

NONLINEAR IMAGE PROCESSING AND FILTERING: A UNIFIED APPROACH BASED ON VERTICALLY WEIGHTED REGRESSION

EWARYST RAFAJŁOWICZ *, MIROŚLAW PAWLAK *, ANGSAR STELAND **,***

* Institute of Computer Engineering, Control and Robotics
Wrocław University of Technology
ul. Wybrzeże Wyspiańskiego 27, 50–370 Wrocław, Poland
e-mail: ewaryst.rafajlowicz@pwr.wroc.pl

** Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg, Manitoba, R3T5V6, Canada

*** RWTH Aachen University
Aachen, Germany

A class of nonparametric smoothing kernel methods for image processing and filtering that possess edge-preserving properties is examined. The proposed approach is a nonlinearly modified version of the classical nonparametric regression estimates utilizing the concept of vertical weighting. The method unifies a number of known nonlinear image filtering and denoising algorithms such as bilateral and steering kernel filters. It is shown that vertically weighted filters can be realized by a structure of three interconnected radial basis function (RBF) networks. We also assess the performance of the algorithm by studying industrial images.

Keywords: Image filtering, vertically weighted regression, nonlinear filters.

1. Introduction

Image filtering and reconstruction algorithms have played the most fundamental role in image processing and analysis. The problem of image filtering and reconstruction is to obtain a better quality image $\hat{\theta}$ from a noisy image $y = \{y_i, 1 \leq i \leq N\}$ recorded over N pixel points $\{\mathbf{x}_i, 1 \leq i \leq N\}$. In image filtering we are only interested in getting the image $\hat{\theta}$ at pixel points, whereas in image reconstruction one wishes to compute $\hat{\theta}$ at any point of an image plane. As such, the former task is a digital-to-digital mapping, whereas the latter one is a digital-to-analog mapping. There are numerous specialized algorithms in the image processing literature dealing either with image filtering or image reconstruction. The nature of these methods depends critically on the assumed image and noise models. In image filtering techniques utilizing deterministic noise models and variational framework, one finds a cleaner image $\hat{\theta}$ by minimizing the image regularity term penalized by a difference between

the clean image and the observed noisy image, i.e., we seek $\hat{\theta}$ as a solution to the following variational problem:

$$\hat{\theta} = \arg \min_{\alpha \in \Theta} (\|\alpha\|_{\Theta} + \lambda \|y - \alpha\|),$$

where Θ is the image space equipped with the norm $\|\cdot\|_{\Theta}$. Here λ is the regularized parameter and the image space Θ must be specified (Buades *et al.*, 2005) by the user. In the Rudin-Osher-Fatemi theory, Θ is the space of functions of bounded variation and λ is specified by *ad-hoc* methods. An image being a solution to the variational problem cannot be given in an explicit way and must be obtained by numerical algorithms, see (Buades *et al.*, 2005) and the references cited therein. Furthermore, in many applications (e.g., images transmitted by a communication channel) the noise present in an image has a stochastic nature and we must develop statistical methods for image filtering and reconstruction.

In this article we focus on statistical methods employing a modern nonparametric regression analysis. Classical image processing methods (Jain, 1989) rely on

a specific parametric model of an underlying image and, additionally, it is commonly assumed that the noise process is an additive white Gaussian random variable. On the contrary, nonparametric statistical methods rely on the observed data themselves and consequently they are able to adapt to virtually any image shape and noise distribution. Furthermore, the noise process need not be additive and may have a complex dependence structure. Nonparametric inference such as density estimation, nonparametric regression, wavelets, and bootstrapping have been extensively examined in the statistical literature, see (Efromovich, 1999; Wasserman, 2006) for an extensive overview of various nonparametric techniques. Surprisingly, the use of nonparametric methods in image processing has been very limited, see (Hall and Koch, 1992; Pawlak and Rafajłowicz, 2001; Pawlak and Liao, 2002; Chiu *et al.*, 1998; Polzehl and Spokoiny, 2000) for some preliminary studies. In the context of image filtering and reconstruction it is natural to view the underlying image model as a nonparametric regression function. Nevertheless, standard nonparametric regression algorithms are linear in observed data. In fact, most nonparametric regression estimates can be written as a generalized kernel method of the following generic form:

$$\hat{\theta}(\mathbf{x}) = \sum_{i=1}^N w_i(\mathbf{x}, \mathbf{x}_i, h) y_i,$$

where the weights $\{w_i(\mathbf{x}, \mathbf{x}_i, h)\}$ define the local neighborhood at the point \mathbf{x} controlled by the smoothing parameter h . The linearity of the above scheme yields a number of limitations in the accuracy. Indeed, linear methods over-smooth edges—an essential component of an image structure. This fact has been recognized recently in the image processing literature (Takeda *et al.*, 2007), where a nonlinear version of kernel smoothers has been proposed.

In this paper we propose a class of image processing filtering methods that generalizes the previous existing techniques which have only been intuitively justified. In particular, our approach covers, as a special case, neighborhood filters, bilateral filters, adaptive smoothing, and the SUSAN algorithm (Yaroslavsky, 1985; Saint-Marc and Medioni, 1991; Smith and Brady, 1997; Tomasi and Manduchi, 1998; Barner *et al.*, 1999; Barash, 2002; Elad, 2002; Buades *et al.*, 2005). It generalizes also the frequently overlooked methodology of nonparametric vertical smoothing, due originally to Lee (1983). This is done by putting our approach into a formal nonparametric framework (Chiu *et al.*, 1998; Polzehl and Spokoiny, 2000; Pawlak and Rafajłowicz, 1999; Pawlak and Rafajłowicz, 2001). In particular, in (Pawlak and Rafajłowicz, 1999; Pawlak and Rafajłowicz, 2001), a general nonparametric vertical regression algorithm was proposed for jump preserving signal reconstruction and filtering. This scheme was extended to a robust version of

the mean-squared error based filters by the introduction of a vertically clipped conditional median filter (Krzyżak *et al.*, 2001; Pawlak *et al.*, 2004; Steland, 2005).

This paper aims at extending the previous nonlinear approaches to the problem of image filtering and reconstruction. We also demonstrate the usefulness of our approach for the problem of industrial image processing. The paper is organized as follows. In Chapter 2 we introduce the concept of a Vertically Weighted Regression (VWR) function. The applicability of this function in forming various image processing tasks is pointed out. A fundamental nonlinear equation for an image characterization is derived. Image filtering formulas with respect to both L_2 and L_1 loss functions are obtained. Section 3 gives empirical versions of the filters derived in Section 2. Important special cases of these filters are discussed in Section 4. In Section 5 we propose a specialized neural network structure utilizing radial basis functions (RBF) to design nonlinear vertically weighted filters. Section 6 demonstrates the practical aspects of our methodology in the context of filtering industrial images.

2. Vertically weighted regression

2.1. Vertically weighted estimator of a location parameter.

2.1.1. Estimating a location parameter from a parameter dependent noise. In order to introduce the basic concepts of our paper, let us first consider a vertically weighted estimator of a location parameter. Let θ^* be a sought parameter observed in the presence of noise, i.e., we have the following observation model:

$$Y_j = \theta^* + \sigma(\theta^*) \cdot \varepsilon_j, \quad j = 1, 2, \dots, N, \quad (1)$$

where ε_j s are i.i.d. random variables such that $\mathbf{E} \varepsilon_j = 0$, $\mathbf{Var} \varepsilon_j = 1$. In (1) $\sigma(\theta)$ is a function, which is tentatively assumed to be known (later we admit $\sigma(\theta)$ to be more freely selected by the user). We do not require $\sigma(\theta)$ to be nonnegative, but $\sigma^2(\theta^*) > 0$ can be interpreted as the variance of errors.

Suppose that θ^* is a gray level of a selected pixel. Then one may interpret the above model as the estimation problem of a pixel gray level from the repeated observations, which are corrupted by random errors which possess the variance dependent on the pixel gray level. Clearly, in practice we usually do not have repeated observations from the same pixel, but it is useful to consider the observations from surrounding pixels as repeated observations of θ^* (gray level of the central pixel), which are additionally biased by the variability of the image in the vicinity of the selected pixel.

2.1.2. ML estimator of a location parameter. Although the idea of vertical weighting does not require the

existence of a probability density of $\varepsilon_{j,s}$, we tentatively assume that such a density, denoted by f_ε , exists, since it is convenient for interpretation purposes. Then by (1) the probability density function $f_Y(y)$ of the observed image Y is given by

$$f_Y(y) = f_\varepsilon((y - \theta^*)/\sigma(\theta^*)/\sigma(\theta^*)). \quad (2)$$

Then the likelihood function for estimating θ^* in (1) is of the form

$$L(\theta) = \prod_{j=1}^N \frac{1}{\sigma(\theta)} f_\varepsilon\left(\frac{Y_j - \theta}{\sigma(\theta)}\right). \quad (3)$$

The maximum likelihood (ML) estimator of θ^* , denoted further by $\hat{\theta}$, is the one for which $L(\theta)$ or, equivalently, the log-likelihood function

$$\bar{l}(\theta) = \sum_{j=1}^N \log f_\varepsilon\left(\frac{Y_j - \theta}{\sigma(\theta)}\right) - N \log(\sigma(\theta)) \quad (4)$$

attains its maximum. Note that if f_ε is the standard normal p.d.f., then the maximization of $l(\theta)$ is equivalent to the minimization of the following function:

$$l(\theta) = \sum_{j=1}^N \left(\frac{Y_j - \theta}{\sigma(\theta)}\right)^2 + N \log(\sigma(\theta)). \quad (5)$$

This equation is a first basic step in introducing a concept of generalized M -type parametric estimators and then their vertically weighted modifications. This is examined in the next subsection.

2.1.3. M -estimators of a location parameter. The formula (5) is justified if the noise process is Gaussian. The theory of M estimators (van der Vaart, 1998) relaxes this assumption by replacing $\log f_\varepsilon$ by a more general function, Ψ . Hence (5) takes the following form:

$$crit(\theta) = \sum_{j=1}^N \Psi\left(\frac{Y_j - \theta}{\sigma(\theta)}\right) + N \log(\sigma(\theta)), \quad (6)$$

where Ψ is selected in such a way that the existence of a minimum of (6) is guaranteed.

We need another informal step in order to make minimization of (6) easier. To do so, let us assume that $\sigma(\theta)$ is a slowly changing function, i.e., $|\mathrm{d}\sigma(\theta)/\mathrm{d}\theta|$ is small. Then, by Taylor's formula and the fact that $EY_j = \theta^*$, we can replace $\sigma(\theta)$ in (6) by $\sigma(Y_j)$. This leads to the following modified criterion:

$$\widetilde{crit}(\theta) = \sum_{j=1}^N \Psi\left(\frac{Y_j - \theta}{\sigma(Y_j)}\right) + N \log(\sigma(Y_j)) \quad (7)$$

Now, we can neglect the last term in (7) as being independent of θ and we finally use

$$Crit(\theta) = \sum_{j=1}^N \Psi\left(\frac{Y_j - \theta}{\sigma(Y_j)}\right) \quad (8)$$

as a criterion for estimating θ^* . The flexibility of this general estimation method is gained by various choices of Ψ and $\sigma(\cdot)$.

Let us also note that under mild conditions imposed on Ψ and $\sigma(\cdot)$, the law of large numbers implies that (with probability 1):

$$N^{-1} \sum_{j=1}^N \Psi\left(\frac{Y_j - \theta}{\sigma(Y_j)}\right) \longrightarrow \mathbf{E} \Psi\left(\frac{Y - \theta}{\sigma(Y)}\right), \quad (9)$$

as $N \rightarrow \infty$, where the expectation is calculated with respect to Y . The value of θ minimizing the right-hand side of (9) will be used in this paper as a preliminary step to design various vertically weighted filtering algorithms. This issue will be discussed in the next section.

2.2. Image processing and vertically weighted regression.

2.2.1. Image model.

Consider the image model

$$y_{ij} = \theta^*(x_{ij}) + \sigma(\theta^*(x_{ij})) \cdot \varepsilon_{ij} \quad (10)$$

for $i = 1, 2, \dots, n_1$ and $j = 1, 2, \dots, n_2$, where $\theta^* : \mathbb{R}^2 \rightarrow [0, 1]$ is an image function, which represents gray levels scaled, without loss of generality, to the interval $[0, 1]$. Next $x_{ij} \in \mathbb{R}^2$ is a 2-D vector, which represents a known location of the (i, j) -th pixel, $i = 1, 2, \dots, n_1$, $j = 1, 2, \dots, n_2$, where $n_1 n_2$ is the total number of pixels representing the image. Throughout the paper we assume that the noise process ε_{ij} s and the function $\sigma(\cdot)$ satisfy the conditions introduced in Section 2.1.

2.2.2. Vertically weighted regression. Our principal problem in this paper is to estimate the function $\theta^*(\cdot)$ from the noisy observations y_{ij} , $i = 1, 2, \dots, n_1$, $j = 1, 2, \dots, n_2$. As a by-product of our studies we also examine some image processing problems like image matching, segmentation, and motion. In these problems one wishes to recover a function being closely related to the original image $\theta^*(\cdot)$.

Let us begin with the generalized criterion, see (9), which characterizes our image recovery techniques, i.e., we have

$$Q(\theta) = \mathbf{E} \Psi\left(\frac{y_{ij} - \theta}{\sigma(y_{ij})}\right), \quad (11)$$

where θ is treated as a decision variable. Assuming that the minimum of $Q(\cdot)$ exists, let us denote by θ_{ij}^+ the minimizer of this criterion function, i.e.,

$$\theta_{ij}^+ = \arg \min_{\theta} \mathbf{E} \Psi \left(\frac{y_{ij} - \theta}{\sigma(y_{ij})} \right). \quad (12)$$

Note that θ_{ij}^+ depends on the true image $\theta^*(x_{ij})$. We shall call θ_{ij}^+ the vertically weighted regression function since the function $\sigma(\cdot)$ appearing in the definition of $Q(\cdot)$ depends on the observed image y_{ij} . It is important to note that, in general, θ_{ij}^+ differs from $\theta^*(x_{ij})$. In some important cases, however, we show that $\theta_{ij}^+ = \theta^*(x_{ij})$. As a result, estimates of θ_{ij}^+ can provide image recovery methods which are more accurate in terms of preserving edges and other image singularities.

2.3. Examples of vertically weighted regression functions. Before entering into the problem of estimating θ_{ij}^+ in (12), it is expedient to explore the flexibility of Q in 11 as a theoretical criterion for obtaining various forms of θ_{ij}^+ . In this respect, it is convenient to define the weight function $w(y) = 1/\sigma(y)$.

2.3.1. L_2 loss. In this section we choose $\Psi(t) = t^2$ in (11). This could correspond to the classical mean-squared error if, additionally, $w(y) = 1$. Otherwise we have the vertically weighted counterpart of this criterion, i.e., we have

$$Q_2(\theta) = \mathbf{E} \left[w^2(y_{ij}) \cdot (y_{ij} - \theta)^2 \right]. \quad (13)$$

It is straightforward to show that $Q_2(\theta)$ is minimized by

$$\theta_{ij}^+ = \mathbf{E} \left[y_{ij} \cdot w^2(y_{ij}) \right] / \mathbf{E} \left[w^2(y_{ij}) \right]. \quad (14)$$

Selecting different $w(\cdot)$, we obtain the following important special cases of vertically weighted regression:

MEAN VALUE. For $w(y) \equiv 1$ we have $\theta_{ij}^+ = \mathbf{E} (y_{ij})$. This is a classical solution yielding linear mean-type filters. Note that in this case the noise process is image independent.

HARMONIC MEAN. Selection $w(y) = \text{const}/\sqrt{y}$ yields

$$\theta_{ij}^+ = \frac{1}{\mathbf{E} (y_{ij}^{-1})}. \quad (15)$$

In this case $\sigma(y) \sim \sqrt{y}$, i.e., we have a moderate influence of the image on noise dispersion.

RELATIVE ERROR. If the dependence of the dispersion on y is linear ($w(y) = \text{const}/y$), then from (14) we obtain

$$\theta_{ij}^+ = \mathbf{E} (y_{ij}^{-1}) / \mathbf{E} (y_{ij}^{-2}). \quad (16)$$

This kind of averaging does not have a commonly accepted name, but we can give the following interpretation of (13). After substituting $w(y) = 1/y$ we obtain

$$Q_2(\theta) = \mathbf{E} \left(1 - \frac{\theta}{y_{ij}} \right)^2, \quad (17)$$

i.e., (16) minimizes the mean relative estimation error.

L_p MEAN. A greater influence of the image amplitude on the noise dispersion can be obtained if $\sigma(y) = y^{p/2}$ or, equivalently, $w(y) = y^{-p/2}$, where $p > 0$. Then

$$\theta_{ij}^+ = \mathbf{E} \left(y_{ij}^{(-p+1)} \right) / \mathbf{E} (y_{ij}^{-p}). \quad (18)$$

An interesting case occurs if $p \rightarrow \infty$. Then it can be easily shown that the solution in (18) corresponds to the average of the *Min* operation on the image y_{ij} .

Yet another L_p mean can be obtained if the smaller influence of the image amplitude on the noise dispersion is required. Thus, let $\sigma(y) = y^{-p/2}$ or $w(y) = y^{p/2}$, where $p > 0$. Then

$$\theta_{ij}^+ = \mathbf{E} \left(y_{ij}^{(p+1)} \right) / \mathbf{E} (y_{ij}^p). \quad (19)$$

It again can be easily shown that for $p \rightarrow \infty$ the empirical counterpart of (19) corresponds to the *Max* operation on the image function.

Hence we can conclude that for p ranging from $-\infty$ to ∞ , the family of operators in (19) smoothly covers the whole range from *Min* to *Max* operations, including the classical mean ($p = 0$) and the harmonic mean ($p = -1$).

INCORPORATING INFORMATION FROM EARLIER IMAGES. Let $\theta^*(\cdot)$ be the last image in a sequence to be processed. Denote by $\theta_{\text{old}}(\cdot)$ the image before last in this sequence. Suppose that we may expect that $\theta^*(\cdot)$ does not differ too much from $\theta_{\text{old}}(\cdot)$. In such a case it is reasonable to use information contained in $\theta_{\text{old}}(\cdot)$ for processing $\theta^*(\cdot)$. Assume for a while that ε_{ij} s are normally distributed with a zero mean and the dispersion $\sigma_\varepsilon > 0$. Let us choose

$$w^2(y_{ij}) = \sigma_a^{-1} \phi((y_{ij} - \theta_{\text{old}}(x_{ij}))/\sigma_a), \quad (20)$$

where $\phi(t) = \exp(-t^2/2)/\sqrt{2\pi}$, $\sigma_a > 0$ reflects the level of confidence that $\theta^*(\cdot)$ does not differ too much from $\theta_{\text{old}}(\cdot)$. Then

$$\theta_{ij}^+ = \lambda \theta^*(x_{ij}) + (1 - \lambda) \theta_{\text{old}}(x_{ij}), \quad (21)$$

where $\lambda \stackrel{\text{def}}{=} \sigma_a^2 / (\sigma_\varepsilon^2 + \sigma_a^2)$. If $\sigma_a \rightarrow \infty$, i.e., we have not enough confidence in small differences between subsequent images, then $\theta_{ij}^+ \rightarrow \theta^*(x_{ij})$, otherwise, if $\sigma_a \rightarrow 0$, then $\theta_{ij}^+ \rightarrow \theta_{\text{old}}(x_{ij})$.

PATTERN MATCHING AND THE SEGMENTATION OF IMAGES. Vertical weighting can be used for verifying whether θ^* is close to a given pattern image, denoted further by θ^0 . If the answer is positive, then the knowledge of θ^0 can be used to improve a filtering algorithm.

Denote by $U(t)$ the kernel uniform in $[-1, 1]$, i.e., $U(t) = 1$ in this interval and zero otherwise. Let $U_H(t) = U(t/H)/H$. Select $w^2(y_{ij}) = U_H(y_{ij} - \theta^0(x_{ij}))$, where $H > 0$ is a parameter which reflects the level of tolerance for differences between the observed image and the pattern θ^0 . If $\mathbf{E} [U_H(y_{ij} - \theta^0(x_{ij}))] > 0$, then

$$\theta_{ij}^+ = \frac{\mathbf{E} [y_{ij} \cdot U_H(y_{ij} - \theta^0(x_{ij}))]}{\mathbf{E} [U_H(y_{ij} - \theta^0(x_{ij}))]} \quad (22)$$

If y_{ij} is, in the mean, too far from $\theta^0(x_{ij})$, which yields

$$\mathbf{E} [U_H(y_{ij} - \theta^0(x_{ij}))] = 0,$$

then we set $\theta_{ij}^+ = 0$.

Setting $\theta^0(x_{ij})$ to be a constant, say $0 < c < 1$, we can select objects or parts of an image having (approximately) the same gray level. In such a case, (22) selects all the parts of an image which have a gray level contained in $[c - H, c + H]$.

2.3.2. L_1 loss. In this section, in (11) we choose $\Psi(t) = |t|$. This defines a vertically weighted counterpart of the classical L_1 criterion. The latter yields a well studied class of median and rank filters (K.E. Barner and G.R. Arce). Hence we have the following criterion:

$$Q_1(\theta) = \mathbf{E} [w(y_{ij}) \cdot |y_{ij} - \theta|] \quad (23)$$

Let $f_{ij}(y)$ be the probability density function of y_{ij} . Define

$$f_{ij}^{(\text{mod})}(y) = \frac{w(y)f_{ij}(y)}{\int w(y)f_{ij}(y) dy} \quad (24)$$

assuming the convergence of the integral in the denominator. Note that (23) can be equivalently rewritten as

$$Q_1(\theta) = \mathbf{E} [|Z_{ij} - \theta|], \quad (25)$$

where the expectation is calculated w.r.t. Z_{ij} , which is defined as a random variable having the p.d.f. $f_{ij}^{(\text{mod})}$. This form of $Q_1(\theta)$ immediately leads to the conclusion that its minimizer, denoted further by $\theta_{ij}^\#$, has the form

$$\theta_{ij}^\# = \text{Med} [Z_{ij}], \quad (26)$$

where $\text{Med}[\cdot]$ denotes the theoretical median of a random variable in the brackets. The analysis of (24) and (26) leads to the following simple conclusions:

- If $w \equiv 1$, then $\theta_{ij}^\#$ reduces to the usual median, i.e., $\theta_{ij}^\# = \text{Med}[y_{ij}]$.

- For

$$w(y) = \begin{cases} 1 & \text{if } y \in [a, b], \\ 0 & \text{if } y \notin [a, b], \end{cases} \quad (27)$$

where $0 \leq a < b \leq 1$, Z_{ij} is the version of y_{ij} that is truncated to $[a, b]$, i.e., $Z_{ij} = y_{ij}$, if $a \leq y_{ij} \leq b$ and zero otherwise. The truncated version of y_{ij} is further denoted as $y_{ij}|_a^b$. According to (24), the p.d.f. of $y_{ij}|_a^b$ is proportional to $f_{ij}(y)|_a^b$ and suitably normalized.

Thus, for the weight in (27) we have

$$\theta_{ij}^\# = \text{Med} [y_{ij}|_a^b] \quad (28)$$

Remark 1. Selecting $w(y) = y^p$ and setting large $p > 0$ magnifies the largest element in the support of $f_{ij}(y)$, i.e., $\text{supp}\{y : f_{ij}(y) > 0\}$. Simultaneously, the normalization in (24) leads to reducing $w(y)f_{ij}(y)$ in other areas. Thus, for $p \rightarrow \infty$, $\theta_{ij}^\# = \text{Med} [Z_{ij}]$ acts as the Max operator on the support of $f_{ij}(y)$. The analogous reasoning for $w(y) = y^p$ and $p \rightarrow -\infty$ provides the Min operator.

2.4. Image filtering from VWR. The classical equation which is commonly used for constructing filters for θ^* follows directly from (10) and has the familiar form

$$\mathbf{E} [y_{ij}] = \theta^*(x_{ij}). \quad (29)$$

The main advantage of this equation lies in its simplicity and the linearity of the expectation operator, which provides “automatic” smoothing. At the same time, the expectation yields unwanted smoothing of edges, corner points, and other image details.

Our aim is to derive nonlinear equations for θ^* , which further provides alternative ways of filtering, which are able to preserve sharp changes in the filtered image.

2.4.1. L_2 loss. As we have already noted, the minimizer θ_{ij}^+ of (13) is not equal to $\theta^*(x_{ij})$. Nevertheless, we can still use (14) to characterize the true image. This leads to some nonlinear equation for θ^* .

Hence, let us specify the following weight function:

$$w^2(y_{ij}) = V((y_{ij} - \theta^*(x_{ij}))/H) \quad (30)$$

where $V(\cdot)$ is a given kernel function, and $H > 0$ is a parameter which controls the width of V . The kernel $V(\cdot)$ is assumed to be chosen in such a way that

$$V(t) \geq 0, \quad \int_{-\infty}^{\infty} V(t) dt = 1, \\ V(t) = V(-t), \quad t \in \mathbb{R}. \quad (31)$$

Assume that f_ϵ , being the p.d.f. of ϵ_{ij} s, is also symmetric $f_\epsilon(-t) = f_\epsilon(t)$. Substituting (30) into (14), using (31) and changing the variables, we obtain $\theta_{ij}^+ = \theta^*(x_{ij})$ and

$$\theta^*(x_{ij}) = \frac{\mathbf{E} [y_{ij} \cdot V_H(y_{ij} - \theta^*(x_{ij}))]}{\mathbf{E} [V_H(y_{ij} - \theta^*(x_{ij}))]}, \quad (32)$$

where $V_H(t) \stackrel{\text{def}}{=} V(t/H)$. This equation is fundamental for further studies. If V is a uniform kernel, then (32) simplifies to

$$\theta^*(x_{ij}) = p_H^{-1} \cdot \mathbf{E} [y_{ij} \cdot U_H(y_{ij} - \theta^*(x_{ij}))], \quad (33)$$

where $p_H = \int_{-H}^H f_\epsilon(t) dt$ is the probability of the event $\epsilon_{ij} \in [-H, H]$.

2.4.2. L_1 loss. Let the same assumptions as in the previous subsection hold for f_ϵ and V . Additionally, assume the continuity of V with possible exception of the end points of the interval $\{t : V(t) > 0\}$ if this interval is finite. Choosing $w(y) = V_H(y - \theta^*(x_{ij}))$ in (23), we obtain

$$Q_1(\theta) = \mathbf{E} [V_H(y_{ij} - \theta^*(x_{ij})) \cdot |y_{ij} - \theta|]. \quad (34)$$

Changing the variables $\eta = y_{ij} - \theta^*(x_{ij})$ we have

$$Q_1(\theta) = \int V_H(\eta) |\theta^*(x_{ij}) - \theta + \eta| f_\epsilon(\eta) d\eta. \quad (35)$$

From (35) it follows that $Q_1(\theta)$ is minimized by $\theta_{ij}^\# = \theta^*(x_{ij})$ and this minimizer is unique, yielding $Q_1(\theta^\#) = \int V_H(\eta) |\eta| f_\epsilon(\eta) d\eta$. Comparing this equality with (26), we conclude that

$$\theta^*(x_{ij}) = \text{Med} [Z_{ij}], \quad (36)$$

where Z_{ij} is defined as in (26) and its p.d.f. has the form

$$f_{ij}^{(\text{mod})}(y) = \frac{V_H(y - \theta^*(x_{ij})) \cdot f_\epsilon(y - \theta^*(x_{ij}))}{\int V_H(\eta) \cdot f_\epsilon(\eta) d\eta}. \quad (37)$$

We can be more explicit if we select $V_H(t) = U(t/H)$. In this case we have

$$\theta^*(x_{ij}) = \text{Med} \left[y_{ij} \Big|_{\theta^*(x_{ij})-H}^{\theta^*(x_{ij})+H} \right]. \quad (38)$$

From the above it follows that, under the symmetry of f_ϵ and the kernel V , one can use both (38) and (32) for constructing filters.

Remark 2. We mainly confine our attention to local operations on images. Note, however, that the proposed approach is sufficiently flexible to perform also some global operations. For example, selecting $w(y)$ in (24) as either $w(y) = \sin(\pi y/2)$ or $w(y) = \cos(\pi y/2)$ for $y \in [0, 1]$, we can either decrease or increase the content of black or gray-black parts of an image, respectively.

3. Empirical VWR filters

Our aim in this section is to propose a practical class of filters stemming from (14). This will be obtained by estimating the optimal solution obtained in the previous section based on the observed record of noisy image data. As a result, we derive a rich family of nonlinear filters which includes some known solution.

3.1. Image filtering methods. Let us again recall (14), i.e., we have

$$\theta_{ij}^+ = \mathbf{E} [y_{ij} \cdot w^2(y_{ij})] / \mathbf{E} [w^2(y_{ij})]. \quad (39)$$

Replacing the expectations w.r.t. y_{ij} by empirical averages from neighborhood pixels in a rectangular window $(2M+1) \times (2M+1)$, $M \geq 1$, we obtain the estimator $\hat{\theta}_{ij}^+$ of θ_{ij}^+ , given by

$$\hat{\theta}_{ij}^+ = \frac{\sum_{k,l=-M}^M y_{i+k,j+l} \cdot w^2(y_{i+k,j+l}) \cdot \kappa_{(i,j,k,l,h)}}{\sum_{k,l=-M}^M w^2(y_{i+k,j+l}) \cdot \kappa_{(i,j,k,l,h)}}, \quad (40)$$

where $\sum_{k,l=-M}^M$ is a shorthand notation for the double sum $\sum_{k=-M}^M \sum_{l=-M}^M$. In (40),

$$\kappa_{(i,j,k,l,h)} \stackrel{\text{def}}{=} K_h(x_{i+k,j+l} - x_{ij}). \quad (41)$$

For $t \stackrel{\text{def}}{=} [t^{(1)}, t^{(2)}]$, the kernel $K_h : \mathbb{R}^2 \rightarrow \mathbb{R}$ is defined as follows:

$$K_h(t) = \frac{1}{h^{(1)} h^{(2)}} K^{(1)} \left(\frac{t^{(1)}}{h^{(1)}} \right) K^{(2)} \left(\frac{t^{(2)}}{h^{(2)}} \right), \quad (42)$$

where $h = [h^{(1)}, h^{(2)}]$, $h^{(1)} > 0$, $h^{(2)} > 0$ control the width and height of a rectangular neighborhood of pixel x_{ij} , respectively. This rectangular neighborhood is placed horizontally in the image plane. The kernels $K^{(u)}(t) \geq 0$, $u = 1, 2$ are selected according to the rules which are commonly used in the area of nonparametric regression estimation, i.e., they are required to fulfill the following conditions:

$$\int K^{(u)}(t) dt = 1, \quad \int t K^{(u)}(t) dt = 0, \quad \int t^2 K^{(u)}(t) dt < \infty, \quad (43)$$

where $u = 1, 2$ and the integrals are calculated either from -1 to $+1$ (bounded support kernels) or from $-\infty$ to ∞ (unbounded support case). In the latter case, unbounded support is, treated formally in the sense that in fact the support does not extend beyond the image range. It is, however, convenient to permit, e.g., the Gaussian kernel $K^{(u)}(t) = \frac{1}{\sqrt{2\pi}} \exp(-t^2/2)$. Examples of bounded support kernels, for which the conditions (31) hold, include

the uniform kernel $K^{(u)}(t) = U(t)/2$ and the Epanechnikov kernel $K^{(u)}(t) = (3/4)(1 - t^2)$ for $|t| < 1$ and zero otherwise.

Remark 3. In (40) we have intentionally omitted the well-known details concerning the necessary corrections when pixel x_{ij} is near the boundaries of the image. For simplicity of formulas, this convention is also used later on.

Let us select both the kernels as rectangular, i.e., $K^{(u)}(t) = U(t)/2$, $u = 1, 2$. Let $h^{(1)} = h^{(2)}$ be chosen in such a way that $K_h([t^{(1)}, t^{(2)}])$ is nonzero only in the rectangle $[-M, M] \times [-M, M]$, where $M > 0$ is the number of pixels. Under these assumptions, (40) can be expressed as

$$\hat{\theta}_{ij}^+ = \frac{\sum_{k,l=-M}^M y_{i+k,j+l} \cdot w^2(y_{i+k,j+l})}{\sum_{k,l=-M}^M w^2(y_{i+k,j+l})}. \quad (44)$$

The problem of the convergence of $\hat{\theta}_{ij}^+$ to θ_{ij}^+ can be tackled in a rather standard form but this issue is outside the scope of this paper.

3.2. Case studies. In parallel to the discussion in Sections 2.3.1 and 2.3.2, we briefly summarize empirical counterparts of the formulas presented there. For simplicity, we temporarily set K_h to be the uniform weight.

3.2.1. L₂ born filters. Special cases of (44) provide the following filters:

EMPIRICAL MEAN. Setting $w(\cdot) \equiv 1$ in (44) we arrive at the well-known moving average filter

$$\hat{\theta}_{ij}^+ = \frac{1}{(2M+1)^2} \sum_{k,l=-M}^M y_{i+k,j+l}. \quad (45)$$

EMPIRICAL HARMONIC MEAN. Selecting in (44) $w(y) = 1/\sqrt{y}$ we obtain a filter which is not in common use:

$$\hat{\theta}_{ij}^+ = (2M+1)^2 / \sum_{k,l=-M}^M y_{i+k,j+l}^{-1}. \quad (46)$$

Trying to apply this filter we have to correct the formula (46) if some of $y_{i+k,j+l}$ s are zero.

EMPIRICAL MINIMIZER OF THE RELATIVE ERROR. The empirical counterpart of (16) has the form

$$\hat{\theta}_{ij}^+ = \sum_{k,l=-M}^M y_{i+k,j+l}^{-1} / \sum_{k,l=-M}^M y_{i+k,j+l}^{-2}. \quad (47)$$

EMPIRICAL MAX AND MIN OPERATIONS. The empirical versions for (19) and (18) have the form

$$\hat{\theta}_{ij}^+ = \sum_{k,l=-M}^M y_{i+k,j+l}^{p+1} / \sum_{k,l=-M}^M y_{i+k,j+l}^p, \quad (48)$$

where, for $p \rightarrow \infty$,

$$\hat{\theta}_{ij}^+ \rightarrow \max_{k,l} \{y_{i+k,j+l} : |k| \leq M, |l| \leq M\}, \quad (49)$$

and, for $p \rightarrow -\infty$,

$$\hat{\theta}_{ij}^+ \rightarrow \min_{k,l} \{y_{i+k,j+l} : |k| \leq M, |l| \leq M\}. \quad (50)$$

To justify (49), recall that for $a_i > 0$ we have

$$\left(\sum_{i=1}^n a_i^p \right)^{1/p} \rightarrow \max_i \{a_i : i = 1, 2, \dots, n\}. \quad (51)$$

Dividing the nominator and denominator of (49) by $(2M+1)^2$ and letting p to be sufficiently large, we can approximate the nominator by $(\max_{k,l} \{y_{i+k,j+l}\})^{p+1}$, where the maximum is taken over the same set as in (49). Analogously, the denominator is close to $(\max_{k,l} \{y_{i+k,j+l}\})^p$. The justification for (50) is quite similar.

EMPIRICAL SEGMENTATION BY THRESHOLDING. The empirical counterpart of (22) has the form

$$\hat{\theta}_{ij}^+ = \frac{\sum_{k,l=-M}^M y_{i+k,j+l} \cdot U_H(y_{i+k,j+l} - \theta^0(x_{ij}))}{\sum_{k,l=-M}^M U_H(y_{i+k,j+l} - \theta^0(x_{ij}))}, \quad (52)$$

if

$$\sum_{k,l=-M}^M U_H(y_{i+k,j+l} - \theta^0(x_{ij})) > 0, \quad (53)$$

i.e., if the set of all those observation $y_{i+k,j+l}$ s, $k, l = -M, \dots, 0, 1, \dots, M$, for which $|y_{i+k,j+l} - \theta^0(x_{ij})| < H$ is nonempty. Otherwise, we set $\hat{\theta}_{ij}^+ = \text{white}$.

Remark 4. In (52) the empirical mean is, in fact, calculated from those $y_{i+k,j+l}$ s, which are close to the pattern $\theta^0(x_{ij})$ both horizontally (k, l) (ranging from $-M$ to M) and vertically (with the accuracy to H). Simultaneously, the actual number of these neighbors is calculated in the denominator, since U is 1 only for vertically close observations.

3.2.2. L_1 born filters. Empirical versions of (26) and (24) are based on the empirical median of observations, which is denoted by $\widehat{Med}\{a_i, i = 1, 2, \dots, k\}$, for the observations as indicated in the curly brackets. We shall denote by $\hat{\theta}_{ij}^\#$ the estimators of $\theta_{ij}^\#$.

- If $w(y) \equiv 1$, then as the estimator of $\theta_{ij}^\# = Med[y_{ij}]$ it is natural to take

$$\hat{\theta}_{ij}^\# = \widehat{Med}\{y_{i+k,j+l}, k, l = -M, \dots, 0, 1, \dots, M\}, \quad (54)$$

i.e., the well-known empirical median from neighborhood pixels.

- The estimator for (28) is given by

$$\hat{\theta}_{ij}^\# = \widehat{Med}\{[y_{i+k,j+l}, k, l = -M, \dots, M]_a^b\}. \quad (55)$$

In (55) and further, the notation $[a_i, i = 1, 2, \dots, n]_a^b$ is interpreted as follows: after applying the operation $|\cdot|_a^b$ to a sequence $a_i, i = 1, 2, \dots, n$ we obtain a subsequence which contains only those a_i s for which the condition $a \leq a_i \leq b$ holds. Then, in (55), the empirical median is calculated if the set $\{[y_{i+k,j+l}, k, l = -M, \dots, 0, 1, \dots, M]_a^b\}$ is nonempty. Otherwise, we set $\hat{\theta}_{ij}^\# = white$.

Remark 5. Note that, in general, (55) differs from (54), since $[a, b]$ must not be centered near the empirical median taken from all the pixels in the current window. Selecting a and b appropriately, we can use (55) in a way similar to (52) for the segmentation of images, but with the mean replaced by the median. We shall use this fact also later.

It is also possible to give an empirical version of the median for weights $w(\cdot)$ more general than uniform ones, but its applicability in image processing seems to be limited.

4. Jump-preserving filters based on pilot estimators

In this section we return to the general class of horizontal weights $K_h(\cdot)$. Our aim is to discuss a new family of nonlinear filters, which includes the well known sigma-filter as a special case.

Recall that under the assumed symmetry of the kernel V and f_ϵ we have obtained (see (32)):

$$\theta^*(x_{ij}) = \frac{\mathbf{E}[y_{ij} \cdot V_H(y_{ij} - \theta^*(x_{ij}))]}{\mathbf{E}[V_H(y_{ij} - \theta^*(x_{ij}))]}. \quad (56)$$

One can interpret this expression as a nonlinear equation for unknown $\theta^*(x_{ij})$. It cannot be solved, since we are unable to evaluate the expectations. However, we can replace the expectations by the corresponding empirical means hoping that a solution of the empirical counterpart of (56)

is close to $\theta^*(x_{ij})$. The above-sketched approach is outside the scope of this paper. We shall take a more direct approach, namely, $\theta^*(x_{ij})$ on the r.h.s. of (56) is replaced by a pilot estimator, then the expectations are replaced by the empirical means, and finally the resulting expression is taken as the estimator of $\theta^*(x_{ij})$ on the l.h.s. of this equation. The idea of using pilot estimators in nonparametric setting is a well-established concept. In the context of the vertical weighting, the pilot estimators, which are based on the median and the mean, were used in (Krzyżak et al., 2001; Steland, 2003).

Recalling that $\varepsilon_{ij} = y_{ij} - \theta^*(x_{ij})$, one can rewrite (56) also as follows:

$$\theta^*(x_{ij}) = \mathbf{E}[y_{ij} \cdot V_H(\varepsilon_{ij})] / \mathbf{E}[V_H(\varepsilon_{ij})]. \quad (57)$$

According to this equation, $\theta^*(x_{ij})$ is expressed in terms of the expectations with respect to y_{ij} and ε_{ij} . The approach to estimating $\theta^*(x_{ij})$, which is based on the estimation of errors ε_{ij} , is discussed below.

Let $\check{\theta}(x_{ij})$ be a pilot estimator of $\theta^*(x_{ij})$, which is a measurable function of (not necessarily all) gray level y_{ij} of the pixels $x_{(i+k,j+l)}$, $|k| \leq M$, $|l| \leq M$. We shall use $\check{\theta}(x_{ij})$ for estimating errors in a vicinity of ε_{ij} . Then the expectation with respect to ε_{ij} can be estimated by empirical averaging estimates of $\varepsilon_{i+k,j+l}$ from a neighborhood of pixel (i, j) . These estimates have the form

$$\check{\varepsilon}_{i+k,j+l} = y_{i+k,j+l} - \check{\theta}(x_{ij}). \quad (58)$$

Replacing the expectations in (57) with their empirical counterparts, we get the following general class of nonlinear filters:

$$\hat{\theta}_{ij}^* = \frac{\sum_{k,l=-M}^M y_{i+k,j+l} V_H(y_{i+k,j+l} - \check{\theta}(x_{ij})) \cdot \kappa_{(i,j,k,l,h)}}{\sum_{k,l=-M}^M V_H(y_{i+k,j+l} - \check{\theta}(x_{ij})) \cdot \kappa_{(i,j,k,l,h)}}, \quad (59)$$

where $\hat{\theta}_{ij}^*$ is the filter output, which estimates $\theta^*(x_{ij})$, i.e., the image gray level at pixel (i, j) .

Selecting various pilot estimators $\check{\theta}(x_{ij})$, we can generate a large class of filters, some of which are listed in the next two subsections. We present formulas for the uniform kernels K_h and V_H , but each filter has its more general counterpart for the general kernels K_h and V_H .

4.1. Mean-based filters. MEAN-RAW FILTER (σ -FILTER). Selecting in (59) $\check{\theta}(x_{ij}) = y_{ij}$, we obtain the so called σ -filter (see (Lee, 1983)). If we additionally se-

lect V and K to be the uniform kernels, then

$$\hat{\theta}_{ij}^* = \frac{\sum_{k,l=-M}^M y_{i+k,j+l} U_H(y_{i+k,j+l} - y_{ij})}{\sum_{k,l=-M}^M U_H(y_{i+k,j+l} - y_{ij})}, \quad (60)$$

which is the σ -filter originally proposed in (Lee, 1983). Note the similarity of (60) and (52), but this time the condition (53) always holds.

In (60), filtering is carried out by calculating the empirical mean of those observations which fall into the box

$$[i-M, i+M] \times [j-M, j+M] \times [y_{ij}-H, y_{ij}+H], \quad (61)$$

which is centered in the raw observation y_{ij} . Therefore, one can call this filter the mean-raw filter.

The disadvantage of centering the box in y_{ij} comes from the fact that y_{ij} contains random errors, which can move out the box far from $\theta^*(x_{ij})$.

MEAN-MEAN FILTER. The above disadvantage can be reduced by selecting $\check{\theta}(x_{ij})$ to be the empirical mean from all the pixels contained in $[i-M, i+M] \times [j-M, j+M]$ (see (45)). We do not discuss this filter further, since the empirical mean as the pilot estimator rules out edges from the image and the box (61) slides too smoothly across edges.

MEAN-MEDIAN FILTER. The empirical median is known to be less destructive for edges than the empirical mean. Thus, selecting

$$\check{\theta}(x_{ij}) = \widehat{Med}\{y_{i+k,j+l}, k, l = -M, \dots, M\} \quad (62)$$

and substituting this expression to (59), we obtain a filter which is expected to be less sensitive to gross errors than the σ -filter and more jump preserving than the mean-mean filter. The simplest form of this filter is obtained by substituting (62) into the following expression:

$$\hat{\theta}_{ij}^* = \frac{\sum_{k,l=-M}^M y_{i+k,j+l} U_H(y_{i+k,j+l} - \check{\theta}(x_{ij}))}{\sum_{k,l=-M}^M U_H(y_{i+k,j+l} - \check{\theta}(x_{ij}))}. \quad (63)$$

The only difference between this filter and the σ -filter (60) is replacing y_{ij} by the median.

4.2. Median-based filters. MEDIAN-RAW FILTER. Our starting point is (38), repeated here for convenience:

$$\theta^*(x_{ij}) = Med \left[y_{ij} \Big|_{\theta^*(x_{ij})-H}^{\theta^*(x_{ij})+H} \right]. \quad (64)$$

As the pilot estimate for $\theta^*(x_{ij})$ on the r.h.s. of (64) we take a raw observation y_{ij} with the hope that the possible gross error contained in it is reduced by using the empirical median, which is truncated to the box (61). These deliberations lead to

$$\hat{\theta}_{ij}^* = \widehat{Med} \left\{ \left[y_{i+k,j+l}, |k|, |l| \leq M \right] \Big|_{y_{ij}-H}^{y_{ij}+H} \right\}. \quad (65)$$

Remark 6. At this point we skip the discussion of the median-mean filter for the same reasons as explained for the mean-mean filter.

MEDIAN-MEDIAN FILTER. We start again from (64). Using (55) as a guideline and substituting (62) as the pilot estimate into the r.h.s. of (64), we obtain the following filter:

$$\hat{\theta}_{ij}^* = \widehat{Med} \left\{ \left[y_{i+k,j+l}, |k|, |l| \leq M \right] \Big|_{\check{\theta}^*(x_{ij})-H}^{\check{\theta}^*(x_{ij})+H} \right\}. \quad (66)$$

We have called this filter the median-median filter, since we firstly calculate the empirical median from gray values of all the pixels in the window $[i-M, i+M] \times [j-M, j+M]$, which is treated as a pilot estimate $\check{\theta}^*(x_{ij})$, and then the empirical median is calculated from gray values in the following box:

$$[i-M, i+M] \times [j-M, j+M] \times [\check{\theta}^*(x_{ij})-H, \check{\theta}^*(x_{ij})+H].$$

Remark 7. The filters mean-mean, mean-median and median-median have their counterparts, which are based on calculating the pilot estimate from all the pixels in the horizontal window $[i-M, i+M] \times [j-M, j+M]$ with the exception of the central one. This may reduce the bias of the corresponding estimator.

5. RBF net structure of vertically weighted filters

Our aim in this section is to sketch the structure of a net which corresponds to the family of filters described by (59).

Firstly, we rewrite (59) as follows:

$$\hat{\theta}_{ij}^* = \sum_{k,l=-M}^M y_{i+k,j+l} \beta_{(i,j,k,l,h)} \cdot V_H(y_{i+k,j+l} - \check{\theta}(x_{ij})) \cdot K_h(x_{i+k,j+l} - x_{ij}), \quad (67)$$

where

$$\beta_{(i,j,k,l,h)}^{-1} \stackrel{\text{def}}{=} \sum_{k,l=-M}^M V_H(y_{i+k,j+l} - \check{\theta}(x_{ij})) \cdot K_h(x_{i+k,j+l} - x_{ij}). \quad (68)$$

Formula (67) looks similar to the description of RBF nets. There are, however, also important differences, which are pointed out below:

1. The weights $y_{i+k,j+l} \beta_{(i,j,k,l,h)}$ depend also indirectly on the smoothing kernel K_h acting in the horizontal direction.
2. The weights $\beta_{(i,j,k,l,h)}$ depend also on the pilot estimator $\check{\theta}(x_{ij})$.

These differences can be taken into account in constructing an RBF net for (67) by noticing that also $\beta_{(i,j,k,l,h)}^{-1}$ has an RBF like structure. Furthermore, as the initial estimate in (67) and (68) one can take the classical kernel smoother of the following form:

$$\check{\theta}(x_{ij}) = \sum_{k,l=-M}^M y_{i+k,j+l} \cdot K_h(x_{i+k,j+l} - x_{ij}). \quad (69)$$

In fact, in (69) one can take a different smoothing parameter h than in (67), (68), but it seems reasonable to keep the same value for them.

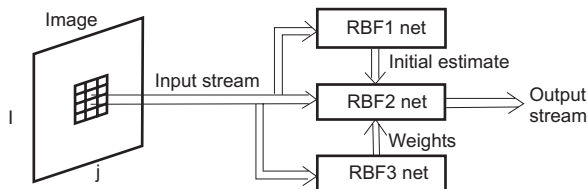


Fig. 1. Structure of three RBF nets which realize vertical filtering.

Summarizing, vertically weighted filters can be realized by the three RBF networks given by (67)–(69). Additionally, the structure of all these nets is the same, which facilitates their implementation in, e. g., FPGA technology. The overall scheme of these networks is shown in Fig. 1, where the RBF 2 net corresponds to (67), the RBF 1 net is described by (69), while the RBF 3 net is defined by (68). For transparency of this figure, the connection between the output of the RBF 1 net and the RBF 3 net is not shown.

Let us note that the above RBF structure has a much smaller number of tunable weights than usual RBF nets. The reason is that we have established values of many weights through the analysis of vertically weighted estimators. The only tunable parameters are $0 < H < 1$ and $h > 0$ (or, equivalently, $M > 0$).

From our simulations it follows that the rules for selecting the horizontal window (half-)width M should be the same as for the classical image processing filters.

We have to stress that the choice of H is crucial for the proper performance of the filters proposed above. At least the following approaches to the choice of H can be considered.

The first one is based on *a-priori* knowledge concerning the heights of edges to be preserved after filtering.

The *a-priori* information can be collected if we have a learning sequence of typical images to be filtered. Such a learning sequence is frequently available when industrial images are processed, since a camera usually provides very similar images of a production process. One can establish H by selecting parts of these images which contain defects and read out heights of typical defects.

The second approach admits H to be tuned locally for each window centered at (i, j) . Thus, we admit $H = H_{ij}$ and permit a data driven choice. A number of different procedures can be used at this stage. We shall describe one of the simplest of them, which is based on estimating the range (or the standard deviation) in a vicinity of (i, j) in order to decide whether the image intensity changes are sufficiently high.

Denote by \hat{r}_{ij} the difference between the maximum and the minimum of the intensities in $[i - M, i + M] \times [j - M, j + M]$. By $\hat{\sigma}_{ij}$ we denote the empirical standard deviation of the intensities in this window. Then the rule for selecting local vertical parameters is as follows:

$$H_{ij} = \begin{cases} R/\hat{r}_{ij} & \text{if } \hat{r}_{ij} > 0, \\ H_{ij} = 1 & \text{otherwise,} \end{cases} \quad (70)$$

where $R > 0$ is a preselected parameter.

Alternatively, one can consider the rule

$$H_{ij} = \begin{cases} c/\hat{\sigma}_{ij} & \text{if } \hat{\sigma}_{ij} > 0, \\ H_{ij} = 1 & \text{otherwise,} \end{cases}$$

where $c > 0$ is a preselected parameter.

In the next section we shall demonstrate how the former rule performs.

6. Examples of filtering industrial images

The aim of the first example is to demonstrate the performance of simple versions of vertically weighted filters applied to the copper slab shown in Fig. 2 (top left panel). Larger and smaller defects are clearly visible, but one can also notice cracks which emerged during the burnishing of the metal surface. Our task is to filter out these cracks, leaving the defects visible.

In the middle panel of the second row the result of applying the classical 11×11 running mean filter is shown, while the bottom right panel contains the result of applying the running median filter with the window size 11×11 . In both the cases the cracks are highly reduced, but small defects either vanished or became much less visible. The boundaries of larger defects are oversmoothed, although the median filter corrupted the boundaries much less than the running mean filter.

The results of applying 11×11 vertically weighted median and mean filters are shown in the middle and rightmost panels in the first row, respectively. In both the

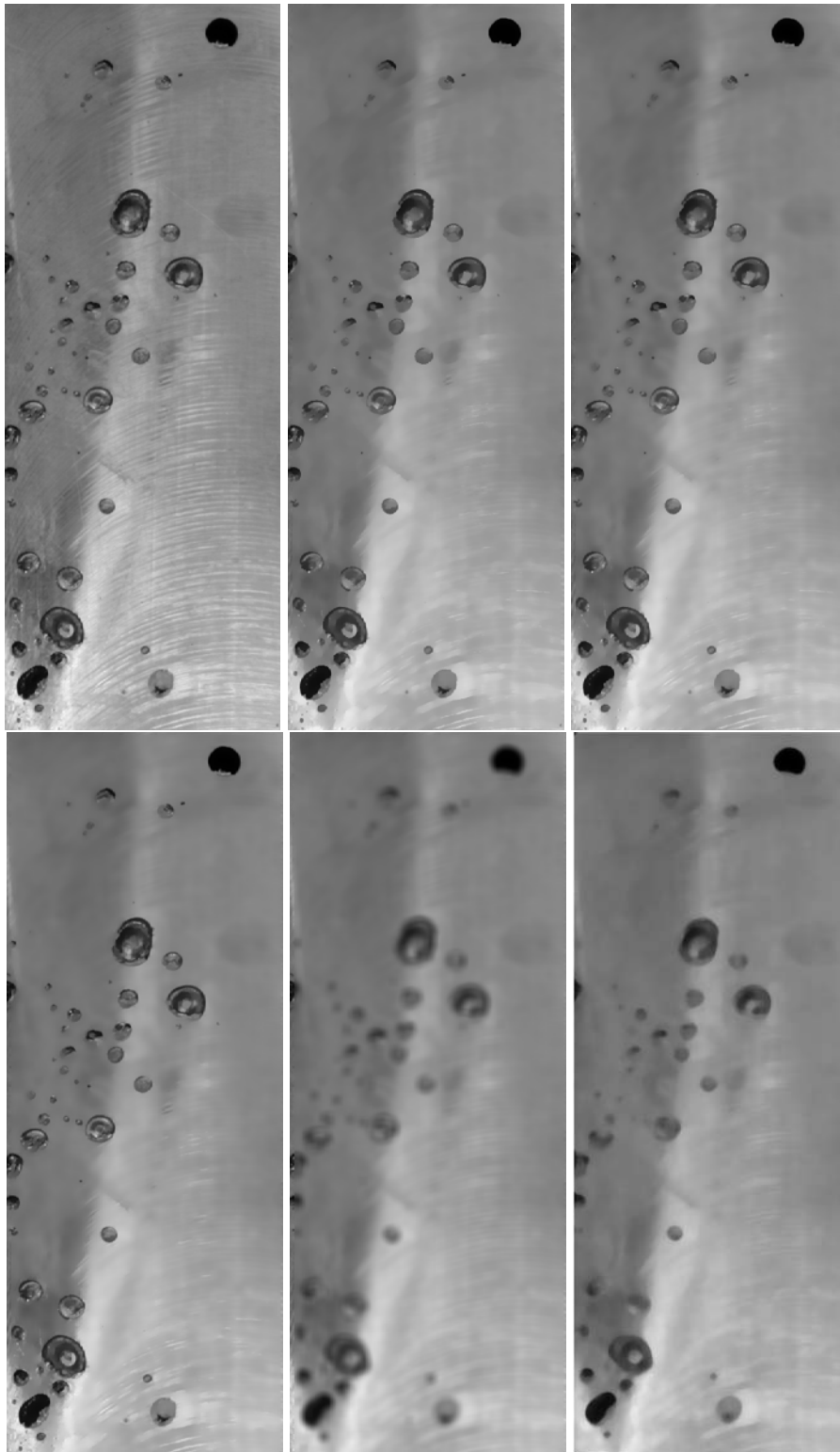


Fig. 2. Copper slab with defects and unwanted cracks to be filtered out (top left panel). The results of filtering by the 11×11 classical running mean (middle panel, bottom row) and median (bottom right panel) corrupt the boundaries of the defects. The vertically weighted median (middle panel, first row) and mean (top right panel) provide better results ($H = 0.1$). The data driven choice of the vertical window height (bottom left panel) provides the best result.

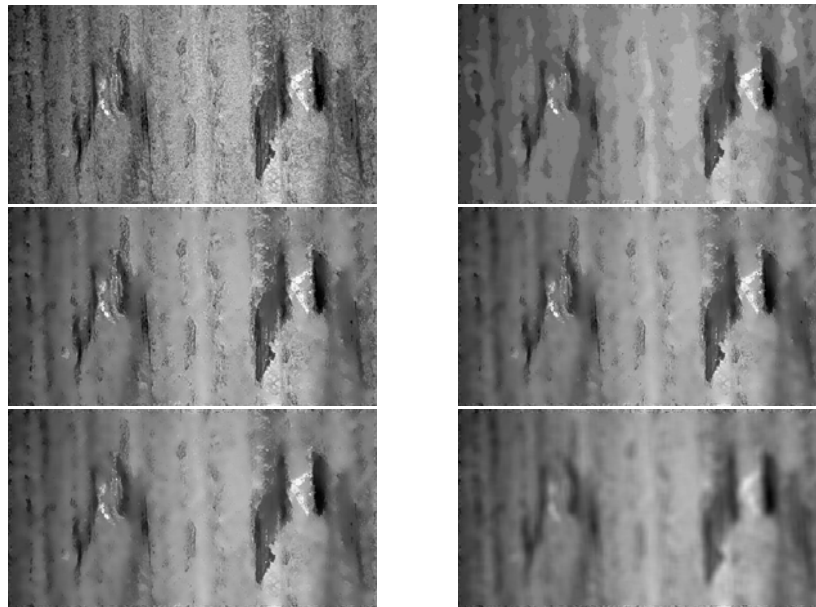


Fig. 3. Piece of a steel wall with rust stains (top left panel). Filtering by the vertically weighted median (top right panel) and mean (bottom left panel), $H = 0.2$. The classical mean filtering in the 11×11 window is reflected in the bottom right panel. The vertically weighted mean with a data driven choice of H_{ij} is shown in the middle row for $R = 0.06$ (left panel) and $R = 0.1$ (right panel).

cases the vertical window height $H = 0.1$ was applied and the results of reproducing the boundaries of the defects are satisfactory. The cracks were filtered, but they are still visible even in the areas which are far from defects.

It seems that the best result was obtained in the bottom left panel. This image was obtained using the vertically weighted mean 11×11 filter with a data driven choice of the vertical window height described by (70). $R = 0.015$ was used in this experiment.

In all of the above-applied vertical filters the raw observations were used as the initial estimates.

Similar conclusions can be drawn from the second series of experiments. In the top left panel of Fig. 3 a piece of steel with rust stains is shown. Two kinds of cracks and patches caused by the rust can be distinguished: small ones, which are not yet dangerous for a steel wall, and larger cracks and patches, which should be kept unchanged after filtering. In the bottom right panel the result of filtering using the classical 11×11 filter is shown for comparisons. The top right and the bottom left panels contain the image filtered by the 11×11 vertically weighted median and mean, respectively. In both the cases $H = 0.2$ was used. In the middle row we collected the results of filtering by the vertically weighted mean with a local, data driven choice of H , according to (70). The image on the r.h.s was obtained using $R = 0.1$, while the one on the

l.h.s. panel with $R = 0.06$. It seems that in the latter case the best result was obtained, since large cracks and patches are clearly visible and smaller stains were filtered out.

7. Concluding remarks

In this paper a thorough analysis of filtering algorithms using the idea of Vertical Weighted Regression (VWR) was conducted. It was shown that nonparametric estimators of VWR provide a framework for designing a large family of nonlinear filters. This gives new filter structures and includes many filters recently considered in the image analysis literature. It is argued that filters based on vertical weighting can be implemented as three interconnected neural networks of the RBF type. A data driven choice of vertical weighting was also proposed (see (70)), and its performance was verified using industrial images with low contrasts. It was demonstrated that a vertically weighted mean filter with a data driven choice of vertical weighting gives precise tuning. This allows us to filter out small defects or inaccuracies, simultaneously keeping the larger ones almost unchanged.

Acknowledgements

The work of E. Rafajłowicz was supported by a grant of the Ministry of Science and Higher Education 2006–2009.

The acquisition of images was supported by a grant from the Foundation for Polish Science.

References

- Barash D. (2002). A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(6): 844–847.
- Barner K., Sarham A. and Hadie R. (1999). Partition-based weighted sum filters for image restoration, *IEEE Transactions on Image Proceedings* **8**(5): 740–745.
- Barner K.E. and Arce G.R. (2004). *Nonlinear Signal and Image Processing: Theory, Methods, and Applications*, CRC Press, Boca Raton, FL.
- Buades A., Coll B. and Morel J. (2005). A review of image denoising algorithms, with a new one, *SIAM Journal on Multiscale Modeling and Simulation* **4**(2): 490–530.
- Chiu C., Glad K., Godtliebsen F. and Marron J. (1998). Edge-preserving smoother for image processing, *Journal of the American Statistical Association* **93**(442): 526–541.
- Efromovich S. (1999). *Nonparametric Curve Estimation: Methods, Theory and Applications*, Springer-Verlag, New York.
- Elad M. (2002): On the origin of the bilateral filter and ways to improve it, *IEEE Transactions on Image Processing* **11**(10): 1141–1150.
- Hall P. and Koch S. (1992). On the feasibility of cross-validation in image analysis, *SIAM Journal on Applied Mathematics* **52**(1): 292–313.
- Jain A. (1989): *Fundamentals of Digital Image Processing*, Prentice Hall, New York.
- Krzyżak A., Rafajłowicz E. and Skubalska-Rafajłowicz E. (2001). Clipped median and space-filling curves in image filtering, *Nonlinear Analysis* **47**(1): 303–314.
- Lee J. (1983). Digital image smoothing and the sigma filter, *Computer Vision, Graphics and Image Processing* **24**(2): 255–269.
- Mitra S. and Sicuranza G. (2001). *Nonlinear Image Processing*, Academic Press, San Diego.
- P. Saint-Marc J.S., and Medioni G. C. (1991). Adaptive smoothing: A general tool for early vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13**(6): 514–529.
- Pawlak M. and Liao S. X. (2002). On the recovery of a function on a circular domain, *IEEE Transactions on Information Theory* **48**(10): 2736–2753.
- Pawlak M. and Rafajłowicz E. (1999). Vertically weighted regression – A tool for nonlinear data analysis and constructing control charts, *Statistical Archives* **84**: 367–388.
- Pawlak M. and Rafajłowicz E. (2001). Jump preserving signal reconstruction using vertical weighting, *Nonlinear Analysis* **47**(1): 327–338.
- Pawlak M., Rafajłowicz E. and Steland A. (2004). On detecting jumps in time series: Nonparametric setting, *Nonparametric Statistics* **16**(3–4): 329–347.
- Polzehl J. and Spokoiny V. (2000). Adaptive weights smoothing with applications to image restoration, *Journal of the Royal Statistical Society B* **62**(2): 335–354.
- Smith S. and Brady J. M. (1997). SUSAN – A new approach to low level image processing, *International Journal of Computer Vision* **23**(1): 45–78.
- Steland A. (2003). Jump-preserving monitoring of dependent processes using pilot estimators, *Statistics and Decision* **21**(4): 343–366.
- Steland A. (2005). On the distribution of the clipping median under a mixture model, *Statistics and Probability Letters* **71**(1): 1–13.
- Takeda H., Farsiu S., and Milanfar P. (2007). Kernel regression for image processing and reconstruction, *IEEE Transactions on Image Processing* **16**(2): 349–366.
- Tomasi C. and Manduchi R. (1998). Bilateral filtering for gray and color images, *IEEE International Conference on Computer Vision*, pp. 839–845.
- van der Vaart A. (1998). *Asymptotic Statistics*, Cambridge University Press, Cambridge, 1998.
- Wasserman L. (2006). *All of Nonparametric Statistics*, Springer-Verlag, New York.
- Yaroslavsky L. (1985). *Digital Picture Processing*, Springer-Verlag, New York.

