5. Median Filtering: Statistical Properties

B. I. Justusson

With 15 Figures

Median filtering is a nonlinear signal processing technique useful for noise suppression. It was suggested as a tool in time series analysis by *Tukey* [5.1] in 1971 and has later on come into use also in picture processing. Median filtering is performed by letting a window move over the points of a picture (sequence) and replacing the value at the window center with the median of the original values within the window. This yields an output picture (sequence) which usually is smoother than the original one.

The classical smoothing procedure is to use a linear low-pass filter and in many cases this is the most appropriate procedure. However, in certain situations median filtering is better and two of its main advantages are: I) Median filtering preserves sharp edges, whereas linear low-pass filtering blurs such edges. II) Median filters are very efficient for smoothing of spiky noise. We illustrate these properties in Fig. 5.1.

The chief objective of this chapter is to present various theoretical results about median filtering. It is hoped that these results are helpful in judging the practical applicability of median filters.

In Sect. 5.1 basic definitions concerning median filters are given. The ability of median filters to reduce noise is examined in Sect. 5.2, and formulas which yield quantitative information about how much the noise is reduced are presented. White noise, nonwhite noise, impulse noise, and salt-and-pepper noise are considered. In Sect. 5.3 we compare the performance of moving averages and median filters on pictures of the form "edge plus noise". Second-order properties of median filters on random noise are treated in Sect. 5.4. Exact results are given for white-noise input, whereas for nonwhite noise, approximate results are obtained through limit theorems. Frequency response is discussed for simple cosine wave input and also for more general input. In

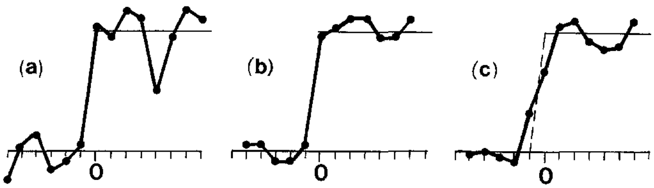


Fig. 5.1. (**a**) Edge plus noise sequence, (**b**) after median filtering, (**c**) after moving average filtering, $n=3$

Sect. 5.5 we present some modifications of median filters which also have the property of preserving edges but differ from simple median filters in other respects. Further uses of medians and other order statistics are discussed in Sect. 5.6.

We conclude this introduction with a short review of earlier work on medians and median filtering.

Medians have long since been used and investigated in statistics as alternatives to sample means in estimation of population means. Most investigations have been concerned with medians and other order statistics of independent random variables; see, e.g., the well-known monographs [5.2, 3]. However, medians of dependent variables have also been treated (see [5.4], where further references are given).

As mentioned above, moving (or running) medians were proposed by *Tukey*, who used them to smooth economic time series. *Tukey* also discussed iterated median filtering and pointed out that median filtering preserves large sudden changes of level (i.e., edges) in time series. *Rabiner* et al. [5.5] and *Jayant* [5.6] used moving medians in speech processing to clean pitches from noise. *Frieden* [5.7] developed a signal processing technique for edge enhancement in which a median filter is used as post-filter to eliminate spurious oscillations. Median filters were later on applied to pictures by several authors. In 1975 *Pratt* examined the effects of median filtering of pictures with normal white noise and with impulse noise. He also investigated the effects of different shapes of the filter windows. His report has been published in [Ref. 5.8, Sect. 12.6]. Median filters were used to correct scanner noise by *Wecksung* and *Campbell* [5.9].

5.1 Definition of Median Filters

5.1.1 One-Dimensional Median Filters

The *median* of n numbers $x_1, \ldots, x_n$ is, for n odd, the middle number in size. For n even we define it as the mean of the two middle numbers. For n even other definitions can be found in the literature but since they differ only slightly and since n will be odd in most of our applications we will not discuss this topic further. We denote the median by

$$\mathrm{Median}(x_1, \ldots, x_n). \tag{5.1}$$

For example: $\mathrm{Median}(0, 3, 4, 0, 7) = 3$.

A *median filter of size* n on a sequence $\{x_i, i \in \mathbb{Z}\}$ is for n odd defined through

$$y_i = \underset{n}{\mathrm{Median}}\, x_i \triangleq \mathrm{Median}(x_{i-v}, \ldots, x_i, \ldots, x_{i+v}), \quad i \in \mathbb{Z}, \tag{5.2}$$

where $v=(n-1)/2$ and $\mathbb{Z}$ denotes the set of all natural numbers. Other terminology in use is *moving medians* and *running medians.*

It is easily seen that this median filter preserves edges, whereas the corresponding moving average filter

$$z_i=(x_{i-v}+\ldots+x_i+\ldots+x_{i+v})/n, \qquad i\in\mathbb{Z}, \tag{5.3}$$

changes an edge into a ramp with width n (see Chap. 6).

5.1.2 Two-Dimensional Median Filters

Digital pictures will be represented by sets of numbers on a square lattice $\{x_{ij}\}$ where (i,j) runs over $\mathbb{Z}^2$ or some subset of $\mathbb{Z}^2$.

A *two-dimensional median filter with filter window A* on a picture $\{x_{ij}, (i,j)\in\mathbb{Z}^2\}$ is defined through

$$y_{ij}=\underset{A}{\mathrm{Median}}\,x_{ij}\triangleq\mathrm{Median}[x_{i+r,j+s};(r,s)\in A], \qquad (i,j)\in\mathbb{Z}^2. \tag{5.4}$$

Various forms of filter windows A can be used, e.g., line segments, squares, circles, crosses, square frames, circle rings. Some examples are shown in Fig. 5.2. The "circle rings" in Fig. 5.2f have been chosen so as to make the number of points in each ring approximately proportional to the area of a corresponding perfect circle ring.

The definitions of median filters given above do not specify how to compute the output close to end points and border points in finite sequences and

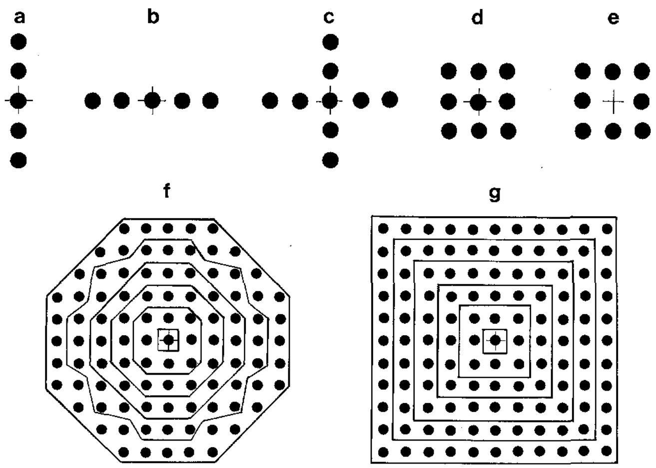


Fig. 5.2a–g. Filter windows. (**a, b**) Line segments, (**c**) a cross, (**d**) a square, (**e**) a square frame, (**f**) circles and circle rings, (**g**) squares and square frames

pictures. One simple convention, which we have used for the generated pictures, is to compute medians of the points lying inside both the picture and the window. Thus for points close to the borders the medians will be computed from fewer than the number of points in A.

5.1.3 Edge Preservation

By an *edge picture* we mean a picture in which all points on one side of a line have a common value a and all points on the other side of the line have a common value b, $b \neq a$. The following result presents a fundamental property of median filters.

If the window set A is symmetric around the origin and includes the origin, i.e., if

$$(r,s) \in A \Rightarrow (-r,-s) \in A\,, \tag{5.5}$$

$$(0,0) \in A\,, \tag{5.6}$$

then the median filter (5.4) preserves any edge picture.

We refer to Chap. 6 by *Tyan* for a thorough discussion of the effects of median filtering of other deterministic signals than just edge pictures. Conditions (5.5, 6) are fulfilled for the windows in Fig. 5.2 except for square frames and circle rings which do not include the origin. However, square frames and circle rings will change edges only slightly. Unless otherwise stated filter windows will be assumed to fulfill conditions (5.5, 6). Note that these conditions imply that the number of points n in A is odd.

5.2 Noise Reduction by Median Filtering

As stated before, median filters can be used for noise suppression. *Pratt* [5.8] discussed in a rather qualitative way their effect on white noise and impulse noise. We shall here present some variance formulas which yield quantitative information about how much the noise will be suppressed.

Median filters are nonlinear and this fact complicates the mathematical analysis of their performance. It is not possible to separate signal effects and noise effects as simply as for linear filters. Throughout this section we confine ourselves to the easiest case with a constant signal.

5.2.1 White Noise

White-noise model The picture values $\{x_{ij}\}$ or the sequence values $\{x_i\}$ are independent identically distributed (i.i.d.) random variables with mean m

$$x = m + z\,, \tag{5.7}$$

where $E(z)=0$ and thus $E(x)=m$. Let $F(x)$ and $f(x)=F'(x)$ denote the distribution and density functions of the x variables.

Next we write down two well-known results from probability theory about medians of i.i.d. random variables (see [Ref. 5.3, Chaps. 2, 9]).

The density of $y=\text{Median}\,(x_1,\ldots,x_n)$ is for n odd given by

$$g(y)=n\binom{n-1}{(n-1)/2}f(y)F(y)^{(n-1)/2}[1-F(y)]^{(n-1)/2}. \tag{5.8}$$

The distribution of $y=\text{Median}\,(x_1,\ldots,x_n)$ is for large n approximately normal $N(\tilde{m},\sigma_n)$ where $\tilde{m}$ is the theoretical median, i.e., is determined by $F(\tilde{m})=0.5$, and

$$\sigma_n^2=\frac{1}{n4f^2(\tilde{m})}\approx\text{Var}\,[\text{Median}\,(x_1,\ldots,x_n)]. \tag{5.9}$$

For small n values one often gets a better approximation of the variance by replacing the factor $1/n$ in (5.9) with $1/(n+b)$, where $b=1/[4f^2(\tilde{m})\sigma_x^2]-1$. This modification is obtained by chosing b so that (5.9) becomes exact for $n=1$.

The above results yield information for one-dimensional as well as two-dimensional filtering by taking n equal to the number of points in the filter window. It may be noted that if $f(x)$ is symmetric around m then (5.8) will also be symmetric around m and thus the following simple formula holds:

$$E[\text{Median}(x_1,\ldots,x_n)]=E(x_i)=m. \tag{5.10}$$

Example 5.1: Uniform Distribution. If the x variables are i.i.d. and uniformly distributed on $[0, 1]$ then one can compute the variance of the median exactly using (5.8).

$$\text{Var}[\text{Median}(x_1,\ldots,x_n)]=\frac{1}{(n+2)\cdot 4}=\frac{\sigma_x^2}{n+2}\cdot 3. \tag{5.11}$$

Formula (5.9) with small sample modification yields the same result.

Example 5.2: Normal Distribution. If the x variables are i.i.d. and $N(m,\sigma)$ then $\tilde{m}=m$ and the variance can only be computed by numerical integration using (5.8). The entries $(n,m,k)=(n,n,n)$ in Table 5.4 give variances for medians of $N(0,1)$ variables. Formula (5.9) together with the modification for small n yields

$$\text{Var}[\text{Median}(x_1,\ldots,x_n)]\approx\frac{\sigma^2}{n+\pi/2-1}\cdot\frac{\pi}{2},\qquad n=1,3,5,\ldots. \tag{5.12}$$

This formula has good accuracy for all odd n.

The average (mean) $\bar{x}$ of n i.i.d. random variables has variance σ^2/n. Equation (5.12) thus yields that for normal white noise the variance of the median is approximately $(\pi/2-1)=57\%$ larger than that of the mean. Hence a moving average suppresses normal white noise somewhat more than a median filter with the same filter window. Otherwise formulated: To get a median filter with the same variance as a given moving average one has to take 57% more points in the filter window.

Figure 5.3 illustrates median filtering and moving average filtering with square 3×3 windows. Each picture has 45×30 points and each point is 1×1 mm. (a_1) is the original test picture. (b_1, c_1, d_1) have been obtained by changing the grey-scale of (a_1) and adding normal white noise with σ-values $h/5$, $h/3$, h where h is the largest edge height. Figure 5.3 will be discussed further in Sect. 5.3.

Example 5.3: Double Exponential Distribution. Let the x variables have a double exponential distribution with mean m and variance σ^2, i.e., have density function

$$f(x)=\frac{\sqrt{2}}{\sigma}\mathrm{e}^{-\sqrt{2}\cdot|x-m|/\sigma}, \qquad x\in R. \tag{5.13}$$

Then by (5.9) the asymptotic variance of Median $(x_1, \ldots, x_n)$ is

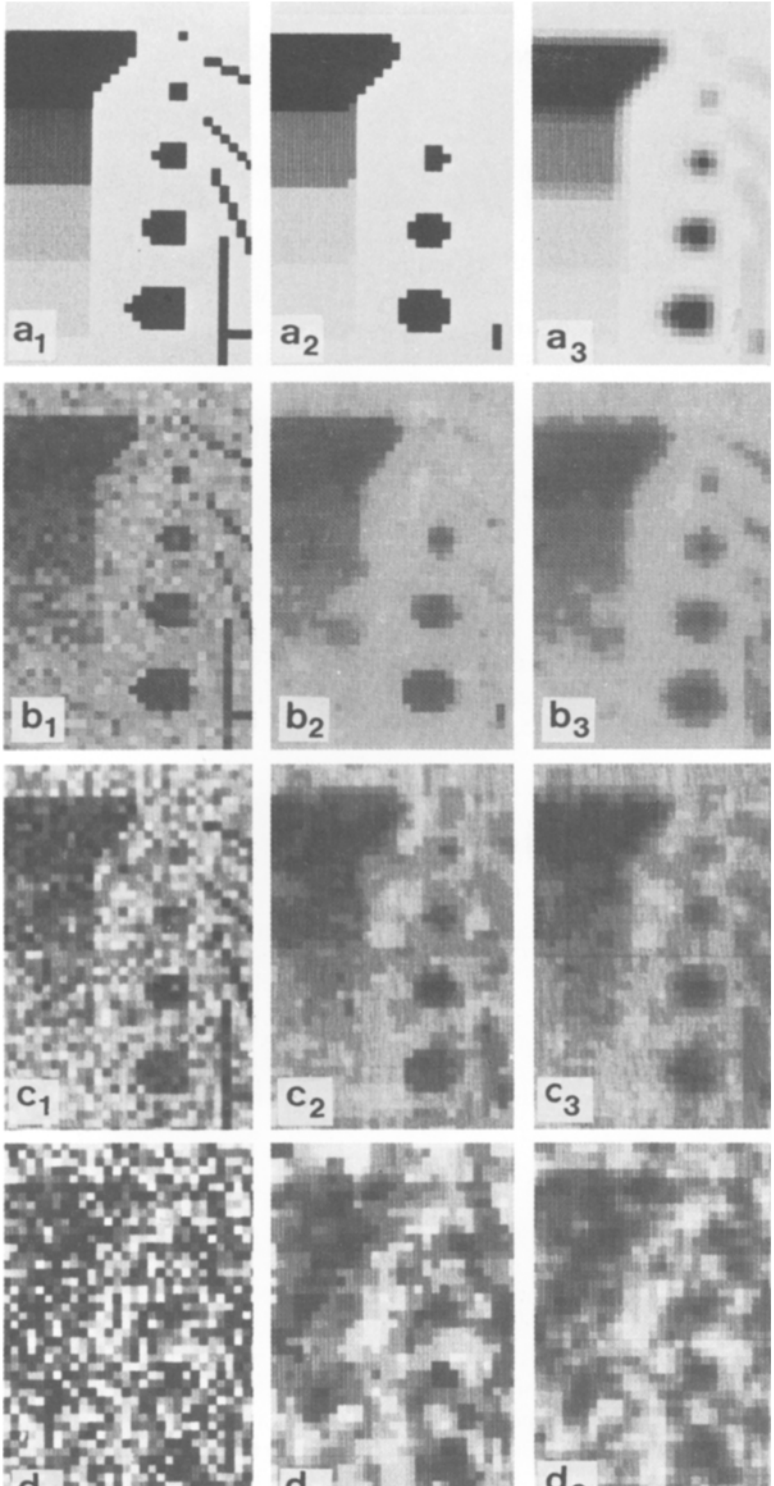
$$\sigma_n^2=\frac{\sigma^2}{(n-1/2)}\cdot\frac{1}{2}\approx \mathrm{Var}\,(\mathrm{Median}) \tag{5.14}$$

which is 50% smaller than the variance σ^2/n of the mean $\bar{x}$. Thus, for this type of noise the median is a better estimator of m than the mean $\bar{x}$. In fact the median is the maximum-likelihood estimator of m and hence the mean-square-error optimal estimator of m (at least asymptotically). The fact that the median here is the maximum-likelihood estimator is an easy consequence of the general result that the median is the least-absolute-deviation estimator of the center of a distribution, i.e.,

$$\min_a \sum_{i=1}^{n} |x_i-a| \tag{5.15}$$

is attained for $a=\mathrm{Median}\,(x_1, \ldots, x_n)$. The above results suggest the more general conclusion that medians are better than means for suppression of noise with heavy-tailed distributions. One extreme is impulse noise which will be discussed in Sect. 5.2.3.

Fig. 5.3a–d. Filtering of pictures with added normal noise $N(0,\sigma)$. (a_1-d_1) Input pictures with $\sigma=0, h/5, h/3, h$; (a_2-d_2) median filtered pictures; (a_3-d_3) moving average filtered pictures



a1
a2
a3
b1
b2
b3
c1
c2
c3
d1
d2
d3

5.2.2 Nonwhite Noise

For input sequences (pictures) that are general random processes (random fields), i.e., with nonindependent variables, one cannot obtain simple exact formulas for the distribution of medians. However, there are limit theorems analogous to (5.9) (see [5.4, 10] where further references can also be found). The conditions needed for the limit theorems are that the processes $\{x_i\}$, $\{x_{ij}\}$ are stationary and mixing. The essence of the mixing condition is that it requires process variables lying far apart to be almost independent (for details see [5.4, 10]). For a stationary mixing normal process with covariance function

$$\mathrm{Cov}(x_i, x_{i+\tau}) = r_x(\tau) = \sigma_x^2 \cdot \varrho_x(\tau), \quad \tau = 0, \pm 1, \ldots \tag{5.16}$$

we have the following approximate expression for the variance of a median

$$\mathrm{Var}[\mathrm{Median}(x_1, \ldots, x_n)] \approx \frac{\sigma_x^2}{n + \pi/2 - 1} \sum_{j=-(n-1)}^{n-1} \left(1 - \frac{|j|}{n}\right) \arcsin[\varrho_x(j)]\,. \tag{5.17}$$

For two-dimensional filtering a similar result holds. In Sect. 5.4 these kinds of approximations and limit theorems will be discussed further.

It is interesting to compare (5.17) with the variance of a mean $\bar{x} = (\sum x_i)/n$ of n variables

$$\mathrm{Var}(\bar{x}) = \frac{\sigma_x^2}{n} \sum_{j=-(n-1)}^{n-1} \left(1 - \frac{|j|}{n}\right) \varrho_x(j)\,. \tag{5.18}$$

The similarity of (5.17) and (5.18) is striking. For normal processes with nonnegative correlations

$$\varrho_x(\tau) \geqq 0, \quad \tau = 0, \pm 1, \ldots \tag{5.19}$$

we obtain, using (5.17, 18) and the fact that $\varrho_x(\tau) \leqq \arcsin \varrho_x(\tau) \leqq \varrho_x(\tau) \cdot \pi/2$, that for large n

$$1 \leqq \frac{\mathrm{Var}(\mathrm{Median})}{\mathrm{Var}(\bar{x})} \leqq \frac{\pi}{2}\,. \tag{5.20}$$

(This result holds for two-dimensional filtering as well.) Thus, for normal processes with nonnegative correlations, the variance of a median is at most 57% larger than the variance of a mean. For processes with both negative and positive correlations the value of the variance quotient in (5.20) can, however, be much larger than $\pi/2$. This point is illustrated in Table 5.1 which presents values of the variance quotient for a normal AR(1) process (first-order

Table 5.1. Variance quotients for normal AR(1) processes

a	0.9	0.5	0.0	-0.5	-0.9
$\frac{\text{Var (Median)}}{\text{Var }(\bar{x})}$	1.10	1.21	1.57	2.59	6.59

autoregressive process) with

$$\varrho_x(\tau)=a^{|\tau|}, \quad \tau=0, \pm 1, \dots . \tag{5.21}$$

Justusson [5.10] reported on findings from computer simulations of normal AR(1) processes which show that also for small n the quotient values in Table 5.1 are valid approximately, except for $a=-0.9$. For $a=-0.9$, $n=9$ a quotient value as large as 14.9 was obtained.

5.2.3 Impulse Noise and Salt-and-Pepper Noise

By *impulse noise* we mean that a signal is disturbed by impulses (spikes), i.e., very large positive or negative values of short duration. Moving medians are well suited to suppress such noise [5.5, 8] provided that the size of the window is chosen to be at least twice the width of the impulses. Then noise impulses which are sufficiently separated will be completely deleted by the median filter. However, impulses lying close to each other may remain.

In picture processing impulse noise stems, for example, from decoding errors, giving rise to black-and-white spots in the picture, and is therefore often called *salt-and-pepper noise*. The errors become especially prominent in very dark or very bright parts of the picture. For such parts of the picture we can derive some simple formulas for the probability of correct reconstruction. We shall consider two models; in the first model all errors get the same value. In the second model the errors get values which are taken at random from the grey scale.

Impulse Noise: Model 1. At each picture point (i,j) an error occurs with probability p independent both of the errors at other picture points and of all the original picture values. An erroneous point gets the (fixed) value d (e.g., the grey scale value for black). Let $\{x_{ij}\}$ be the distorted picture. Then

$$x_{ij}=\begin{cases} d & \text{with probability } p \\ s_{ij} & \text{with probability } 1-p, \end{cases} \tag{5.22}$$

where s_{ij} denotes the values of the original picture.

Assume now that the point (i',j') is located in a part of the picture where the grey values $\{s_{ij}\}$ of the original picture are constant [at least within the window

Table 5.2. Probability of erroneous reconstruction of impulse noise, $1-Q(n, p)$

Error rate p	Window size n				
	3	5	9	25	49
0.01	0.00030	0.0000099	0.0000000	0	0
0.05	0.00725	0.00116	0.000033	0.0000000	0
0.1	0.028	0.0086	0.00089	0.0000002	0
0.15	0.0608	0.0266	0.00563	0.000017	0
0.2	0.104	0.058	0.0196	0.00037	0.000013
0.3	0.216	0.163	0.099	0.017	0.00165
0.4	0.352	0.317	0.267	0.154	0.0776
0.5	0.500	0.500	0.500	0.500	0.500

A centered at (i',j')], i.e.,

$$s_{i'+r,j'+s}=s_{i'j'}=c\neq d, \quad (r,s)\in A\,. \tag{5.23}$$

Apply a median filter with window A to $\{x_{ij}\}$

$$y_{ij}=\operatorname*{Median}_{A}(x_{ij})\,. \tag{5.24}$$

Then the output value $y_{i'j'}$ will be correct, i.e., $y_{i'j'}=s_{i'j'}=c$, if and only if the number of errors within the window A centered at (i',j') is less than half the number of points in A, i.e., less than or equal to $(n-1)/2$, where n is the size of A. The fact that the number of erroneous points in the window has a binomial distribution yields the following result:

$$\begin{aligned} &P\,[\text{correct reconstruction at } (i',j')]=P\,(y_{i'j'}=s_{i'j'}) \\ &\quad=\sum_{k=0}^{(n-1)/2}\binom{n}{k}p^k(1-p)^{n-k}\triangleq Q(n,p)\,. \end{aligned} \tag{5.25}$$

Values of $1-Q(n,p)$ for some different values of n and p are shown in Table 5.2. It is seen that if the error rate p is not too large, say not larger than 0.3, then median filtering with a rather small window will reduce the portion of errors considerably. Larger windows will of course reduce the noise even more but will also give more signal distortion. The result of a performed median filtering is shown in Fig. 5.4a.

Impulse Noise: Model 2. This model differs from Model 1 above only in the respect that the error points get random, instead of fixed, grey values z_{ij}. These are assumed to be independent random variables with uniform distribution on

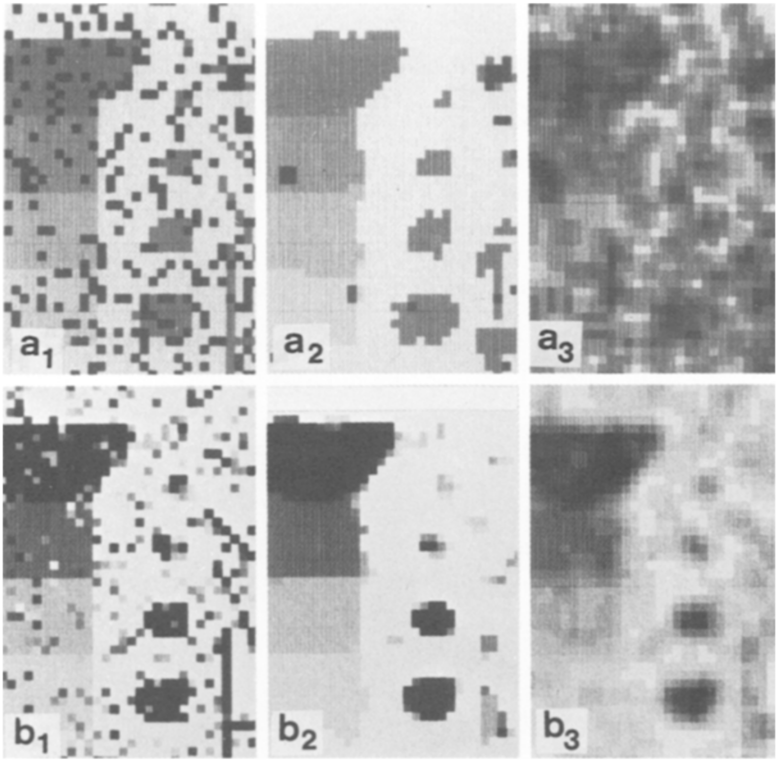


Fig. 5.4a, b. Filtering of impulse noise pictures. (a_1, b_1) Input pictures with model 1 and model 2 noise, respectively, error rates $p=0.2$; (a_2, b_2) median filtered pictures; (a_3, b_3) moving average filtered pictures

the continuous interval $[0, d]$. In short,

$$x_{ij}=\begin{cases} z_{ij} & \text{with probability } p \\ s_{ij} & \text{with probability } 1-\mathrm{p}\,. \end{cases} \tag{5.26}$$

To obtain simple formulas we assume that the original picture is totally white (or totally black) in the neighborhood of (i', j'), i.e., $c=0$ (or $c=d$) in (5.23). This is in a sense the most difficult case for a median filter since all erroneous values then fall on the same side of the correct value. The probability of correct reconstruction is the same as $Q(n, p)$ in (5.25) above, but moreover the magnitude of the remaining errors is diminished. The expected values of the output variables and of the remaining errors are given by

$$E[\mathrm{Median}(x_{i'j'})]=d\cdot\sum_{k=(n+1)/2}^{n}\frac{k-(n-1)/2}{k+1}\binom{n}{k}p^k(1-p)^{n-k} \tag{5.27}$$

and

$$E[\text{Median}(x_{i'j'})|\text{Erroneous reconstruction at } (i',j')]$$
$$= E[\text{Median}(x_{i'j'})]/[1-Q(n,p)]\,. \tag{5.28}$$

Proof: Let $N_{i'j'}$ denote the number of errors in A centered at (i',j'). $N_{i'j'}$ is binomially distributed. The conditional distribution of the median at (i',j'), given that $N_{i'j'}$ equals k, is the same as the distribution of $\text{Median}(0,\ldots,0,z_1,\ldots,z_k)$ where the number of zeros is $n-k$, and $z_1,\ldots,z_k$ are independent and uniformly distributed on $(0,d)$. Further

$$\text{Median}(0,\ldots,0,z_1,\ldots,z_k) = \begin{cases} 0 & \text{if} \quad k \leqq (n-1)/2 \\ z_{(r;k)} & \text{if} \quad k \geqq (n+1)/2\,, \end{cases} \tag{5.29}$$

where $z_{(r;k)}$ denotes the rth-order statistic of $z_1,\ldots,z_k$, and $r=k-(n-1)/2$.

Finally

$$\begin{aligned} E_A[\text{Median}(x_{i'j'})] &= \sum_{k=0}^{n} E[\text{Median}(x_{i'j'})|N_{i'j'}=k]P(N_{i'j'}=k) \\ &= \sum_{k=(n+1)/2}^{n} E\{z_{[k-(n-1)/2;k]}\}\binom{n}{k}p^k(1-p)^{n-k} \\ &= \sum_{k=(n+1)/2}^{n} d\cdot\frac{k-(n-1)/2}{k+1}\binom{n}{k}p^k(1-p)^{n-k}\,. \end{aligned} \tag{5.30}$$

In the last step we used the expected values of uniform order statistics; see, e.g., [Ref. 5.3, Chap. 3]. Q.E.D.

Figure 5.4b illustrates the case $n=9$ and $p=0.2$. According to the above formulas the proportion of erroneous points ought to be reduced from $p=0.2$ to $1-Q(n,p)=0.0196$ and the expected size of an error ought to be reduced from $E(z_{ij})=0.5\cdot d$ to [by (5.28)]

$$\frac{0.00366}{0.0196}\cdot d = 0.187\cdot d\,. \tag{5.31}$$

The result of the actual filtering agrees well with those estimates.

Moving averages are not well suited for filtering of impulse noise and salt-and-pepper noise as can be seen in Fig. 5.4. Some filters intended for reduction of salt-and-pepper noise were suggested by *Rosenfeld* and *Kak* [5.11]. We have not carried out a comparison between those filters and the median filter.

A kind of noise which is very similar to impulse noise is *missing line noise* which occurs when pictures are scanned and entire scan lines are lost or when their values are erroneously decoded [5.9]. A median filter with rectangular window with size $n\times m$ (m points on n scan lines) will, as for impulse noise, reduce the number of errors. Assume that erroneous lines occur with probability p independent of other lines, that all values in an erroneous line get the same value d, and that $s_{ij}=c$. Then the probability of correct reconstruction is $Q(n, p)$. Note that here n denotes the number of lines in the rectangular window. Computationally, the simplest window is $m=1$ but larger m values can sometimes be advantageous for other reasons.

5.3 Edges Plus Noise

So far we have seen that median filters preserve edges (with no noise added) whereas moving averages blur such edges. We have also seen that for normal white noise (with constant background) moving averages reduce the noise somewhat more than median filters with the same window size. In this section we study filtering of edges with added white noise, i.e., sequences or pictures with variables

$$x=s+z\,, \tag{5.32}$$

where s denotes deterministic signal values which equal 0 on one side of the edge and equal h on the other side, and z are white-noise variables. In Sect. 5.3.1 we compare the effects of moving medians and moving averages on such sequences (pictures). Section 5.3.2 contains the mathematical derivation of the distribution of order statistics on such sequences. The derivation is included in this presentation since it gives an example of the kind of arguments to use when deriving results on order statistics from independent random variables, for example (5.8) and the results in Sect. 5.4.1.

5.3.1 Comparison of Median Filters and Moving Averages

In this subsection we assume that the noise variables z are normal $N(0, \sigma)$. To begin with we consider one-dimensional filtering and assume that the jump of the edge is at point $i=1$ (cf. Fig. 5.5). Thus, for $i \leqq 0$, x_i is $N(0, \sigma)$ and for $i \geqq 1$, x_i is $N(h, \sigma)$.

The density $g(x)$ of an n point median with k variables x_i being $N(h, \sigma)$ and $n-k$ variables x_i being $N(0, \sigma)$ can be obtained from (5.37–39) in the next subsection. Though the formula for $g(x)$ is rather complicated, it is quite simple to compute means and standard deviations by numerical integration. This has been done for $\sigma=1$ and some values of n and k. The results are shown in Table 5.3. Values for $\sigma \neq 1$ can be obtained by a suitable change of scale, and values for $k \geqq (n+1)/2$ can be obtained by symmetry arguments.

The distribution of a moving average is easily obtained as $N(hk/n, \sigma/\sqrt{n})$ where k is the number of variables with $s=h$ within the actual window.

Table 5.3. Moments of medians on edge-plus-noise variables $z_1+h, \ldots, z_k+h, z_{k+1}, \ldots, z_n$, where the z_is are i.i.d. $N(0, 1)$

			h					
n	k		0.0	1.0	2.0	3.0	4.0	≧5.0
3	1	E (Median)	0.000	0.305	0.486	0.549	0.562	0.564
		σ (Median)	0.670	0.697	0.760	0.806	0.822	0.826
9	3	E (Median)	0.000	0.318	0.540	0.626	0.641	0.642
		σ (Median)	0.408	0.424	0.471	0.513	0.527	0.529
5	1	E (Median)	0.000	0.179	0.270	0.294	0.297	0.297
		σ (Median)	0.536	0.551	0.580	0.596	0.600	0.600
5	2	E (Median)	0.000	0.386	0.676	0.808	0.841	0.846
		σ (Median)	0.536	0.560	0.631	0.705	0.740	0.748
25	5	E (Median)	0.000	0.184	0.286	0.312	0.315	0.315
		σ (Median)	0.248	0.256	0.271	0.280	0.282	0.282
25	10	E (Median)	0.000	0.391	0.719	0.900	0.944	0.948
		σ (Median)	0.248	0.260	0.295	0.346	0.371	0.375

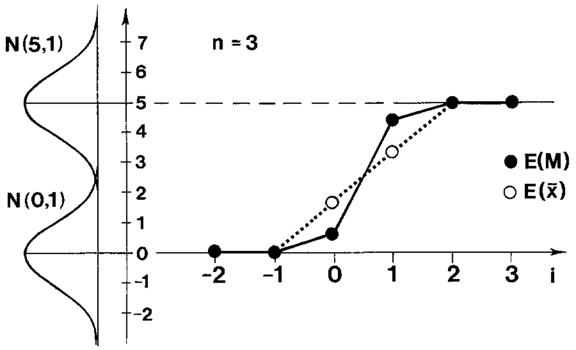


Fig. 5.5. Edge plus noise. Expectations for a moving median (M), and for a moving average ($\bar{x}$), $n=3$, $h=5$, $\sigma=1$

Figure 5.5 shows the successive expected values of medians and means with $n=3$, around an edge with height $h=5$. The expected values of the moving average follow a ramp and indicate substantial blurring of the edge. The expected values of the moving median also indicate some blurring, though much smaller than for the moving average.

To be able to compare the efficiencies of filters on edge-plus-noise sequences we need a goodness-of-fit measure. We shall use the average of the mean-square-errors (MSE) at N points close to the edge

$$\frac{1}{N}\sum_i E(y_i - s_i)^2, \tag{5.33}$$

where y_i denotes the filter output values. For the case shown in Fig. 5.5, i.e., $n=3$ and $N=2$, the expression (5.33) equals $E(y_0^2)$. In Fig. 5.6 we have plotted

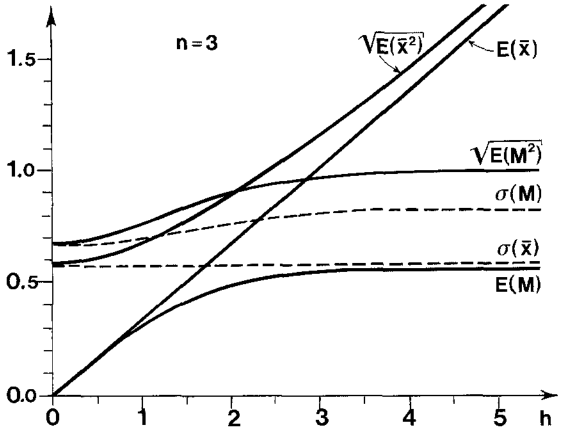


Fig. 5.6. Edge plus noise. Moments for a moving median (M), and for a moving average ($\bar{x}$), $n=3$, $k=1$, $\sigma=1$

$E(y_0^2)^{1/2}$ against the edge height h, both for a median filter and for a moving average filter. It is seen that for $h<2\,(h<2\cdot\sigma)$, the MSE for the moving average is somewhat smaller than the MSE for the median filter, but for $h>3\,(h>3\cdot\sigma)$ the MSE of the median is considerably smaller than that of the mean. This result suggests that moving medians are much better than moving averages for large edge heights, $h>3\sigma$, and that for smaller edge heights there is rather little difference between the two filters. Very similar results are obtained for larger windows, $n=5$, and for two-dimensional filtering, 3×3 and 5×5 windows. The conclusions are also confirmed by the filtered pictures in Fig. 5.3 where h is fixed and $\sigma=0$, $h/5$, $h/3$, and h. The goodness-of-fit measure we have used can only measure the sharpness *across* the edge. It does not say anything about the smoothness of the filtered picture *along* the edge. Figure 5.3 indicates that moving averages give results which are smooth along the edges whereas the median filtered edges are slightly rugged.

We now make some further comments on the behavior of medians as h varies. Throughout we assume that k, the number of variables with mean h, is less than $(n+1)/2$. From Fig. 5.6 and Table 5.3 it is seen that the standard deviations increase with h and have bounded asymptotic values. The expectations are for small h close to the expectations of the corresponding means

$$E(\text{Median})\approx E(\bar{x})=h\cdot\frac{k}{n}, \tag{5.34}$$

but for large h they have a bounded asymptote and thus behave radically differently from the expectations of the means. The explanation is that for large h, say $h>4$, the x variables with mean 0 (say $x_1, \ldots, x_{n-k}$ here) will be separated from the x variables with mean h (say $x_{n-k+1}, \ldots, x_n$ here) and then

$$\text{Median}(x_1, \ldots, x_{n-k}, \ldots, x_n)\approx x_{[(n+1)/2,\,n-k]}, \tag{5.35}$$

where the expression on the right-hand side denotes the $(n+1)/2$th order statistic of $x_1, \ldots, x_{n-k}$. Expectations and variances of normal order can be found in *Owen* [5.12]. An approximate formula for the expectation of $x_{(r,n)}$ is

$$E[x_{(r,n)}] \approx F^{-1}\left(\frac{r}{n+1}\right), \tag{5.36}$$

where $F(x)$ is the distribution function of the x variables.

Finally, it is of course possible to use the above results in analyzing objects other than edges. The model $x = s + z$ can be used to describe, for example, pulses with added noise.

5.3.2 Distribution of Order Statistics in Samples from Two Distributions

Let $x_1, \ldots, x_n$ be independent random variables, let $x_1, \ldots, x_k$ have distribution function $F_1(x)$ and density $f_1(x) = F'_1(x)$, and let $x_{k+1}, \ldots, x_n$ have distribution function $F_2(x)$ and density $f_2(x) = F'_2(x)$. Then $x_{(r,n)}$, the rth order statistic of $x_1, \ldots, x_n$ [$r = (n+1)/2$ gives the median] has density

$$g(x) = g_1(x) + g_2(x), \tag{5.37}$$

where

$$\begin{aligned} g_1(x) = \sum_j k \binom{k-1}{j} \binom{n-k}{r-j-1} f_1(x) F_1(x)^j F_2(x)^{r-j-1} \\ \cdot [1 - F_1(x)]^{k-j-1} [1 - F_2(x)]^{n-k-r+j+1}, \end{aligned} \tag{5.38}$$

and

$$\begin{aligned} g_2(x) = \sum_j (n-k) \binom{k}{j} \binom{n-k-1}{r-j-1} f_2(x) F_1(x)^j F_2(x)^{r-j-1} \\ \cdot [1 - F_1(x)]^{k-j} [1 - F_2(x)]^{n-k-r+j}. \end{aligned} \tag{5.39}$$

The summations are to be carried out over all natural numbers j for which all involved binomial coefficients $\binom{p}{q}$ fulfill $p \geqq q \geqq 0$.

Proof: The density $g(x)$ of $x_{(r,n)}$ can be obtained by

$$g(x) = \lim_{dx \to 0+} \frac{1}{dx} P[x \leqq x_{(r,n)} \leqq x + dx]. \tag{5.40}$$

The method of proof is to split up the event in (5.40) into subevents whose probabilities can be calculated. The number of different subevents is calculated using combinatorics. We shall also use the fact that at most one variable x_i can fall into an infinitesimal interval $[x, x+dx]$. The following subevents are used:

A_1: One of $x_1, \ldots, x_k$ falls into $[x, x+dx]$
A_2: One of $x_{k+1}, \ldots, x_n$ falls into $[x, x+dx]$
B_j: Exactly j variables of $x_1, \ldots, x_k$ fall into $(-\infty, x)$, $j=0, \ldots, k$.

By the law of total probability

$$\begin{aligned}&\frac{1}{dx} P[x \leqq x_{(r,n)} \leqq x+dx] \\ &\quad = \sum_j \frac{1}{dx} P[\{x \leqq x_{(r,n)} \leqq x+dx\} \cap A_1 \cap B_j] \\ &\quad + \sum_j \frac{1}{dx} P[\{x \leqq x_{(r,n)} \leqq x+dx\} \cap A_2 \cap B_j]\,. \end{aligned} \tag{5.41}$$

Consider an event in the first sum. It occurs if and only if one variable of $x_1, \ldots, x_k$ falls into $[x, x+dx]$, j variables of those remaining of $x_1, \ldots, x_k$ fall into $(-\infty, x)$, $r-j-1$ variables of $x_{k+1}, \ldots, x_n$ fall into $(-\infty, x)$, and all remaining variables fall into $(x+dx, +\infty)$. The probability of this event is

$$\begin{aligned}&kf_1(x)dx \cdot \binom{k-1}{j} F_1(x)^j \cdot \binom{n-k}{r-j-1} F_2(x)^{r-j-1} \\ &\quad \cdot [1-F_1(x+dx)]^{k-j-1} \cdot [1-F_2(x+dx)]^{n-k-r+j+1}\,. \end{aligned} \tag{5.42}$$

Inserting this into (5.41) and letting $dx \to 0+$, we obtain $g_1(x)$. By similar arguments one obtains $g_2(x)$ from the second sum in (5.41). In this case one variable from $x_{k+1}, \ldots, x_n$ falls into $[x, x+dx]$. Q.E.D.

5.4 Further Properties of Median Filters

In this section we examine covariance and spectral properties of median filters. The obtained results show that the second-order properties of median filters are very similar to those of moving average filters. The last subsection contains some results on sample path behavior.

5.4.1 Covariance Functions; White-Noise Input

Covariance functions for median filtered white-noise sequences were computed in *Justusson* [5.10]. This was done by first deriving formulas for the distribution of pairs of order statistics from partly overlapping samples and then computing covariances by numerical integration using these formulas. The distribution formulas were derived using elaborations of the ideas used in Sect. 5.3.2 but since the formulas are rather complicated we will not reproduce them here. We will, however, present the resulting covariance functions for normal white noise.

Let $\{x_i\}$ be independent $N(0, 1)$-variables and set

$$C(n,m,k) = \mathrm{Cov}[\underset{n}{\mathrm{Median}}(x_1, \ldots, x_n), \underset{m}{\mathrm{Median}}(x_{n-k+1}, \ldots, x_{n+m-k})], \tag{5.43}$$

i.e., $C(n,m,k)$ is the covariance of medians from samples of sizes n and m with overlap size k. Numerical values of $C(n,m,k)$ are presented in Table 5.4 together with probabilities $P(n,m,k)$ for the two medians in (5.43) to be equal. For nonoverlapping samples, i.e., $k=0$, we of course have $C(n,m,k)=0$. Note also that $C(n,n,n)=\mathrm{Var}(\mathrm{Median})$. Covariances of general normal variables, $N(m_x, \sigma)$, are given by $\sigma^2 \cdot C(n,m,k)$.

The autocovariance function for the output from an n point median filter, $y_i = \mathrm{Median}(x_i)$, on normal $N(m_x, \sigma)$ white noise is

$$r_y(\tau) = \mathrm{Cov}(y_{i+\tau}, y_i) = \sigma^2 C[n, n, (n-|\tau|)^+], \tag{5.44}$$

where $(a)^+ = \max(a, 0)$. Figure 5.7 shows r_y for $n=3$, $\sigma=1$ together with the covariance function r_z of a three-point moving average. The formula for r_z is for general n given by (white-noise input)

$$r_z(\tau) = \sigma^2 \left(1 - \frac{|\tau|}{n}\right)^+. \tag{5.45}$$

In Fig. 5.7 it is seen that r_y and r_z are similar in form or equivalently that the normalized correlation functions are similar, but that r_y has larger values, i.e., $r_y(0) > r_z(0)$.

Covariance functions for two-dimensional median filters can also be expressed in terms of $C(n,m,k)$. For a $n \times n$ square window we get

$$r_y(\tau_1, \tau_2) = \sigma^2 C[n^2, n^2, (n-|\tau_1|)^+ \cdot (n-|\tau_2|)^+]. \tag{5.46}$$

Figure 5.8 shows r_y for a 3×3 window. Only the values in the first quadrant are shown. The others can be obtained by symmetry.

The similarity of the correlation functions of moving medians and moving averages is to some extent explained by the relatively high correlation between a median and a mean. (Further explanation is given in the next section.) If $\{x_i\}$ are independent $N(m, \sigma)$ distributed variables then

$$\mathrm{Cov}\left[\mathrm{Median}(x_1, \ldots, x_n), \frac{1}{n}\sum_1^n x_i\right] = \mathrm{Var}(\bar{x}) = \frac{\sigma^2}{n}; \tag{5.47}$$

Table 5.4. Covariances $C(n, m, k)$ and probabilities of equality $P(n, m, k)$ for pairs of medians from $N(0,1)$-samples of sizes n and m with overlap size k [$C(n, n, n) = \mathrm{Var}$ (Median)]

n	m	k	$C(n, m, k)$	$P(n, m, k)$
1	1	1	1.0000	1.0000
3	1	1	0.3333	0.3333
3	3	1	0.1177	0.1333
3	3	2	0.2480	0.3333
3	3	3	0.4487	1.0000
5	1	1	0.2000	0.2000
5	3	1	0.0721	0.0857
5	3	2	0.1494	0.2000
5	3	3	0.2398	0.4000
5	5	1	0.0445	0.0571
5	5	2	0.0914	0.1286
5	5	3	0.1422	0.2286
5	5	4	0.2003	0.4000
5	5	5	0.2868	1.0000
7	1	1	0.1428	0.1429
7	3	1	0.0520	0.0635
7	3	2	0.1069	0.1429
7	3	3	0.1672	0.2571
7	5	1	0.0323	0.0433
7	5	2	0.0660	0.0952
7	5	3	0.1016	0.1619
7	5	4	0.1402	0.2571
7	5	5	0.1848	0.4286
7	7	1	0.0235	0.0333
7	7	2	0.0478	0.0722
7	7	3	0.0732	0.1195
7	7	4	0.1000	0.1810
7	7	5	0.1290	0.2698
7	7	6	0.1621	0.4286
7	7	7	0.2104	1.000
9	1	1	0.1111	0.1111
9	3	1	0.0407	0.0505
9	3	2	0.0832	0.1111
9	3	3	0.1286	0.1905
9	5	1	0.0253	0.0350
9	5	2	0.0516	0.0758
9	5	3	0.0791	0.1255
9	5	4	0.1081	0.1905
9	5	5	0.1398	0.2857
9	7	1	0.0186	0.0272
9	7	2	0.0376	0.0583
9	7	3	0.0573	0.0949
9	7	4	0.0778	0.1400
9	7	5	0.0995	0.1991
9	7	6	0.1230	0.2857
9	7	7	0.1499	0.4444
9	9	1	0.0146	0.0224
9	9	2	0.0296	0.0476
9	9	3	0.0449	0.0766
9	9	4	0.0608	0.1110
9	9	5	0.0774	0.1536
9	9	6	0.0949	0.2100
9	9	7	0.1138	0.2929
9	9	8	0.1352	0.4444
9	9	9	0.1661	1.0000
25	25	5	0.0105	0.0253
25	25	10	0.0214	0.0581
25	25	15	0.0329	0.1061
25	25	20	0.0453	0.1972
25	25	23	0.0538	0.3452
25	25	24	0.0572	0.4800
25	25	25	0.0617	1.0000
49	49	7	0.0040	0.0125
49	49	14	0.0081	0.0272
49	49	21	0.0123	0.0456
49	49	28	0.0166	0.0701
49	49	35	0.0209	0.1069
49	49	42	0.0258	0.1797
49	49	47	0.0296	0.3597
49	49	48	0.0305	0.4898
49	49	49	0.0318	1.0000
81	81	9	0.0018	0.0074
81	81	18	0.0039	0.0158
81	81	27	0.0059	0.0256
81	81	36	0.0079	0.0373
81	81	45	0.0099	0.0521
81	81	54	0.0120	0.0721
81	81	63	0.0142	0.1028
81	81	72	0.0165	0.1649
81	81	79	0.0185	0.3658
81	81	80	0.0188	0.4938
81	81	81	0.0193	1.0000
1	1	1	1.0000	1.0000
9	1	1	0.1111	0.1111
9	9	9	0.1661	1.0000
25	1	1	0.0400	0.0400
25	9	9	0.0519	0.1199
25	25	25	0.0617	1.0000
49	1	1	0.0206	0.0204
49	9	9	0.0261	0.0550
49	25	25	0.0286	0.1157
49	49	49	0.0318	1.0000
81	1	1	0.0123	0.0123
81	9	9	0.0157	0.0320
81	25	25	0.0171	0.0594
81	49	49	0.0180	0.1091
81	81	81	0.0193	1.0000
121	1	1	0.0083	0.0083
121	9	9	0.0105	0.0210
121	25	25	0.0114	0.0372
121	49	49	0.0119	0.0598
121	81	81	0.0123	0.1027
121	121	121	0.0129	1.0000
2	2	2	0.5000	1.0000
4	4	4	0.2982	1.0000
6	6	6	0.2147	1.0000
8	8	8	0.1682	1.0000
12	12	12	0.1175	1.0000
16	16	16	0.0904	1.0000
20	20	20	0.0734	1.0000
24	24	24	0.0619	1.0000
28	28	28	0.0535	1.0000
32	32	32	0.0471	1.0000
36	36	36	0.0420	1.0000
40	40	40	0.0380	1.0000

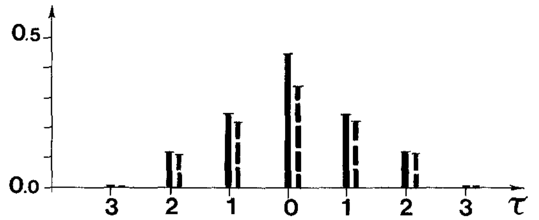


Fig. 5.7. Covariance functions for a three-point moving median (solid lines) and for a three-point moving average (dashed lines) on normal white noise, $N(0, 1)$

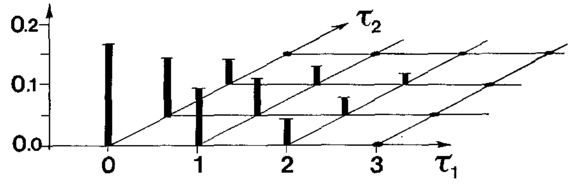


Fig. 5.8. Covariance function for a two-dimensional median filter with a 3×3 square window on normal white noise, $N(0, 1)$

see [Ref. 5.3, p. 31]. Using (5.12) we get for large n the correlation coefficient

$$\varrho(\text{Median}, \bar{x}) \approx \sqrt{\frac{2}{\pi}} = 0.80\,. \tag{5.48}$$

5.4.2 Covariance Functions; Nonwhite-Noise Input

For autocovariances of median filtered nonwhite noise it is not possible to give general exact formulas. We shall here present some approximate formulas which were derived in *Justusson* [5.10] by considering limit results when the window size tends to infinity. The approximations often work surprisingly well also for small window sizes. For details in the derivations we refer to [5.10].

Assume that $\{x_i\}$ is a stationary mixing sequence with marginal distribution function $F(x)$ with density $f(x)$. We have

$$P[\text{Median}(x_{i-v}, \ldots, x_i, \ldots, x_{i+v}) \leqq x] = P\left[\sum_{j=-v}^{v} \text{sign}(x_{i+j} - x) \leqq 0\right]. \tag{5.49}$$

By "inversion" of (5.49) one can obtain the following approximate representation formula (Bahadur representation) for large n:

$$\begin{aligned} y_i &= \text{Median}(x_{i-v}, \ldots, x_i, \ldots, x_{i+v}) \\ &\approx \tilde{m} + \frac{1}{2f(\tilde{m})n} \sum_{j=-v}^{v} \text{sign}(x_{i+j} - \tilde{m})\,, \end{aligned} \tag{5.50}$$

where $v = (n-1)/2$ and $F(\tilde{m}) = 0.5$. Thus, a moving average behaves asymptotically as a moving average of sign variables (hard clipped variables). Now an approximation formula for the covariance function of the median filtered sequence can be obtained by computing the covariance function for the moving average to the right in (5.50). This yields

$$r_y(\tau) \approx \frac{1}{nf^2(\tilde{m})} \sum_{j=-(n-1)}^{n-1} \left(1 - \frac{|j|}{n}\right) c_{j+\tau}, \tag{5.51}$$

where $c_k = P(x_0 \leqq \tilde{m},\ x_k \leqq \tilde{m}) - 1/4$. For normal noise with covariance function $r_x(\tau) = \sigma^2 \varrho(\tau)$ the quantities c_k can be computed exactly. By doing this and employing the small sample modification mentioned in Sect. 5.2.1 we get

$$r_y(\tau) \approx \frac{\sigma^2}{n + \pi/2 - 1} \sum_{j=-(n-1)}^{n-1} \left(1 - \frac{|j|}{n}\right) \arcsin[\varrho(j+\tau)]\,. \tag{5.52}$$

In [5.10] we examined the accuracy of the approximation (5.52) for normal white noise and normal AR(1) processes. For input processes with zero, positive, or moderately negative correlations, the accuracy is good even for very small n values. On the other hand, for a process with correlation function $\varrho(\tau) = (-0.9)^{|\tau|}$ the accuracy is bad. Moving medians have only minor smoothing effects on such a process which behaves like $x_i \approx (-1)^i y$ over long time intervals. In fact, an input sequence $x_i = (-1)^i y$, $i \in \mathbb{Z}$, will be unaltered in form by a moving median [though shifted one step for some n values (cf. Chap. 6)]. On the other hand moving averages have a very strong smoothing effect on such a process since the regularly fluctuating x values give rise to cancellations. In general the approximation formulas for the covariances of moving medians can be expected to work well only for sequences on which the median filters act similar to moving averages. In particular they should not be expected to work well on highly oscillatory sequences and on edge sequences.

We can now explain the similarity between the correlation properties of median filters and moving average filters. For large n (5.51) can be approximated by

$$r(\tau) \approx \text{const} \cdot \left(1 - \frac{|\tau|}{n}\right)^{+}. \tag{5.53}$$

Furthermore, (5.53) holds asymptotically for all moving averages although with different constants. Thus, they have the same normalized correlation function but may have different asymptotic variances.

In Sect. 5.2.2 we mentioned that medians over large windows are approximately normally distributed. This result can be proved by using the Bahadur representation and applying a central limit theorem for stationary mixing processes to the right-hand side of (5.50).

The above ideas can also be applied to two-dimensional median filtering. We get the following Bahadur representation

$$y_{ij} = \operatorname*{Median}_{A}(x_{ij}) \approx \tilde{m} + \frac{1}{2f(\tilde{m})n} \sum_{(r,s)\in A} \operatorname{sign}(x_{i+r,j+s} - \tilde{m}), \tag{5.54}$$

where n is the size of the window A. For normal noise $\{x_{ij}\}$ the corresponding approximation of the covariance function $r_y(\tau_1, \tau_2)$ becomes

$$r_y(\tau_1, \tau_2) \approx \frac{\sigma^2}{n+\pi/2-1} \sum_{(r,s)\in A} \sum_{(r',s')\in A} \arcsin[\varrho(\tau_1 + r - r', \tau_2 + s - s')]. \tag{5.55}$$

For some windows (5.55) can be simplified further.

5.4.3 Frequency Response

Impulse response, step response, and frequency response functions are often used to describe filters. Since a median filter wipes out impulses and preserves edges, the impulse response is zero, and the step response is unity. We shall in this subsection compute the power spectrum distribution of median filtered cosine waves in the cases $n=3$ and $n=5$.

To begin with we consider continuous time filtering. Let, for $0 \leqq \omega_0 \leqq \pi$,

$$x(t) = \cos(\omega_0 t), \qquad t \in R, \tag{5.56}$$

$$y(t) = \operatorname{Median}[x(t-1), x(t), x(t+1)], \qquad t \in R. \tag{5.57}$$

It is easily seen (cf. Fig. 5.9a) that

$$y(t) = \begin{cases} \cos[\omega_0(t-1)] & 0 \leqq t \leqq 1/2 \\ \cos(\omega_0 t) & 1/2 \leqq t \leqq T/2 - 1/2 \\ \cos[\omega_0(t+1)] & T/2 - 1/2 \leqq t \leqq T/2, \end{cases} \tag{5.58}$$

where $T = 2\pi/\omega_0$. Since $y(t)$ is an even periodic function with period T, (5.58) determines $y(t)$ for all t.

The variance of $y(t)$ is obtained by straightforward integrations

$$\sigma_y^2 = \frac{1}{T} \int_0^T y^2(t)\,dt \tag{5.59a}$$

$$= \frac{1}{2}\left[1 + \frac{1}{\pi}(\sin 2\omega_0 - 2\sin\omega_0)\right], \qquad 0 \leqq \omega_0 \leqq \pi. \tag{5.59b}$$

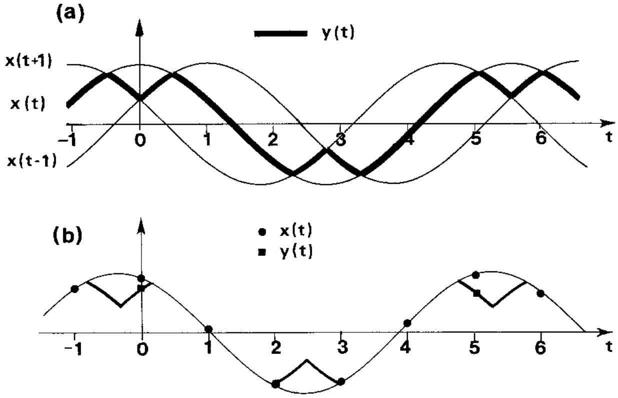


Fig. 5.9a, b. Median filtering of a cosine wave, $n=3$. (**a**) Continuous time, $x(t)=\cos(\omega_0 t)$; (**b**) discrete time, $x(t)=\cos(\omega_0 t+\theta)$, $\omega_0=9/8$, period $T=2\pi/\omega_0=5.6$, $y(t)=\text{Median}[x(t-1), x(t), x(t+1)]$

We can expand $y(t)$ as a Fourier series with coefficients

$$c_k=\frac{1}{T}\int_0^T e^{itk\omega_0}y(t)dt, \qquad k=0, \pm 1, \pm 2, \ldots. \tag{5.60}$$

Simple calculations give

$$c_1=\frac{1}{2}\left(1-\frac{\omega_0}{\pi}-\frac{\omega_0}{\pi}\cos\omega_0\right), \qquad 0\leqq\omega_0\leqq\pi. \tag{5.61}$$

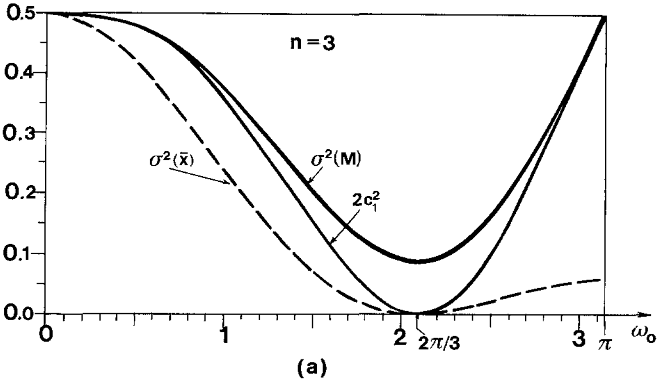
Note that $2c_1^2$ is the spectral mass (effect) at the frequencies $\pm\omega_0$, and that σ_y^2 is the total spectral mass (effect). Figure 5.10a shows σ_y^2 and $2c_1^2$ for $0\leqq\omega_0\leqq\pi$. Also shown is σ_y^2 for a corresponding three-point moving average ($2c_1^2=\sigma_y^2$ for linear filters).

It is seen that for low-frequency cosine input ($\omega_0\leqq 2\pi/3$ or $T\geqq n$) the moving median and moving average have similar response, whereas for $\omega_0>2\pi/3$ both σ_y^2 and $2c_1^2$ increase for the median, and at $\omega_0=\pi$ they reach the same value as for $\omega_0=0$. This is explained by the fact that a three-point median filter will preserve the form of a sequence $x_i=(-1)^i x$ but shift it one step.

In discrete time the choice of zero as phase angle in (5.56) is rather arbitrary. If, instead, the phase angle θ is chosen at random with uniform distribution on $[0, 2\pi]$, one obtains a stationary process

$$x(t)=\cos(\omega_0 t+\theta), \quad t=0, \pm 1, \pm 2, \ldots, \tag{5.62}$$

and the median filtered output (cf. Fig. 5.9b) will have the same covariance properties as the continuous time process (5.57). The power spectrum



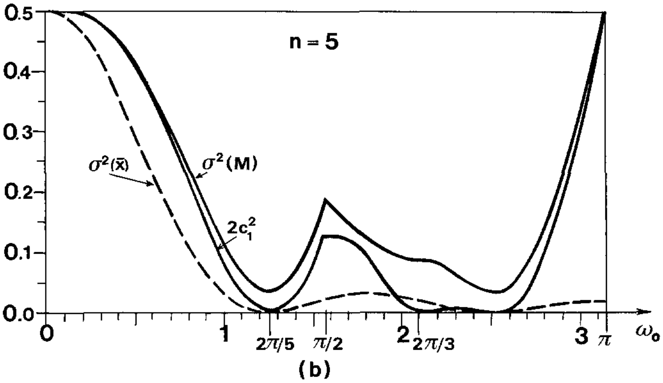


Fig. 5.10a, b. Filtering of a cosine wave. (**a**) $n=3$, (**b**) $n=5$. Variance $\sigma^2(M)$ and spectral mass at ω_0, $2c_1^2$, for a median filtered cosine, and variance $\sigma^2(\bar{x})$ for a moving average filtered cosine

distribution of the median filtered sequence is obtained from (5.60) by folding the spectral masses c_k^2 on $k\omega_0$ into the interval $(-\pi, \pi]$. For ω_0 values which are rational multiples of π, this may result in overlays of spectral masses, but apart from this we can still interpret $2c_1^2$ as the spectral mass (effect) on $\pm\omega_0$.

We have carried out the same kind of analysis for median filters with $n=5$. The results are presented in Fig. 5.10b. The derivations were rather cumbersome. The results agree with simulation experiments performed by *Velleman* [5.13]. For frequencies $\omega_0 \leqq 2\pi/5$, i.e., $T \geqq n=5$, the spectral responses of moving medians and moving averages are similar. For general n this will hold for $T \geqq n$. Results for larger values of n can be obtained by numerical integration of (5.59a) and (5.60) with $y(t)$ as in (5.57).

Two-dimensional median filtering with 3×3 and 5×5 square windows of a cosine wave along the axis of the point grid yields the same variances and spectral components as shown above.

Median filters are nonlinear and hence the above frequency response functions for single cosines do not correspond to transfer functions for general

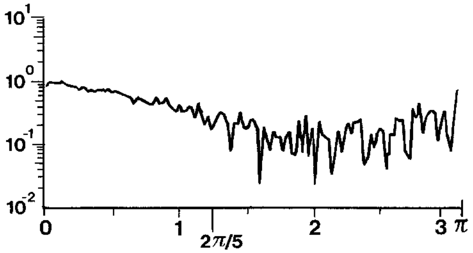


Fig. 5.11. Cross section of an empirical "transfer function" for a 5×5 median filtered picture (reproduced from [Ref. 5.14, Fig. 3.6] with kind permission of G. Heygster)

signals (mixtures of cosines). *Heygster* [5.14] has computed empirical "transfer functions" as quotients of Fourier transforms of output and input pictures. One example for a 5×5 median filter is shown in Fig. 5.11. It shows a cross section of the bivariate transfer function with one spectral coordinate set to zero. Note that for frequencies $\omega_1 \leqq 2\pi/5$ the curve is very smooth, and can thus be interpreted as a transfer function, whereas for $\omega_1 \geqq 2\pi/5$ the curve is very irregular due to interferences from different frequencies. Of course these transfer functions depend on the actual input pictures chosen. Heygster's results confirm the previously noted similarity of spectral response of moving medians and moving averages for frequencies $\omega \leqq 2\pi/n$ or $T \geqq n$.

5.4.4 Sample-Path Properties

We have already discussed the effect of median filtering of edges and highly oscillatory sequences. Excluding such sequences the general shape of median filtered sequences is similar to that of moving average filtered sequences. This follows from the Bahadur representation (5.50).

Although the general appearance of moving median and moving average sequences is the same there are differences in details. Several authors have noted that median filtered sequences include many runs of equal values. The following result gives a simple estimate of the frequency of equal successive values. Let the input sequence $\{x_i\}$ be a stationary mixing random process. Then

$$\lim_{n \to \infty} P[\underset{n}{\mathrm{Median}}(x_i) = \underset{n}{\mathrm{Median}}(x_{i+1})] = 0.5 . \tag{5.63}$$

Thus, roughly 50% of all the values in a median filtered sequence are equal to their right neighbors when n is large. (Note that this is not the same as being equal to any one of the two neighboring values.)

Proof: Going from i to $(i+1)$ in the filter computations involves replacing x_{i-v} by x_{i+1+v} in the filter window. If these two variables lie on the same side of $y = \mathrm{Median}(x_i)$ then $\mathrm{Median}(x_{i+1}) = \mathrm{Median}(x_i)$. The mixing condition implies that x_{i-v} and x_{i+1+v} are asymptotically independent. Together with

Median $x_i \approx \tilde{m}$ this gives

$$\begin{aligned} &P[\mathrm{Median}(x_i) = \mathrm{Median}(x_{i+1})] \\ &\quad \approx P(x_{i-v} < \tilde{m}, x_{i+1+v} < \tilde{m}) + P(x_{i-v} > \tilde{m}, x_{i+1+v} > \tilde{m}) \\ &\quad \approx 0.5 \cdot 0.5 + 0.5 \cdot 0.5 = 0.5\,. \end{aligned} \tag{5.64}$$

which proves (5.63).

For white-noise input the probability in (5.63) can be computed exactly and equals $0.5(1-1/n)$, [5.10]. For two-dimensional filtering there is no such simple result like (5.63) but for white-noise input and square windows we can use Table 5.4 to compute probabilities of equality.

$$P[\mathrm{Median}(x_{ij}) = \mathrm{Median}(x_{i,j+1})] = P(n^2, n^2, n^2 - n) \tag{5.65}$$

which for a 3×3 window yields 0.2100 and for a 9×9 window 0.1649.

Another feature of median filtered sequences is the occurrence of small occasional edges, to be compared with moving average filtered sequences which vary in a very "continuous" way. To remove such occasional edges *Rabiner* et al. [5.5] applied a three-point linear filter after the median filtering.

5.5 Some Other Edge-Preserving Filters

In this section we describe some filters which have the same basic properties as median filters: preservation of edges and reduction of noise. All filters to be considered, except those in the last subsection, are variations on the median theme. The previous median filters will here be called *simple median filters*. Variance formulas for the filter output are given for white-noise input.

5.5.1 Linear Combination of Medians

Let A_k, $k = 1, \ldots, K$ be different windows. Then a *linearly combined median filter* [5.15, 16] is defined through

$$y_{ij} = \sum_{k=1}^{K} a_k \underset{A_k}{\mathrm{Median}}(x_{ij}), \tag{5.66}$$

where a_k are real coefficients. The window sets can, for example, be squares with side lengths $1, 3, \ldots, 2K-1$, or circles with diameters $1, 3, \ldots, 2K-1$. One can of course also choose window sets which do not include the origin, for example square frames or circle rings (cf. Fig. 5.2).

If all the window sets A_k are symmetric around and include the origin, conditions (5.5, 6), then an edge picture $\{x_{ij}^0\}$ will be preserved in form but the height of the edge will be modified by the factor $\sum a_k$,

$$y_{ij} = \left(\sum_{k=1}^{K} a_k\right) \cdot x_{ij}^0 . \tag{5.67}$$

This follows from the fact that each median in the combination preserves the edge. Note that if $\sum a_k = 0$ then $y_{ij} \equiv 0$.

For normal white-noise input $\{x_{ij}\}$ one can compute the variance of y_{ij} using the results in Sect. 5.4.1

$$\mathrm{Var}(y_{ij}) = \sum_{k=1}^{K} \sum_{m=1}^{K} a_k a_m C(n_k, n_m, n_{km}) , \tag{5.68}$$

where $C(\cdot, \cdot, \cdot)$ is as in Sect. 5.4.1 and n_k, n_m, n_{km} denote the numbers of points in A_k, A_m, and $A_k \cap A_m$, respectively. Table 5.4 contains values of $C(\cdot, \cdot, \cdot)$ needed when the windows are squares, circle rings, and square frames. The two latter collections of window sets consist of mutually disjoint window sets for which the variance formula simplifies to

$$\mathrm{Var}(y_{ij}) = \sum_{k=1}^{K} a_k^2 C(n_k, n_k, n_k) . \tag{5.69}$$

Another way to obtain information about the behavior of linearly combined median filters is to use the fact that simple median filters and moving averages have many similar properties. If one replaces each median in (5.66) with a mean of the same x values, one obtains a linear filter,

$$z_{ij} = \sum_{k=1}^{K} a_k \operatorname*{Mean}_{A_k}(x_{ij}) . \tag{5.70}$$

The second-order properties of this filter are easy to calculate and will give at least some information about the properties of the corresponding median filter. One can also go backwards and start with a linear filter with desirable properties and construct a corresponding linearly combined median filter. This is most simply done if the windows are mutually disjoint, e.g., square frames or circle rings.

5.5.2 Weighted-Median Filters

In simple median filters all x values within the window have the same influence on the resulting output. Sometimes one may want to give more weight to the central points. The following type of filter has this possibility. For the sake of simplicity we consider only one-dimensional filters.

The basic idea is to change the number of variables in each window set $(x_{i-v}, \ldots, x_i, \ldots, x_{i+v})$ by taking not one number x_{i+r} but k_r numbers all equal to x_{i+r} and then compute the median of the extended sequence of numbers. We call the resulting filter a *weighted-median filter.*

A simple example may clarify the construction. For $n=3$, $k_{-1}=k_1=2$, $k_0=3$, we have

$$\begin{aligned} y_i &= \text{Weighted-Median}(x_{i-1}, x_i, x_{i+1}) \\ &= \text{Median}(x_{i-1}, x_{i-1}, x_i, x_i, x_i, x_{i+1}, x_{i+1}). \end{aligned} \tag{5.71}$$

If the weights are symmetric, $k_{-r}=k_r$, and if k_0 is odd then a weighted-median filter preserves edges.

If the input $\{x_i\}$ is white noise with density $f(x)$ then a weighted median with weights $\{k_r\}$ is for large n approximately normally distributed $N(\tilde{m}, \sigma_n)$ where $\tilde{m}$ is the theoretical median and

$$\sigma_n^2 = \frac{\sum k_r^2}{(\sum k_r)^2} \cdot \frac{1}{4f^2(\tilde{m})}. \tag{5.72}$$

Since this result has not been treated elsewhere we give a brief indication of how it can be proved: By using an extended Bahadur representation [see (5.50)] we get

$$\begin{aligned} y_i &= \text{Weighted-Median}(x_i) \\ &\approx \tilde{m} + \frac{1}{2f(\tilde{m})\sum k_r} \sum_{r=-v}^{v} k_r \operatorname{sign}(x_{i+r} - \tilde{m}). \end{aligned} \tag{5.73}$$

By the central limit theorem the right-hand side in (5.73) is approximately normally distributed provided that the Lindeberg condition is fulfilled, which, for a weight sequence $k_r = k_r^{(n)}$, here amounts to

$$\max_r \frac{[k_r^{(n)}]^2}{\sum_i [k_i^{(n)}]^2} \to 0 \quad \text{as} \quad n \to \infty. \tag{5.74}$$

This condition essentially requires that none of the factors k_r is considerably larger than the others.

5.5.3 Iterated Medians

Since median filters preserve edges so do *iterations of median filters* [5.1, 8]. *Tukey* also suggested the following remarkable smoothing procedure: Repeat median filtering until no more changes occur, i.e., until one reaches a median-filtering invariant sequence (fixed-point sequence) (cf. Chap. 6). Note that this invariant sequence need not be constant, in contrast to iterations of moving averages on stationary sequences $\{x_i\}$ which ultimately result in constant sequences.

The statistical properties of iterated medians seem difficult to analyze. We can only present some experiences from simulations of one-dimensional AR(1) processes. For nonnegatively correlated AR(1) sequences, parameter $a \geqq 0$, only a few iterations were needed to reach a fixed-point sequence and there were only small changes after the first median filtering. Therefore, it seems likely that the variance formulas for simple medians hold approximately also for iterated medians on processes with nonnegative covariances. For AR(1) processes with alternating positive and negative correlations, $a<0$, a large number of iterations were needed to reach an invariant sequence and great changes occurred during the filtering process. The final sequences were much smoother and closer to the mean level than the one-step filtered sequences.

When using iterated medians it is of course possible to use different windows in the successive iterations. *Pratt* [5.8] and *Narendra* [5.17] investigated a two-dimensional filtering method in which one first applies a one-dimensional median filter to each line of the picture and then a one-dimensional median filter to each row of the resulting picture, i.e., first

$$z_{ij} = \mathrm{Median}(x_{i,j-v}, \ldots, x_{ij}, \ldots, x_{i,j+v}), \tag{5.75}$$

and then

$$y_{ij} = \mathrm{Median}(z_{i-v,j}, \ldots, z_{ij}, \ldots, z_{i+v,j}). \tag{5.76}$$

The filter is called a *separable median filter*. Its statistical properties can be analyzed theoretically if $\{x_{ij}\}$ is white noise with density $f(x)$ (see [5.17]). The fundamental point is that the z variables in (5.76) are independent since they are computed from x variables in different lines. *Narendra* computed the exact density f_z of the z variables by (5.8) and inserted f_z into the approximate variance formula (5.9) for y_{ij}. We shall present a somewhat simpler formula using the fact that f_z is approximately a normal density $N(\tilde{m}, \sigma_n)$ according to (5.9). Insertion into (5.12) yields

$$\mathrm{Var}(y_{ij}) \approx \frac{1}{n+\pi/2-1} \cdot \frac{\pi}{2} \cdot \frac{1}{4f^2(\tilde{m})n}. \tag{5.77}$$

For $\{x_{ij}\}$ being normal $N(m, \sigma)$ we get by using (5.77) and the small sample modification in Sect. 5.2.1

$$\mathrm{Var}(y_{ij}) \approx \frac{\sigma^2}{(n+\pi/2-1)^2} \left(\frac{\pi}{2}\right)^2. \tag{5.78}$$

For large n this yields output variance $(\sigma/n)^2 \cdot 2.47$ to be compared with the output variance of a simple $n \times n$ median, $(\sigma/n)^2 \cdot 1.57$, and that of a $n \times n$ moving average, $(\sigma/n)^2 \cdot 1$. The variance of a separable median is (roughly) 57% larger than that of an $n \times n$ median and (roughly) 147% larger than that of a moving average.

5.5.4 Residual Smoothing

Assume that the x variables are generated by a signal-plus-noise model

$$x_i = s_i + n_i , \tag{5.79}$$

where the signal $\{s_i\}$ varies slowly compared with the noise $\{n_i\}$. Median filtering gives an estimate of s_i

$$y_i = \text{Median}(x_i) \approx s_i . \tag{5.80}$$

Thus, the residuals

$$\hat{n}_i = x_i - \text{Median}(x_i) \approx n_i \tag{5.81}$$

give estimates of the noise variables. Further median filtering of the residuals could reduce the noise further

$$z_i = \text{Median}(\hat{n}_i) \approx 0 . \tag{5.82}$$

Addition of y_i and z_i now gives a hopefully good estimate of s_i,

$$\hat{s}_i = y_i + z_i = \text{Median}(x_i) + \text{Median}[x_i - \text{Median}(x_i)] . \tag{5.83}$$

This *residual smooting* (or *double smoothing*) technique has been suggested and studied by *Rabiner* et al. [5.5], *Velleman* [5.13], and others.

It is easily realized that this smoothing technique preserves edges. Some further combined smoothing methods can be found in [5.18].

5.5.5 Adaptive Edge-Preserving Filters

The filters described earlier are general-purpose smoothing filters. For restricted classes of pictures special filters can be designed. One important class of pictures occurs in remote sensing in connection with classification of picture points into a limited number of classes. The pictures can be described as consisting of compact regions of essentially constant grey level, i.e., underlying the observed picture is an ideal picture consisting of regions of constant grey levels, but the observed pictures have been degraded by addition of noise. Several smoothing algorithms have been proposed for this type of pictures; see, e.g., *Davis* and *Rosenfeld* [5.19] and *Nagao* and *Matsuyama* [5.20] where further references can be found. The main principles used in the proposed algorithms are as follows:

A window is centered at the point (i, j). If there is little variation between the values in the window, it is decided that (i, j) is an interior point and the value at (i, j) is replaced by the average of the values within the window. If, instead, there is much variation between the values in the window, it is decided that (i, j) is a boundary point and (i, j) is assigned the average value of the points which lie on the same side of the border as (i, j) does.

As measures of variation one has used local variances, Laplacians, or gradients. The computational efforts increase very rapidly with window size so it has been suggested to iterate the algorithms instead of using large windows. The algorithms not only preserve sharp edges but also sharpen blurred edges. For example, ramps get changed into step edges. Thus the algorithms do not really "preserve" slanting edges. Some of the algorithms are designed to preserve not only edges but also sharp corners of the region boundaries. For details concerning the aspects touched upon in this subsection we refer to [5.19, 20].

5.6 Use of Medians and Other Order Statistics in Picture Processing Procedures

In the preceding sections we have examined the smoothing ability of median filters. We will now give some examples of how medians can be combined with other picture processing procedures. The examples are taken from edge detection, object extraction, and classification. The final subsection contains a brief account of general order statistics and their use in picture processing.

5.6.1 Edge Detection

The edge-preservation property of median filters makes them suitable as *prefilters* in edge detection, i.e., smoothing the picture with a median filter before applying an edge detector. Many edge detectors are of the type

$$w_{ij} = \begin{cases} 1 & \text{if} \quad |g_1| > \Delta \quad \text{or} \quad |g_2| > \Delta \\ 0 & \text{otherwise}, \end{cases} \tag{5.84}$$

where g_1, g_2 are gradients at the point (i, j) and Δ is a threshold value. When presmoothing has been performed, it is advisable to use gradients which are symmetric and have span larger than one unit. For example

$$\begin{aligned} g_1 &= y_{i+u,j} - y_{i-u,j} \\ g_2 &= y_{i,j+u} - y_{i,j-u} \end{aligned} \tag{5.85}$$

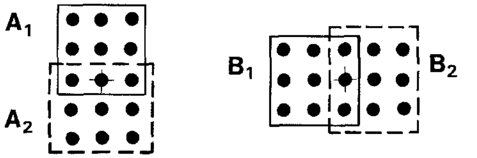


Fig. 5.12. Edge detection. Prefiltering with a median filter (here 3×3) corresponds to a block-type edge detector. The centering point is marked with a small cross

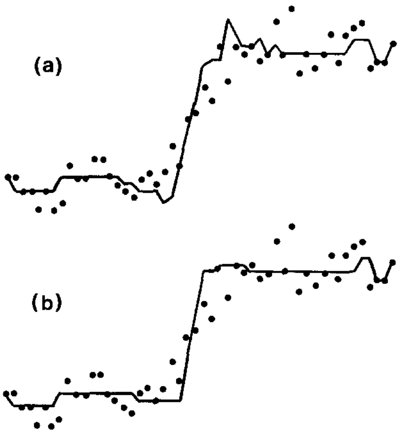


Fig. 5.13a, b. Restoration of a step edge. (**a**) Restoration by a linear filter; (**b**) restoration by five iterations of a linear filter and a median filter. (Dots show the input data; the output points are connected by lines) (reproduced from [Ref. 5.7, Fig. 2.6] with kind permission of the Optical Society of America © and B. R. Frieden)

with $1/2 \leqq u \leqq n/2$, n = side length of the smoothing window. Note that using median filtering and such an edge detector is equivalent to using a block-type edge detector in which gradients are computed from medians over blocks,

$$\begin{aligned} g_1 &= \underset{A_1}{\mathrm{Median}}(x_{ij}) - \underset{A_2}{\mathrm{Median}}(x_{ij}), \\ g_2 &= \underset{B_1}{\mathrm{Median}}(x_{ij}) - \underset{B_2}{\mathrm{Median}}(x_{ij}). \end{aligned} \tag{5.86}$$

Figure 5.12 illustrates square 3×3 blocks. Gradient span equals 2 ($u = 1$).

Moving medians can also be used as *post-filters* after edge enhancement [5.7]. A sharp edge which has been degraded by a low-pass linear filter can be sharpened by inverse filtering or some modification of inverse filtering. Since the impulse responses of linear filters of inverse type often have large side lobes, they give rise to spurious oscillations at edges (overshots) (cf. Fig. 5.13a). *Frieden* [5.7] used a median filter to eliminate such spurious oscillations, see Fig. 5.13b, in which the linear restoration and median filtering were iterated five times. The window size used, $n = 11$, was equal to the period of the oscillations. Note that the edge in Fig. 5.13b not only has fewer spurious oscillations, but also is sharper than the edge in Fig. 5.13a.

5.6.2 Object Extraction

Extraction of objects in a picture is a common problem in picture processing. We shall consider objects spread out on a smoothly varying background, and

		9	9					9	9		
		9	9					9	9		
		9						9			

(a)

1	1	1	1	1	1	1	2	3	4	5	6
1	1	1	1	1	1	1	2	3	4	5	6
1	1	10	10	1	1	1	2	12	13	5	6
1	1	10	10	1	1	1	2	12	13	5	6
1	1	1	1	1	1	1	2	3	4	5	6
1	1	1	1	1	1	1	2	3	4	5	6
1	1	1	1	1	1	1	2	3	4	5	6
1	1	10	1	1	1	1	2	12	4	5	6
1	1	1	1	1	1	1	2	3	4	5	6

(b)

1	1	1	1	1	1	1	2	3	4	5	6
1	1	1	1	1	1	1	2	3	5	5	6
1	1	1	1	1	1	1	2	4	5	6	6
1	1	1	1	1	1	1	2	4	5	6	6
1	1	1	1	1	1	1	2	3	5	5	6
1	1	1	1	1	1	1	2	3	4	5	6
1	1	1	1	1	1	1	2	3	4	5	6
1	1	1	1	1	1	1	2	3	4	5	6
1	1	1	1	1	1	1	2	3	4	5	6

(c)

1	1	1	1	1	1	1	2	3	4	5	6
1	2	3	3	2	1	1	3	5	6	6	6
1	3	5	5	3	1	1	4	7	8	7	6
1	3	5	5	3	1	1	4	7	8	7	6
1	2	3	3	2	1	1	3	5	6	6	6
1	1	1	1	1	1	1	2	3	4	5	6
1	2	2	2	1	1	1	3	4	5	5	6
1	2	2	2	1	1	1	3	4	5	5	6
1	2	2	2	1	1	1	3	4	5	5	6

(d)

0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	−1	0	0
0	0	9	9	0	0	0	0	8	8	−1	0
0	0	9	9	0	0	0	0	8	8	−1	0
0	0	0	0	0	0	0	0	0	−1	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	9	0	0	0	0	0	9	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

(e)

0	0	0	0	0	0	0	0	0	0	0	0
0	−1	−2	−2	−1	0	0	−1	−2	−2	−1	0
0	−2	5	5	−2	0	0	−2	5	5	−2	0
0	−2	5	5	−2	0	0	−2	5	5	−2	0
0	−1	−2	−2	−1	0	0	−1	−2	−2	−1	0
0	0	0	0	0	0	0	0	0	0	0	0
0	−1	−1	−1	0	0	0	−1	−1	−1	0	0
0	−1	8	−1	0	0	0	−1	8	−1	0	0
0	−1	−1	−1	0	0	0	−1	−1	−1	0	0

(f)

Fig. 5.14a–f. Object extraction. (**a**) Original objects; (**b**) filter input picture, consisting of objects with added background; (**c**) 3×3 median filtering of the input picture; (**d**) 3×3 moving average filtering of the input picture; (**e**) subtraction of (**c**) from the input picture; (**f**) subtraction of (**d**) from the input picture we assume that the objects are relatively well separated. An illustrative example is an aerial photo of stones spread out on hills. An appropriate extraction technique is: first smooth the picture to obtain an estimate of varying background, then subtract this estimate from the original picture. Ideally, this leaves the objects on a constant background (with zero level) and they can then be detected by thresholding. As smoothing operator one can use, for example, a moving median or a moving average, i.e.,

$$y_{ij}=x_{ij}-\underset{A}{\mathrm{Median}}\,(x_{ij}), \tag{5.87}$$

or

$$z_{ij}=x_{ij}-\underset{A}{\mathrm{Mean}}\,(x_{ij}). \tag{5.88}$$

The windows can be squares or crosses. If the windows are at least twice as large as the objects then the moving median wipes out the objects and hence the subtraction in (5.87) extracts the objects perfectly. This holds in those parts of the picture where the background is constant. In other parts some distortion of the objects occurs. However, the moving average based filter (5.88) gives considerably more distortion at all points. Figure 5.14 illustrates these aspects. Finally, we note that (5.87) is a linearly combined median filter with $\sum a_k=1-1=0$ and thus cancels all edges.

5.6.3 Classification

Multispectral classification of remote sensing pictures is conventionally performed as pointwise classification, i.e., in classifying a point only the spectral values at that point are used. However, for noisy pictures some kind of smoothing is often needed and several methods have been proposed.

Post-smoothing of the result from a pointwise classification through local majority decisions is an easily implemented and frequently used method. Another simple method is presmoothing of each spectral component by moving averages or moving medians. All these methods have the drawback that a pixel which is uniquely classified by its own spectral values can be reclassified due to its neighbors. By combining the spectral values at each point with the median of the neighboring points and using the combined values for classification we obtain an algorithm without the mentioned drawback.

Let $x_{ij}^{(k)}$, $k=1,\dots,K$, denote the spectral values at point (i,j) and let

$$y_{ij}^{(k)}=\underset{A}{\mathrm{Median}}\,[x_{ij}^{(k)}]. \tag{5.89}$$

Then use the extended feature vector

$$(x_{ij}^{(k)}, y_{ij}^{(k)};\ k=1,\dots,K)$$

in classification with conventional classification rules for multivariate normal distributions.

The use of medians for smoothing is motivated by its edge preservation which in this context amounts to preservation of class boundaries.

5.6.4 General Order Statistics

The rth-*order statistics* $x_{(r,n)}$ of n numbers $x_1, \ldots, x_n$ is defined as the rth number in size. In particular $x_{(1,n)} = \min x_i$, $x_{(n,n)} = \max x_i$, and $x_{[(n+1)/2,n]} = \text{Median}(x_i)$. Filtering with *moving order statistics* is performed similar to median filtering: a window is moved over the picture and the rth order statistic of the values within the window is computed. These filters are also called *percentile filters*.

Moving order statistics can be used for shrinking and expanding of objects in pictures (erosion and dilation). Applications have been made, for example, to cell pictures [5.14]. For shrinking $r < (n+1)/2$ is used, e.g., $r = 1$, and for expanding $r > (n+1)/2$, e.g., $r = n$. Note that for $r = (n+1)/2$, i.e., for medians, the average level of the picture is not changed. Shrinking and expanding are often performed by first thresholding the pictures and then applying moving r-out-of-n decision rules. Note that filtering by moving order statistics followed by thresholding gives identically the same results. The intermediate picture may give valuable information. Figure 5.15 illustrates filtering with 3×3 windows and $(r, n) = (2, 9)$.

The statistical literature on order statistics is voluminous. Most of the results on medians presented earlier have counterparts for general order statistics [5.3, 10]. For white-noise input both exact and asymptotic distributions can be obtained, whereas for nonwhite noise only asymptotic results are available. However, the limiting behavior of extreme order statistics [with $r/n \to 0$ or 1, e.g., $x_{(1,n)}$ and $x_{(n,n)}$] are different in nature. In particular, the limiting distributions are nonnormal.

1	0	1	1	0	0	1
0	0	0	7	6	9	0
0	0	0	6	3	5	0
0	6	7	6	5	6	0
1	3	9	4	0	0	1
1	4	5	5	0	1	0
0	0	1	1	0	1	0

(a)

·	·	·	·	·	·	·
·	0	0	0	0	0	·
·	0	0	0	5	0	·
·	0	0	0	0	0	·
·	1	4	0	0	0	·
·	0	1	0	0	0	·
·	·	·	·	·	·	·

(b)

·	·	·	·	·	·	·
·	0	0	0	0	0	·
·	0	0	0	1	0	·
·	0	0	0	0	0	·
·	0	1	0	0	0	·
·	0	0	0	0	0	·
·	·	·	·	·	·	·

(c)

0	0	0	0	0	0	0
0	0	0	1	1	1	0
0	0	0	1	0	1	0
0	1	1	1	1	1	0
0	0	1	1	0	0	0
0	1	1	1	0	0	0
0	0	0	0	0	0	0

(d)

·	·	·	·	·	·	·
·	0	0	0	0	0	·
·	0	0	0	1	0	·
·	0	0	0	0	0	·
·	0	1	0	0	0	·
·	0	0	0	0	0	·
·	·	·	·	·	·	·

(e)

Fig. 5.15a–e. Shrinking of objects. (**a**) Input picture; (**b**) $x_{(2,9)}$ filtering of the input picture; (**c**) thresholding of (**b**) with threshold = 4; (**d**) thresholding of the original picture with threshold = 4; (**e**) 2-out-of-9 filtering of (**d**)

For moving order statistics on stationary mixing processes $\{x_i\}$ with r/n staying away from 0 and 1, the mean of the output is asymptotically equal to the λ percentile of the marginal distribution function $F(x)$ of x_i with $\lambda = r/(n+1)$, i.e.,

$$E[x_{(r,n)}] \approx x_\lambda = F^{-1}(\lambda), \tag{5.90}$$

and the autocovariance function is asymptotically

$$r(\tau) \approx \text{const} \cdot \left(1 - \frac{|\tau|}{n}\right)^+, \tag{5.91}$$

similar to (5.53) for medians. By (5.90) one can estimate how much the average level of the sequence or picture will be changed. Formula (5.91) shows that moving order statistics have a smoothing effect if r/n is not too close to 0 or 1.

Heygster [5.14] investigated empirical transfer functions (cf. Sect. 5.4.3) also for moving order statistics. For r/n not too close to 0 or 1 these transfer functions have a similar low-pass character as those of moving medians, but for r/n close to 0 or 1 they were not smooth even at low frequencies.

Acknowledgements. The author wishes to thank Prof. B. Rosén and Dr. T. Orhaug for helpful discussions in the course of this work. The author also wants to thank Gunilla Johnson for drawing the figures.

References

5.1 J.W.Tukey: *Exploratory Data Analysis* (Addison-Wesley, Reading, Mass., 1977, preliminary ed. 1971)
5.2 A.E.Sarhan, B.G.Greenberg (eds.): *Contributions to Order Statistics* (Wiley, New York 1962)
5.3 H.A.David: *Order Statistics* (Wiley, New York 1970)
5.4 J.L.Gastwirth, H.Rubin: Ann. Stat. **3**, 1070–1100 (1975)
5.5 L.R.Rabiner, M.R.Sambur, C.E.Schmidt: IEEE Trans. ASSP-**23**, 552–557 (1975)
5.6 N.S.Jayant: IEEE Trans. COM-**24**, 1043–1045 (1976)
5.7 B.R.Frieden: J. Opt. Soc. Am. **66**, 280–283 (1976)
5.8 W.K.Pratt: *Digital Image Processing* (Wiley, New York 1978)
5.9 G.W.Wecksung, K.Campbell: Computer **7**, 63–71 (1974)
5.10 B.Justusson: "Order Statistics on Stationary Random Processes, with Applications to Moving Medians"; Tech. Rpt. TRITA-MAT-1, Royal Institute of Technology, Stockholm, Sweden (1979)
5.11 A.Rosenfeld, A.C.Kak: *Digital Picture Processing* (Academic Press, New York 1976)
5.12 D.B.Owen: *Handbook of Statistical Tables* (Addison-Wesley, Reading, Mass. 1962)
5.13 P.F.Velleman: "Definition and Comparison of Robust Nonlinear Data Smoothing Algorithms"; Tech. Rpt. Economic and Social Statistics Dept., Cornell University (1978)
5.14 G.Heygster: "Untersuchung von Zweidimensionalen Rangordnungsoperatoren im Orts- und Frequenzbereich", in *Bildverarbeitung und Mustererkennung*, ed. by E. Triendl, Informatik Fachberichte, Vol. 17 (Springer, Berlin, Heidelberg, New York 1978) pp. 204–208
5.15 B.Justusson: "Noise Reduction by Median Filtering" in Proc. 4th Intern. Joint Conf. on Pattern Recognition (1978) pp. 502–504
5.16 S.G.Tyan: "Fixed Points of Running Medians"; Tech. Rpt. Dept. of Electrical Engineering and Electrophysics, Polytechnic Institute of New York (1977)
5.17 P.M.Narendra: "A Separable Median Filter for Image Noise Smoothing" in Proc. IEEE Conf. on Pattern Recognition and Image Processing (1978) pp. 137–141
5.18 P.F.Velleman: Proc. Nat. Acad. Sci. USA **74**, 434–436 (1977)
5.19 L.S.Davis, A.Rosenfeld: IEEE Trans. SMC-**8**, 704–710 (1978)
5.20 M.Nagao, T.Matsuyama: "Edge Preserving Smoothing" in Proc. 4th Intern. Joint Conf. on Pattern Recognition (1978) pp. 518–520