IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197 Volume 1, Issue 6 (Mar. – Apr. 2013), PP 37-41
www.iosrjournals.org

Removal of High Density Impulse Noise Using Cloud Model Filter

Easwara.M IV Sem, M.Tech, Guide: Satish Babu. J Asst. Professor
SJCIT, Chickballapur

Abstract: *The fact that makes image denoising a difficult task is uncertainties in the impulse noise. The most knowledge in dayflies is uncertainty and erratic, unfortunately it is similar to impulse noise. The mathematic implements for handling uncertainty mostly are probability theory and fuzzy mathematics. That means, among the uncertainties involved in impulse noise, the randomness and the fuzziness are the two most important features. In this paper we use a detail-preserving filter based on the Cloud Model (CM) to remove severe impulse noise. CM is an uncertain conversion model, between qualitative and quantitative description that integrates the concept of randomness and fuzziness. The normal random number generation method in normal cloud generator algorithm overcomes the insufficiency of common method to generate random numbers. It can produce random numbers which can be predictable and replicated, and this random numbers present to be a random sequence as a whole. The digital features of the normal cloud characterized by three values with the expectation Ex, entropy En and Hyper entropy He and are good enough to represent a normal cloud. First, an uncertainty-based detector, normal cloud generator, identifies the pixels corrupted by impulse noise. Then, the identified noise pixels are replaced by a fuzzy mean estimation of the processed noise free pixels within the detection window. Compared with the traditional switching filters, the CM filter makes a great improvement in image denoising. Especially, at high density noise level. Thus, the cloud model filter can remove severe impulse noise while preserving the image details.*
Key words—*uncertainty, fuzzy median, normal cloud generator, cloud drops.*

I. Introduction

Digital images are often corrupted by impulsive noise during data acquisition, transmission, and processing**.** The main sources of noise are malfunctioning pixel sensors, faulty memory units, imperfection encountered in transmission channels, external disturbances in a transmission channels, electro-magnetic interferences, timing errors in ADC, etc. The noise may seriously affect the performance of image processing techniques. Hence, an efficient de-noising technique becomes a very important issue in image processing. Impulse noise produces small dark and bright spots on an image. Grasping the noise characteristics is helpful to remove the noise. Noise reductions are basically classified into two types: linear and non-linear techniques. Mean filters are linear filters. Their estimation alter the good pixels, thus produce image blurring. Non-linear noise reduction is a two step process: 1) noise detection and 2) noise replacement. Median filters and its variants [3], [4], [5], [8] have an effective noise suppression and high computational efficiency at low noise density (< 50%), but they fail to account local features such as thin lines, edges at high noise density. Also, they think about only the randomness. The randomness mainly shows in two aspects: 1) the pixels are randomly corrupted by the noise and 2) the noise pixels are randomly set to the maximum or minimum value.
Some decision based filters [7] are good even at high noise density (80%), however, many jagged edges appear in the restored images. It requires more processing time since it uses 21 x 21 window. Sorting fixed window filter [6] uses the median values or the left neighborhood values to replace the noise pixels. This filter creates mainly stripe regions, because it often replaces the corrupted pixel by the left neighborhood pixel. It smears the image details seriously and also sharply decreases the qualities of restored images.

This reveals that the early de-noising techniques fail to understand the uncertainties of noise completely. The better solution is that the pixels those identified as good ones would remain unchanged, while those identified as noisy are replaced with an appropriate estimation. Since, most knowledge in dayflies is uncertainty and erratic, unfortunately it is similar to impulse noise. Among the uncertainties involved in impulse noise, the randomness and the fuzziness are the two most important features. On the other hand, the fuzziness focuses on the pixels with the extreme values whether they belong to the noise or not. Not all of the pixels, which are set to the extreme values, will be the noise pixels [1]. The mathematic implements for handling uncertainty mostly are probability theory and fuzzy mathematics. In fact, CM is an uncertain conversion model, between qualitative and quantitative description that integrates the concept of randomness from probability theory and fuzziness from fuzzy set theory [2]. To represent the uncertainties better and resolve the afore mentioned problems, this paper presents a novel effective filter based on the CM for impulse noise removal. It is compared with the traditional switching filters, the CM filter has the better performance in image de-noising across a wide range of noise levels with good detail preservation. The outline of the paper is as follows. The cloud model and noise detection-estimation illustrations are reviewed in Section II and III. Simulation results and conclusions are presented in Sections IV and V, respectively.

II. Cloud Model And Its Parameters

Cloud is model described by linguistic values for representation of uncertain relationships between a specific qualitative concept and its quantitative expression. Cloud integrates the concept of randomness and fuzziness. The normal random number generation method in normal cloud generator overcomes the insufficiency of common method to generate random numbers. It can produce random numbers which can be predictable and replicated, and this random numbers present to be a random sequence as a whole. CM is defined as :

Let U be a universal set expressed by exact numbers, and C be the qualitative concept associated with U. If number $x \in$ U exists, which is the random realization of concept, and the certainty degree of x for C, i.e., $\mu(x) \in [0,1]$, is a random value with stabilization tendency, $\mu : U \rightarrow [0,1] \ \forall x \in U \rightarrow \mu(x)$ (1)

then the distribution of x on U is called the cloud, and each x is called a drop. The cloud can be characterized by three parameters, the expected value Ex, entropy En, and hyper entropy He. Ex is the expectation of the cloud drops' distribution in the domain, it points out which drops can best represent the concept and reflects the distinguished feature of the concept. En is the uncertainty measurement of the qualitative concept, which is determined by both the randomness and the fuzziness of the concept. It represents the value region in which the drop is acceptable by the concept, while reflecting the correlation of the randomness and the fuzziness of the concept [1]. The greater the En is, the range of values represented by the concept is the greater, the more vague the concept is. Hyper entropy He, is entropy En of entropy, reflecting the degree of dispersion of the cloud droplets [2]. En is derived from Mean Deviation about the mean for n independent random variables x_i. $i=1,2,3,\ldots,n,$ with mean X. The cloud employs its three parameters to represent the qualitative concept as shown in Fig. 1.

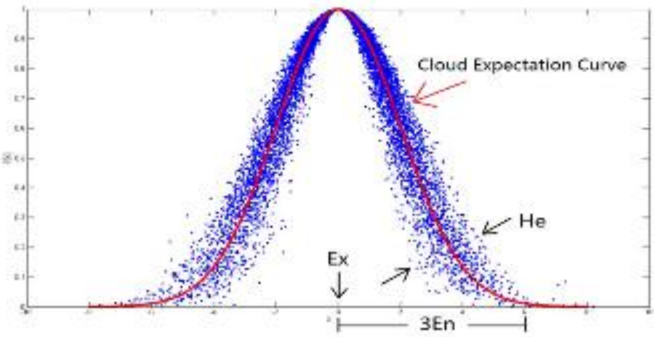

Fig 1. Illustration of three digital cloud parameters

The distribution of pixel values on domain is called cloud and each pixel in the domain is called cloud drop. According to the normal cloud generator [2], the certainty degree of each drop is a probability distribution rather than a fixed value. It means that the certainty degree of each drop is a random value in a dynamic range. If He of the cloud is 0, then the certainty degree of each drop will change to be a fixed value. The fixed value is the expectation value of the certainty degree. In fact, the value is also the unbiased estimation for the average value of the certainty degrees in the range. All the drops and their expectations of certainty degrees can compose a curve, and the curve is the Cloud Expectation Curve (CEC) [1]. All drops located within [Ex+3En, Ex-3En] take up to 99.99% of the whole quantity and contribute 99.74% to the concept. Thus, the drops are located out of domain [Ex+3En, Ex-3En], and then, their contributions to the concept can be neglected. The certainty degree of each pixel is calculated through the CEC, given by,

$$\mu=\exp(-(x_i-\mathrm{Ex})^2/2\mathrm{En}^2) \quad (2)$$

Where x_i is cloud drops, Ex is their expectation value and En is entropy.

Noise Model:

Due to faulty switching devices, pixels are randomly corrupted by the two extreme values. Thus, the noise pixels are usually set to the maximum and minimum values in the dynamic range. Let $x(i,j)$ for $(i,j) \in \mathcal{A}$ be the gray value of image X at pixel location (i,j) and $[S_{min}, S_{max}]$ be the dynamic range of X, i.e., $S_{min} \leq x(i,j) \leq S_{max}$ for all (i,j). Denote **y** by a noisy image. In the salt and pepper impulse noise model, the observed gray level at location (i,j) is

$$Y(i,j) = \begin{cases} S_{min}, & \text{with probability } p \ ; \text{ for } (i,j) = 0 \\ S_{max}, & \text{with probability } q \ ; \text{ for } (i,j) = 255 \\ x\,(i,j) & \text{with probability } 1-(p+q) \ ; \text{ for } 0 < (i,j) < 255 \end{cases} \quad \text{where } p+q = \text{noise level} \tag{3}$$

III. Noise Detection And Estimation

We consider all the pixels in the window as a set and use CM to represent it. Let each pixel of image X_{MxN} be a cloud drop and input them into Backward Cloud Generator (BCG). It generates three parameters of the cloud C. These Ex, En and He, will be inputs to Forward Cloud Generator (FCG), which generates cloud drops (random numbers). It is as shown in fig.2. This is basis for uncertainty reasoning.

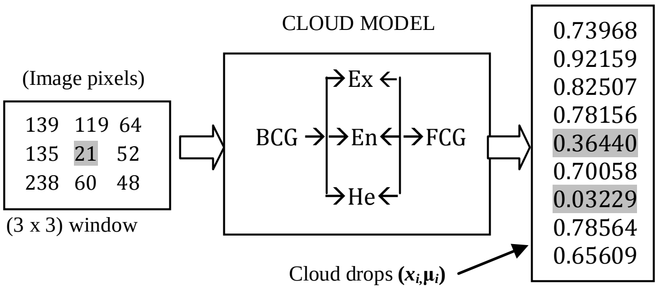

Fig 2. Calculated the cloud drop (x_i,μ_i) for a (3 x 3) window using normal cloud generator.

It is seen that in the fig. 2 at the output of FCG, the certainty degrees of the noise pixels (shaded figures) are far less than that of the uncorrupted pixels. Also, the noise pixels are usually distributed on the both sides of the cloud, and the uncorrupted pixels are located near the central region of the cloud. The CM uses all the pixels in the window to detect the noise pixel and the certainty degrees of each pixels in the proposed detector are "soft" values between [0,1]. Hence, cloud model filter is capable to overcome the drawbacks of existing filters. According to "the 3En rule," the drops out of domain [Ex±3En] can be neglected, which is helpful to identify the noise. Based on this idea, this section presents a novel impulse noise detector using only a fixed 3x3 window, and its details shown as follows. Let w (i,j) be a window of size (2N+1) x (2N+1) centered at location (i,j). $w\,(i,j) = x\,(i+p, j+q)\ \forall\ p,q \in (-N, N)$ where N = 1.

Step 1: Impose 3x3 window with N = 1 for $w(i,j)^{(2N+1)\times(2N+1)}$ on image X.

Step 2: Compute expectation Ex:

$$Ex = 1/n \sum_{x(i+p,\, j+q)\ \in\ w_{(i,j)3x3}} x\,(i+p, j+q) \tag{4}$$

Step 3: Compute entropy En

$$En = \sqrt{\frac{\pi}{2}} * \frac{1}{n} \sum_{x(i+p,\, j+q)\ \in\ w_{(i,j)3x3}} |\ x\,(i+p, j+q) - x(i,j)\ | \tag{5}$$

Step 4: Calculate w_{min} and w_{max} in $w(i,j)^{(3x3)}$ which are extreme operations to recover the smallest and the largest of two values, respectively. i.e., w_{max} = min ($S_{max,}$ Ex+3En) and

w_{min} = max ($S_{mim,}$ Ex -3En)

Step 5: If $w_{min} < x(i,j) < w_{max}$, x(i,j) is uncorrupted pixel (it has to remain unchanged). Then, $y(i,j) = x(i,j)$. Otherwise, x(i,j) is a corrupted pixel. Go to step 6.

Step 6: Noisy pixel x(i,j) is replaced by weighted mean of already processed previous four uncorrepted pixel values , X_{nbp}, within the $w(i,j)^{(3x3)}$. i.e., $X_{nbp} = [x(i-1,j-1), x(i,j-1), x(i+1,j+1), x(i-1,j),]$

Step 7: Calculate the weights for X_{nbp} within the $w(i,j)^{(3x3)}$ $\mu_{nbp} = \exp[-(X_{nbp} - Ex)^2 / 2En^2]$. (6)

Step 8: Then, calculate the weighted mean, $y(i,j) = 1/m \sum X_{nbp} * \mu_{nbp}$. (7)

The CM filter replaces the noise pixel by using the weighted mean of the neighborhood pixels, and their weights are the certainty degrees of them. For understanding of the above steps, a 3x3 windowed sub-image, shown in fig. 2, as an example, is illustrated as follows: Assume that the central pixel **21** lies at an edge of the image. Since, the certainty degrees of each pixels in the proposed detector are "soft" values between [0,1], the noisy pixel values will be replaced by an appropriate pixel values.

*The pixel under test is x(i,j) = 21; Expectation, Ex = 97; Entropy, En = 53.7; S_{min} = 21 ; S_{max} = 238.
*Computing w_{max} = min (238, 258.4) = 238; w_{min} = max (21, -63.8) = 21.
*If $w_{min} < x(i,j) < w_{max}$; 21 < 21< 238 ; hence, x(i,j) = 21, is a noisy pixel.
*To replace the noisy pixel, x(i,j), collect already processed previous four good pixel values, X_{nbp} = [139, 119, 64, 135] and compute the weights of X_{nbp} using equation (5).
i.e., μ_{nbp} = [0.73968, 0.92159, 0.82507, 0.78156].
Then, noisy pixel value 21 is replaced by equation (6).
i.e., y(i,j) = 92, is an appropriate pixel value, this provides higher correlation between the corrupted pixel and neighborhood pixel. Higher correlation gives rise to better edge preservation. to preserve the edge of the given image [6].
The CM detector has three major differences with the traditional detectors. First, the proposed detector uses all the pixels in the window to detect the pixel. Second, the traditional filters usually discard the extreme values in the detection window. However, not all of the pixels that are set to the maximum or minimum values will be the noise pixels, the CM does not. Third, the proposed detector identifies if the detected pixel is a noise pixel or not and replacel the noise candidate in $w(i,j)^{(3x3)}$ at the same time. It is a pretreatment to increase the computational efficiency of the post-processing, because those pixels with lower contribution degrees play a small role in the post-filtering.

IV. Simulation Results

An 8-bit gray scale image Lena of 512 x 512 size, has been used to test the performance of the CM filter with dynamic range of values. Image will be corrupted by salt-and-pepper noise at different noise densities, 10% to 80%. The restoration performances are quantitatively measured by peak signal-to-noise ratio (PSNR),

$$\text{PSNR}= 20 \log_{10}\frac{255}{MSE} \text{ dB} \tag{8}$$

$$\text{MSE} = \frac{1}{MN}\sum_{ij}^{mn} (y(i,j)-x(i,j))^2 \tag{9}$$

Where y(i,j) and x(i,j) denote the pixel values of the restored image and the original image, respectively.

The experiment aims to study the detail-preserving abilities of the filter when the images are affected by a severe noise. In this case, since, an effective result would be obtained the window size of the CM filter is limited to 3x3, this causes to increase the computational efficiency. It removes a pixel immediately after the pixel has been identified as a corrupted candidate. Therefore, in the CM filter, the noise detector and the postfilter (replacing noisy pixel) use the same windows.

For comparison, the boundary discriminative noise detection (BDND) filter [7], and the fast median (FM) filter [6] are used. When the noise level is lower than 60%, the performance of the CM filter is similar to the BDND filter, at high noise densities the CM filter proves that having good detail preserving ability. For the FM filter, the decrease in the PSNR is more pronounced than the others and it creates many stripe regions, because it often replaces the corrupted pixel by the left neighborhood pixel. However, the CM filter is a switching fuzzy mean filter, which restores the images and preserves the details well without any jagged edges. To study the detail-preserving abilities of two filters CM and BDND filter, they are tested by the noise image with the noise level 90% (see Figs. 3). Although the BDND filter restores the images without noise, however, many jagged edges appear in the image details at the high noise levels, particularly in the Lena hat region.

(a) (b)
(a) Lena with the noise level of 90% (b) Original image.


(b) CM filter (26.85 dB). d) BDND filter (25.45 dB)

Fig. 3. Restoration results of different filters.

All these are because theBDND filter is a switching median filter, which makes the filter often smear the image details. Obviously, in those regions, the images restored by the CM filter basically keep the same gray levels with the original images. Table-I lists comparison of restored images in PSNR (in decibels)

Table-I (comparison of restored images in PSNR) (in dB)

.	Noise Density(%)				
Filter	**10**	**30**	**50**	**70**	**80**
CM	42.23	37.13	33.26	30.61	28.36
BDND	41.91	35.95	32.62	29.53	27.08
FM	41.64	34.01	29.83	25.82	23.08

To make a reliable comparison, each filter is run 20 times in the same running environment; it is MATLAB 7.0.1 on a personal computer equipped with the 3.2-GHz CPU and 2 GB RAM. Table-II lists the average runtimes in milliseconds for each filter operating on the Lena.

Table-II (average runtimes in milliseconds)

	Noise Density(%)				
Filter	**10**	**30**	**50**	**70**	**80**
CM	440	439	439	440	441
BDND	12324	11390	12074	11509	11341
FM	186	187	186	187	187

V. Conclusion

There are three important aspects in image denoising: First, the accuracy of the noise detection, it will directly influence the results of the image denoising. Second, the computational efficiency, for the real-time work, the filters with lower computational efficiency may not obtain the satisfactory results. Finally, large uncertainties exist in the noise. Thus, understanding the uncertainties can completely help to improve the qualities of the restored images. In this paper, a novel filter with uncertainty for impulse noise removal has been proposed. It represents the uncertainties of the noise perfectly by using the CM, which is helpful in detecting and removing the noise. In addition, the proposed filter identifies the noise pixel without needing to sort the pixel gray values, using 3x3 window, which immensely increases the computational efficiency in noise detection. No matter whether, in noise detection, the image details preservation or computational complexity, the CM filter makes a great improvement and has the higher performances. In sum, the CM filter is a moderately fast denoising filter with good detail preservation.

References:

[1]. Zhe Zhou, "Cognition and Removal of Impulse Noise With Uncertainty", IEEE *Transactions on image processing*, vol. 21, no. 7, pp. 3157-3167, July 2012.

[2]. Zhaohong Wang, Cloud Theory and Fractal Application in Virtual Plants *I.J. Intelligent Systems and Applications*, 2, 17-23 , MECS (http://www.mecs- press.org/) March 2011.

[3]. S.-J. Ko and S.-J. Lee, "Center weighted median filters and their applications to image enhancement," *IEEE Trans. Circuits.*,vol.38, no. 9, pp. 984–993, Sep. 1991.

[4]. H. Hwang and R. A. Haddad, "Adaptive median filters: New algorithms and results," *IEEE Trans. ImageProcess.*, vol. 4, no. 4, pp. 499–502, Apr. 1995.

[5]. Z. Wang and D. Zhang, "Progressive switching median filter for the removal of impulse noise from highly corrupted images," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 46, no. 1, pp. 78–80,Jan. 1999.

[6]. K. S. Srinivasan and D. Ebenezer, "A new fast and efficient decisionbased algorithm for removal of high- density impulse noises," *IEEE Signal Process. Letters*, vol. 14, no. 3, pp. 189–192, Mar. 2007.

[7]. P.-E. Ng and K.-K. Ma, "A switching median filter with boundary discriminative noise detection for extremely corrupted images," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp 1506–1516, Jun. 2006.

[8]. T. Chen, K.-K. Ma, and L.-H. Chen, "Tri-state median filter for image denoising," *IEEE Trans. Image Process.*, vol. 8, no. 12, pp. 1834–1838, Dec. 1999.