This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2805289, IEEE Transactions on Fuzzy Systems

Adaptive Type-2 Fuzzy Approach for Filtering Salt and Pepper Noise in Grayscale Images

Vikas Singh, *Student Member, IEEE,* Raghav Dev, Narendra K. Dhar, *Student Member, IEEE,* Pooja Agrawal, and Nishchal K. Verma, *Senior Member, IEEE*

***Abstract*—This paper proposes a novel adaptive Type-2 fuzzy filter for removing salt and pepper noise from the images. The filter removes noise in two steps. In the first step, the pixels are categorized as good or bad based on their primary membership function (MF) values in the respective filter window. In this paper, two approaches have been proposed for finding threshold between good or bad pixels by designing primary MFs. a) MFs with distinct Means and same Variance and b) MFs with distinct Means and distinct Variances. The primary MFs of the Type-2 fuzzy set is chosen as Gaussian membership functions (GMFs). Whereas, in the second step, the pixels categorized as bad are denoised. For denoising, a novel Type-1 fuzzy approach based on a weighted mean of good pixels is presented in the paper. The proposed filter is validated for several standard images with the noise level as low as 20% to as high as 99%. The results show that the proposed filter performs better in terms of peak signal-noise-ratio (PSNR) values compared to other state-of-the-art algorithms.**

***Index Terms*—Type-1 fuzzy set, Type-2 fuzzy set, Salt and pepper noise, *Mean of k-middle*, PSNR.**

I. Introduction

REMOVING noise from images is an essential task since good quality images are required in various applications such as medical imaging, satellite imaging, recognition, etc. There are several types of noises which can degrade the quality of images. One such type is salt and pepper (SAP) noise. This noise can be represented as randomly occurring white (1) and black (0) pixels in the image. The main sources of this noise are electrical conditions, light intensity, imperfection in imaging sensors, transmission errors, etc. [1], [2].

The state-of-the-art suggests that various approaches have been proposed for removal of SAP noise. Median filter [3] and adaptive median filter [4] are popularly used for SAP noise removal. In these approaches, noisy pixel intensity is replaced by the median of intensities of neighborhood pixels. Although these filters are efficient to remove noise but fails to preserve details of the image due to blurring at the edges. The weighted mean, weighted fuzzy mean, adaptive fuzzy mean and iterative fuzzy filters are also used to reduce SAP noise [5]–[8]. In all these methods, the problem lies with weighing of good pixels which may lead to loss of actual image details to a certain extent. Ahmed *et al.* [9] have proposed an iterative adaptive fuzzy filter for removal of high-density SAP noise. The drawback of this approach seems to be assignment of weight to good pixels in window during denoising using inverse distance weighting function. Therefore, this method fails to preserve the image details. The other problem of this method is use of many heuristic parameters such as ϵ, K_1, and K_2 which are not consistent to give best result for different noise levels.

To overcome these problems others have proposed fuzzy filter based on Type-2 fuzzy set [10]–[12]. Liang and Mendel [13] proposed Type-2 adaptive filter using an unnormalized Type-2 Takagi Sugeno Kang (TSK) fuzzy logic system (FLS) for the application of equalization of a nonlinear time-varying channel. John *et al.* [14] have applied neuro-fuzzy clustering techniques for classifying images where an image is represented by Type-2 fuzzy set. Yıldırım *et al.* [15] have proposed a Type-2 fuzzy filter for suppressing noise in the image while at the same time preserving thin lines, edges, texture, and other useful features within the image. In [16], [17], Type-2 fuzzy filter for edge, corner detector and noise reduction from the color images have been proposed. The drawback of these methods are formation of big fuzzy rule base (FRB) matrices and use of rigorous fuzzification and defuzzification, which increase computation time and complexity of the filter. The filter window is also not adaptive with respect to noise level.

In this paper, we propose an adaptive Type-2 fuzzy filter with a combination of Type-1 fuzzy set to eliminate the problem of FRB matrix formation, fuzzification and defuzzification. The proposed filter consists of two steps. In the first step, pixels are categorized as good or bad. In the second step, the pixel categorized as bad in step-1 is denoised. For step-1, two approaches based on adaptive threshold using primary MF values of Type-2 fuzzy logic are developed. Either of the two approaches may be used in the first step. In the second step, for denoising the bad pixels, good pixels are weighted in their respective filter window. To assign proper weight to good pixel a novel Type-1 fuzzy logic based approach is proposed. The proposed filter is simple and very effective to remove SAP noise. This filter also preserves image features such as edges, corners, etc. Moreover, it doesn't require any parameter tuning. The proposed filter is validated on several standard grayscale images and provides improved peak signal-to-noise ratio.

The paper is organized as follows. The required preliminaries for proposed methodology are presented in Section II. The proposed adaptive Type-2 fuzzy filter is explained in Section III. In Section IV, experimental results, discussion, and comparison with existing filters are presented. The concluding remarks are drawn in Section V.

II. Preliminaries

A. Type-2 fuzzy set

A Type-2 fuzzy set is an extension of Type-1 fuzzy set, which was originally introduced by Zadeh [10]. The set theoretic operations and properties of membership grades are

Department of Electrical Engineering, IIT Kanpur, India - 208016

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2805289, IEEE Transactions on Fuzzy Systems

evaluated in the form of algebraic product and sum given by Mizumoto and Tanaka [11]. Karnik and Mendel in [12] have extended the concept of Type-2 set for performing union, intersection and complement. The Type-2 fuzzy set $\tilde{M}_{ij}^H$, is characterized by membership function $\mu_{\tilde{M}_{ij}^H}(p_{ij}, \mu_{M_{ij}^H})$ and is defined as:

$$\tilde{M}_{ij}^H = \{ \, (p_{ij}, \mu_{M_{ij}^H}), \;\; \mu_{\tilde{M}_{ij}^H}(p_{ij}, \mu_{M_{ij}^H}) \;\; \forall \; p_{ij} \in I, \;\; \forall \;\; \mu_{M_{ij}^H} \in J_{p_{ij}} \subseteq [0\,1] \, \} \tag{1}$$

where $0 \leq \mu_{M_{ij}^H}, \mu_{\tilde{M}_{ij}^H}(p_{ij}, \mu_{M_{ij}^H}) \leq 1$ and I is the universe of discourse.

B. Mean of k-middle

The *mean of k-middle* is similar to α trimmed mean as defined in [9], [18]. Let $R = \{r_1, r_2, \cdots, r_N\}$ is an N element set, then the *mean of k-middle*, $m_k(R)$ is given by

$$m_k(R) = \begin{cases} \frac{1}{2k-1} \sum\limits_{i=h-k+1}^{h+k-1} r_i, & \text{if } N \text{ is odd } (N = 2h-1) \\ \frac{1}{2k} \sum\limits_{i=h-k+1}^{h+k} r_i, & \text{if } N \text{ is even } (N = 2h) \end{cases} \tag{2}$$

where r_i is the i^{th} element in set R and $k = 1, 2, \cdots, h$. For $k = 1$, it is same as classical median, and for $k = h$, it is same as classical mean.

C. Neighborhood pixel set

A neighborhood pixel set R_{ij}^H associated with pixel $p_{ij} \in I$ with *half filter window* of size H is defined as:

$$R_{ij}^H = \{ \, p_{i+k,j+l} \; \forall \, k, l \in [-H, H] \, \} \tag{3}$$

where, R_{ij}^H has a size of $(2H+1) \times (2H+1)$. For example, if r_n is an element in R_{ij}^H then $n = 1, 2, \cdots, N$, where, $N = (2H+1) \times (2H+1)$.

III. Proposed Adaptive Type-2 Fuzzy Filter

The proposed Type-2 adaptive fuzzy filter for SAP noise removal has been discussed in this section. It has two steps. In the first step, pixels are categorized as 'good' or 'bad'. Two approaches have been proposed for this categorization. Both have their own advantages. Either of them can be used. In the second step an approach for denoising bad pixels is proposed. The schematic steps of methodology is shown in Fig. 1.

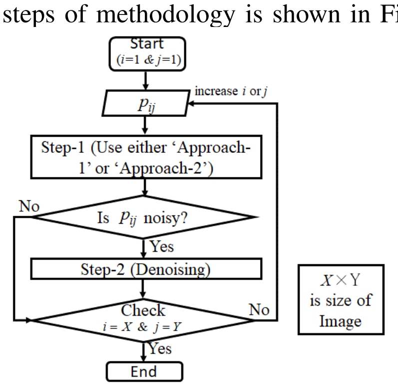


Fig. 1: Schematic diagram of proposed filter

Both the approaches of first step comprise of three parts. In the first part, set R_{ij}^H is evaluated using (3) for pixel p_{ij}. The primary MF values of Type-2 fuzzy set are evaluated for each element of R_{ij}^H in the second part. The upper membership function (UMF) and lower membership function (LMF) values of Type-2 fuzzy set are used to decide the threshold. In the third part, elements of R_{ij}^H are categorized as 'good' and 'bad' pixels by comparing their primary MF values with the threshold. The pixel (within filter window) having primary MF value greater than the threshold is considered as 'good' otherwise 'bad'. If the center pixel p_{ij} is a good pixel, then its value is retained and filtered in next iteration. If the center pixel p_{ij} is bad then it is denoised using good pixels of R_{ij}^H in the second step which is Type-1 fuzzy set [10], [29], [30].

The two approaches for the first step differ in design of UMF and LMF of Type-2 fuzz set. The algorithm designed for noise removal uses anyone of the two approaches. The two approaches are explained in following sub-sections.

A. MFs with distinct Means and same Variance

In this approach, UMF and LMF of Type-2 fuzzy set are designed to obtain a threshold which will ensure proper categorization of pixels in a window. The pixel having intensity value greater than the threshold is considered as good, whereas one with value lesser than the threshold is a bad or noisy pixel. The paper deals with SAP noise whose intensity values are either 0 or 1. Hence the pixels having intensity values $p_{ij} \notin \{0, 1\}$ are retained.

For each pixel $p_{ij} \in \{0, 1\}$ at the location (i, j) in an image I, a filter window of size $(2H+1) \times (2H+1)$ is considered for the evaluation of neighborhood set R_{ij}^H. The pixel p_{ij} is at the center of the window and H is its half-length. A Type-1 fuzzy set M_{ij}^H is defined where each element $r_n \in R_{ij}^H$ is associated with a membership value using a GMF $\mu_{M_{ij}^H}(r_n) : R_{ij}^H \to [0, 1]$. It is known as primary MF. The membership grade of primary MF is again a fuzzy set in the unit interval [0,1]. It is known as secondary MF. The Type-2 fuzzy set $\tilde{M}_{ij}^H$ is characterized by MF $\mu_{\tilde{M}_{ij}^H}(r_n, \mu_{M_{ij}^H})$ as defined in (1). Every element r_n in the set R_{ij}^H belongs to Type-1 fuzzy set M_{ij}^H and it is associated with a GMF,

$$\mu_{M_{ij}^{(H,k)}}(r_n) = e^{-(r_n - \nu_{ij}^{(H,k)})^2 / 2(\sigma_{ij}^H)^2} \tag{4}$$

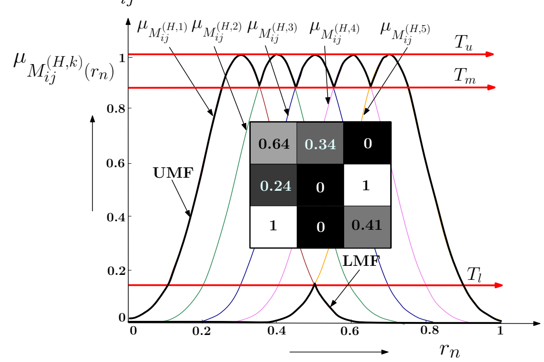


Fig. 2: MFs with distinct Means and same Variance

The GMF parameters, Means $(\nu_{ij}^{(H,k)})$ are varied with respect to k and Variance (σ_{ij}^H) is kept constant. They are calculated as follows:

$$\nu_{ij}^{(H,k)} = m_k(R_{ij}^H), \;\; k = 1, 2, 3, \ldots, h \tag{5}$$

$$\sigma_{ij}^H = m_h(\Omega_{ij}^H) \tag{6}$$

where, m_k is the *mean of k-middle* and m_h is the classical mean (*mean of k-middle* at $k = h$) as defined in (2). The parameter Ω_{ij}^H is calculated using l_1 norm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2805289, IEEE Transactions on Fuzzy Systems

$$\Omega_{ij}^{H} = \{|r_n - \nu_{avg}|,\ \forall\ r_n \in R_{ij}^{H}\};\ \nu_{avg} = \frac{1}{h}\sum_{k=1}^{h}\nu_{ij}^{(H,k)} \quad (7)$$

where, ν_{avg} is the average mean of all the *means of k-middle*.

The equations (4)-(7) are described below. A neighborhood vector R_{ij}^{H} for a $p_{ij} \in I$ of a half filter window of size H using (3) is created. Then, using (5), h different means are evaluated for all possible values of k. Using (6) and (7) σ_{ij}^{H} and Ω_{ij}^{H} are evaluated. Based on the values of σ_{ij}^{H} and $\nu_{ij}^{(H,k)}$, we can plot h number of GMFs for a filter window using (4). For example, if $H = 1$, then $N = 9$ and $h = 5$. Five GMFs are plotted using *mean of k-middle* ($k = 1, 2, \cdots, h$) for a 3×3 window as shown in Fig. 2. Now, let Δ_{ij} is a matrix consisting membership values of elements $r_n \in R_{ij}^{H}$ evaluated using (4), (5) and (6). Basically, Δ_{ij} contains h membership values of each element of filter window corresponding to h GMFs. Hence, the size of matrix is $h \times N$ and can be written as:

$$\Delta_{ij} = \begin{bmatrix} \mu_{M_{ij}^{(H,1)}}(r_1) & \mu_{M_{ij}^{(H,1)}}(r_2) & \cdots & \mu_{M_{ij}^{(H,1)}}(r_N) \\ \mu_{M_{ij}^{(H,2)}}(r_1) & \mu_{M_{ij}^{(H,2)}}(r_2) & \cdots & \mu_{M_{ij}^{(H,2)}}(r_N) \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{M_{ij}^{(H,h)}}(r_1) & \mu_{M_{ij}^{(H,h)}}(r_2) & \cdots & \mu_{M_{ij}^{(H,h)}}(r_N) \end{bmatrix} \quad (8)$$

A column-wise S-norm (max operation) is performed on the matrix Δ_{ij} followed by T-norm (min operation) on the outcomes to obtain a MF value T_m as shown in Fig. 2 for a 3×3 window. It is the minimum MF value of UMF. In the designed algorithm T_m is used as threshold for categorizing a pixel. Mathematically, it can be expressed as,

$$T_m = \wedge(\vee(\Delta_{ij})) \quad (9)$$

where, $\wedge$ and $\vee$ are the min and max operators respectively. It is adaptive in nature as compared to threshold based on Type-1 which is heuristic [9]. In the matrix Δ_{ij}, h number of MF values are associated with each pixel intensity. A set of MF values $\mu_{M_{ij}^{H}}$ associated with neighborhood vector R_{ij}^{H} is the column-wise mean value of the matrix Δ_{ij}. Mathematically, it can be expressed as,

$$\mu_{M_{ij}^{H}} = \frac{\sum_{k=1}^{h}\Delta_{ij}}{h} \ \ \forall\ \ i = 1 \cdots N \quad (10)$$

The MF value of every single pixel in filter window obtained from (10) is compared with threshold computed in (9). If the MF value $\mu_{M_{ij}^{H}}$ is greater than the threshold T_m, then the pixel is considered as good else bad or noisy. If the center pixel p_{ij} has MF value greater than the threshold then it is deemed to be good for this iteration and supposed to be rechecked for denoising in the forthcoming iterations. T_u is maximum MF value of the UMF and T_l is the maximum MF value of the LMF as shown in Fig. 2 and Fig. 3. The complete approach is given in Algorithm 1.

B. MFs with distinct Means and Variances

The second approach is similar to the first one, with only difference in determining the MF values of each pixel. The MF values are obtained by varying both Mean and Variance. Every element r_n in the set R_{ij}^{H} belongs to fuzzy set M_{ij}^{H} similar to the first approach with a GMF,

$$\mu_{M_{ij}^{(H,k)}}(r_n) = e^{-(r_n - \nu_{ij}^{(H,k)})^2/2(\sigma_{ij}^{H})^2} \quad (11)$$

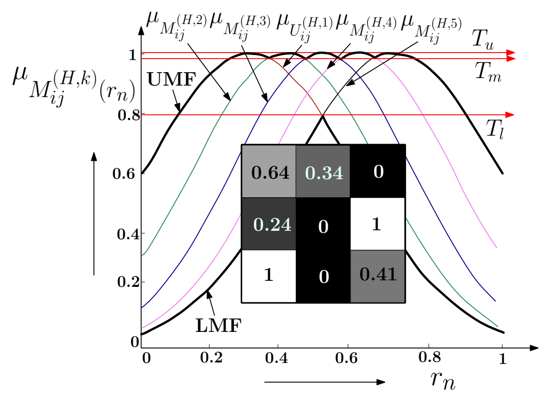


Fig. 3: MFs with distinct Means and Variances

Both the GMF parameters Means ($\nu_{ij}^{(H,k)}$) and Variance (σ_{ij}^{H}) are varied with respect to k and are calculated as follows:

$$\nu_{ij}^{(H,k)} = m_k(R_{ij}^{H}),\ \ k = 1, 2, 3, \cdots, h \quad (12)$$

$$\sigma_{ij}^{(H,k)} = m_k(\Omega_{ij}^{(H,k)}),\ \ k = 1, 2, 3, \cdots, h \quad (13)$$

where, m_k is the *mean of k-middle* defined in (2). The parameter Ω_{ij}^{H} is defined using l_1 norm as follows:

$$\Omega_{ij}^{(H,k)} = \{\ |r_n - \nu_{ij}^{(H,k)}|,\ \ \forall\ \ r_n \in R_{ij}^{H}\ \} \quad (14)$$

The matrix Δ_{ij} defined in (8) are calculated using (12), (13) and (14). The operations given in (9) and (10) are performed on matrix Δ_{ij} to obtain the threshold T_m and MF value $\mu_{M_{ij}^{H}}$ for 3×3 window. They are shown in Fig. 3. The categorization of pixels are done in similar way as the first approach as mentioned in Algorithm 1 with one change in step 8.

C. Denoising noisy pixels using Type-1 fuzzy logic

The pixels categorized as bad (noisy) in step-1 are denoised in this step. The good pixels in the window play an important role in denoising the bad pixel. The selection of appropriate weights for these good pixels is very important. The Type-1 fuzzy set has been used to determine the desired weights for good pixels. The set of good pixels G is considered to be a fuzzy set and each element in it is mapped to $[0, 1]$ by MF μ_G. Using this approach each pixel in the fuzzy set G has a different MF value. GMF is considered for the set of good pixels in G. First, mean of GMF is computed by applying *mean of k-middle* for all the good pixels in G and then an average is taken for all the k-means. The variance is found out using l_1 norm of the good pixels with respect to the average mean. The MF plot for the good pixels across a 3×3 window is shown in Fig. 4.

The *mean of k-middle* for good pixels in a particular window is computed using (2). The average mean m_{avg} and variance σ_G of MF μ_G are determined as follows:

$$m_{avg} = \frac{\sum_{k=1}^{h} m_k}{h};\quad \sigma_G = |G - m_{avg}| \quad (15)$$

$$\mu_G(g_i) = e^{-(g_i - m_{avg})^2/2\sigma_G^2} \quad (16)$$

where, m_k is the k^{th} ($k = 1, 2, \cdots, h$) middle mean of good pixels in G. Finally, the denoised pixel intensity p^{new} is computed in (17).

$$p^{new} = \frac{\sum_{\forall g_i \in G} w_i g_i}{W};\quad W = \sum_{i=1}^{\rho} w_i \quad (17)$$

$w_i \in \mu_G$ is the weight corresponding to the i^{th} good pixel, ρ is the number of good pixels in a filter window and W is the normalizing term.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2805289, IEEE Transactions on Fuzzy Systems

Table I: Comparison of performance of proposed filter with several state-of-the-art algorithms (in term of PSNR (in dB))

Dataset [28]	Noise (In%)	FM [21]	CEF [24]	PWS [23]	AMEPR [19]	PB [26]	BDND [20]	CM [27]	SATV [25]	IAF [9]	Proposed Approaches A	Proposed Approaches B
Lena	20	37.05	37.46	36.85	38.21	38.31	38.52	39.42	39.20	39.92	**40.75**	***40.79***
	50	29.81	30.71	29.57	33.46	32.04	32.74	33.57	33.88	34.10	**34.88**	***34.90***
	80	23.11	23.22	22.68	27.16	25.97	27.11	28.45	27.14	28.84	***28.92***	**28.89**
Bridge	20	30.41	28.47	29.18	32.12	27.53	30.66	31.35	31.98	31.59	***32.56***	**32.55**
	50	24.24	22.35	22.79	26.64	23.75	25.22	26.52	26.89	27.01	**27.62**	***27.63***
	80	20.67	19.52	20.03	21.98	19.45	21.39	22.28	22.41	22.92	***23.21***	***23.21***
Peppers	20	36.21	35.03	35.46	37.45	36.32	34.44	37.54	36.87	37.99	**41.00**	***41.01***
	50	29.53	30.38	29.26	31.25	29.25	30.23	32.03	31.62	32.34	***35.17***	**35.14**
	80	22.21	23.65	22.84	27.32	25.64	26.61	27.46	26.42	27.54	**29.22**	***29.26***
Baboon	20	27.22	26.85	26.82	29.87	24.22	27.73	28.47	28.49	29.75	***29.30***	**29.29**
	50	22.26	21.93	20.42	24.52	21.27	23.46	24.05	23.91	24.84	**24.59**	***24.60***
	80	18.69	17.60	17.86	19.73	17.38	19.92	20.36	20.59	20.73	**20.79**	***20.81***
Barbara	20	29.46	29.58	28.72	29.72	29.24	29.85	30.78	30.70	31.95	***33.22***	**33.20**
	50	23.46	23.37	22.69	25.33	23.49	25.17	26.10	25.91	26.74	**28.24**	***28.26***
	80	19.35	19.31	18.91	21.41	21.64	21.74	22.54	22.66	22.78	***23.82***	**23.81**
Boat	20	34.75	30.87	33.78	34.89	32.73	34.83	35.31	35.97	36.03	***36.67***	**36.62**
	50	27.96	25.65	26.80	29.34	27.83	29.68	29.99	30.38	30.69	***31.39***	**31.38**
	80	23.65	21.46	22.50	24.75	22.29	24.93	25.58	25.18	25.88	**26.23**	***26.26***
House	20	37.84	37.91	37.74	37.42	37.45	37.23	38.32	38.66	38.97	**47.48**	***47.54***
	50	29.45	29.52	29.49	31.51	30.70	31.72	32.45	32.53	33.19	***39.90***	**39.88**
	80	22.65	22.63	22.54	25.45	24.84	25.81	27.52	26.16	27.82	***32.04***	**32.01**

**The values in bold represent better PSNR as compared to several state-of-the-art algorithms. **The values in bold-italic represent better PSNR among the two proposed approaches.*

Table II: Comparison of mean run time of proposed filter with several state-of-the-art algorithms (in seconds)

Noise (In%)	FM [21]	CEF [24]	PWS [23]	AMEPR [19]	PB [26]	BDND [20]	CM [27]	SATV [25]	IAF [9]	Proposed Approaches A	Proposed Approaches B
20	2.31	18.59	26.48	3957.94	895.72	219.36	12.58	23.96	12.04	12.76	12.80
50	2.31	18.59	26.48	3957.94	895.72	219.36	12.58	23.96	12.04	51.43	51.55
80	2.31	37.09	34.63	6486.45	1804.51	220.47	17.02	28.78	26.32	155.40	156.10

Algorithm 1 SAP noise removal using Type-2 fuzzy filter

for all pixels $p_{ij} \in I$ **do**
  if $p_{ij} \notin \{0,1\}$ **then**
    retain p_{ij}
    continue;
  while $p_{ij} \in \{0,1\}$ **do**
    initialize $H = 1$;
    Compute R_{ij}^H by eq. (3);
    Compute $\mu_{M_{ij}^H}, \sigma_{ij}^H$ based on Approach A using $(4)-(8)$; **or**,
    Compute $\mu_{M_{ij}^H}, \sigma_{ij}^H$ based on Approach B using $(12)-(15)$;
    Compute $T_m, \mu_{M_{ij}^H}$ using $(9)-(11)$;
    if $\mu_{M_{ij}^H}(p_{ij}) \geq T_m$ **then**
      retain p_{ij}
      break;
    if $\sigma_{ij}^H \leq \epsilon$ **then**
      $p_{ij} = m_{avg}$
      break;
    Compute G_{ij}^H
    $\rho = |G_{ij}^H|$
    if $\rho < 1$ **then**
      $H = H + 1$;
      continue;
    Compute p_{ij}^{new} using $(15)-(17)$;

In case of high noise level, there is a chance that ρ (Cardinality of G_{ij}^H) will become zero. In such cases H is increased by 1 and the whole process is repeated. If the deviation σ_G is below a very small threshold ϵ, i.e., the window consists of pixels with intensities very near to that of p_{ij} then the value of p_{ij} is simply replaced by m_{avg}. This will restrict the division by zero which may arise due to uniform intensity and make σ_G zero.

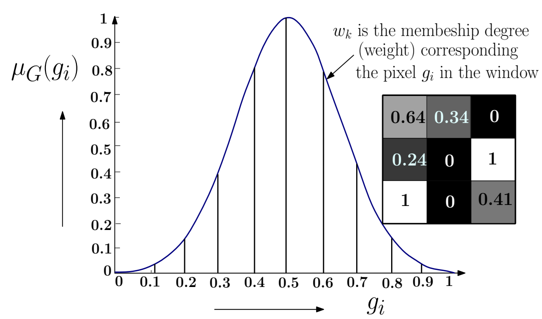


Fig. 4: Denoising noisy pixels using Type-1 fuzzy logic (where, g_i is i^{th} good pixel in the set G)

D. Stopping Criterion

The stopping criteria for algorithm is defined by percentage of denoised pixels α given in (18). The number of noisy pixels detected in succeeding iterations is denoted by d_m. The algorithm terminates when α falls below 0.05%.

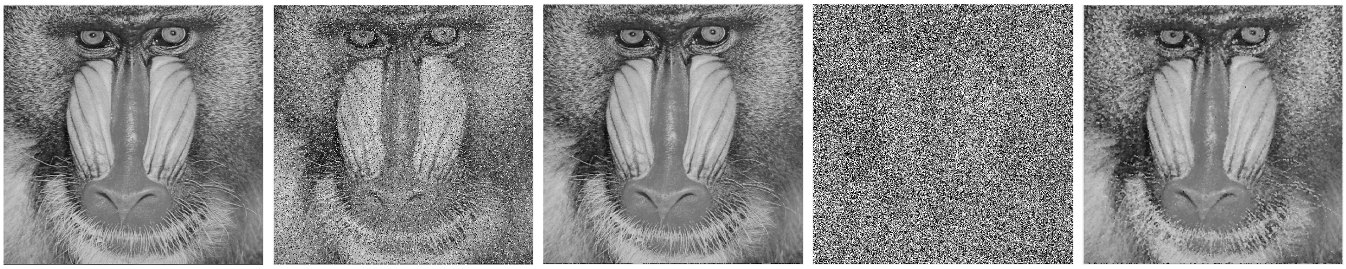
$$\alpha = \frac{d_m^{new} - d_m^{old}}{X \times Y} \tag{18}$$

where, $X \times Y$ represents the dimension of 8-bit images.

IV. Results and Validations

The proposed filter has been used for removing SAP noise from seven standard grayscale images of resolution 512×512 [28]. The images considered are Baboon, Barbara, Boat, Bridge, House, Lena, and Peppers. The validations were carried out on a system with Intel Core i7, 3.2 Ghz processor

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2018.2805289, IEEE Transactions on Fuzzy Systems



(a) (b) (c) (d) (e)

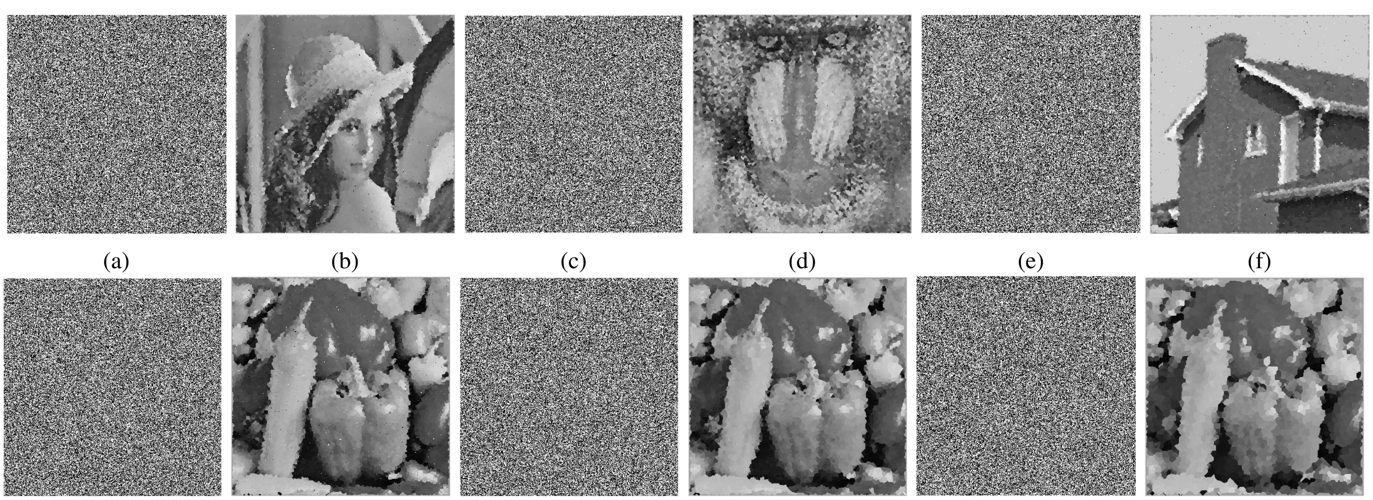
Fig. 5: Baboon image: (a) actual image, (b) at 20% noise, (c) filtered image; (d) at 80% noise, (e) filtered image.

(a) (b) (c) (d) (e)

Fig. 6: Barbara image: (a) actual image, (b) at 20% noise, (c) filtered image; (d) at 80% noise, (e) filtered image.

(a) (b) (c) (d) (e)

Fig. 7: Lena image: (a) actual image, (b) at 20% noise, (c) filtered image; (d) at 80% noise, (e) filtered image.



(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

Fig. 8: Filtered images at higher noise level: (a) Lena at 97% noise, (b) filtered Lena image; (c) Baboon at 97% noise, (d) filtered Baboon image; (e) House at 97% noise, (f) filtered House image; Filtered gray scale Peppers image at higher noise level (g) at 97% noise, (h) filtered image; (i) at 98% noise, (j) filtered image; (k) at 99% noise, (l) filtered image.

and 8 GB RAM. The performance of proposed filter has been compared with various state-of-the-art algorithms. For all these images the parameter k is varied form $1, 2, \cdots, h$ in both the approaches of step 1 of filter to compute the mean and variance. The MFs are plotted for both the approaches to decide respective threshold. The threshold is adaptive in nature which depends on SAP noise level. The SAP noise level is varied from 20% to 99%. For comparative analysis the results for three different noise levels viz. 20%, 50%, and 80% are shown in Table I. For the noise level of 97% the results are shown (Fig. 8) with the images of Lena, Baboon, House and Peppers. The filter operation at noise level of 98% and 99% are tested only for peppers. The proposed approaches preserve significant image details even for 99% of noise level (shown in Fig. 8). To perform the experimentation we have set a minimum number of good pixel (ρ_{min}) as 1 in filter window to estimate the pixel intensity of bad pixels. This is because a large number of "good" pixels are required for denoising bad pixel. Having ρ_{min} more than 1, the results are poor for high noise level. The performance of proposed approaches are numerically evaluated using peak signal-to-noise ratio (PSNR). The PSNR is defined using filtered image (I_f) with respect to original image (I_o) as follows:

$$\text{PSNR}(I_o, I_f) = 10\log_{10} \frac{255^2}{\frac{1}{XY}\sum_{i,j}(I_o(i,j) - I_f(i,j))^2} \quad . \qquad (19)$$

where, 255 is the maximum pixel intensity of 8 bit images.

Table I shows the performance of proposed approaches A and B with respect to various state-of-the-art algorithms, namely, Iterative adaptive fuzzy filter (IAF) [9], adaptive median with edge-preserving regularization (AMEPR) [19], boundary discriminative noise detection (BDND) [20], fast median (FM) [21], wavelet neural network (WNN) [22], pixel-wise S-estimate of variance (PWS) [23], contrast enhancement based filter (CEF) [24], spatially adaptive total variation filter (SATV) [25], patch-based (PB) [26], and the cloud model (CM) filter [27]. The PSNR values given in Table I are averaged over 20 experiments for each image. The computational time of the proposed approaches are comparable to state-of-the-art algorithms for low noise levels as shown in Table II. With the increase in noise level the filter window size increases. For large windows more MF's are drawn to compute the threshold. The computational time is relatively higher in such cases.

V. Conclusion

This paper proposes a novel adaptive Type-2 fuzzy filter for removing SAP noise from grayscale images. The use of either of the two proposed approaches in first step of fuzzy filter detects noisy pixels in the filter window. In the subsequent step, the proposed weighted mean Type-1 fuzzy approach denoises bad pixels in the respective filter window. The proposed approach is validated on several grayscale images. The experimental results show that proposed filter outperforms other state-of-the-art algorithms. Moreover, the filter preserves meaningful image details even at noise level as high as 99%.

References

[1] W. Luo, "Efficient removal of impulse noise from digital images," *IEEE Trans. Consumer Electron.,* 52(2), pp. 523-527, 2006.

[2] A. Bovik, *Handbook of Image and Video Processing,* New York, NY, USA: Academic, 2000.

[3] T. A. Nodes and N. C. Gallagher, "Median filters: Some modifications and their properties," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* 30(5), pp. 739-746, 1982.

[4] R. C. Gonzalez and R. E. Woods, *Digital Image Processing,* 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002

[5] C. S. Lee, Y. H. Kuo, and P. T. Yu, "Weighted fuzzy mean filters for image processing," *Fuzzy Sets Syst.,* no. 89, pp. 157–180, 1997.

[6] P. Zhang and F. Li, "A new adaptive weighted mean filter for removing salt and pepper noise," *IEEE Signal Process. Lett.,* 21(10), pp. 1280–1283, 2014

[7] C. S. Lee and Y. H. Kuo, "Adaptive fuzzy filter and its application to image enhancement," *Fuzzy Techniques in Image Processing*, Physica-Verlag HD, 2000, pp. 172–193.

[8] F. Farbiz and M. B. Menhaj, "A fuzzy logic control based approach for image filtering," *Fuzzy Techniques in Image Processing*, Physica-Verlag HD, 2000, pp. 194–221.

[9] F. Ahmed and S. Das, "Removal of high-density salt-and-pepper noise in images with an iterative adaptive fuzzy filter using alpha-trimmed mean," *IEEE Transactions on Fuzzy Systems,* 22(5), pp. 1352-1358, 2014.

[10] A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning-1," *Inform. Sci.,* vol. 8, pp. 199-249, 1975.

[11] M. Mizumoto and K. Tanaka, "Some properties of fuzzy sets of type-2," *Inform. Control,* vol. 31, pp. 312-340, 1976.

[12] N. Karnik and J. M. Mendel, "Operations on type-2 fuzzy sets," *Int. J. Fuzzy sets System.,* vol. 122, pp. 327-348, 2001.

[13] Q. Liang and J.M. Mendel, "Equalization of nonlinear time-varying channels using type-2 fuzzy adaptive filters." *IEEE Transactions on Fuzzy Systems*, 8(5), pp. 551-563, 2000.

[14] R.I John, P.R Innocent and M.R Barnes, "Neuro-fuzzy clustering of radiographic tibia image data using type 2 fuzzy sets". *Information Sciences*, 125(1), pp. 65-82, 2000.

[15] A. Ba and M.E YÜksel, "Impulse noise removal from digital images by a detail-preserving filter based on type-2 fuzzy logic". *IEEE Transactions on Fuzzy Systems*, 16(4), pp. 920-928, 2008.

[16] M.E Yuksel and A. Basturk, "Application of type-2 fuzzy logic filtering to reduce noise in color images". *IEEE Computational intelligence magazine*, 7(3), pp. 25-35, 2012.

[17] P. Murugeswari and D. Manimegalai, "Noise reduction in color image using interval type-2 fuzzy filter (IT2FF)," *Int. J. Eng. Sci. Technol.,* vol. 3, pp. 1334–1338, 2011

[18] J. Bednar and T. Watt, "Alpha trimmed means and their relationship to median filters," *IEEE Transactions on acoustics, speech, and signal processing,* 32(1), pp. 145-153, 1984.

[19] R. H. Chan, C. W. Ho, and M. Nikolova, "Salt and pepper noise re moval by median type noise detectors and detail preserving regularization," *IEEE Trans. Image Process.,* 14(10), pp. 1479–1485, 2005.

[20] P. E. Ng and K. K. Ma, "A switching median filter with boundary dis criminative noise detection for extremely corrupted images," *IEEE Trans. Image Process.,* 15(6), pp. 1506–1516, 2006.

[21] K. S. Srinivasan, D. Ebenezer, "A new fast and efficient decision-based algorithm for removal of high density impulse noises," *IEEE Signal Process. Lett.,* 14(3), pp. 189–192, 2007.

[22] C. Deng and J. Y. An, "An impulse noise removal based on a wavelet neural network," *Proc. 2nd Int. Conf. Inf. Comput. Sci.,* vol. 2, pp. 71–74, May 21–22, 2009.

[23] V. Crnojević, and N. Petrović, "Impulse noise filtering using robust pixel wise S-estimate of variance," *EURASIP journal on Advances in Signal Processing,* 2010(1), pp. 830702. 2010.

[24] U. Ghanekar, A. K. Singh, and R. Pandey, "A contrast enhancement-based filter for removal of random valued impulse noise," *IEEE Signal Process. Lett.,* 17(1), pp. 47–50, 2010.

[25] R. Rojas and P. Rodrguez, "Spatially adaptive total variation image de-noising under salt and pepper noise," *Proc. the Eur. Signal Process.Conf., Barcelona, Spain,* pp. 278–282, 2011.

[26] J. Delon and A. Desolneux, "A patch-based approach for random valued impulse noise removal," *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.,* pp. 1093–1096, Mar. 25–30, 2012.

[27] Z. Zhou, "Cognition and removal of impulse noise with uncertainty," *IEEE Trans. Image Process.,* 21(7), pp. 3157–3167, 2012.

[28] A. G. Weber, "The USC-SIPI image database version 5." *USC SIPI Report* 315, pp. 1-24, 1997.

[29] N. K. Verma and M. Hanmandlu, "From Gaussian mixture model to non-additive fuzzy systems," *IEEE Trans. Fuzzy Systems,* (15)5, pp. 809-827, 2007.

[30] N. K. Verma and M. Hanmandlu, "Additive and Non-Additive Fuzzy Hidden Markov Models," *IEEE Trans. Fuzzy Systems,* 18(1), pp. 40-56, 2010.