly into one another. They can be characterized by a single wavelength of the corresponding electromagnetic wave that ranges from approximately 400 to 700nm. The superposition of such chromatic light is perceived by humans as another color rather than recognizing the individual wavelengths separately Malacara (2002). Thus the mixture of chromatic red and green light is seen as yellow. In the human eye the cones are responsible for color vision. It has been experimentally found that they can be mostly divided into three principal sensing categories, corresponding roughly to red, green, and blue. Other colors can therefore be seen as combinations of these three primary colors. This naturally leads to the definition of the RGB color space in which each color is characterized by the amounts of the primary colors necessary to build it up.

Many displaying devices like computer monitors directly make use of this color space through excitation of different amounts of red, green and blue areas on the screen. Digital cameras usually filter the three primary colors from the incoming light in the image acquisition process using a Bayer mask Bayer (1976). Encoding color in the internet or in imaging software follows this scheme too. It should be noted, that not all visible color can be created by such mix, but it should be sufficient for most applications.

However, this does not match the way humans *describe* color. For example bright or pale yellow are seen as two variants of the same color rather than different mixtures of the three primary colors. Taking this into account we can characterize color in terms of *brightness*, *hue*, and *saturation*. Hue is associated with the dominant wavelength and corresponds to the notion of color as we perceive it. The amount of white light mixed with the hue is referred to as saturation. Therefore the pure spectral colors are fully saturated. Other colors like pink are mixtures of a pure color with white and thus less saturated. The combination of hue and saturation is called chromaticity. The brightness corresponds to the intensity as known from grayscale images. The three characteristics mentioned above form the HSI color space.

Other color spaces like YUV or YIQ are related to the way, signal transport for television is performed. One of the components (Y) again reflects the brightness of the image while the other two refer to the chromaticity. Therefore black and white TV sets can simply ignore the color information in the signal and nevertheless produce a reasonable output.

A specific color can be transferred from one color space into another. For example the conversion from RGB to YUV is a linear transformation and can thus be represented as a matrix multiplication. This makes the conversion a very fast operation. Details on the conversion between the color spaces can be found in the literature (see Gonzalez and Woods (2002)).

In the package **adimpro** grayscale and color images are represented by objects of class "adimpro". Image data is stored as two and three dimensional arrays, respectively. The third dimension for color images is 3 and contains the color information in a specific color space as described above. The color space is coded in the attribute `type` and taken from the following list: `yuv, yiq, hsi, xyz, rgb, greyscale`. It is possible to get information on an adimpro-object via the generic `summary`-function. The package contains several function to transform images from one colorspace to another like:

```
R> img.yuv <- rgb2yuv(img.rgb)

R> img.rgb <- hsi2rgb(img.hsi)

R> img.grey <- rgb2grey(img.rgb)
```

One can view any image with:

```
R> show.image(img.yuv)
```

3 Smoothing using structural adaptation

Efficient image denoising requires a reduction of random noise without compromising the content of the image. The Propagation-Separation approach implemented in the package **adimpro** is designed to achieve this goal. Image denoising is ideally done on images that have been exposed to little preprocessing, like images coded in RAW or 16-Bit TIFF format. Loss afflicted preprocessing like conversion to JPEG should be avoided ahead of adaptive image denoising to improve the quality of adaptation of the algorithm. Image acquisition in digital cameras is usually done by measuring the intensity of light for the three primary colors using a Bayer mask Bayer (1976). The process of reconstruction of the missing color components for the full resolution is called demosaicing but will not be considered here.

Let the distribution of Y_i be specified up to a parameter θ_i that may depend on the coordinates i of a pixel and some additional parameters ν. Such parameters are possibly heterogeneous error variances. Let us also assume for a moment that we know how estimates of these parameters can be obtained.

We presume that for each pixel i the function $\theta(.)$ can be well approximated by a constant within a local vicinity U_i containing pixel i. This serves as our structural assumption.

Our estimation problem can now be viewed as consisting of two parts. In order to efficiently estimate the function $\theta(.)$ at pixel coordinates i we need to describe a local model, i.e. to assign weights $W_i = \{w_{i1}, \dots, w_{in}\}$. If we knew the neighborhood U_i by an oracle we would define local weights as $w_{ij} = w_j(i) = I_{j \in U_i}$ and use these weights to estimate θ_i as

$$\widehat{\theta}_i = \sum_j w_{ij} Y_j. \tag{1}$$

Since θ and therefore U_i are unknown the assignments will have to depend on the information on θ that we can extract from the observed data. If we have good estimates $\widehat{\theta}_j$ of θ_j we can use this information to infer on the set U_i by testing the hypothesis

$$H : \theta_j = \theta_i. \tag{2}$$

A weight w_{ij} can be assigned based on the value of a test statistic T_{ij}, assigning zero weights if $\widehat{\theta}_j$ and $\widehat{\theta}_i$ are significantly different. This provides us with a set of weights $W_i = \{w_{i1}, \dots, w_{in}\}$ that determines a local model in i.

Given the local model we can then estimate our function $\theta(.)$ in each pixel i by (1).

We utilize both steps in an iterative procedure. We start with a very local model in each point i given by weights

$$w_{ij}^{(0)} = K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(0)}) \tag{3}$$

with $\boldsymbol{l}_{ij}^{(0)} = |i-j|/h^{(0)}$ and $|i-j|^2 = (i_h - j_h)^2 + (i_v - j_v)^2$. The initial bandwidth $h^{(0)}$ is chosen very small. K_{loc} is a kernel function supported on $[-1, 1]$, i.e. weights vanish outside a ball $U_i^{(0)}$ of radius $h^{(0)}$ centered in i. We then iterate two steps, estimation of $\theta(x)$ and refining the local models. In the kth iteration new weights are generated as

$$\begin{aligned} w_{ij}^{(k)} &= K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(k)}) K_{\mathrm{st}}(\boldsymbol{s}_{ij}^{(k)}) \quad \text{with} && (4) \\ \boldsymbol{l}_{ij}^{(k)} &= |i-j|/h^{(k)} \quad \text{and} \quad \boldsymbol{s}_{ij}^{(k)} = T_{ij}^{(k)}/\lambda. && (5) \end{aligned}$$

The bandwidth h is increased by a constant factor with each iteration k. The test statistic for (2)

$$T_{ij}^{(k)} = N_i^{(k-1)} \mathcal{D}(\widehat{\theta}_i^{(k-1)}, \widehat{\theta}_j^{(k-1)}; \widehat{\nu}_i) \tag{6}$$

with $N_i^{(k-1)} = \sum_j w_{ij}^{(k-1)}$ is used to specify the penalty $\boldsymbol{s}_{ij}^{(k)}$. This term effectively measures the statistical difference of the current estimates in i and j. In (6) the term $\mathcal{D}(\theta, \theta'; \nu)$ denotes a distance between the probability measures $P_{\theta,\nu}$ and $P_{\theta',\nu}$.

3.1 Adaptive weights smoothing

We now formally describe the resulting algorithm. For clarity of presentation we, for a moment, defer the problem of estimating ν_i and modifications due to spatial correlation in the image.

Image type	Stochastic penalty $\boldsymbol{s}_{ij}^{(k)}$	ν_i
Gray scale images 1.1	$\frac{N_i^{(k-1)}}{\lambda \widehat{\sigma}_i^2}(\widehat{\theta}_i^{(k-1)} - \widehat{\theta}_j^{(k-1)})^2$	σ_i^2
Color images 1.2	$\frac{N_i^{(k-1)}}{\lambda}(\widehat{\theta}_i^{(k-1)} - \widehat{\theta}_j^{(k-1)})^T \widehat{\Sigma}_i^{-1}(\widehat{\theta}_i^{(k-1)} - \widehat{\theta}_j^{(k-1)})$	Σ_i

Table 1: Stochastic penalties for examples in Section 1.

- **Initialization:** Set the initial bandwidth $h^{(0)}$ and compute, for every i the statistics

$$N_i^{(0)} = \sum_j w_{ij}^{(0)}, \quad \text{and} \quad S_i^{(0)} = \sum_j w_{ij}^{(0)} Y_j \tag{7}$$

and the estimates

$$\widehat{\theta}_i^{(0)} = S_i^{(0)} / N_i^{(0)} \tag{8}$$

using $w_{ij}^{(0)} = K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(0)})$. Set $k = 1$ and $h^{(1)} = c_h h^{(0)}$ for some constant $c_h > 1$.

- **Adaptation:** For every pair i, j, compute the penalties

$$\begin{aligned} \boldsymbol{l}_{ij}^{(k)} &= |i - j| / h^{(k)}, \qquad (9) \\ \boldsymbol{s}_{ij}^{(k)} &= \lambda^{-1} T_{ij}^{(k)} = \lambda^{-1} N_i^{(k-1)} \mathcal{D}(\widehat{\theta}_i^{(k-1)}, \widehat{\theta}_j^{(k-1)}; \widehat{\nu}_i). \qquad (10) \end{aligned}$$

Now compute the weights $w_{ij}^{(k)}$ as

$$w_{ij}^{(k)} = K_{\mathrm{loc}}\big(\boldsymbol{l}_{ij}^{(k)}\big) K_{\mathrm{st}}\big(\boldsymbol{s}_{ij}^{(k)}\big)$$

and specify the local model by $W_i^{(k)} = \{w_{i1}^{(k)}, \ldots, w_{in}^{(k)}\}$.

- **Local estimation:** Now compute new local MLE estimates $\widehat{\theta}_i^{(k)}$ of $\theta(x_i)$ as

$$\widehat{\theta}_i^{(k)} = S_i^{(k)} / N_i^{(k)} \qquad \text{with} \qquad N_i^{(k)} = \sum_j w_{ij}^{(k)}, \quad S_i^{(k)} = \sum_j w_{ij}^{(k)} Y_j .$$

- **Stopping:** Stop if $h^{(k)} \geq h_{\max}$, otherwise set $h^{(k)} = c_h h^{(k-1)}$, increase k by 1 and continue with the adaptation step.

3.2 Examples

The algorithm provided in 3.1 can be adjusted to meet the properties of different types of images, that is different types of distributions of Y, by an appropriate choice of the statistical penalty $\boldsymbol{s}_{ij}$ in (9). For gray scale images 1.1 and color images 1.2 the appropriate penalty is given in Table 1.

3.3 Choice of parameters - Propagation condition

The proposed procedure involves several parameters. The most important one is the scale parameter λ in the statistical penalty s_{ij}. The special case $\lambda = \infty$ simply leads to a kernel estimate with bandwidth $h_{\max}$. We propose to chose λ as the smallest value satisfying a propagation condition. This condition requires that, if the local assumption is valid globally, i.e. $\theta(x) \equiv \theta$ does not depend on x, then with high probability the final estimate for $h_{\max} = \infty$ coincides in every point with the global estimate. More formally we request that in this case for each iteration k

$$\boldsymbol{E}_I |\widehat{\theta}^{(k)}(I) - \theta| < (1+\alpha) \boldsymbol{E}_I |\check{\theta}^{(k)}(I) - \theta| \tag{11}$$

for a specified constant $\alpha > 0$. Here

$$\check{\theta}^{(k)}(i) = \sum_j K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(k)}) Y_j / \sum_j K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(k)}) \tag{12}$$

denotes the nonadaptive kernel estimate employing the bandwidth $h^{(k)}$ from step k. The value λ provided by this condition is characteristic for the selected class of error distributions, but does not depend on the unknown model parameter θ. A default value for λ can therefore be found by simulations. In the examples below they are selected for a value of $\alpha = 0.1$.

The second parameter of interest is the maximal bandwidth $h_{\max}$ which controls both numerical complexity of the algorithm and smoothness within homogeneous regions.

Additionally we specify a number of parameters and kernel functions that have less influence on the resulting estimates. As a default the kernel functions are chosen as $K_{\mathrm{loc}}(x) = (1 - x^2)_+^2$ and $K_{\mathrm{st}}(x) = (1 - x^2)_+$. The initial bandwidth $h^{(0)}$ is chosen as 1. The bandwidth is increased after each iteration by a default factor $c_h = 1.25^{1/2}$.

3.4 Correction for spatial correlation

In a CCD sensor, the common device within a consumer digital camera, a Bayer mask is used to obtain color information. Within a square of four pixel one is red filtered, one blue and two green, that is red and blue are recorded in every fourth pixel while green is obtained in every second. To obtain a color image in full resolution the missing color values are filled in by interpolation (demosaicing). This results in spatial correlation within the image. Other sources of spatial resolution are noise reduction filters applied within the camera as well as effects from camera movements or their automatic corrections.

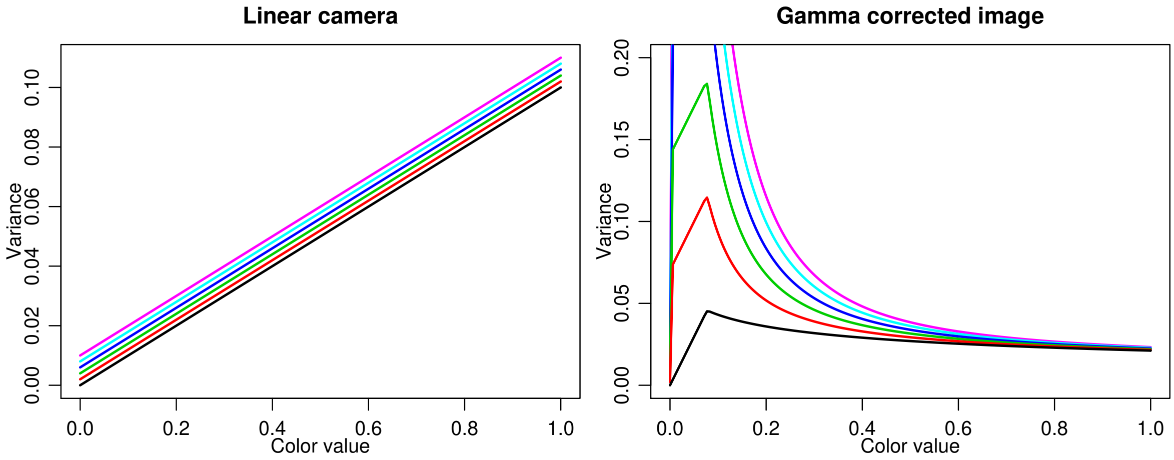

Figure 1: Mean-variance dependence for a perfectly linear camera and for corresponding gamma-corrected images $\gamma = 2.4$.

The statistical penalty $\boldsymbol{s}_{ij}$ effectively measures the probability of observing the estimate $\widehat{\theta}_i$ if the correct parameter θ_i equals $\widehat{\theta}_j$. The penalty heavily depends on a correct assessment of the variability of $\widehat{\theta}_i$. In case of (positive) spatial correlation the variance reduction achieved will be reduced and therefore we need to adjust the statistical penalty.

As a rough model we assume the spatial correlation to be caused by convolution with a Gaussian kernel with unknown bandwidths $g = (g_h, g_v)$ in horizontal and vertical direction. This leads to a multiplicative correction factor $C_i(g, h) = \frac{\sum_l \left[\sum_j K(\frac{|i-j|}{h}) K(\frac{|j-l|}{g})\right]^2}{Q_{h;i} \quad Q_{g;i}}$ with $Q_{h;i} = \sum_j K(\frac{|j-i|}{h})^2$ in the statistical penalty.

3.5 Variance mean dependence

Error variances in the recorded images are usually unknown and to be estimated. Depending on the imaging process error variances may be heteroscedastic. Special examples consists of confocal microscopy or perfectly linear cameras where the signal can be viewed as a linear combination of the number of recorded photons (Poisson counts) and termal noise (approximately homoscedastic Gaussian). Such a situation is characterized by a linear dependence between variance and the expected signal Tian et al. (2001); Hirakawa and Parks (2005). A linear behavior may also serve as an approximation in case of RAW-images. In contrast images recorded in common formats like TIFF or JPEG are *gamma corrected*, that is a nonlinear transformation has been applied to the measured intensities. Except for low color values a constant variance may be a reasonable assumption for such images, see Figure 1.

3.6 Estimation of variances and spatial correlation

Problems frequently encountered in image processing are unknown error variances, or even variance inhomogeneity, spatial correlation and correlated errors between channels in color images.

We try, in each color channel, to estimate both the spatial correlation and the, possibly heteroscedastic, error variances. These estimates are improved within the estimation process.

Let l specify the index of the color channel and $r_{il}^{(k)} = Y_{il} - \widehat{\theta}_{il}^{(k)}$ denote the residuals with respect to the reconstructed image $\widehat{\theta}_{il}^{(k)}$. For small bandwidths $h^{(k)} < 2$ $\widehat{\theta}_{il}^{(k)}$ is replaced by a non-adaptive kernel estimated using bandwidth 2.

We estimate the spatial correlation as in color channel l as

$$
\begin{aligned}
\mu_l^{(k)} &= 1/n \sum_{i=1}^{n} r_{il}^{(k)} && (13)\\
v_l^{(k)} &= 1/n \sum_{i=1}^{n} (r_{il}^{(k)})^2 \quad - (\mu_l^{(k)})^2 && (14)\\
\widetilde{\rho}_h &= \sum_{i_h=1}^{n_h-1} \sum_{i_v=1}^{n_v} (r_{(i_h,i_v)l}^{(k)} - \mu_l^{(k)})(r_{(i_h+1,i_v)l}^{(k)} - \mu_l^{(k)})/v_l^{(k)} && (15)\\
\widetilde{\rho}_v &= \sum_{i_h=1}^{n_h} \sum_{i_v=1}^{n_v-1} (r_{(i_h,i_v)l}^{(k)} - \mu_l^{(k)})(r_{(i_h,i_v+1)l}^{(k)} - \mu_l^{(k)})/v_l^{(k)} && (16)
\end{aligned}
$$

These estimates are severely biased due to the spatial correlation of the original data. The bias vanishes as $h_{(k)}$ increases. We apply an approximate bias correction using a polynomial model for the correlation in $\widetilde{\rho}$, $(h^{(k)})^{-1/2}$, and $(h^{(k)})^{-1}$.

In case of color images estimates of the correlations between channels, that is $\rho_{1,2}, \rho_{1,3}, \rho_{2,3}$, can be obtained as

$$\widehat{\rho}_{l,m} = \sum_{i=1}^{n} (r_{il}^{(k)} - \mu_l^{(k)})(r_{im}^{(k)} - \mu_m^{(k)})/\sqrt{v_l^{(k)} v_m^{(k)}}. \qquad (17)$$

Variances within channels can again be estimated from adjusted residuals

$$r_{il}^{(k)} = (Y_{il}^{1/2} - \widehat{\theta}_{il}^{(k)}) \qquad (18)$$

as

$$\widehat{\sigma}_l^2 = \sum_{i:N_i^{(k)}>1} \frac{N_i^{(k)}}{N_i^{(k)}-1} (r_{il}^{(k)})^2 (N_i^{(k)} - 1) / \sum_{i:N_i^{(k)}>1} (N_i^{(k)} - 1) \qquad (19)$$

in case of constant variances. The weights $(N_i^{(k)} - 1)$ reflect the varying information content of the residuals while the factor $N_i^{(k)}/N_i^{(k)} - 1$ corrects for the bias of the squared residuals.

A linear variance model can be evaluated by solving

$$\frac{N_i^{(k)}}{N_i^{(k)} - 1}(r_{il}^{(k)})^2 = a_l + b_l \widehat{\theta}_{il}^{(k)}, \quad i : N_i^{(k)} > 1 \tag{20}$$

by weighted least squares with weights $(N_i^{(k)} - 1)$ leading to variance estimates

$$\widehat{\sigma}_{il}^2 = \widehat{a}_l + \widehat{b}_l \widehat{\theta}_{il}^{(k)}. \tag{21}$$

For color images estimates of covariance matrices Σ_i are then obtained as

$$\widehat{\Sigma}_i = \begin{pmatrix} \widehat{\sigma}_{i1}^2 & \widehat{\rho}_{12}\widehat{\sigma}_{i1}\widehat{\sigma}_{i2} & \widehat{\rho}_{13}\widehat{\sigma}_{i1}\widehat{\sigma}_{i3} \\ \widehat{\rho}_{12}\widehat{\sigma}_{i1}\widehat{\sigma}_{i2} & \widehat{\sigma}_{i2}^2 & \widehat{\rho}_{23}\widehat{\sigma}_{i2}\widehat{\sigma}_{i3} \\ \widehat{\rho}_{13}\widehat{\sigma}_{i1}\widehat{\sigma}_{i3} & \widehat{\rho}_{23}\widehat{\sigma}_{i2}\widehat{\sigma}_{i3} & \widehat{\sigma}_{i3}^2 \end{pmatrix} \tag{22}$$

assuming that correlations between channels do not depend on θ_i.

The proposed procedure, employing a local constant structural assumption, is implemented in function `awsimage`.

3.7 Local polynomial Propagation-Separation (PS) approach

Until now we assume that the gray or color value is locally constant. This assumption is essentially used in the form of the stochastic penalty s_{ij}. The effect can be viewed as a regularization in the sense that in the limit for $h_{max} \to \infty$ the reconstructed image is forced to a local constant gray value or color structure even if the true image is locally smooth. Such effects can be avoided if a local polynomial structural assumption is employed. Due to the increased flexibility of such models this comes at the price of a decreased sensitivity to discontinuities.

The Propagation-Separation approach from Polzehl and Spokoiny (2004) assumes that within a homogeneous region containing pixel i, i.e. for $j = (j_h, j_v) \in U_i$, the gray value or color Y_j can be modelled as

$$Y_j = \theta_i^\top \Psi(j_h - i_h, j_v - i_v) + \varepsilon_{j_h,j_v}, \tag{23}$$

where the components of $\Psi(\delta_h, \delta_v)$ contain values of basis functions

$$\psi_{m_1,m_2}(\delta_h, \delta_v) = (\delta_h)^{m_1}(\delta_v)^{m_2} \tag{24}$$

for integers $m_1, m_2 \geq 0$, $m_1 + m_2 \leq p$ and some polynomial order p. For a given local model W_i estimates of θ_i are obtained by local Least Squares as

$$\widetilde{\theta}_i = B_i^{-1} \sum_j w_{ij} \Psi(j_h - i_h, j_v - i_v) Y_j, \tag{25}$$

with

$$B_i = \sum_j w_{ij} \Psi(j_h - i_h, j_v - i_v) \Psi(j_h - i_h, j_v - i_v)^\top. \tag{26}$$

The parameters θ_i are defined with respect to a system of basis functions centered in i. The component of the estimate of θ in the local model W_j, corresponding to the basis function ψ_{m_1,m_2} centered at i can be obtained by a linear transformation as

$$\widehat{\theta}^{(k-1)}_{j(i),m} = \sum_{m':|m'| \leq p-|m|} \binom{m+m'}{m} \widehat{\theta}^{(k-1)}_{j,m+m'} (i-j)^{m'}, \qquad |m| \leq p. \tag{27}$$

In iteration k a statistical penalty can now, for grayscale images, be defined as

$$s^{(k)}_{ij} = \frac{1}{\lambda 2\sigma^2} \big(\widehat{\theta}^{(k-1)}_i - \widehat{\theta}^{(k-1)}_{j(i)}\big)^\top B_i \big(\widehat{\theta}^{(k-1)}_i - \widehat{\theta}^{(k-1)}_{j(i)}\big)\big). \tag{28}$$

In case of color images with independent errors between channels a natural generalization occurs as a possibly weighted sum of the statistical penalties for the three color channels.

A detailed description and discussion of the resulting algorithm and corresponding theoretical results can be found in Polzehl and Spokoiny (2004).

Local linear and local quadratic models are implemented for both grayscale and color images in function `awspimage`. The implementation allows for homogeneous spatial correlation and unknown variances that are either constant or a linear function of the gray- or colorvalues.

4 Examples

The package **adimpro** can basically read and write the pgm and ppm image format. However the package allows to read and write all image formats supported by ImageMagick ImageMagick Studio LLC (2006), if this is installed on the system. For reading camera RAW formats the package **adimpro** uses dcraw Coffin (2006). Both programs are available for many operating systems. However, if you can not or don't want to install them on your system you must rely on some other software or R package to read the image data into the R session or to transform the image data into portable pixmap format. Since image conversion is not the target of the package and excellent software for this purpose exists we do not plan to extend its capabilities in the future.

Reading an image file from disk is done by one of the following statements:

```
R> img <- read.image(”image.tif”)
```

```
R> img <- read.raw(”image.raw”)
```

Both statements create an object of class `”adimpro”`. Such objects are lists containing the gray- or color values as well as image metadata.

Alternatively an object of class `”adimpro”` can be generated from image data provided as an array, with the first two dimensions corresponding to the image dimension and the third being 1, or 3 for grayscale and color valued images, respectively. This is achieved by

```
R> img <- make.image(imgdata).
```

In the package **adimpro** we provide two functions which implement the Propagation-Separation approach described in the preceeding sections. The function `awsimage` uses a local constant model while `awspimage` considers local polynomial models up to a degree of 2.

The complexity of the algorithm is mainly determined by h_{max} which corresponds to the maximum bandwidth used in the iteration process (see Section 3 for details).

```
R> img.smooth <- awsimage(img, hmax = 4)
```

The function `awsimage` returns again an object of class ”adimpro”.

In Figure 2 we show an example of image smoothing under controlled noise conditions. An image of good quality and low noise has been taken using a CANON 300D with ISO 100. Spatially correlated homogeneous noise has been added.

As we already mentioned before, the assumption of homogeneous noise may only be suitable for gamma corrected images. However, if the camera provides RAW format, noise is characterized by a linear dependence between its variance and the mean intensity of light at each pixel, aside from saturation effects. Thus for images in RAW format, it is possible to specify a model for the variance mean dependence, namely a linear model.

Figure 3 shows a test image with nine homogeneous colored regions taken by a Panasonic Lumix LX2 under extreme lighting conditions. ISO 1600 has been used, therefore the image has a relatively strong noise component. The result of adaptive smoothing is given on the right side of the figure. We read and processed the RAW image, as obtained by dcraw Coffin (2006) using bilinear interpolation for demosaicing. The estimates for the spatial correlation and the correlation between the color channels are given in Table 2 and 3.

Furthermode we demonstrate the effects of the use of the linear variance model, which is

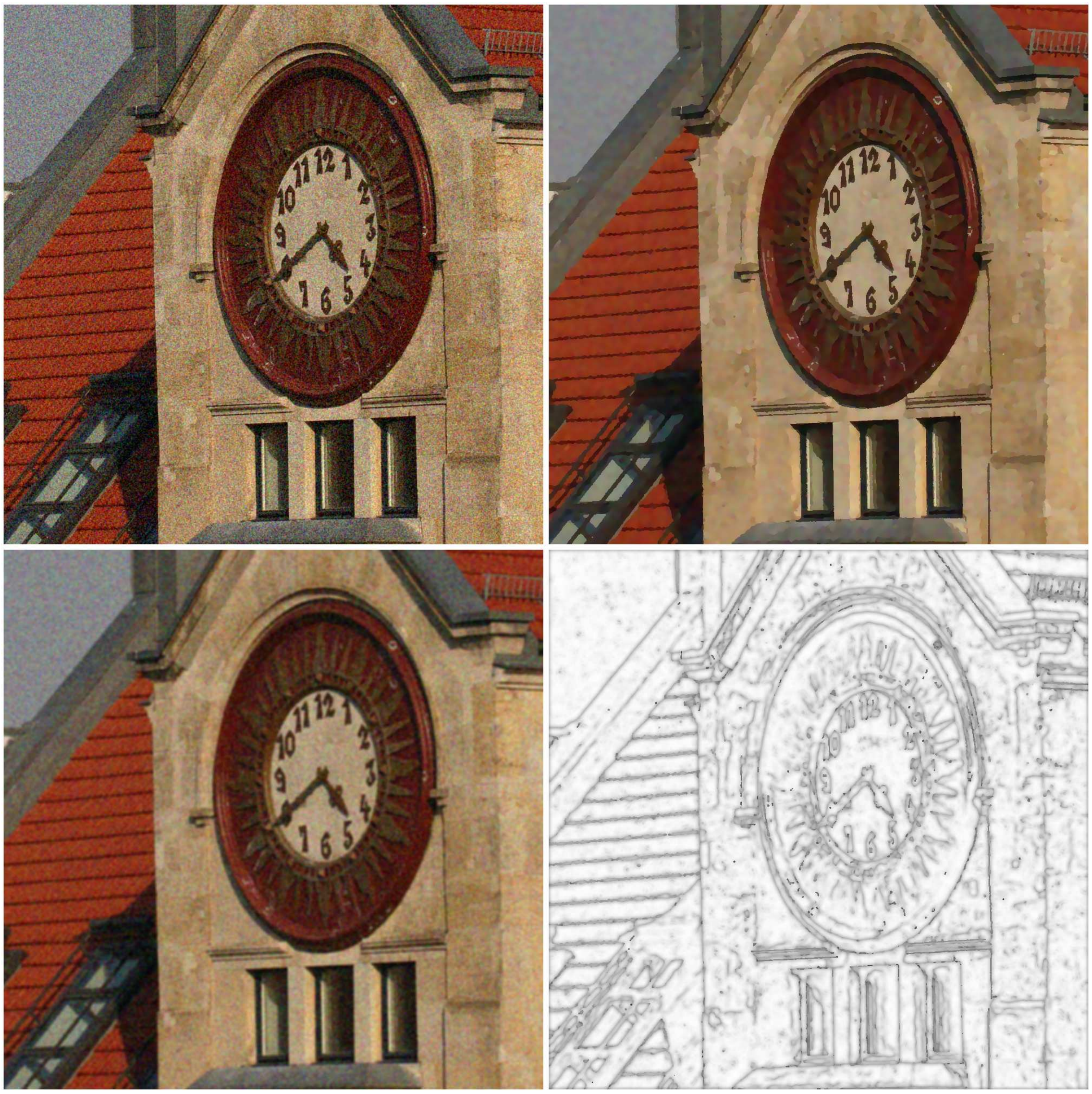

Figure 2: Smoothing a noisy image with a local constant model as structural assumption. Upper left: Noisy image. Uniform but spatially correlated noise has been added to an image taken with a digital camera. Upper right: Result of the `awsimage` function with $h_{max} = 10$. Lower left: non-adaptive kernel smoothing using a bandwidth that minimizes MSE on the whole image. Lower right: Image illustrating the amount of adaptation used in the iteration procedure in `awsimage`. Dark areas correspond to stronger adaptation, while white areas show non-adaptive behavior. The latter correspond to homogeneous regions with respect to the specified model, while darks lines are caused by detected structure.

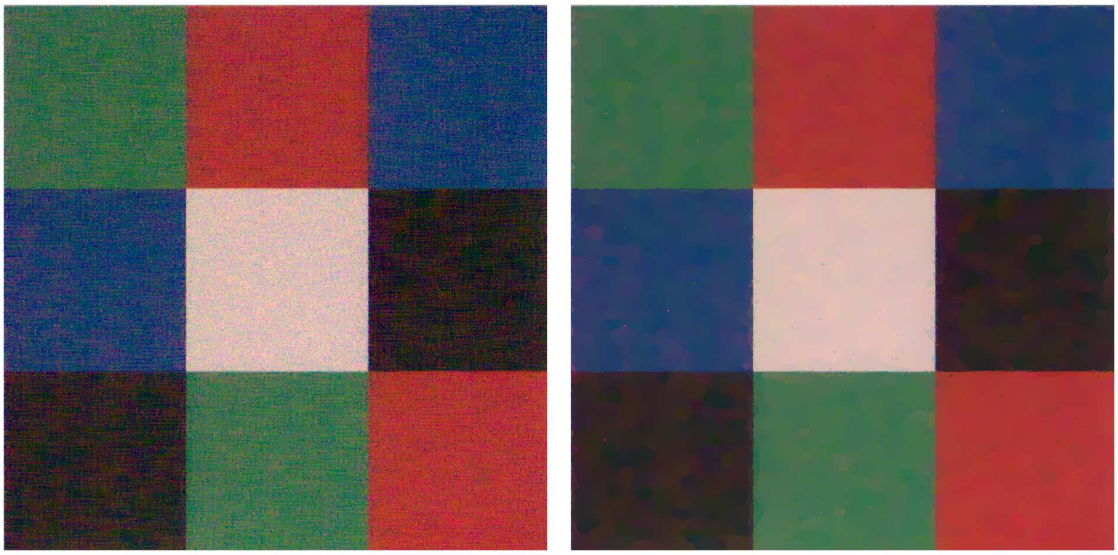

Figure 3: Adaptive smoothing of a noisy test image using a local constant model for the color values and a linear model for the variance mean dependence. $h_{max} = 22.4$ has been used. Left: original image Right: smoothed image.

	red channel	green channel	blue channel
horizontal	0.66	0.42	0.63
vertical	0.62	0.40	0.61

Table 2: Estimated spatial correlation in horizontal and vertical direction in the three color channels for the example image used in Figure 3.

red/green	red/blue	green/blue
-0.15	-0.0023	-0.58

Table 3: Estimated correlation between the color channels for the image used in Figure 3.

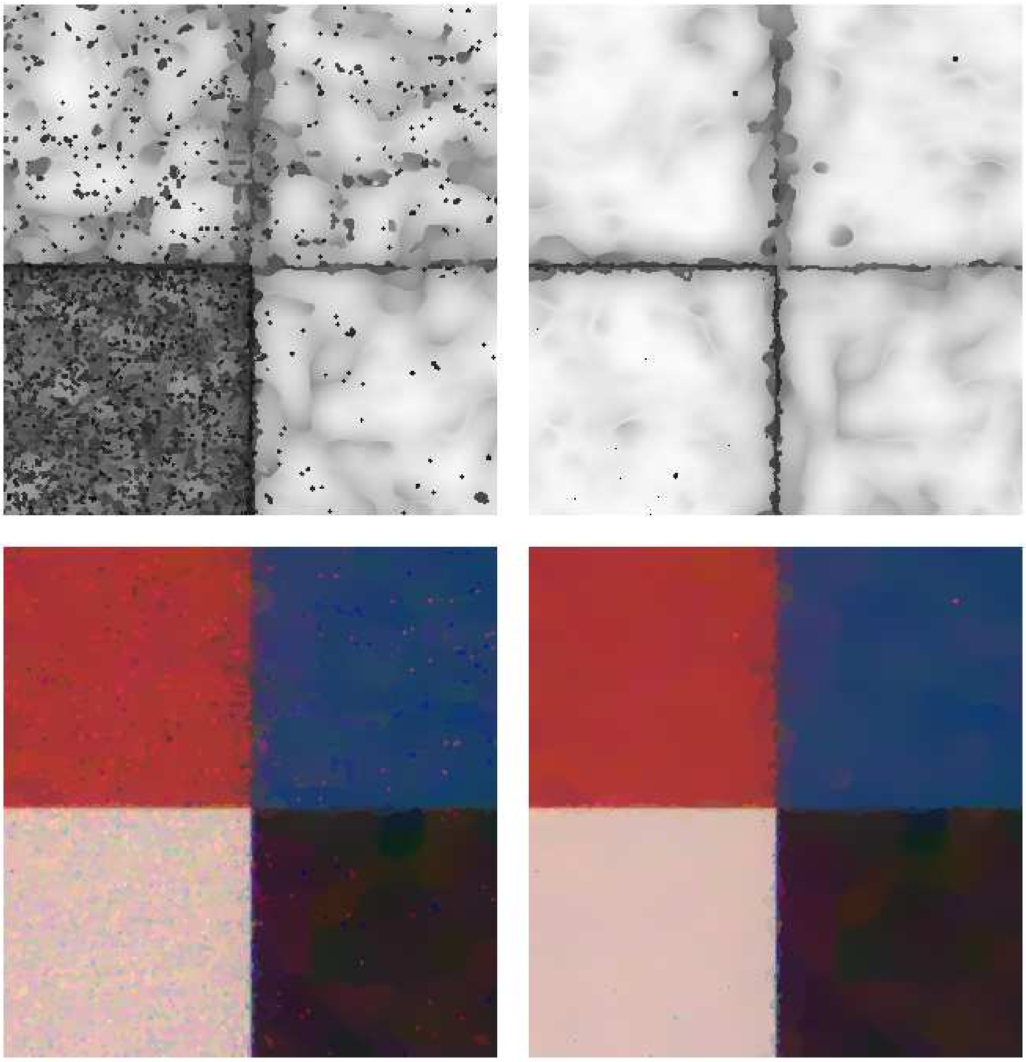

Figure 4: Denoising the image in Figure 3 using a constant (left) and a linear variance model (right). Part of the image in Figure 3 is shown here. Again $h_{max} = 22.7$ has been used. The left column corresponds to a constant variance model, while the right column shows the corresponding results for a linear variance model. The upper images show the degree of adaptation to structure through the sum of weights in the two cases as in Figure 2. The edges between the colored regions can be clearly identified in both models. While there is only little structure in the homogeneous region, which is mainly due to the heavy noise, the bright square shows a lot of structure when smoothing with a constant variance model. Since noise for RAW images consists of two additive components which are approximately proportional to the mean (photon counts) and constant (termal noise) the linear variance model is correct, a constant model underestimates the variance in the bright region. Thus smaller differences of the color value are detected as significant with higher probability. The procedure therefore partly adapts to the noise structure.

	red channel	green channel	blue channel
black area	0.0229	0.0122	0.0121
white area	0.0507	0.0351	0.0329
blue area	0.0218	0.0157	0.0216
red area	0.0414	0.0145	0.0141
	0.0234	0.0167	0.0151

Table 4: Estimated value for the variance in the three color channels in four areas of different mean color for the example image used in Figure 3. We considered the four areas in the upper right corner of the image. The first four rows of the table give the mean estimates for a linear variance mean model, while the last row contains the corresponding values for a constant variance mean dependence. The red value show cases, where in a constant model, the variance is underestimated in contrast to the (true) linear model and thus the smoothing procedure partly adapts to the noise. The effects are illustrated in Figure 4.

	red channel	green channel	blue channel
Estimated constant parameter	0.000378	0.0000824	0.0000775
Estimated linear parameter	0.00573	0.00404	0.00382

Table 5: Estimated parameters for the linear variance model in the three color channels for the image used in Figure 3.

the correct one for RAW-images as described above. When smoothing an image using an inadequate constant variance model we underestimate the variance where color values are large. This leads to a segmentation of bright homogeneous regions through adaptation to the noise component. A further negative bias in the estimated variance is introduced when estimating the variance again from residuals in the iterative process. We demonstrate the effect in Figure 4 and Tables 4 and 5, where the use of a linear and constant variance model is compared.

As a last example we compare in Figure 5 the results of adaptively smoothing an image taken by a CANON PowerShot S30 at 800 ISO with a local constant and a local quadratic model.

5 Acknowledgments

This work is supported by the DFG Research Center Matheon.

Figure 5: Smoothing an image with different structural assumptions. Left: Original image. Center: Result of smoothing with a local constant model and $h_{max} = 11.6$. Right: Same for a local quadratic model and $h_{max} = 18.4$. This demonstrates that a local quadratic model leads to smoother impression at homogeneous regions at the cost of a slightly decreased sensitivity at edges.

References

Bayer, B. (1976). Color imaging array. U.S. Patent 3 971 065.

Coffin, D. (2006). *dcraw*.

Gonzalez, R. C. and Woods, R. (2002). *Digital Image Processing*. Prentice Hall.

Hirakawa, K. and Parks, T. (2005). Image denoising for signal-dependent noise. In *Proc. Accoustics, Speech, and Signal Processing*, volume 2, pages 29–32.

ImageMagick Studio LLC (2006). *ImageMagick*.

Malacara, D. (2002). *Color Vision and Colorimetry: Theory and Applications*. SPIE Optical Engineering Press, Bellingham, Wash.

Polzehl, J. and Spokoiny, V. (2000). Adaptive weights smoothing with applications to image restoration. *J. Roy. Statist. Soc. Ser. B*, 62:335–354.

Polzehl, J. and Spokoiny, V. (2001). Functional and dynamic magnetic resonance imaging using vector adaptive weights smoothing. *J. Roy. Statist. Soc. Ser. C*, 50:485–501.

Polzehl, J. and Spokoiny, V. (2004). Spatially adaptive regression estimation: Propagation-separation approach. Preprint 998, WIAS.

Polzehl, J. and Spokoiny, V. (2006). Local likelihood modeling by adaptive weights smoothing. *Probab. Theory Relat. Fields*, 135:335–362.

Polzehl, J. and Spokoiny, V. (2007). *Handbook of Data Visualization III*, chapter Structural adaptive smoothing by Propagation-Separation-methods. Handbooks of Computational Statistics. Springer.

R Development Core Team (2005). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Tabelow, K., Polzehl, J., Voss, H. U., and Spokoiny, V. (2006). Analysing fMRI experiments with structure adaptive smoothing procedures. *NeuroImage*, 33(1):55–62.

Tian, H., Fowler, B., and Gamal, A. (2001). Analyis of temporal noise in cmos photodiode active pixel sensor. *IEEE Journal Solid-State Circuits*, 36(1):92–101.