

A New Efficient Approach for Removal of High Density Salt and Pepper Noise in Videos Using Modified Decision Based Unsymmetric Trimmed Median Filter

M. Jyothi

Dept. of ECE, Aurora's Scientific Technological Research Academy, Bandlaguda, India

Abstract

A new and efficient algorithm for high-density salt and pepper noise removal in videos is proposed. The proposed algorithm uses a modified decision based unsymmetric trimmed median filter to remove the high density salt and pepper noise. This filter restores the gray scale, and color images that are highly corrupted by salt and pepper noise. The existing methods are Standard median filter (MF), Decision based algorithm (DBA), Modified decision based algorithm (MDBA), and Progressive switched median filter (PSMF). The proposed method can remove the high density noise, its computational speed is also higher than existing filters, it will take the feedback from noisy or corrupted pixels. The proposed algorithm is tested against Different gray scale and color images and it gives better Peak Signal-to-Noise Ratio (PSNR) and Image Enhancement Factor (IEF). Results of the algorithm is compared with various existing algorithms and it is proved that the new method has better visual appearance and quantitative measures at higher noise densities as high as 90%.

Keywords

Trimmed Median Filter, Salt and Pepper Noise, Unsymmetrical Trimmed Median Filter

I. Introduction

Images and videos are often corrupted by impulse noises during acquisition and transmission. This impulse noise present in an image due to bit errors in transmission or introduced during the signal acquisition stage. There are two types of impulse noise, they are salt and pepper noise and random valued noise. This is based on the noise values. The noise which is easier-to-restore is called salt-and-pepper noise and the noise more difficult random valued is called impulse noise [2]. The noise which sprinkles on the images like white and black dots significantly reduces the visual effects of images. The common filtering algorithms are not effective to remove the noise, especially for heavy noise. To suppress the high density noise and improve image quality, it's very important to keep the margins and details as well as removing noises in images. Among these standard median filter has been established as reliable method to remove the salt and pepper noise without damaging the edge details. However, the major drawback of standard.

Median Filter (MF) is that the filter is effective only at low noise densities [1]. When the noise level is over 50% the edge details of the original image will not be preserved by standard median filter. Adaptive Median Filter (AMF) [2] perform well at low noise densities. But at high noise densities the window size has to be increased which may lead to blurring the image.

Among all the methods for removal of impulse noise, the median filter [3, 4] is used widely because of its effective noise suppression capability and high computational efficiency [7]. non-linear digital filters, based on order statistics are median filters. Median filters are known for their capability to remove impulse noise without damaging the edges. Median filters are known for their capability

to remove impulse noise as well as preserve the edges. The main drawback of a Standard Median Filter (SMF) [4-5] is that it is effective only for low noise densities. At high noise densities, SMFs often exhibit blurring for large window sizes and insufficient noise suppression for small window sizes. However, most of the median filters operate uniformly across the image and thus tend to modify both noise and noise-free pixels. Consequently, the effective removal of impulse often leads to images with blurred and distorted features. Ideally, the filtering should be applied only to corrupted pixels while leaving uncorrupted pixels intact [1]. Applying median filter unconditionally across the entire image as practiced in the conventional schemes would inevitably alter the intensities and remove the signal details of uncorrupted pixels. Therefore, a noise-detection process to discriminate between uncorrupted pixels and the corrupted pixels prior to applying nonlinear filtering is highly desirable. Adaptive Median is a "decision-based" or "switching" filter that first identifies possible noisy pixels and then replaces them using the median filter or its variants, while leaving all other pixels unchanged.

This filter is good at detecting noise even at a high noise level. The adaptive structure of this filter ensures that most of the impulse noises are detected even at a high noise level provided that the window size is large enough. The performance of AMF is good at lower noise density levels, due to the fact that there are only fewer corrupted pixels that are replaced by the median values [6-8]. At higher noise densities, the number of replacements of corrupted pixel increases considerably; increasing window size will provide better noise removal performance; however, the corrupted pixel values and replaced median pixel values are less correlated. As a consequence, the edges are smeared significantly recently. The DBA processes the corrupted image by first detecting the impulse noise. The detection of noisy and noise-free pixels is decided by checking whether the value of a processed pixel element lies between the maximum and minimum values that occur inside the selected window [9-10].

A. Unsymmetric Trimmed Median Filter

In this UTMF, the selected 3*3 window elements are arranged in either increasing or decreasing order. Then the pixel values 0's and 255's in the image (i.e., the pixel values responsible for the salt and pepper noise) are removed from the image. Then the median value of the remaining pixels is taken. This median value is used to replace the noisy pixel. This filter is called trimmed median filter because the pixel values 0's and 255's are removed from the selected window.

II. Review of Sortings

Sorting is the most important operation used to find the median of a window. There are various sorting algorithms such as binary sort, bubble sort, merge sort, quick sort etc. In the proposed algorithm, shear sorting technique is used since it is based on parallel architecture. In practice the parallel architectures help to reduce the number of logic cells required for its implementation.

The illustration of shear sorting is shown in fig. 1- 4. In the odd phases (1, 3, 5) even rows are sorted in descending order and rows are sorted out in ascending order. In the even phases columns are sorted out independently in ascending order.

13	17	12
18	11	19
15	14	16

Fig. 1: Original Matrix before Sorting

12	13	17
19	18	11
14	15	16

Fig. 2: Step 1- Row Sorting

12	13	11
14	15	16
19	18	17

Fig. 3: Step 3- Column Sorting

11	12	13
16	15	14
17	18	19

Fig. 4: Step 4 - Row Sorting

In order to improve the computational efficiency of above Sorting algorithm is modified as follows:

Step 1) A 2-D Window "S" of size 3×3 is selected.

Assume the pixel to be processed is P(X, Y).

xy

Step 2) the pixel values inside the window are sorted, And Pmin, Pmax and Pmed determined as follows.

- The rows of the window are arranged in Ascending order.
- The columns of the window are arranged Ascending order.
- The right diagonal of the window is Arranged in ascending order.

Now the first element of the window is the Minimum value Pmax

the last element of the Window is the Maximum value Pmin and the middle element of the window is the median value Pmed

Step 3). there are 2 cases

Case i): If the selected window contain all the elements as 0's and 255's. Then replace with the mean of the element of window.

Case ii): If the selected window contains not all elements as 0's and 255's. Then eliminate 255's and 0's and find the median value of the remaining elements. Replace with the median value.

Step 4: Repeat steps 1 to 3 until all the pixels in the entire image are processed.

III. Proposed Algorithm

The proposed Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) algorithm processes the corrupted images by first detecting the impulse noise. The processing pixel is checked whether it is noisy or noisy free. That is, if the processing pixel lies between maximum and minimum gray level values then it is noise free pixel, it is left unchanged. If the processing pixel takes the maximum or minimum gray level then it is noisy pixel which is processed by MDBUTMF. Unsymmetric Trimmed Median Filter (MDBUTMF) algorithm removes this drawback at high noise density and gives better Peak Signal-to-Noise Ratio (PSNR) and Image Enhancement Factor (IEF) values than the existing algorithm.

A. Examples for Salt and Pepper Noise for Images

In our proposed algorithm MDBUT filter is first applied to the static images.

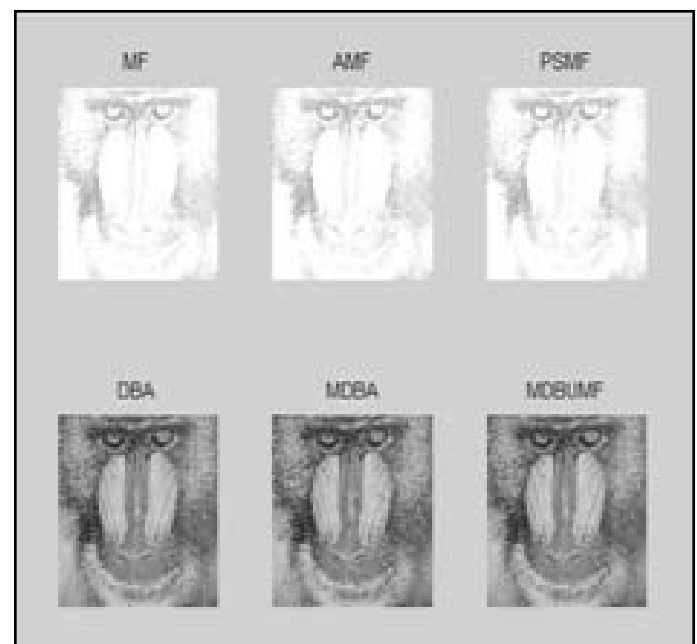


Fig. 5: Results of Different Algorithms for Baboon Image. (a) Output of MF. (b). Output of AMF. (c). Output of PSMF. (d) Output of DBA. (e) Output of MDBA. (f) Output of MDBUTMF.

Those are represented in fig 5 and fig 6. In fig 5 the results for image is obtained from different algorithms like MF, AMF, PSMF, DBA, MDBA and MDBUTMF. Similarly the plots for different algorithms are shown in fig 6.

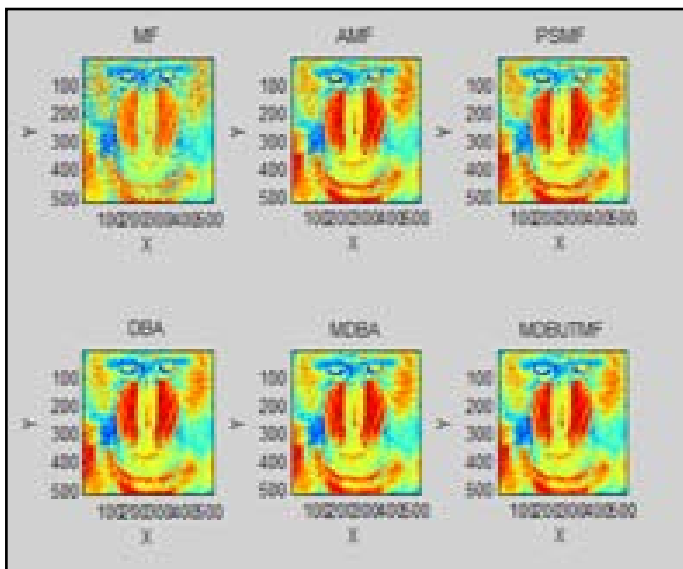


Fig. 6: Plots for Baboon Image (a) Output of MF. (b) Output of AMF. (c) Output of PSMF. (d) Output of DBA. (e) Output of MDBA. (f) Output of MDBUTMF.

IV. Block diagram of MDBUT Filter for Video

Video signals are often contaminated by noise during acquisition and transmission. De-noising of video is highly desirable, which can enhance perceptual quality, increase compression effectiveness, facilitate transmission bandwidth reduction, and improve the accuracy of the subsequent feature extraction and pattern recognition processes.

A corner can be defined as the intersection of two edges. A corner can also be defined as a point for which there is two dominant and different edge directions in a local neighbourhood of the point. An interest point is a point in an image which has a well-defined position and can be robustly detected. This means that an interest point can be a corner but it can also be, for example, an isolated point of local intensity maximum or minimum, line endings, or a point on a curve where the curvature is locally maximal.

This paper presents a much simpler method based on a combination of temporal differencing and image template matching which achieves highly satisfactory tracking performance in the presence of clutter and enables good classification. Hence the use of Kalman filtering or other probabilistic approaches is avoided. Two of the basic methods for target tracking in real time video applications are temporal differencing (DT) [1] and template correlation matching. In the former approach, video frames separated by a constant time t are compared to find regions which have changed. In the latter approach each video image is scanned for the region which best correlates to an image template. Independently, these methods have significant shortcomings.

DT tracking is impossible if there is significant camera motion, unless an appropriate image stabilisation algorithm is employed [5]. It also fails if the target becomes occluded or ceases its motion. Template correlation matching generally requires that the target object's appearance remains constant. The method is generally not robust to changes in object size, orientation or even changing lighting conditions. However, the tracking properties of these two methods are complementary. When the target is stationary, template matching is at its most robust while DT will fail. And when the target is in motion, DT will be successful where template matching will tend to "drift". This is the motivation for combining the two methods. The idea is to use DT to detect moving targets and train the template matching algorithm.

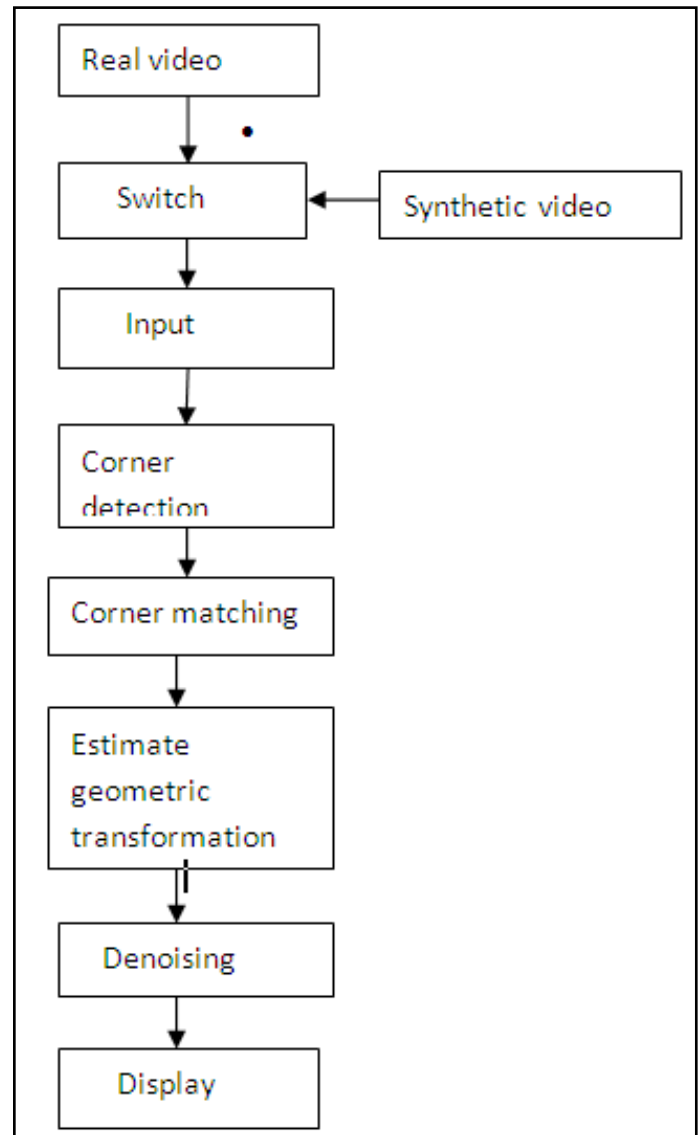


Fig. 7: Block Diagram of Video Denoising

These targets are then tracked using template matching guided by the DT stage. This combination obviates the need for any predictive filtering in the tracking process as the tracking is guided by motion detection. This simple paradigm produces remarkably robust results.

This paper describes a system for robustly tracking targets in a video stream and classifying the targets into "humans" and "vehicles" for an outdoor video surveillance application. Target tracking is based on two main principles;

- Temporal consistency which provides a robust way of classifying moving targets while rejecting background clutter,
- The combination of motion detection with image based template matching which provides a highly robust target tracking scheme. Target classification is based on a simple application of maximum likelihood estimation after computing a simple shape based metric for each target.

A. Corner Detection

Corner detection is an approach used within computer vision systems to extract certain kinds of features and infer the contents of an image. Corner detection is frequently used in motion detection, image registration, video tracking, image mosaicing, panorama stitching, 3D modelling and object

recognition. Corner detection overlaps with the topic of interest point detection.

This is one of the earliest corner detection algorithms and defines a corner to be a point with low self-similarity.[2] The algorithm tests each pixel in the image to see if a corner is present, by considering how similar a patch centered on the pixel is to nearby, largely overlapping patches. The similarity is measured by taking the Sum of Squared Differences (SSD) between the two patches. A lower number indicates more similarity.

If the pixel is in a region of uniform intensity, then the nearby patches will look similar. If the pixel is on an edge, then nearby patches in a direction perpendicular to the edge will look quite different, but nearby patches in a direction parallel to the edge will result only in a small change. If the pixel is on a feature with variation in all directions, then none of the nearby patches will look similar.

The corner strength is defined as the smallest SSD between the patch and its neighbours (horizontal, vertical and on the two diagonals). If this number is locally maximal, then a feature of interest is present.

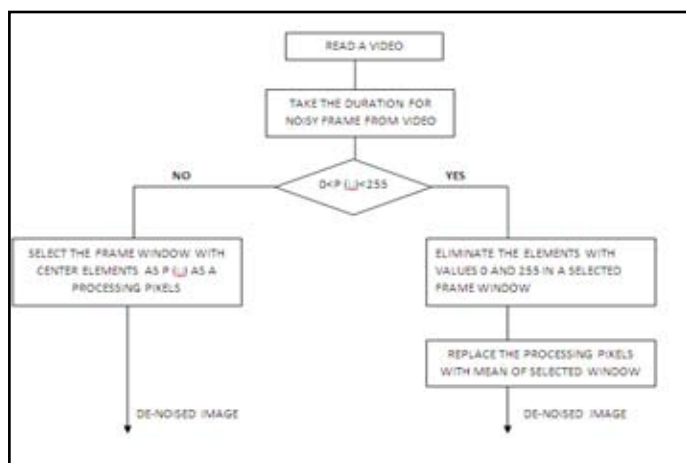
As pointed out by Moravec, one of the main problems with this operator is that it is not isotropic: if an edge is present that is not in the direction of the neighbours, then it will not be detected as an interest point

B. Corner Matching

While corner features are typically found to sub-pixel accuracy, this information is not always used in the matching stage. It is easier to round the corner location to the nearest pixel and use this image pixel value, and the neighbouring pixel values, in calculating the similarity measure.

We compare this approach with the more “correct” approach of using the corner’s sub pixel information and calculating intensity values for the correlation window by bilinear interpolation of the image pixels across a grid centred on the corner feature. This calculation increases the calculation time, but if implemented efficiently does not increase the time by a substantial amount the performance of the corner matcher was evaluated over a series of different images. Test images were of real scenes, some also containing independent motion, and were captured with video cameras or digital stills cameras. The image pairs conformed to the necessary criteria for correlation matching, i.e. similar lighting conditions and little deformation of corners between frames.

1. Algorithm of Flow Chart



V. Experimental Results

The performance of the proposed algorithm is tested for various levels of noise corruption and compared with standard filters namely Standard Median Filter (SMF), Adaptive Median Filter (AMF) and Modified Decision Based Algorithm (MDBA). Each time the test image is corrupted by salt and pepper noise of different density ranging from 10 to 90 with an increment of 10 and it will be applied to various filters. In addition to the visual quality, the performance of the developed algorithm and other standard algorithms are quantitatively measured by the following parameters such as Peak Signal-to-Noise Ratio (PSNR), Mean square error (MSE) and Image Enhancement Factor (IEF). All the filters are implemented in MATLAB 7.5 mon a PC equipped with 2.4 GHz CPU and 1 GB RAM memory for the evaluation of computation time of all algorithms.

A. For Videos



Fig. 8: Noisy Video



Fig. 9: Noise Removed from the Above Video by MDBUT Filter

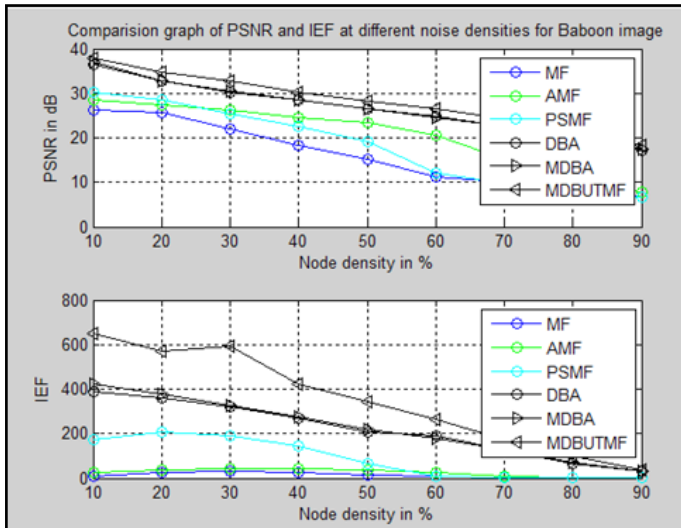
VI. Conclusion

An efficient non-linear algorithm to remove high-density salt and pepper noise is proposed. The modified sheer sorting architecture reduces the computational time required for finding the median. This increases the efficiency of the system. The algorithm removes noise even at higher noise densities and preserves the edges and fine details. The performance of the algorithm is better when compared to the other architecture of this type.

VII. Results and Discussion

The developed algorithms are tested using 512X512, 8bits/pixel image Lena (Gray), Parrot (color), Barbara (color). The

performance of the Proposed algorithm is tested for various levels of noise corruption and compared with standard filters namely Standard Median Filter (SMF), Adaptive Median Filter (AMF) and Decision Based Algorithm (DBA). Each time the test image is corrupted by salt and pepper noise of different density ranging from 10 to 90 with an increment of 10 and it will be applied to various filters. In addition to the visual quality, the performance of The developed algorithm and other standard algorithms are quantitatively measured by the following parameters such as Peak Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE) and Image Enhancement Factor (IEF). All the filters are implemented in MATLAB 7.5 on a PC equipped with 2.4 GHz CPU and 1 GB RAM memory for the evaluation of computation time of all algorithms.



- [10] Srinivasan K. S., Ebenezer D., "A New Fast and Efficient Decision-Based Algorithm for Removal of High Density Impulse Noises", IEEE signal processing letters, Vol. 14, No. 3, pp.189-192, 2007.

References

- [1] W.K.Pratt, "Digital Image Processing", (New York: Wiley 1978).
- [2] I. Pitas, A.N. Venetsanopoulos, "Non-linear Digital Filter Principles and Applications", (Boston: Kluwer Academic publishers, 1990).
- [3] J. Astola, P. Kuosmanen, "Fundamentals of NonLinear Digital Filtering", Boca Raton, CRC, 1997.
- [4] N.C.Gallagher, Jr., G.L.Wise, "A Theoretical Analysis of the Properties of Median Filters", IEEE Trans. Acoust., Speech and Signal Processing, Vol. ASSP-29, pp.1136-1141, Dec.1981.
- [5] T.A.Nodes, N.C.Gallagher, "Median Filters: Some Modifications and their properties", IEEE Trans. Acoust., Speech and Signal Processing, Vol. ASSP-30, pp.739-746, April 1987.
- [6] S.Manikandan, O.Uma Maheswari, D.Ebenezer, "An Adaptive Recursive Weighted Median Filter with Improved Performance in Impulsive Noisy Environment", WSEAS Transactions on Electronics, Issue 3, Vol.1, July 2004.
- [7] Ho-Ming Lin, "Median Filters with Adaptive Length", IEEE Transactions on Circuits and Systems, Vol. 35, No. 6, June 1988.
- [8] T.S. Huang, G.J. Yang, G.Y. Tang, "Fast two dimensional Median Filtering algorithm", IEEE Transactions on Acoustics, Speech and Signal Processing, 1(1995), pp. 499-502.
- [9] Pok, G. Jyh-Charn Liu, "Decision based median filter improved by predictions", ICIP 99(1999) proceedings, Vol. 2 pp. 410-413