Bull. Spec. Astrophys. Obs., 2002, 53, 114–118

Fast image processing method for PC: 7. From median to mode filtering

T.B. Georgiev

Rozhen National Astronomical Observatory, BG-4700 Smolyan, Bulgaria, tsgeorg@astro.bas.bg

Received July 31, 2001; accepted November 10, 2001.

Abstract. A method for mode filtering of astronomical images with integer type pixels is described. It is implemented as an addition to the method of fast median filtering. The mode is derived in four stages: (1) the median of the pixel values in the filtering window is derived; (2) the shortest histogram interval containing half of the number of pixels is determined; (3) the (new) median in this interval is derived; (4) the weighted average of the pixel values in the vicinity of the new median is calculated. Mode filtering removes small scale images (cosmics, stars) more efficiently than median filtering. Thresholded mode filtering, as well as thresholded median filtering, preserves stellar peaks. The computing time is about 3 times longer than in the case of median filtering. As examples of processing, a simulated and a real frames are given.

Key words: methods: image processing — methods: numerical

1. Introduction

Median filtering is the main representative of the rank statistic methods (Kim & Yaroslavskii, 1986; Pasian, 1991). It was first introduced by Tukey (1971) and used for removing outliers from economic data rows. As opposed to the average, the median is almost independent of single impulse noise events, i.e. appears to be a robust estimation of the mean value. For this reason the median filtering is widely used for smoothing or cleaning of astronomical frames.

Let an approximately round pixel window with diameter W slides over the rows of the frame. Let i and j be the index numbers of the rows and columns of the frame, M(i, j) be the pixel value, corresponding to the current center of the median window and MED(i, j) is the median of the pixel values within the window. Then in its simplest application median filtering MED(i, j) changes M(i, j) and a smoothed frame is produced. In the application of the mode filtering the mode MOD(i, j) must play the role of the median MED(i, j).

The process of median (or mode) filtering may be ruled by two independent parameters. One of them is the diameter W of the smoothing window, besides larger W causes stronger smoothing. The other one is the threshold value $T = S \cdot C$, which is the product of the estimated width S of "the sigma" of the local histogram and an optional threshold coefficient C (usualy C = 3). Then the median MED(i, j) changes the current pixel value M(i, j) if $|MED(i,j) - M(i,j)| > T$. It must be noted that when a stellar image occurs in the median window, the local histogram becomes wider than that in the case of pure background. Then the threshold value T grows up naturally and the peak of the stellar image may be omitted without change. This is an efficient method for cleaning impulse noise (cosmics) from astronomical frames preserving details of interest (peaks of stellar images) (see also Sun & Neuvo, 1994). Though, in the case of a "big pixel sampling" the thresholded median filtering can not preserve efficiently the peaks of the bright stars and it must be applied with care.

Median filtering is a very important method in astronomical image processing with at least three main applications. First, using a big filtering window (e.g. 51–201 pixels) one performs decomposition of the frame into two frames — smoothed and residual. The smoothed frame contains the large scale shape of the background or of the large extended object (galaxy, comet) and the residual frame contains the small and intermediate scale details of interest (stars, stellar aggregates). Second, using an intermediate window (15–35 pixels), one can detach the faint and intermediate stellar images. Then the smoothed frame may show the intermediate scale structures of the object (spiral arms of a galaxy) and the residual frame contains stellar images on an approximately flat background. Third, the thresholded median filtering with a small window diameter (3–5 pixels) is used for removing impulse noise (cosmics, bad pixels).

The fast median filtering algorithm for frames with integer pixel values has been proposed by Huang et al. (1979) (see also Frieden, 1980; Huang, 1981). One implementation of this method was described in detail in the first paper of this series (Georgiev, 1996a). In the case of floating point pixel values the median derivation forces preliminary ranging of the data in the window which is a slow procedure. Our tests show that in the case of integer pixel values the median filtering method is ten times faster than the respective floating point implementation in MIDAS and IDL. For this reason we develop the method of mode filtering for frames with integer pixel values.

The mode of the local histogram of the frame is the most frequent brightness value and for this reason the mode filtering must be more resistant against noise or small "noise-like" details in the frame. Compared to median, mode filtering is a more robust and less impulse noise dependent method. The present paper describes a method for mode filtering based on preliminary use of the fast median algorithm.

2. Certain methods of mode derivation

Although the mode of a histogram is an easy parameter to define, it is difficult to determine it well for frame filtering. In the usual case of poor or sparse local histogram the simple derivation of the mode as the most populated histogram column is useless. The reason is that it may coincide accidentally with the local maximum or minimum pixel value, being too far from the true most probable value. The problem is really very complicated and it seems there are no convenient ways for robust and easy estimation of the mode yet. This situation is discussed widely in the papers of Coleman & Andrews (1979) and Davies (1988). Though, two certain methods may be pointed out.

First, the mean value of the histogram may be used together with the median for mode deriving. In the case of big filtering window, when the histogram is richly populated, the natural range of the "three means" is

```
mode < median < average or average < median < mode.
```

Then the ancient method for mode deriving is the approximation (Yule & Kendal, 1965).

mode = average — 3(average-median)

Unfortunately, in the cases of bimodal, sparse or poorly populated histogram, the natural range of the "three means" may be changed. After testing we found that this method is too inaccurate and not suitable for general application.

Second, iterative solution of the problem of mode deriving, called "truncated median", is proposed by Davies (1988) as follows. We again assume that the natural range of the three means is valid and note that the median lies closer to the mode end of the histogram rather than to the average end. So, the mean value is likely to be on the side where the range of the pixel values is high and where the histogram must be truncated. It is asserted that the most sensible place to truncate the histogram is such that the original median bisects the total range of the truncated histogram. The procedure is repeated till the distances between the median and the ends of the truncated histogram are equalized. However, the algorithm of this process is really very complicated and slow. After testing we decided not to use it.

Further we investigated another, more sophisticated approach. It has been proposed by Rousseew & Leroy (1986) as alternative to the classical method of the least squares for deriving the parameters of a linear regression. This approach is based on the method of the least median of the squares of the deviations. In the particular case, for robust estimation of the mode, it involves deriving the mean value of the most narrow histogram interval, containing half (or slightly more) of the total histogram. Our tests show that a better result is reached by deriving a new median of the found interval and (optionally) deriving a weighted average in a subinterval centered on the new median. We find this approach very powerful and describe it below.

3. The proposed method and its algorithm

Mode filtering is implemented in a computer program written in C-language, as in the case of median filtering, described by Georgiev (1996a). The program applies a circular window on the whole frame, including the periphery. The process of big images is foreseen and only a horizontal band of the frame, with a width of W rows, is stored currently in the processor memory.

The classical fast median algorithm (Frieden, 1980; Huang, 1981; Georgiev, 1996a) based on the local histogram is used for preliminary deriving of the local median MED. In the case of the first pixel in the row the histogram is initialized and the MED is found directly by counting the histogram columns. Further, the process is significantly faster. Only the peripheral pixels of the window are removed and added to the histogram. The median value changes slowly along the frame, and later, the search for the current median may be done by fast algorithm, beginning from the old median. For this reason the auxiliary number SUM, which contains the sum of the histogram columns H[N] with numbers N less or equal to MED, is used. It follows the changes of the left half of the histogram. The value of SUM changes with removing and adding the contribution of the peripheral window. If K1 and K2 are the values of such pixels, the process may be written in C-language as

```
H[K1]--; if (K1<=MED) SUM--; /* removing */
H[K2]++; if (K2<=MED) SUM++; /* adding */
```

Let TOT be the total of the pixels in the window and HLF be half of TOT (the integer number from 0.5*TOT+1). Then the process of defining the median (always from left to right in the histogram body) is

```
while(SUM >=HLF) { SUM-=H[MED]; MED--; }
    /* moving back */
while(SUM < HLF) { MED++; SUM+=H[MED]; }
    /* moving forward */
```

We must derive also the bounds M1 and M2 of the "one sigma" histogram interval, excluding the histogram tails with values Q1 and Q2. Assuming gaussian distribution, we introduce Q1 = Q2 = 0.16*TOT. Then using the auxiliary sums S1 and S2, we find M1 and M2 as follows:

```
M1=MED; S1=SUM;
while (S1>Q1) { S1-=H[M1]; M1--; } M1++;
M2=MED; S2=TOT-SUM+H[MED];
while (S2>Q1) {S2-=H[M2]; M2++;} M2--;
```

Now the inner ("one sigma") interval of the histogram is placed between M1 and M2. It contains TOT-Q1-Q2 = 0.68*TOT pixels. If the mode filtering is not preferred in the application, the values M1 and M2 may be used for thresholded median filtering (see Part 1).

Further we give the most important part of the method — determining the bounds of the shortest interval of the histogram H[N], which contains at least half of the pixel amount HLF. Let us denote the bounds of this interval again by M1 and M2. First we must find the number N0 of the first nonzero column of the histogram. We search for M0 beginning from M1, using the auxiliary sum S:

```
M0=M1;  S=S1;
while (S>0) { S-=H[M0]; M0--; } M0++;
```

Let us denote the bounds and the size of the current checked histogram interval by L1, L2 and L21. The first such interval, containing at least HLF pixels is already known. Its bounds are M0 and MED. That is why we initialize the values of M1, M2, L2 and M21 as follows:

```
M1=M0;  M2=L2=MED;  M21=M2-M1;
```

Generally, we may find a few intervals with minimal length and in such a case we must choose the most populated of them. Therefore we must control also the pixel amount SS in such intervals. If the most populated intervals again are more than one, we use the most left of them.

Let S be the number of pixels in the current checked interval. The left bound L1 of the possible intervals lies between M1 and NED and we change it in a loop. We subtract H[L1], H[L1+1], ... from S till reaching S < HLF. Then we add H[L2], H[L2+1], ... to S and immediately after reaching S >= HLF we compare the size and the pixel amount of the interval with the previous intervals. The respective and the most complicated part of the program is

```
S=SUM-H[M0];  SS=HLF;
for (L1=M0+1; L1<=MED; L1++) if (H[L1]>0) {
while (S<HLF) { L2++; S+=H[12]; }  L21=L2=L1;
if(L21==M21 && S>SS) { SS=S; M1=L1; M2=L2; }
if(L21<M21) {
M21=L21; M1=L1; M2=L2; SS=HLF; }  S-=H[L1]; }
```

The bounds M1 and M2 of the desired interval are found. The next operation is estimation of the mode MOD1 in the interval. First we find it as a (new) median inside the bounds M1 and M2. The value of M21 is small and the direct computation is fast. Here the comparing sum QWT is a quarter of the total histogram:

```
MOD1=M1;  S=H[MOD1]; while (S<QWT) { MOD1++;
 S+=H[MOD1]; }
```

In the end we derive (optionally) another estimation of the mode — MOD2. It is a weighted average with coefficients C[ND], where ND is the distance between the number of the histogram column N and the previous estimation of the mode MOD. The computations spread out on a subinterval, centered on MOD, with half length of NDMAX. The value NDMAX is an input parameter of the program, which must be less than the mean expected "one sigma" width of the local histogram. In our case we use NDMAX=9. The coefficients C[ND] may have, e.g., gaussian distribution. However, we use coefficients corresponding to 2nd degree polynomial fit of the data. Such coefficients are derived for smoothing the data row and frame in the paper of Georgiev (1996b). The text of the program is

```
S=SS=0; for (N=M1; N<=M2; N++) { ND=abs(N-MOD1);
if(ND<NDMAX) { D=(double)H[N]*C[ND]; D+=S; SS+=
S*(double)N; } }
MOD2=(int)(SS/S+0.5);
```

The mode value may be used in the same manner as the median value in the median filtering. The bounds M1 and M2 also may be used for thresholded mode filtering as in the thresholded median filtering. However, in such a case the "one sigma" gaussian interval must be found as 1.5*M21/2 (see Rousseew & Leroy, 1986).

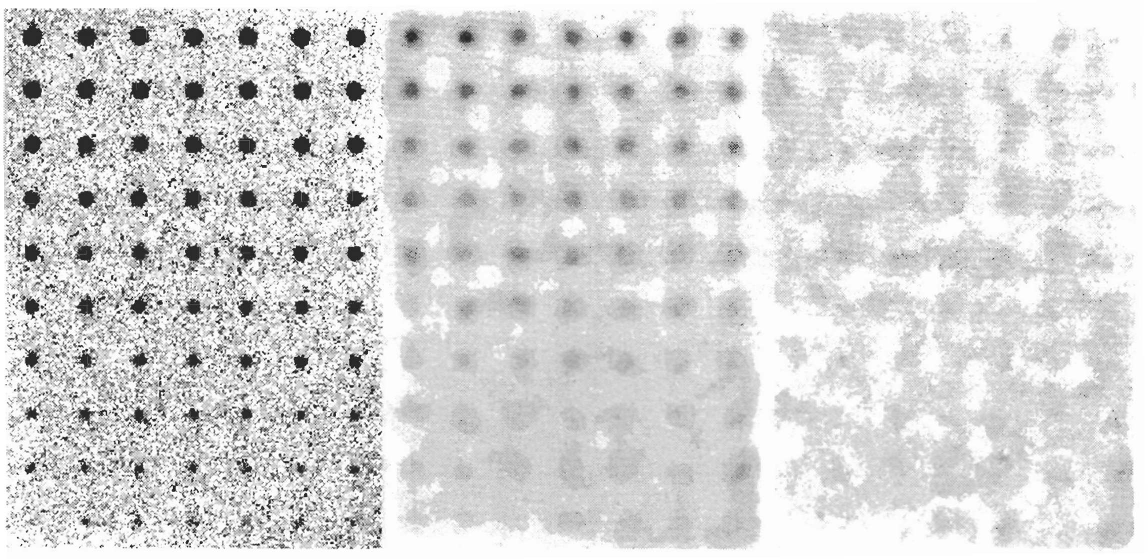

Figure 1: *Removing the stellar images from a simulated frame using window with a diameter of 15 pixels: left part — original; middle part — after median smoothing; right part — after mode smoothing.*

4. Comparisons between median and mode filtering

An example of median and mode filtering in a simulated stellar field is shown in Fig.1. The signal and noise parameters of the field are close to one 20 min CCD exposure of the 2 m RC telescope of the Rozhen NAO in V band. The original frame (on the left part of Fig.1) contains gaussian stellar images of 19–24 mag with sizes (FWHM) about 2.5 pix and added noise. The smoothing window diameter is 15 pix. The result of the median smoothing (on the middle part of Fig.1) contains the pedestals of the stellar images, even in the cases of faint stars. The result of the mode smoothing (on the right part of Fig.1) contains neglected pedestals of stars, even in the cases of relatively bright stars.

Another example of the median and mode filtering is given in Fig.2. It is made on a part of a V frame of the dwarf galaxy Cas 1 with an exposure of 10 min, obtained with the 2 m Rozhen telescope. The telescope is equipped with a Photometrics CCD camera with a size of 1024×1024 pixels. The scale is $0.32''$/pix. The size of the stellar images in this case is $1.6''$ or 5 pix.

The frame in Fig.2 is preliminary processed with the Rozhen software. The cosmics are cleaned by thresholded mode filtering with a window diameter W = 5 pix and threshold T = 3 · sigma. The used part of the original frame (on the left part of Fig.2) contains a few dozen stellar images and noise. The size of the next filtering window is 25 pixels. The result of the median smoothing (on the middle part of Fig.2) shows the large scale shape of the galaxy and remnants of bright stellar images. The result of the mode smoothing (on the right part of Fig.2) shows again the shape of the galaxy. It is not so smooth as in the previous case, but the stellar remnants, even in the cases of bright stars, are very faint.

5. Conclusions

The described method for mode filtering is more efficient than the median filtering in the cases of impulse noise cleaning and stellar images removing. It must be especially noted that the thresholded mode filtering removes outlier pixel values in the periphery of the stellar images more efficiently than the median filtering. At the same time it preserves the stellar peaks as well as the thresholded median filtering does (see Part 1).

One disadvantage of the mode filtering is that the smoothed frame contains slightly more small scale fluctuations than the median filtered one (see Fig.1 and 2). The reason is that the mode is derived by using the number of pixels twice as small as the median. For better smoothing we may derive a mode using more pixels, e.g. 2/3 of the histogram amount, instead half of it. However, then the result of the mode filtering becomes close to that of the median filtering. Though, if a very smoothed frame is necessary, the method of regression smoothing without loss of resolution (Georgiev, 1996b) or other smoothing tool may be used additionally.

Acknowledgements. The author is grateful to P.Grosbol, P.Balester and K.Banse for discussions about the median filtering and the possibilities for implementing the fast mode filtering.

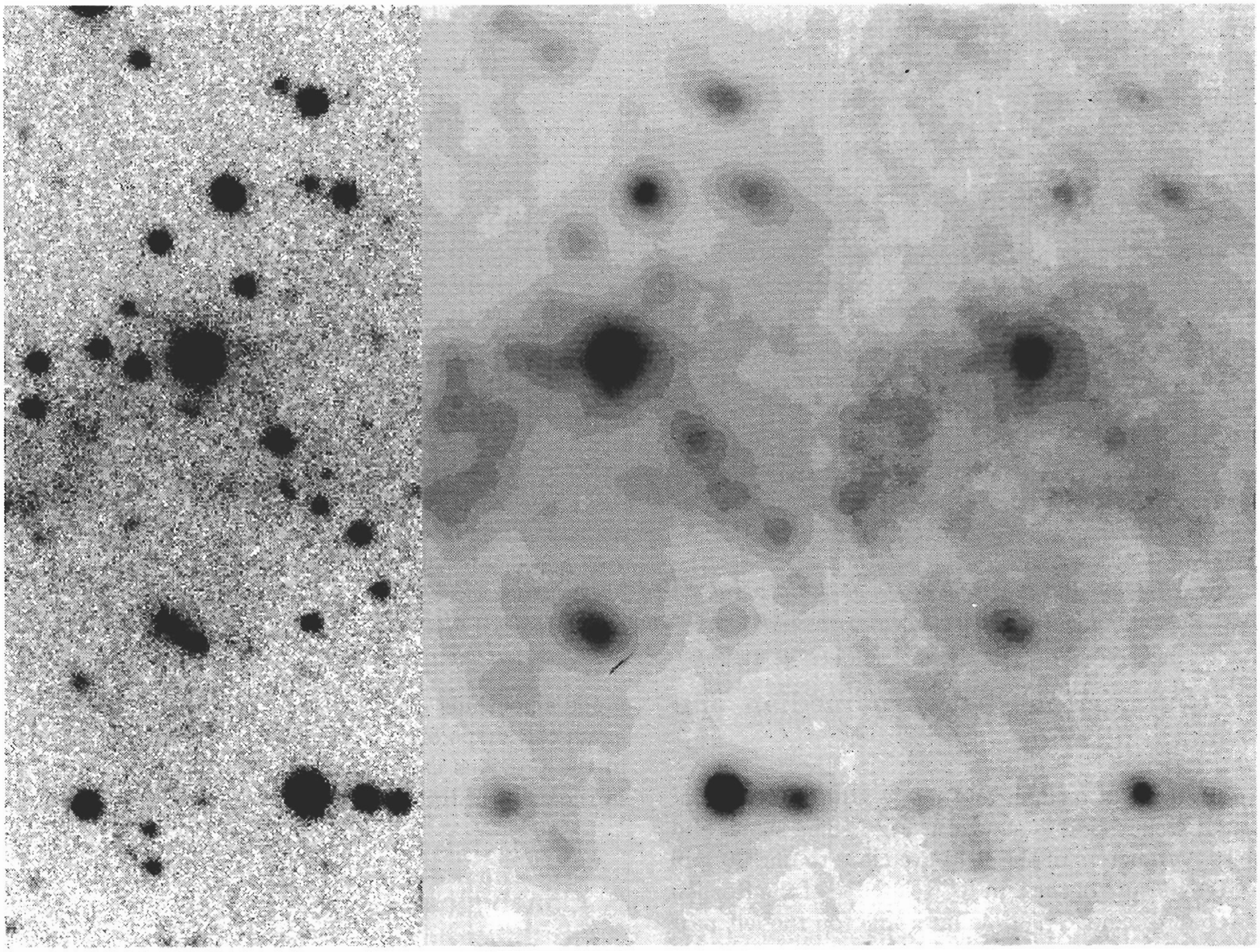

Figure 2: *Unfolding the shape of nearby dwarf galaxy Cas 1 using a window diameter of 25 pixels: left part — original; middle part — after median smoothing; right part — after mode smoothing.*

References

Coleman G.B., Andrews H.C, 1979, Proc. IEEE 67 (5), 773

Davies E.R., 1988, Pattern Recognition Letters, **7**, 87

Frieden B.R., 1980, The computer in optical research. Methods and applications. Springer Verlag. (in Russian: 1985)

Georgiev T.B., 1991, Astrofiz.Issled. (Izv. SAO) **33**, 213

Georgiev T.B., 1996a, Bull. Spec. Astrophys. Obs., **39**, 124

Georgiev T.B., 1996b, Bull. Spec. Astrophys. Obs., **39**, 131

Huang T.S., 1981, Topics in Applied Physics, **43**, Two-dimensional digital signal processing, Springer-Verlag, New York (in Russian: 1985)

Huang T.S., Yang G.F., Tang G.Y., 1979, IEEE Trans. on Acoustic, Speech and signal processing, **27**, 1

Kim V., Yaroslavskii L., 1986, Computer vision, graphics and image processing, **35**, 234

Rousseew J., Leroy M., 1987, Robust estimation of regression and outliers, John Willey, New York, 165

Pasian F., 1991, in: Proc. of 3rd ESO/ST-ECF data analysis workshop, eds.: Grosbol P., Warmels R., 57

Sun T., Neuvo Y., 1994, Pattern Recognition Letters, **15**, 341

Tukey J.W., 1971, in: Exploratory data analysis, Addison-Wesley Publ.Co. 1977 (in Russian: 1981)

Yanshin V., Kalinin G., 1994, Processing of images using C-language, Moscow, Mir

Yule G.U., Kendall M.G., 1965, Introduction of the theory of the statistics, Griffin, London (in Russian: 1968)