PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Fast and robust image processing algorithms based on approximations to order statistics

H. C. Palm

H. C. Palm, "Fast and robust image processing algorithms based on approximations to order statistics," Proc. SPIE 10648, Automatic Target Recognition XXVIII, 1064807 (30 April 2018); doi: 10.1117/12.2304336

SPIE.

Event: SPIE Defense + Security, 2018, Orlando, Florida, United States

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018 Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

Fast and robust image processing algorithms based on approximations to order statistics

H. C. Palm*[a]
[a]Forsvarets forskningsinstitutt, P. O. Box 25, N-2027 Kjeller, Norway

ABSTRACT

A test system with four cameras in the infrared and visual spectra is under development at FFI (The Norwegian Defence Research Establishment). The system can be mounted on a high speed jet aircraft, but may also be used in a land-based version. It can be used for image acquisition as well as for development and test of automatic target recognition (ATR) algorithms. The sensors on board generate large amounts of data, and the scene may be rather cluttered or include anomalies (e.g. sun glare). This means we need image processing and pattern recognition algorithms which are robust, fast (real-time), and able to handle complex scenes. Algorithms based on order statistics are known to be robust and reliable. However, they are in general computationally heavy, and thus often unsuitable for real time applications. But approximations to order statistics do exist. Median of medians is one example. This is a technique where an approximation of the median of a sequence is found by first dividing the sequence in subsequences, and then calculating median (of medians) recursively. The algorithm is very efficient, the processing time is of order $O(n)$. By utilizing such techniques for estimating image statistics, the computational challenge can be overcome. In this paper we present strategies for how approximations to order statistics can be applied for developing robust and fast algorithms for image processing, especially visualization and segmentation.

Keywords: Order statistics, Image processing, visualization, segmentation

1. INTRODUCTION

The origin of this work is a test system with four cameras in the infrared and visual spectra which is under development at FFI (The Norwegian Defence Research Establishment) [1]. This system may be mounted on a high speed jet aircraft, in a land-based version, or used in the lab. Applications are image acquisition as well as development and test of automatic target recognition algorithms (ATR). We need fast and robust algorithms for both image processing and visualization. Such algorithms may involve use of order statistics.

Order statistics for estimating position and deviation, like median and quartile difference, is known to be robust [2]. They are little affected by anomalies in a dataset. Thus, they are very useful in application with data containing outliers. In our applications, sun glare in infrared images is an example of such outliers. However, the drawback with such methods is that they in general are very time-consuming. This is because they often involve sorting of the dataset's elements.

If the dataset consists of integer elements, efficient histogram-based algorithms are available for determining various quantiles. This is the situation if we have 8 bit or 16 bit input from an image sensor. However, this strategy does obviously not work if we have a floating point input. This is the case when the input is a feature image, e.g. a texture image.

We haven't found very much related work reported in the literature. Pratt et al presented the pseudomedian in [3]. This strategy is based on concepts of mathematical morphology, and it can be computed very efficiently. However, it is not robust against outliers. Yang et al has proposed a median filter algorithm without sorting [4]. We haven't studied it further, mainly because it uses integer input. Other approaches which apply medians, (ex. [5]) seem to be very focused on preprocessing of images, especially for removing salt and pepper noise.

In the next section we will describe the median of medians algorithm in detail, and study its characteristics. In section 3 we will present two of our applications of the algorithm. Finally, we will summarize and conclude.

*hans-chr.palm@ffi.no

Automatic Target Recognition XXVIII, edited by Firooz A. Sadjadi, Abhijit Mahalanobis, Proc. of SPIE
Vol. 10648, 1064807 · © 2018 SPIE · CCC code: 0277-786X/18/$18 · doi: 10.1117/12.2304336

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

2. ESTIMATES BASED ON MEDIAN OF MEDIANS

2.1 Calculating the median of medians as a position estimate

Median of medians is far from a new technique. It was first presented by Blum et al back in 1973 [6]. It seems that it has mainly been used as a preprocessing of sorting algorithms. But there is no reason why it can't be used "as is".

A median of medians will of course be an approximation to a median, but our experience is that it has been a sufficiently good approximation in our applications.

It easily shown that the median of a sequence with length 3 can be found as

$$\mathrm{med}(\mathrm{x}_1, x_2, x_3) \quad = \quad \max(\min(x_1, x_2), \min(x_1, x_3), \min(x_2, x_3)) \tag{1}$$

This procedure may be extended to larger sequences, e.g. the median of a sequence of length 5 is given as

$$\begin{aligned} \mathrm{med}(\mathrm{x}_1, x_2, x_3, x_4, x_5) \quad = \quad & \max[\min(x_1, x_2, x_3), \min(x_1, x_2, x_4), \min(x_1, x_2, x_5), \\ & \min(x_1, x_3, x_4), \min(x_1, x_3, x_5), \min(x_1, x_4, x_5), \\ & \min(x_2, x_3, x_4), \min(x_2, x_3, x_5), \min(x_2, x_4, x_5), \\ & \min(x_3, x_4, x_5)] \end{aligned} \tag{2}$$

For large sequences, this median-computing procedure becomes impractical.

In the following, we will restrict ourselves to sequences of length 3^n, $n = 1,2,3,\ldots$. Let's assume we have a sequence of length nine. Then an approximation to the median could be

$$m = \mathrm{med}(\mathrm{med}(x_1, x_2, x_3), \mathrm{med}(x_4, x_5, x_6), \mathrm{med}(x_7, x_8, x_9)) \tag{3}$$

Instead of calculating the median of a sequence of length nine, we can approximate it with calculating four medians, each with length three; three medians of parts of the sequence, and one median of the medians. This strategy is obviously very easy to extend to larger sequence lengths. The following pseudo-code computes the median of medians.

```
double medOfMed(double *x, int length)

{

    int i, k;

    while (length >= 3)
    {
        for (i = 0, k = 0; i < length; i += 3; k++)

            x[k] = max(min(x[i], x[i+1]), min(x[i], x[i+2]), min(x[i+1], x[i+2]));
        length /= 3;
    }

    return x[0];

}
```

The number of medians of length 3 which has to be computed for a sequence of length 3^n is

$$N \quad = \quad \frac{1}{2}\left(3^n - 1\right) \tag{4}$$

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

2.2 Length not equal to 3^n

In the previous section we assumed the sequence length to be a power of 3. In general, this is not the case. So what do we do when the assumption is not fulfilled? As far as we see, there are two strategies. One is simply to pick 3^n samples of the N samples in the sequence, where n is the largest integer such that $3^n < N$. This implies that $N - 3^n$ samples are discarded, which of course is not desirable. The other strategy is to extend the sequence with $3^{n+1} - N$ randomly picked samples. The serious drawback with this approach is that several samples are used more than once. Hence the samples are definitely not independent any longer. See section 2.4 for a comparison between these strategies.

2.3 Deviation estimate

A robust deviation estimate, which we have applied, is based on the median of absolute deviation (MAD) [7]. It is defined as

$$\hat{\sigma} \quad = \quad 1.4826 \cdot \mathrm{MAD}(\mathbf{x}) \quad = \quad 1.4826 \cdot \mathrm{med}(|\mathbf{x} - \mathrm{med}(\mathbf{x})|) \qquad (5)$$

where $\mathbf{x}$ is a vector containing the samples. 1.4826 is a scale factor so that estimates based on samples drawn from Gaussian distributions will be unbiased.

2.4 Characteristics

We will in this section study the characteristics of the median of medians. This will be done by use of Monte Carlo simulations.

The first we want to investigate is the mean and standard deviation of the estimates when there are no contamination and Gaussian distributed samples. In this situation, the "usual" estimates for expectation and standard deviation are known to be optimal. The main question of interest is how large increase in the standard deviation of the estimates does the approximation cause. The mean and standard deviation of the position estimates are shown in Fig 1. The mean and standard deviation of the deviation estimate are shown in Fig 2.

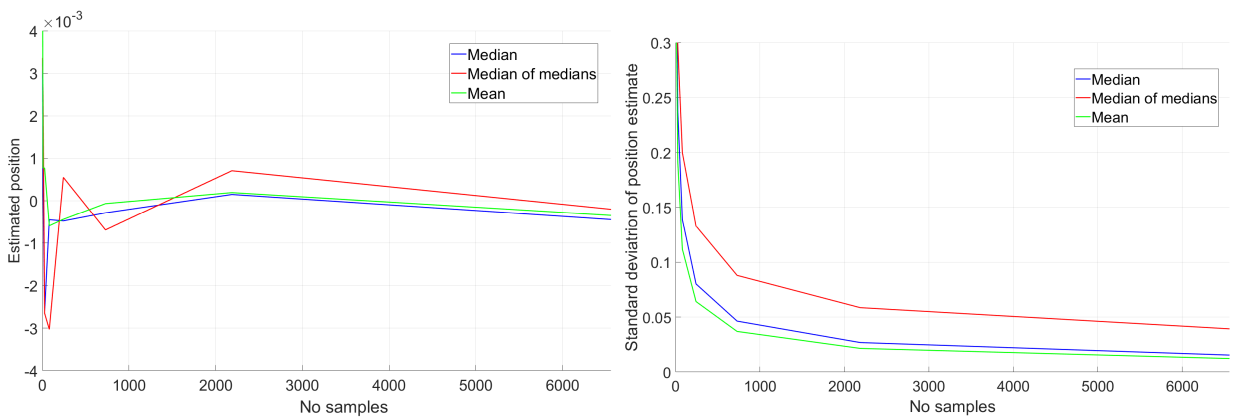

Figure 1. Estimates of mean and standard deviation of position estimates. 10000 replications are used in the simulations.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

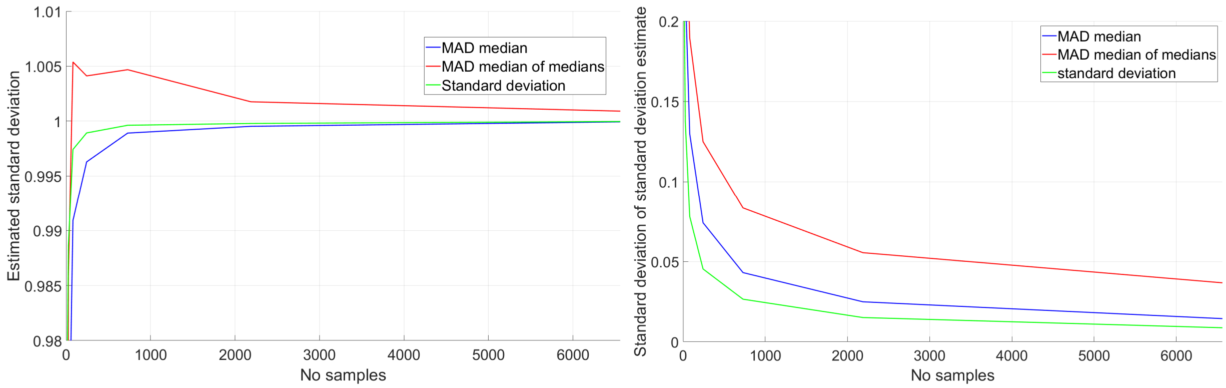

Figure 2. Estimates of mean and standard deviation of deviation estimates. 10000 replications are used in the simulations.

Fig 3 shows the (estimated) probability density of the position estimates for N = 729 samples. The samples are drawn from a standard Gaussian distribution.

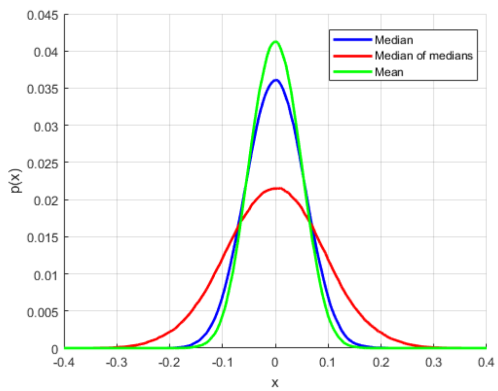

Figure 3. Probability density for the three different position estimates. N = 729. 10000 replications are used in the simulations.

Not surprisingly, the standard deviation of median of medians estimates is higher than the "pure" median-based estimate. It is to be expected that an estimate utilizing all samples in a sequence simultaneously will have a lower standard deviation than an estimate based on subsets of the sequence. The ratio between the two robust estimates is around a factor 2.

An important question concerning the robustness of the median of medians; will it have the same breakdownpoint as the median? In Fig 4, we have added "salt noise" to a standard Gaussian distribution. N = 729 samples are used.

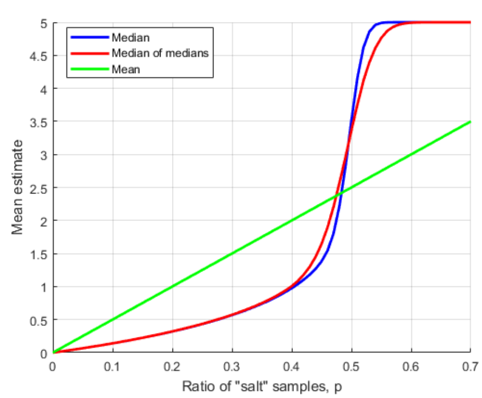

Figure 4. Investigating robustness of the estimators. Position estimate as a function of ratio of "salt samples". N = 729. 10000 replications are used in the simulations.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

Interestingly, we see that both estimators behave rather similarly with respect to robustness. The performance was expected to be a little poorer. This can easily be illustrated for a small sequence. E.g. while four outliers in a sequence of length nine are always handled by the median, one can easily see that they may or may not be handled by the median of medians. It depends on their location in the sequence.

Next, we want to study the standard deviation as a function of contamination. As in the previous examples, all but p*100% are taken from a standard Gaussian distribution. The rest of the samples are drawn from a Gaussian distribution with standard deviation 5.

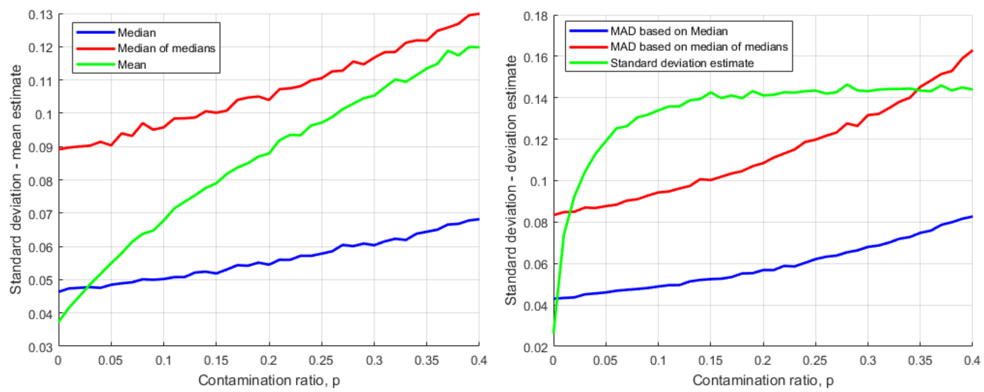

Figure 5. Investigating robustness of the estimators. $N = 729$. Standard deviation estimates of the mean estimates to the left and of deviation estimates to the right. 10000 replications are used in the simulations.

The relative difference between the median and median of median based estimates seems to be relatively constant, and the estimates themselves seem to be little affected of the contamination. These examples also demonstrate the unrobustness of the usual estimates for mean and standard deviation. Especially the standard deviation estimate is highly affected of contamination.

Another issue is the one we addressed in section 2.2; what to do if the sequence length is not a power of three. Does the strategy of extending/filling the sequence with randomly picked samples of the sequence have any negative impact. In Fig 6 we show the results for both strategies as well as the standard deviation of the median/MAD estimate.

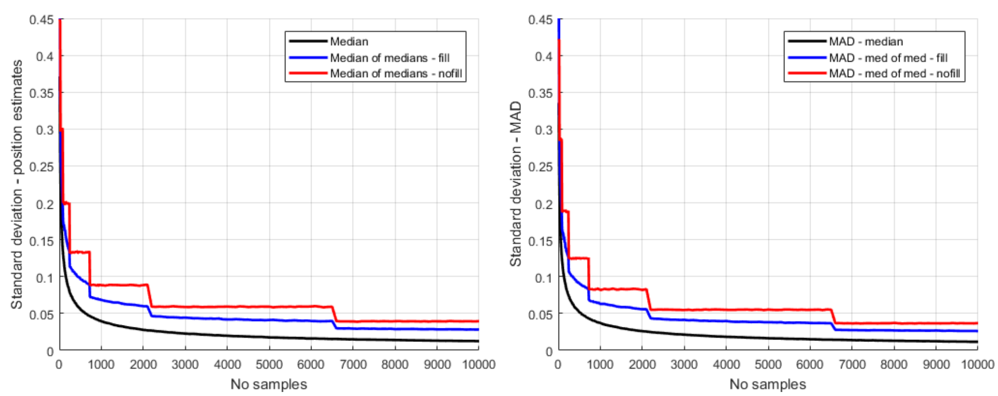

Figure 6. Investigating robustness of the estimators. Position estimate as a function of ratio of “salt samples”. $N = 729$. 10000 replications are used in the simulations.

Finally, we want to find the elapsed time of the median of medians estimate relative to the median-based estimate. The median of medians computation is done using our own C implementation (called from Matlab). For the median, we have used Matlab’s implementation. The reason for this is simply that it is considered to be an efficient implementation. The results are presented in Fig 7.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

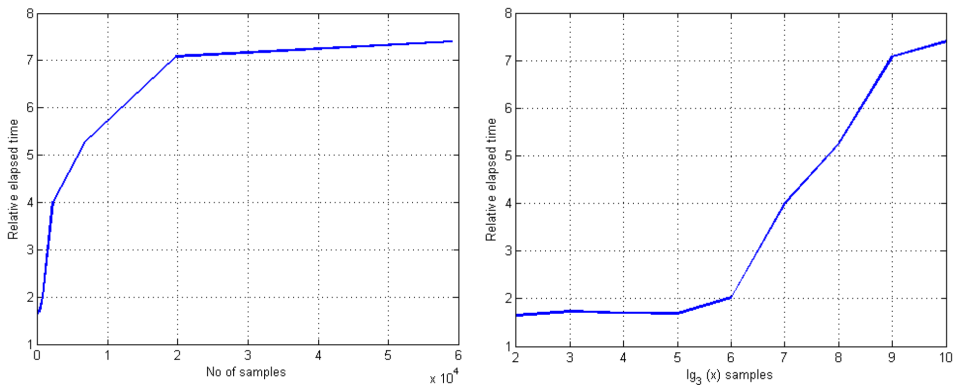

Figure 7. Elapsed time of median relative to median of medians.

We see that for all sample sizes, the median of medians is faster than the median. On the computer we used, for sample sizes larger than 3^6 (729), the difference must be said to be significant.

3. EXAMPLES OF APPLICATIONS TO IMAGE PROCESSING

We will in this section demonstrate two applications of the median of medians. The first example is simply a visualization application. In the second example we have applied the median of medians in a segmentation algorithm.

3.1 Visualization

The cameras used in our test system may be run at a frequency up to 100 Hz. So far, we have used 50 Hz or less. In general each frame is either a 16 bit unsigned int or a float. In both cases they have to be converted to a 8 bit image before they can be displayed. Our approach is based on median and MAD estimates.

Initial experiments revealed that only a small fraction of intensities of randomly chosen pixels is sufficient for performing a good conversion from float or 16 bit unsigned to 8 bit, $1000 < N < 10000$ seems to be enough. Thus, a reasonable number of samples is $N = 6561$. The conversion procedure is simply

1. Draw the intensities from $N = 6561$ samples randomly.
2. Apply the median of medians procedure and calculate (the approximations for) the median and the MAD according to the procedure presented in the last chapter.
3. Let $b_{\min} = \max(\text{med} - \alpha \cdot \text{MAD}, B_{\min})$ and $b_{\max} = \min(\text{med} + \alpha \cdot \text{MAD}, B_{\max})$ denote the intensities to correspond to respectively 0 and 255 in the 8 bit image. $B_{\min}$ and $B_{\max}$ is the minimum and maximum intensity in the image to be converted, and α is a constant. We have used $\alpha = 5$. Now, assign to a pixel i, j in the 8 bit image the intensity $b_{i,j} = \dfrac{255(\max(\min(B_{i,j}, b_{\max}), b_{\min}) - b_{\min})}{b_{\max} - b_{\min}}$.

The following images illustrate the results. Fig 8 shows an IR medium wave image of sun glare. For the left image, min and max intensities are used for the 8 bit conversion, the other are based on the above strategy. Estimates based on median of medians are used for the middle image, and median-based estimates are used for the right image.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

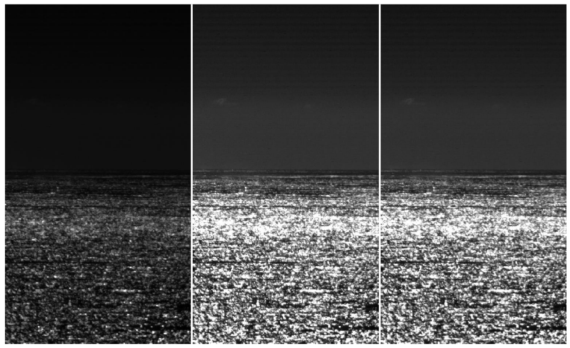

Figure 8. 8 bit conversion for visualization based on min and max intensities (left), median of medians estimates (middle), and median (right). The conversion is based on 6561 randomly drawn samples.

Not surprisingly, the min- max- based conversion performs poor due to the high-intensity sun glare. Concerning the two median-based strategies, there is no practical difference between them with respect to the visualization. However, the median of medians is in this particular case around 5 times faster than the median.

Fig 9-10 show an example where an IR image is contaminated with 10% “salt” pixels. The “salt” pixels are set to an intensity 10 times larger than the maximum intensity in the image. The original image is presented in Fig 9. In Fig 10 the visualization of the contaminated image is shown. The image to the left is based on the median of medians, and the image to the right is based on the median.

Figure 9. The image without any contamination. The visualization is based on min and max intensities. Notice the plume causes the image to be relatively dark.

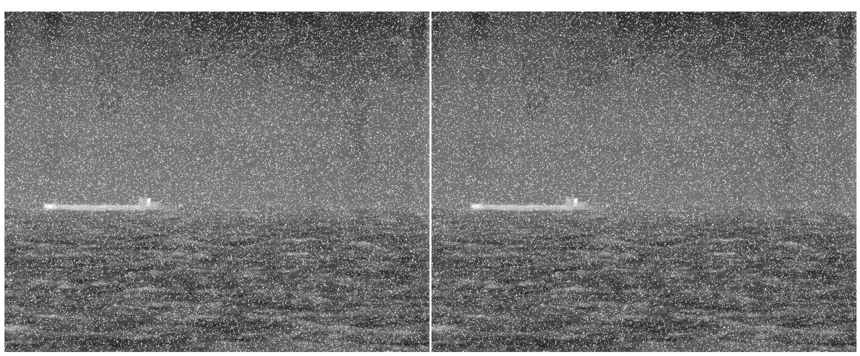

Figure 10. The image contaminated with 10% salt. Visualization based on median of medians to the left, and median to the right.

We see that even though it is no contamination (Fig 9), a small bright spot causes the image to be relatively dark. Furthermore, there are only small, insignificant differences between the two median based visualizations.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

Finally we present an experiment with different number of samples. From left to right, we have used N = 729, 2187, 6561, 19683, 59049, 177147 for the visualization.

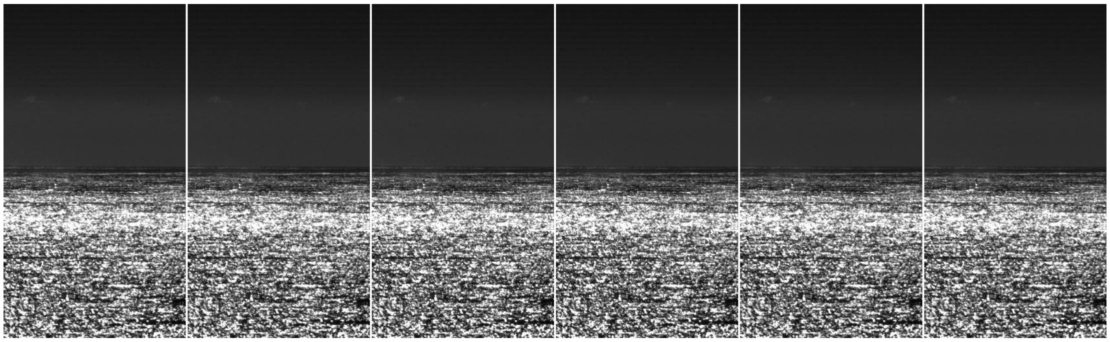

Figure 11. Visualization for different size of the dataset. From left to right, N = 729, 2187, 6561, 1968, 59049, 177147.

This last example demonstrates that the visualization may be based on a small fraction of the image intensities, and still it gives very good results.

3.2 Image segmentation

The idea for this segmentation algorithm is to model the background intensities, and then extract anomalies, i.e. pixels with sufficiently low probability density. The algorithm is heavily inspired by a strategy for generating a classifier based on assumption of mixture of Gaussian distributions [8], i.e. the probability density is given as

$$\mathrm{p}(\mathrm{x}) \quad = \quad \sum_{i=1}^{n} \alpha_i N_d(\mu_i, \sigma_i)(x) \tag{6}$$

where n denotes the number of Gaussian mixtures ("hats") with means μ_i and standard deviations σ_i. α_i are normalizing constants;

$$\sum_{i=1}^{n} \alpha_i = 1$$

The algorithm initially picks a predefined number of randomly chosen samples (i.e. pixels). Based on these samples, the principle of the algorithm is to cluster the intensities of these samples into a predefined number of groups. We assume that the intensities in each group can be modeled with a Gaussian distribution, and thus estimate the parameters (mean, standard deviation, normalizing constant) for each group. Finally, pixels with sufficiently small probability densities are identified.

Given the samples, and the number of groups, the segmentation procedure consists of the following steps:

1. Compute the median of the whole dataset. Divide the dataset into two groups; one above the median, and one below.
2. Apply the MAD estimator, and estimate the standard deviation in each group.
3. Identify the group with the largest standard deviation. Compute the median intensity of the samples in this group, and divide the samples in two subgroups according to the description in step 1.
4. If the number of groups is less than the predefined number: Goto step 2.
5. For each: group: Estimate the parameters μ_i, σ_i, and α_i. The median and MAD are used as estimator for the mean and standard deviation respectively.
6. Segment the image, ie identify the pixels i,j where $b_{i,j} < \tau$. $\tau > 0$ is an anomaly threshold chosen in advance.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

The median of medians is used in the estimation. The number of samples in a group is in general obviously not a power of three. In order to obtain this, we have extended the number of samples in the group as described in section 2.3.

It should be noticed that this algorithm does not give any optimal estimates of μ_i, σ_i, and α_i with respect to maximum likelihood. If that is desirable, an optimization step must be added at the end. In addition, the strategy of extending the number of samples in a group to a power of three should be avoided because it implies that the samples no longer can be assumed to be statistical independent. However, we are not primarily interested in the background distribution; we are merely interested in well performing segmentation. Our experience is that the above procedure gives parameters which reflect the background distribution reasonably well and hence gives good segments.

Fig 12 shows the segmentation of a vessel in a thermal IR image using the algorithm described here. A visual inspection shows that the background intensities are grouped around a small number of modes. Therefore we will expect that it is sufficient with small number of Gaussian components in the segmentation. We have used $n = 3$, and 7 in this experiment. The segmentation is based on the intensities in $N = 6561$ randomly drawn pixels.

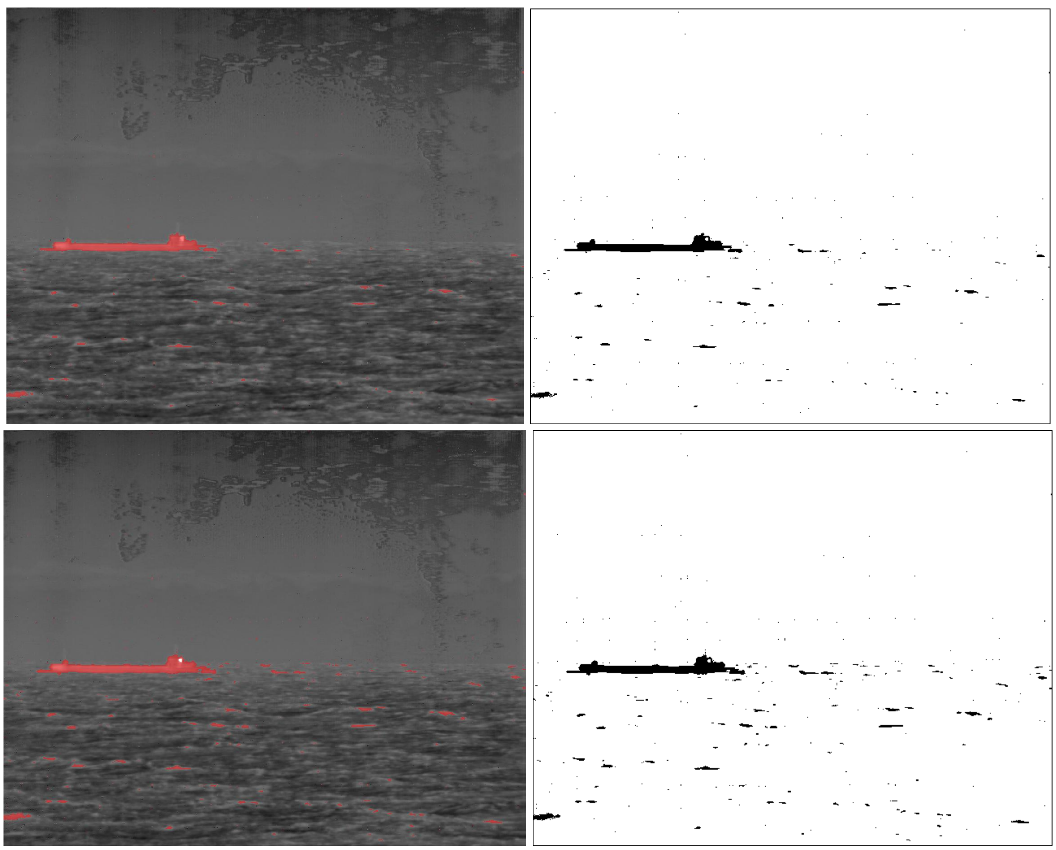

Figure 12. Segmentation with the assumption of background intensities being mixture of Gaussian distributions. Three Gaussian mixtures are used in the upper image, and 7 in the lower image. The segmentation is based on $N = 6561$ samples.

We notice that the segmentation results in the two images in Fig 12 is more or less the same. The vessel and the brightest parts of some waves are segmented. This indicates that the number of Gaussian mixtures is not very critical.

Finally, in Fig 13 we have combined the algorithm in 3.1 for determining outliers with the segmentation algorithm in 3.2. The input image is heavily contaminated (20%) with "salt" noise. A preprocessing identical to the algorithm presented in section 3.1 is applied for removing "salt" samples from the dataset. The method is sufficiently robust to handle this noise so that the following segmentation is unaffected.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

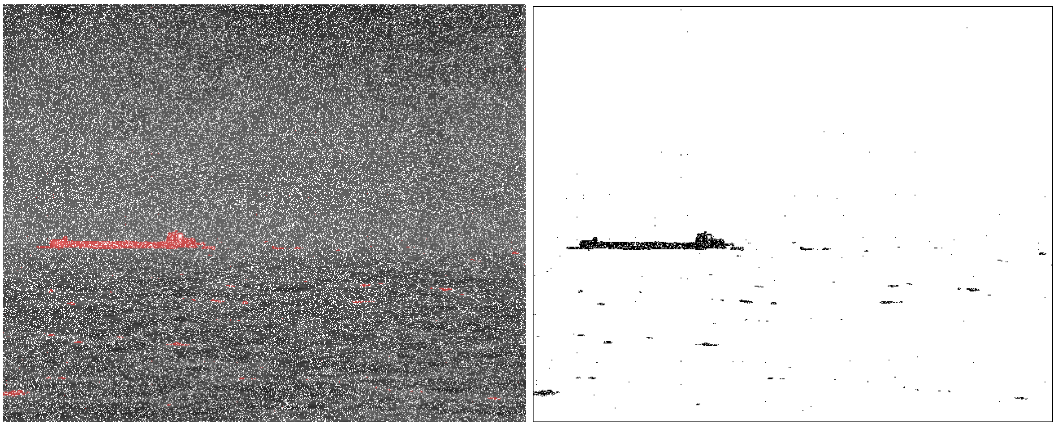

Figure 13. Segmentation with the assumption of background intensities being mixture of Gaussian distributions. The input image is contaminated with 20% “salt” noise. Preprocessing and three Gaussian mixtures are used.

With such high amount of “salt contamination”, the preprocessing is in fact unnecessary. The contamination is so large that it will be modeled in the background estimation, and hence not treated as anomalies. Fig 14 illustrates this. Except for contaminated areas, the vessel is extracted well.

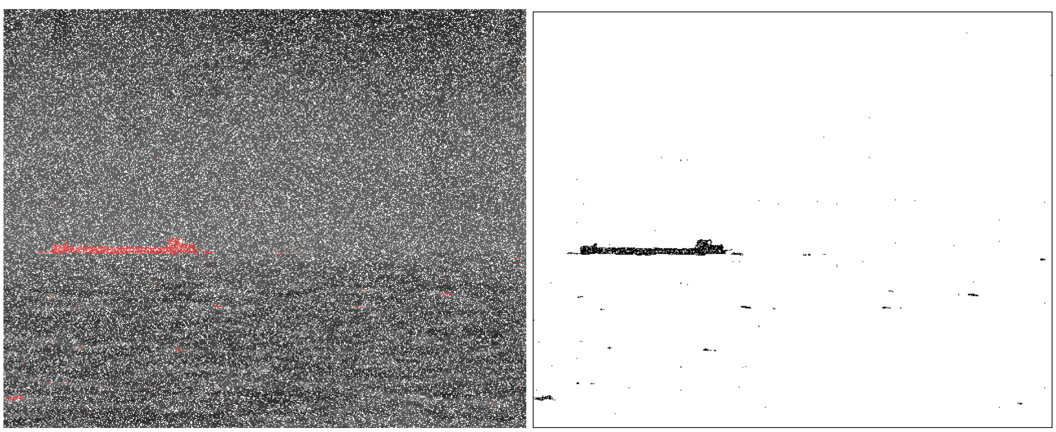

Figure 14. Segmentation with the assumption of background intensities being mixture of Gaussian distributions. The input image is contaminated with 20% “salt” noise. No preprocessing is applied. Three Gaussian mixtures are used.

4. SUMMARY AND CONCLUSION

We have in this paper studied the median of medians in order to see if it applicable in image analysis “as is”. Since it is based on many median computations of small pieces of a dataset instead of using all samples simultaneously, we were not surprised that the standard deviation of the median of medians is higher than of the median. Based on our applications, this increase in uncertainty does not seem to cause any trouble. Whether we apply the median of medians or a “pure” median, doesn’t have much influence of the results. The main drawback with the median of medians is that number of samples has to be a power of 3.In general, this is not fulfilled. In order to obtain a dataset where the number of element is a power of three, we have drawn randomly a number of the input samples and included them for the estimation. This is obviously disputable from a statistical point of view. However, from a practical point of view, we haven’t seen any negative impact. In addition, it must be emphasized that the median of medians is much faster than the median, which is of importance in real time applications.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

We have applied the median of medians in image analysis. Here, we have presented results from visualization and segmentation. So far we have found it to produce fast, robust, and reliable results.

REFERENCES

[1] Breivik, G. M., Løkken, K. H., Brattli, A., Palm, H. C., Haavardsholm, T. V., "An ATR architecture for algorithm development and testing," Proc. SPIE 8744 (2013).

[2] Conover, W. J., [Practical nonparametric statistics], John Wiley and Sons, New York (1980).

[3] Pratt, W. K., Cooper, T. J., Kabir, I., "Pseudomedian filter," Proc. SPIE 534, 34-43 (1985).

[4] Yang, W., Zhang, Z., Lu, X., Li, J., Chen, D., Yang, G., "A novel fast median filter algorithm without sorting," Proc. SPIE 9897 (2016).

[5] Wu, C., C., Chen, D. Y., Zhao, C., Y., "Impulsive noise filter using median- and partition-based operation," Optical Engineereing, 47(11) (2008).

[6] Blum, M., Floyd, R., W., Pratt, V., R., Rivest, R., L., Tarjan, R., E., "Time bounds for selection," Journal of Computer and System Sciences, 7(4) (1973).

[7] Wikipedia, "Median of absolute deviation," <https://en.wikipedia.org/wiki/Median_absolute_deviation> (8 March 2018).

[8] Palm, H. C., "A new method for generating statistical classifiers assuming linear mixtures of Gaussian densities," Proceeding of 12th International Conference on Pattern Recognition, B, 483-486 (1994).

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 11/7/2018
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use