ROBUST B-SPLINE IMAGE SMOOTHING

Marta Karczewicz, Moncef Gabbouj and Jaakko Astola

Tampere University of Technology, Signal Processing Laboratory
P.O. Box 553, 33101 Tampere, Finland

ABSTRACT

In this work we present a new approach to two - dimensional robust spline smoothing. The proposed method is based on M-estimator algorithms but unlike in other M-estimator based image processing algorithms it takes into consideration spatial relations between picture elements. The contribution of the sample to the model depends not only on the current residual of that sample, but also on the neighboring residuals. The smoothing parameter (λ) is estimated separately for each processing window and it adapts to the local structure of the image. In order to test the proposed algorithm we apply it to image filtering problem. We show that the filter based on our algorithm has excellent detail preserving properties while suppressing additive Gaussian and impulsive noise very efficiently.

1 INTRODUCTION

A common assumption on the noise corrupting the image has been that the noise distribution is Gaussian. Real sensor data, however, do not behave as nicely as is assumed in classical statistical procedures. The main purpose of this work is to extend the recent results in signal approximation when we are dealing with a mixture of Gaussian noise and impulsive noise. The noise-suppression process can be performed by fitting a surface to the data. Images are modeled as tensor-product bicubic B-splines. Segmented nature of spline functions gives them flexibility and allows them to adjust very effectively to local characteristics of data. Splines have very good Gaussian noise removal properties. There still remains the problem of the methods which allow to robustify the spline approximation. The only attempt that we are aware of in the field of image processing is the two stage algorithm presented by Sinha and Schunk [1]. The first stage, based on robust regression (the least median of squares (LMedS) of errors), removes outliers; whereas, the second stage, based on approximation using weighted bicubic splines, reduces normal noise. Our method is based on the iterative M-estimation algorithm with modified residuals. This idea has been discussed by Lenth [2] and Huber [3] among others. At each stage the contribution of every data sample to the model depends on the residual (difference between the data and the model) of that sample and the influence function. Two well known influence functions are the soft-limiter and redescending function. The currently used M-estimators have severe drawbacks when applied in image restoration. In the case of an estimator based on a soft limiter, even a small fraction of impulsive noise may have a large effect on the estimate (they have a small breakdown point). M-estimators based on the redescending function can reach a much higher breakdown point, especially when the initial fit is very robust in image processing; however, they tend to remove important details, e.g. lines. The proposed algorithm takes into consideration spatial relations between picture elements. In our method the modified residual computed at each iteration can be equal to zero like in algorithms with redescending influence functions, but the decision does not depend only on the distance of the sample from the current model.

2 SMOOTHING SPLINES

Data approximation by spline functions can be obtained by imposing some smoothness constraints on the solution - smoothing splines or by reducing the number of coefficients - least square approximation. The basic setup for smoothing splines in 2-D is as follows [4]. Let us assume that we have given a set of data values $z_{i,j}$ and a set of weights $w_{i,j} > 0$ corresponding to data points (x_i, y_j), $i, j = 1, \ldots, N$. The smoothing spline $\hat{f}_\lambda$ is defined as the minimizer of

$$\sum_{i=1}^{N}\sum_{j=1}^{N} w_{i,j}^2[z_{i,j} - f(x_i, y_j)]^2 + \lambda J_m(f). \qquad (1)$$

The first term is the residual error in the fit of the surface to the data. The second term $J_m(f)$ is the mth-order Laplacian smoothing functional which is a measure of smoothness. For $m = 2$ the case considered

0-8186-6950-0/94 $4.00 © 1994 IEEE

here is

$$J_2(f) = \int\int [f_{xx}^2 + 2f_{xy}^2 + f_{yy}^2]\, dxdy. \quad (2)$$

The parameter λ governs the trade-off between smoothness and goodness-of-fit. When λ is large this places a premium on smoothness and heavily penalizes estimators with large derivatives. Conversely, a small value of λ corresponds to more emphasis on goodness-of-fit.

We assume that the region of interest Ω is square, with $\Omega = [a,b] \times [a,b]$. Let $t_i; i = 1, \ldots, N+k+1$ be the cubic B-spline knots (degree k=3) over the interval $[a,b]$ defined in the x and y directions. A bicubic B-spline has the form

$$h(x,y) = \sum_{i=1}^{N}\sum_{j=1}^{N} \theta_{i,j} B_i(x) B_j(y), \quad (3)$$

where B_i and B_j are B-spline elements defined on the $\mathbf{t}$ knot sequence. To evaluate the B-spline approximation to $\hat{f}_\lambda$, f is replaced with h given by Eq. (3) in the objective function (1), and the resulting quadratic is minimized by taking partial derivatives with respect to θ_{ij} and setting them to zero [1]. The problem then becomes one of determining the coefficients θ_{ij} to the system of equations:

$$[\mathbf{H} + \lambda\mathbf{\Sigma}]\boldsymbol{\theta} = \mathbf{b}, \quad (4)$$

with

$$\mathbf{b} = (\mathbf{WP})^T \mathbf{z}_w,$$

$$\mathbf{H} = (\mathbf{WP})^T(\mathbf{WP}),$$

$$\boldsymbol{\theta} = \mathbf{cs}\left(\begin{bmatrix} \theta_{1,1} & \ldots & \theta_{N,1} \\ \vdots & & \vdots \\ \theta_{1,N} & \ldots & \theta_{N,N} \end{bmatrix}\right),$$

where

$$\mathbf{P} = \mathbf{P_1} \otimes \mathbf{P_2},$$

$$\mathbf{P}_1 = \begin{bmatrix} B_1(x_1) & \ldots & B_N(x_1) \\ \vdots & & \vdots \\ B_1(x_N) & \ldots & B_N(x_N) \end{bmatrix},$$

$$\mathbf{P}_2 = \begin{bmatrix} B_1(y_1) & \ldots & B_N(y_1) \\ \vdots & & \vdots \\ B_1(y_N) & \ldots & B_N(y_N) \end{bmatrix},$$

$$\mathbf{z}_w = \mathbf{Wz},$$

$$\mathbf{z} = \mathbf{cs}\left(\begin{bmatrix} z_{1,1} & \ldots & z_{N,1} \\ \vdots & & \vdots \\ z_{1,N} & \ldots & z_{N,N} \end{bmatrix}\right),$$

and

$$\mathbf{\Sigma} = \mathbf{B}^{(2)} \otimes \mathbf{B}^{(0)} + 2\mathbf{B}^{(1)} \otimes \mathbf{B}^{(1)} + \mathbf{B}^{(0)} \otimes \mathbf{C}^{(2)},$$

where

$$\mathbf{B}_{ij}^{(l)} = \int B_{i-k-1}^{(l)} B_{j-k-1}^{(l)}(x)dx, \quad l = 0,1,2,$$

By $\mathbf{cs}(\mathbf{A})$ we denote the column vector obtained by putting the columns of the matrix $\mathbf{A}$ underneath each other in their natural order and by $\mathbf{A} \otimes \mathbf{B}$ the Kronecker product of two matrices.

The solution vector $\bar{\mathbf{z}}$

$$\bar{\mathbf{z}} = \mathbf{cs}\left(\begin{bmatrix} h(x_1,y_1) & \ldots & h(x_N,y_1) \\ \vdots & & \vdots \\ h(x_1,y_N) & \ldots & h(x_N,y_N) \end{bmatrix}\right),$$

is given by

$$\bar{\mathbf{z}} = \mathbf{P}[\mathbf{H} + \lambda\mathbf{\Sigma}]^{-1}\mathbf{P}^T\mathbf{W}^2\mathbf{z}. \quad (5)$$

3 CHOOSING THE SMOOTHING PARAMETER

We use the method for choosing λ proposed by Hall and Titterington [5], called *equivalent degrees of freedom choice for* λ. This method is based on the estimate of σ^2

$$\hat{\sigma}^2(\lambda) = \frac{RSS(\lambda)}{(M - df(\lambda))}, \quad (6)$$

where $M = N^2$, the residual sum of squares $RSS(\lambda)$ is given by

$$RSS(\lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N}(w_{i,j}h(x_i,y_j) - w_{i,j}z_{i,j})^2 \quad (7)$$

and $df(\lambda)$ denotes degrees of freedom, which are equal to the trace of the hat matrix $\mathbf{A}(\lambda)$,

$$df(\lambda) = \text{trace}\{\mathbf{A}(\lambda)\} \quad (8)$$

that is, the matrix that maps data into predictions

$$\bar{\mathbf{z}} = \mathbf{A}(\lambda)\mathbf{z}. \quad (9)$$

The parameter $\lambda = \lambda_{EDF}$ can be estimated as the solution to

$$F_{EDF}(\lambda) = \frac{RSS(\lambda)}{(M - df(\lambda))} = \sigma^2. \quad (10)$$

Using eigenanalysis of the underlying matrices to describe the residual sum of squares and degrees of freedom Eq. (10) can be rewritten as [5, 6]

$$F_{EDF}(\lambda) = \frac{\sum_{i=1}^{M} \frac{\lambda^2 v_i^2 d_i^2}{(1+\lambda d_i)^2}}{M - \sum_{i=1}^{M} \frac{1}{(1+\lambda d_i)}} = \sigma^2. \quad (11)$$

The values v_i are the elements of the vector $\mathbf{v} = \mathbf{U}\mathbf{z}$, where matrix $\mathbf{U}$ is the result of the Schur decomposition of the matrix $\mathbf{W}^{-1}(\mathbf{P}^T)^{-1}\mathbf{\Sigma}\mathbf{P}^{-1}\mathbf{W}^{-1}$:

$$\mathbf{W}^{-1}(\mathbf{P}^T)^{-1}\mathbf{\Sigma}\mathbf{P}^{-1}\mathbf{W}^{-1} = \mathbf{U}^T\mathbf{D}\mathbf{U}$$

and $d_1, \ldots, d_M$ are the eigenvalues of the same matrix $\mathbf{W}^{-1}(\mathbf{P}^T)^{-1}\mathbf{\Sigma}\mathbf{P}^{-1}\mathbf{W}^{-1}$ (diagonal elements of $\mathbf{D}$).

It can easily be checked that if $F_{EDF}(\lambda = \infty) > \sigma^2$ there always exists (at least) one value of $\lambda > 0$ and a smoothing spline for which Eq. (11) has a solution.

4 ROBUST SPLINE FITTING

There still remains the problem of how to robustify the spline approximation. Our method is based on the iterative M-estimator algorithm with modified residuals. To overcome the drawbacks of M-estimators when applied to image modeling presented in Section 1, we use the spatial relations between pixel elements.

Given a set of data values $z_{i,j}$ corresponding to data points in the local processing window (x_i, y_j), $i, j = 1, \ldots, N$, we want to determine the estimate $\bar{z}_{i,j}$ of the ideal image.

Algorithm 1

1. Consider the mth iteration ($m > 0$). Let $\bar{z}_{i,j}^{(m)}$ be trial values for $\bar{z}_{i,j}$. Denote by $r_{i,j}^{(m)}$ and $\hat{r}_{i,j}^{(m)}$ the residuals and modified residuals, respectively:

$$r_{i,j}^{(m)} = z_{i,j} - \bar{z}_{i,j}^{(m-1)}, \qquad (12)$$

$$\hat{r}_{i,j}^{(m)} = \psi\left(\frac{r_{i,j}^{(m)}}{\sigma}\right)\sigma, \qquad (13)$$

   where the function ψ is the classical minimax function introduced by Huber

$$\psi(x) = \min(c, \max(x, -c)). \qquad (14)$$

2. Set the modified residual $\hat{r}_{i,j}^{(m)}$ to 0 if $r_{i,j}^{(m)} > k_2\sigma$ (or $r_{i,j}^{(m)} < -k_2\sigma$) and $r_{i,j}^{(m)}$ does not create with the neighboring residuals $r_{s,t}^{(m)} > k_1\sigma$ (or $r_{s,t}^{(m)} < -k_1\sigma$) a structure from the given set of detail preserving requirements. In addition, assign a weighting factor $w_{i,j}^{(m)}$ to the observations. If the sample has the modified residual set to zero, the weight is equal to w_{const} ($0 < w_{const} << 1$), and 1, otherwise.

3. The spline is fitted to the auxiliary variables $\hat{z}_{i,j}^{(m)}$ which are created by adding to the model from the previous iteration the modified residuals

$$\hat{z}_{i,j}^{(m)} = \bar{z}_{i,j}^{(m-1)} + \hat{r}_{i,j}^{(m)}. \qquad (15)$$

   First, we find the weighted planar least square solution (corresponding to the smoothing spline for $\lambda \to \infty$). If the value of the criterion function $F_{EDF}(\lambda)$ for the resulting residuals exceeds the true variance of the noise σ^2, we have to compute the value of $\lambda^{(m)}$ for which the value of the criterion function and the variance σ^2 are the same. This is done by finding the root of Eq. (11) with the vector of observations $\mathbf{z}$ replaced by the vector of auxiliary variables $\hat{\mathbf{z}}^{(m)}$.

   The smoothed data is computed using Eq. (5)

$$\bar{\mathbf{z}}^{(m)} = \mathbf{P}[\mathbf{H} + \lambda^{(m)}\mathbf{\Sigma}]^{-1}\mathbf{P}^T\mathbf{W}^2\hat{\mathbf{z}}^{(m)}, \qquad (16)$$

4. Steps 1-3 are repeated for successive values of integer m until the norm $\|\bar{\mathbf{z}}^{(m+1)} - \bar{\mathbf{z}}^{(m)}\|$ becomes smaller than some predefined threshold, or reach the maximum number of iterations is reached.

The starting values $\bar{z}_{i,j}^{(0)}$ should be very robust, thus

$$\bar{z}_{i,j}^{(0)} = \text{median}(z_{1,1}, \ldots, z_{N,N}) \qquad (17)$$

is a convenient choice. The so-called cutoff value c regulates the degree of robustness; good choices are between 1 and 2 (Huber [3]) for $m > 1$ and between 3.5 and 4, for $m = 1$. At the first iteration $m = 1$ we have to choose the larger c to avoid the planar fit if there is a certain detail in the processing window (e.g. line) that we want to preserve. If the weighted planar fit is chosen, the approximating spline will not start changing its shape into the direction of this feature and the iterations will stop. At the first iteration, the weighted planar surface is fitted to the auxiliary variables which are created by adding to the median of the samples in the processing window the modified residuals $\hat{r}_{i,j}^{(1)}$. The modified residuals $\hat{r}_{i,j}^{(1)}$ corresponding to the samples creating the detail usually have values $\pm c\sigma$. Thus, large enough c can help avoid the choice of the planar surface.

Robust spline fitting is based on the M-estimator algorithm with modified residuals. The modified residual can be equal to zero like in algorithms with redescending ψ-function, but the decision does not depend only on the distance of the sample from the current model. The modified residual is equal to zero if the corresponding sample does not belong to the current model and the probability that this sample belongs to the given set of structures is smaller than some predefined threshold. In our experiments, such structures are clusters of five or more connected pixels or four, creating vertical, horizontal or diagonal lines. This gives a good trade-off between removing clusters of impulses and the minimum size of objects or details that are of interest to the observer. Setting modified residuals to zero is enough to improve detail preservation and to bound the influence of outliers. Outliers, however, will still contribute to the final model. This can cause some artifacts especially near the boundaries of the regions with different grey level values. Such artifacts can be eliminated by introducing additional weights. The sample with the zero modified residual gets weight equal to $0 < w_{const} << 1$; and 1, otherwise. We do not use weights equal to zero to avoid singularities in matrix equation (4). The weighs for the outlying observations are larger than zero, but because they are much smaller than the weights of the remaining observations and the spline is fitted to the auxiliary variables which are already bounded, these observations will be ignored almost completely. The "smoothest" solution at this points will be preferred.

In order to determine λ_{EDF} we have to find the root of Eq. (11). From experiments we found that the solution to this equation can be calculated using the bisection method on the logarithm of λ. There is no need to find λ to any great accuracy and the number of iterations in our implementation of the bisection method was fixed to $n = 10$.

4.1 Choosing the Values of k_1 and k_2.

Let us assume that we have the sample $z_{i,j}$, with the corresponding residual $r_{i,j}^{(m)}$ larger than or equal to $k_2\sigma$, and that this sample belongs to a detail created by $n = 5$ or more pixels with the same gray level value corrupted by Gaussian noise with variance σ^2. In our algorithm to check if the sample is a part of the detail and not a cluster of outliers, we search for at least n connected samples including $z_{i,j}$ with residuals larger than $k_1\sigma$. The probability that we find n such samples is in fact larger than or equal to the probability that the range of n samples (denoted as W) arising from a normal distribution with variance σ^2 is smaller than $w = (k_2-k_1)\sigma$. We have to realize that an estimate corresponding to larger values of $k_2 - k_1$ is more sensitive to outliers. Thus, we choose $k_2 - k_1 = 3.86$ which gives $Prob(W < w)$ equals 0.95. We can not take the value of k_1 too small because then the observations belonging to the current model can have too much influence on the decision that our sample in question really is a part of a certain detail. A reasonable choice is k_1 equals to the cutoff value c which results in values for k_2 between 4.86 and 5.86.

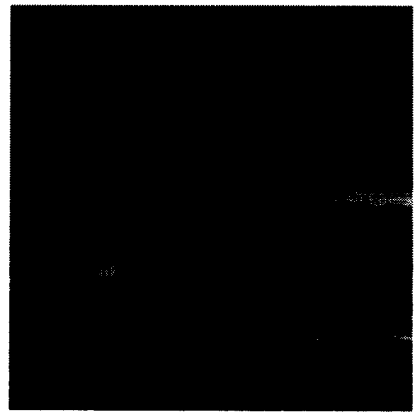

Figure 1: Original image.

5 IMAGE RESTORATION

The filter for image restoration is implemented by sliding a processing window over the input observations. The sample at the center of the window is replaced by its estimate obtained by robust spline fitting. We use the processing window of size 5×5. The knot sequence for tensor-product bicubic B-splines is (-2, -2, -2, -2, 0, 2, 2, 2, 2) in both the x and y directions.

5.1 Simulation results

In the simulations, we use three different images: *Lenna*, *Home* and *Harbor*. As a measure of image quality, we use the peak SNR (PSNR), defined by

$$PSNR = 10 \log \frac{(255)^2}{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}[P(i,j) - Q(i,j)]^2} dB,$$

where $P(i,j)$ and $Q(i,j)$ are, respectively, the $(i,j)th$ pixels in the original and processed images of size $N \times N$. The noisy images are generated by adding zero mean i.i.d. Gaussian noise of variance 64 (PSNR=30.07) and 5% of impulsive noise. We compared our method with the median, the bidirectional multistage median [7] and the center weighted median (CWM) filters. Table 1 summarizes the filtering results. For each filter we compare against, we use the filter parameters which produce the largest PSNR value. Thus, the center weight for the CWM filter is equal to 3 for *Lenna* and *Home* and 5 for *Harbor*. Visual quality is compared using part (350x350) of the image *Harbor*. The original image, the noisy image (Gaussian noise $\sigma^2 = 64$ and 5% of impulsive noise), image filtered with multistage median and our filter are shown respectively in Figs. 1-4.

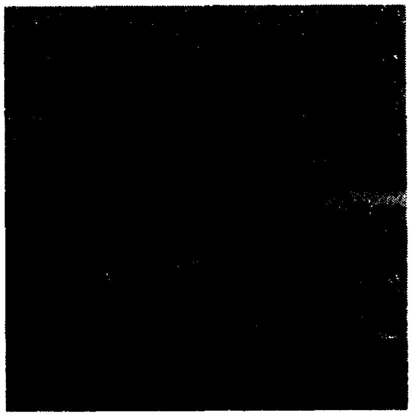

Figure 2: Noisy image (Gaussian noise $\sigma^2 = 64$ and 5% of impulsive noise).

Figure 3: Image filtered with the proposed filter.

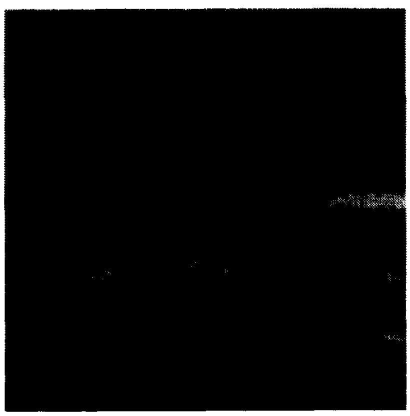

Figure 4: Image filtered with the bidirectional multistage median filter.

Filter Type	Image		
	Lenna	Home	Harbor
Median (3x3)	31.36	26.67	24.65
CWM (3x3)	31.96	28.14	27.63
Bidirectional multistage median (5x5)	31.58	28.73	28.02
Spline filter	32.96	31.89	31.54

Table 1: PSNR [dB] results for Gaussian and impulse noisy images.

6 CONCLUSIONS

A new image approximation scheme is proposed. The structural constraints are incorporated in an iterative M-estimator algorithm. As a result, we get an image modeling method that is not influenced by outliers and reduces Gaussian noise efficiently; while, at the same time important details are retained. As potential application, we presented image restoration. Application of the algorithm in edge detection is presented in [8]. Spline smoothing is closely related to linear filtering which influences the noise attenuation in the detail regions. Thus, directions for further research include the usage of discrete linear splines, which allow discontinuities to be introduced more easily.

7 REFERENCES

[1] S. S. Sinha and B. G. Schunck, "A Two-Stage Algorithm for Discontinuity Preserving Surface Reconstruction," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 14. No. 1, 1992

[2] R. V. Lenth, "Robust splines," *Commun. Statist*, A6, pp. 847-854, 1977.

[3] P.J. Huber, "Robust Smoothing" in *Robustness in Statistics*, New York:Academic Press, 1979.

[4] R. L. Eubank, *Spline Smoothing and Nonparametric Regression*, Dekker: New York, 1988.

[5] A. Thompson, J. Brown and D. Titterington, "A study of methods of choosing the smoothing parameters in image restoration by regularization," *IEEE Trans. Pattern Anal. and Mach. Intell.*, Vol. 13, No. 4, pp. 326-339, 1991

[6] M. Karczewicz, *Robust Multiple-Regression Based Algorithms with Applications in 1-D and 2-D Signal Processing*, M.Sc. Thesis, Tampere University of Technology, 1994

[7] A. Nieminen, P. Heinonen and Y. Neuvo, "A new class of detail-preserving filters for image processing," *IEEE Trans. Pattern Anal. and Mach. Intell.*, Vol. 9, No. 1, pp. 74-90,1987.

[8] M. Karczewicz, M. Gabbouj and I. Defée, "Edge Detection Using Robust B-Spline," *NORSIG-94, Nordic Signal Processing Symposium*, 2-4 June, 1994, Ålesund, Norway