DEVELOPMENT OF A NEW NONLINEAR FILTERING ALGORITHM FOR IMPULSE NOISE REDUCTION

ASSOC PROF. DR MOHD ASHRAF BIN AHMAD

RESEARCH VOTES NO:

RDU1703151

Fakulti Teknologi Kejuruteraan Elektrik & Elektronik

Universiti Malaysia Pahang

2020

ACKNOWLEDGMENTS

The author and fellow researchers would like to thank the Universiti Malaysia Pahang, for financially supporting this research under Research Vote no. RDU1703151.

ABSTRACT

The use of digital image-based visual information has gained a lot of attention due to its flexibility and easy to be manipulated. However, digital images that have been used as basic input to an application system are frequently contaminated by noise. One of the most common types of noise found in digital images is impulse noise. Therefore, a new type of filter based on switching scheme are proposed as impulse noise removal. The proposed filter namely Dual Sliding Statistics Switching Median filter (DSSSM) are two-stage filters which consist of noise detection and noise filtering stages. In the DSSSM method, the filtering process begins with noise detection stage; whereby the statistics of a localized detection window of sorted and non-sorted orders are simultaneously processed. Subsequently, the median of absolute difference obtained from the statistics of both windows will be used to classify the existence of the noise pixel. Next in the filtering stage, the pixels that have been classified as noise will be restored while the noise-free pixels will be left unchanged. By using these processing techniques, the proposed filters are not only able to remove the impulse noise, but are also able to maintain the structure and shape of the object in the image. The simulation results also show that the DSSSM filter outperform other existing conventional filters in the literature, in terms of qualitative and quantitative assessments.

ABSTRAK

Penggunaan informasi visual berasaskan imej digital telah mendapat perhatian yang begitu meluas kerana ciri-cirinya yang fleksibel dan mudah untuk dimanipulasi. Namun, imej digital yang menjadi input asas kepada sesuatu sistem aplikasi sering dicemari oleh hingar. Antara jenis hingar yang lazimnya terdapat pada imej digital ialah hingar impuls. Oleh itu, suatu penapis baharu berasaskan skim pensuisan untuk penyingkiran hingar impuls pada imej digital telah diperkenalkan. Penapis ini yang dinamakan sebagai penapis Median Pensuisan Statistik Dwi-gelongsor (Dual Sliding Statistics Switching Median filter (DSSSM)) Dalam kaedah DSSSM, pengesanan hingar dilaksanakan terlebih dahulu dengan memproses statistik tetingkap pengesan setempat dalam susunan teratur dan tidak teratur secara serentak. Kemudian, median perbezaan mutlak yang diperolehi daripada statistik kedua-dua tetingkap akan digunakan bagi mengklasifikasikan piksel hingar yang wujud. Seterusnya pada peringkat penapisan, piksel-piksel yang telah diklasifikasikan sebagai hingar akan dipulihkan manakala piksel-piksel bebas hingar akan dikekalkan. Dengan cara ini, penapis yang dicadangkan ini bukan sahaja mampu bagi menyingkirkan hingar impuls, malah ia juga mampu mengekalkan struktur dan bentuk objek dalam imej. Hasil daripada keputusan simulasi turut menunjukkan penapis DSSSM mampu mengatasi penapis-penapis konvensional lain yang wujud dalam kajian ilmiah, baik dari segi penilaian kualitatif mahupun kuantitatif.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Background and Research Motivation

In the era of multimedia technology, the use of digital image-based visual information have gained a lot of attention due to its flexibility (i.e. the input data is able to be pre and/or post-processed) and this phenomenon is expected to continue growing. Medical imaging diagnosis, geographical analysis, image-based control and instrumentation are among modern daily life applications which have adopted digital image processing technology. In general, these applications involve numerous image processing operations (e.g. image segmentation, edge detection, classification, etc.) of which are highly dependent on the quality of digital input images in order for them to work perfectly. Unfortunately, digital images are frequently subjected to the contamination of impulse noise that typically due to the interferences generated during transmission/acquisition or storage through electronic medium, poor sensor configuration and timing errors in analog-to-digital conversion [1]. Therefore, it is imperative to remove the impulse noise effect before any subsequent image processing operations can be carried out as the occurrences of impulse noise can severely damage the information in the original image. An important characteristic of this type of noise is that only parts of the pixels are contaminated while the rest are noise-free.

One of the most effective approaches to cater for the occurrence of impulse noise and for the improvement of the quality of the acquired image is by using denoising-based algorithm. Accordingly, a large number of nonlinear filters have been widely exploited to remove the impulse noise as they are generally more superior to linear filtering techniques. For instance, standard median (SM) filter [2] and adaptive median (AM) filter [3] are two of the most basic nonlinear filtering techniques for suppressing impulse noise. Ironically, this SM is implemented unconditionally across the image while its variants (e.g. see AM) inherited this clumsy smoothing property; thus they tend to modify both noise and noise-free pixels simultaneously. Consequently, the detailed regions such as object edges and fine textures in image are smeared and appear blurry or jittered.

To get rid of the problem, various filters under switching scheme have been studied and experimented by a number of recently published works; such as switching median filter I and II (SWM-I and SWM-II) [4], multi-state median (MSM) filter [5], Laplacian switching median (LSM) filter [6], enhanced rank impulse detector (ERID) [7], directional weighted median (DWM) filter [8] and noise-ranking switching filter (NRSF) [9], etc. Basically, this filtering scheme divides its implementation into two stages; which are impulse noise detection stage and impulse noise filtering stage. Impulse noise detection algorithm is implemented prior to the filtering process in order to determine whether a pixel should be modified or left unchanged. With this kind of filtering properties, these techniques are shown to be more effective to preserve most of the image details compared to the conventional non-switching techniques.

In a different way, Chen and Wu [10] have come out with the technique based on the adaptive switching scheme called adaptive center-weighted median (ACWM) filter. Briefly, ACWM is a two phase iterative median filter which uses a fixed processing window size with adaptively filtering process. Apart from the switching scheme and the adaptive switching scheme, the hybrid switching scheme is another class of filters which has been groomed to yield good filtering results. Many researchers have embedded other order statistics (e.g. rank-order statistic, median of absolute deviations, etc.) and image processing techniques (e.g. mathematical morphology, directional or edge detection, etc.) into the hybrid switching scheme filters as part of its filtering mechanism. One of the techniques in this filtering scheme is the work done by Chen et al. [11]. In general, this tri-state median (TSM) filter is formed by a combination of SM and center weighted median (CWM) filter [12]. It uses a set of two predefined thresholds to determine whether the original pixel should be retained or replaced by the SM filtered output or the CWM filtered output. Meanwhile, a more sophisticated filtering technique has been presented by Luo in [13]. This filter incorporates the rank order absolute difference (ROAD) statistics with fuzzy impulse detection algorithm to classify and remove impulse noise from corrupted images. Noticeably, the restoration abilities of those aforementioned techniques are improved but at the cost of lost fine image details and increased complexity.

Of late, in accordance with the evolution in digital image acquisition technologies, the corruption rate of impulse noise in digital images has managed to be reduced to the level that may be regarded as low; i.e. less than 30% noise density [14-20].

1.2 Research Objectives

Based on the aforementioned statements and observations; hence our aim in this project is:

(i) To develop an efficient filtering technique for the range of low level impulse noise

(ii) To analyze the characteristic of the proposed filtering techniques in terms of the images' details and textures preservation, the processing time, and it's robustness to filter out the noise.

1.3 Scope of Project

The scope of this research is limited to the design and development of noise filtering algorithms. It begins with a scientific study of various existing conventional filtering methods in an effort to discover the advantages and disadvantages inherent in each method. Later, this research will propose a new noise filtering method to address the identified weaknesses. In the production of image recovery algorithms by this research, special attention is given to the process of filtering out random values of random impulse noise in gray-scale digital images. The purpose of this focus is limited to gray-scale digital images in order to focus on developing a sound noise filter base first before the framework can be used for the development of further filtering mechanisms that include various types of images such as colored images and medical images. A hundred standard test images have been used in a series of simulation experiments. All of these images are downloaded from various database sources on the internet.

Finally, all proposed algorithms, along with various other conventional noise filtering methods will be implemented in M-file programming using MATLAB 2014 integrated design environment software (IDE).

1.4 Organization of Report

This report is organized as follows.

Overall, this thesis is divided into 5 chapters. Initially, the discussion in Chapter 1 begins with a brief introduction to digital images and their applications. This includes a brief discussion of some common types of digital noise and their recovery techniques. Later, the objective of the research is also explained as a guide to the scope of work carried out. Finally, this chapter concludes with thesis guidelines.

The whole of Chapter 2 is dedicated to the general survey of existing impulse noise filters. The development of conventional impulse noise filters marked by the increasing sophistication of the approaches used has been studied for the identification of their filtering mechanisms. The limitations and disadvantages of these noise filters are also discussed. These noise filters will be used in comparison to the proposed filtering method.

The discussion in Chapter 3, however, focuses on the proposed impulse noise filtering method. Two new types of filters of random-value impulse noise have been proposed and they will serve as the core content of this chapter. Further details of the implementation steps and further details of the two proposed filtering methods are also presented. The final section of this chapter also deals with the selection of appropriate test images to evaluate the performance of the proposed noise filters.

Further in Chapter 4, the results and discussion of the output image obtained from the proposed filtering method and several other conventional filtering methods will be presented. All results are discussed, interpreted and compared qualitatively and quantitatively in an effort to evaluate the performance of each type of filter performed.

Finally, Chapter 5 will summarize and conclude the overall project that has been undertaken. In addition, future suggestions and improvements to the proposed noise filter are also included in this last chapter.

Chapter 2

Literature Review

2.1 Introduction

Basically, noise-induced degradation will result in low-quality and unattractive visual appearance. As stated in the preceding chapter, the image recovery process is usually used on digitally degraded digital images for the purpose of retrieving the original information or details of the image. Modern applications now require an image recovery algorithm to not only eliminate noise, but at the same time maintain the original texture of the image. These two conflicting goals have led to the creation of various noise filtering methods. For many years, there was still a need to develop image recovery algorithms; either by suggesting a new approach or by improving the efficiency of existing image recovery methods.

2.2 Digital Image Representation

Generally, images are divided into two types - digital and analog. Basically, analog images cannot be analyzed directly using a computer because the computing process works with digital data instead of analog. Therefore, an image must first be transformed into a digital representation before it can be processed or analyzed by a computer. The process of digitizing this image is called 'digitization'.

Simply put, a digital image can be imagined as a piece of a picture that is divided into small rectangles, such as 6 vertically and 6 horizontally (see Figure 2.1).

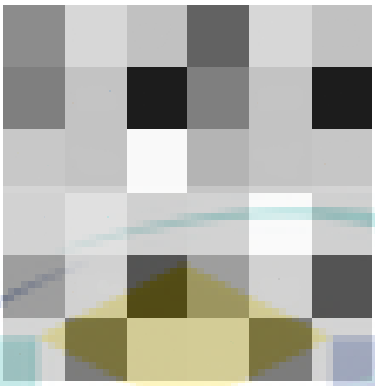


Figure 2.1: The image is divided into 6 × 6 sections.

In digital imaging terms, each of these small rectangles is known as a pixel derived from an English word 'pixel' which stands for term 'picture element'. At each pixel location is the brightness of an image being sampled and merged. This will create an integer (that is, a signed integer) that represents the brightness or darkness of the image at the pixel location. Brightness and darkness of the image are also often referred to as intensity or gray level. In general, the order of the two-dimensional arrays of each pixel with this integer value is referred to as a digital image. Therefore, digital images can be translated into representations of a matrix with coordinates and magnitudes as shown in Figure 2.2.

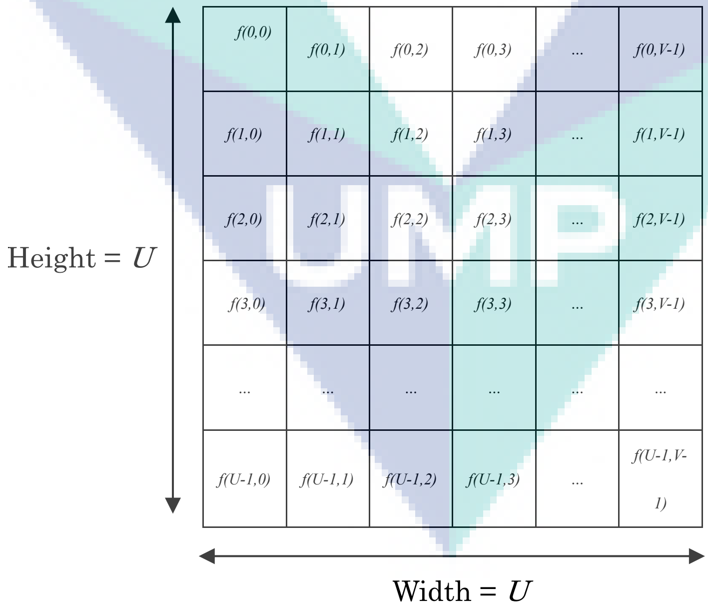


Figure 2.2: $U \times V$ digital image.

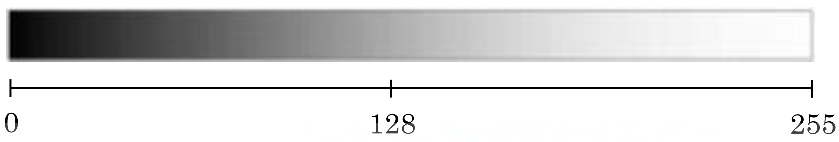


Figure 2.3: 256 gray level in an 8-bit gray scale image.

In this report, digital images are defined as two-dimensional functions $f(i, j)$ where i and j are coordinates, while f is the intensity value of the coordinates (i, j) of the image. In gray scale images, all red, green, and blue components of each pixel are represented by a single color sample that carries intensity information. Typically, the intensity of this gray scale is stored as an 8-bit integer giving 256 different color effects; ranging from black at low intensity to white at highest intensity. In short, the intensity of this gray scale image can also be represented by the integer interval [0, 255], as shown in Figure 2.3.

2.2.1 Impulse Noise Model

Digital images are usually exposed to various types of noise pollution during the process of image acquisition and / or transmission due to several problems or imperfections in image sensors and communication channels. Noise is defined as an unwanted signal and its presence will damage the image quality.

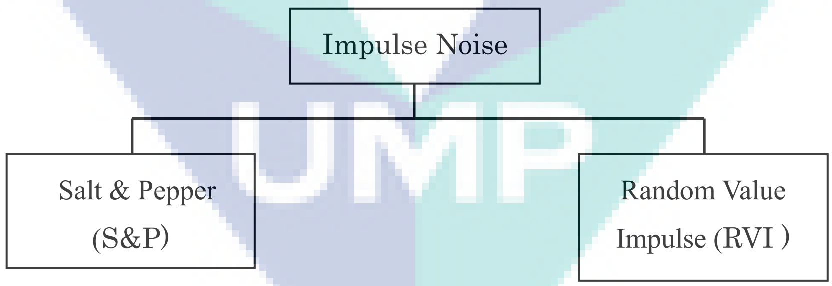


Figure 2.4: Types of impulse noise.

In general, based on the noise distribution in the image histogram, impulse noise can be categorized into two types; which is a random value of impulse noise and a fixed value of impulse noise (see Figure 2.4). Random value impulse noise models (RVI) are also known as uniform impulse noise, while fixed value impulse noise models are often referred to as salt & pepper (S&P) noise. Examples of these two types of noise are shown in Fig. 2.5 (b) and (c). Consider $f(i, j)$ and $x(i, j)$, are the gray level or intensity of the original image and the noise image at location (i, j), respectively. Noise impulses that occur with noise density r can be defined as:

$$x(i,j)=\begin{cases} n(i,j) & :\text{with probability } r \\ f(i,j) & :\text{with probability } 1\text{-}r \end{cases} \tag{2.1}$$

where n (i, j) is the pixel intensity value of the noise. An image is corrupted by a random impulse noise when n (i, j) is uniformly distributed within the dynamic range of the image, namely n (i, j) ∈ [N_{min}, N_{max}]. For example, in an 8-bit gray scale image with 256 gray levels, n (i, j) random-value impulse noise can range from 0 (N_{min}) to 255 (N_{max}). Meanwhile, for salt and pepper noise; n (i, j) is assumed to take the minimum and maximum intensity intensities, namely n (i, j) ∈ (N_{min}, N_{max}).

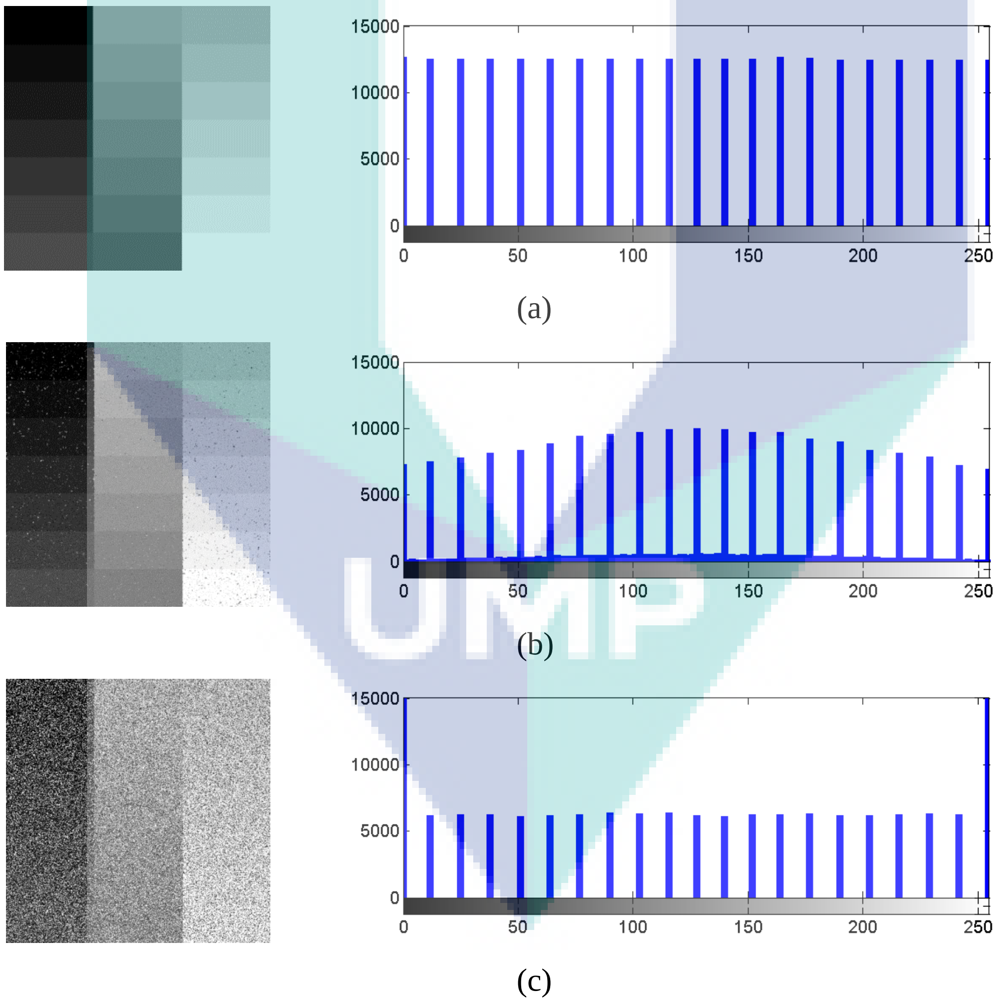


Figure 2.5: (a) The original test image and its histogram, (b) The test image is contaminated by the RVI noise and its histogram and (c) The test image is contaminated by the S&P noise and its histogram.

While many researchers have put a lot of emphasis on producing good filters to remove salt and pepper noise (the simplest form of impulse noise), but this research has taken a step forward with focusing on the detection and removal of random-impulse noise Practically, the process of detecting random-valued impulse noise is more challenging compared to the salt and pepper noise detection process because the intensity of this type of noise pixel is almost identical to the intensity of the surrounding pixels [18]-[20]. The difference between the RVI and S&P impulse noise forms can be seen in the image histogram in Figure 2.5.

2.2.2 Impulse Noise Filters

Generally, noise filters are divided into two main categories. The first category is linear filters. The linear filter output is directly proportional to the intensity value of all the pixels contained in the filtering window. One of the simplest examples of linear filters is the Standard Mean (MF) filter. This MF filter will replace each pixel in the image with an average intensity value calculated based on the pixels in the local filtering window. In addition, there are also linear filters that use the weighted average concepts such as Gaussian (GF) and Binomial (BF) filters [21]. Through this concept, the weighting will be assigned to certain pixels within the filtering window to control the balance between noise removal and edge conservation. The compressor refers to the repetition of a pixel several times in the original image data set. In this way, both the GF filter and the BF filter are able to produce better image quality and have better edge preservation capabilities than conventional MF filters.

Another major category of filters is non-linear filters. In addition to average linear filters, there are also other average based filters that fall into the non-linear filter categories such as Bilateral filter (BLF) [22] and Mean Shift filter (MSF)) [23]. Non-linear average filters usually have a more selective output. For example, the output for the MSF filter is calculated based on the pixel mode values in the local window, whereas for the BF filter, a pixel will be replaced by the weighted average value calculated based on the space and pixel information in the filter window. In general, the performance of both filters is better compared to the average filters in the linear category. However, the use of average values for the pixel recovery process will still cause an image to have serious blurring and is found to be less effective for impulse noise filtering [24], [25]. This is because, from a statistical standpoint, the existence of extreme remote data such as impulse noise in a set of data can significantly affect average values [25].

Therefore, one of the ideal solutions to solve the problem is to use a median-based filter. Some of the earliest examples of non-linear medians based on the earliest and most frequently used filters for impulse noise are such as the Standard Median (SM) filter. In short, this SM filter has the same filtering framework as the linear filter, but the output of this filter is calculated based on the median value instead of the average. This median based filter produces better filter image quality and has better edge preservation capability than the average based filters Comparison of performance between these average and median based filters can be referred in Table 2.1. The evaluation used is the comparison of the authors based on the reading surveys. From Table 2.1, the term 'good' refers to the most effective, the 'moderate' term refers to the most effective, while the term 'poor' refers to the ineffective. Based on the evaluation made in Table 2.1, it is found that non-linear filters are median, SM is better than linear filters and average non-linear filters.

Table 2.1: Comparison of image recovery performance by average and median based filters.

Filter	**Ease of Implementation**	**Edge Preservation**
MF	1	3
GF	2	3
BF	2	3
BLF	3	2
MSF	3	2
SM	1	1

Note: Scales [1] - [3] represent [1] Good, [2] Moderate and [3] Poor.

2.2.3 Conventional Median Filter

In recent decades, median-based filters have attracted the attention of many researchers for their ease of implementation and their capability in preserving edge image objects [26]. One of the median based filters is the SM filter mentioned in the previous subsection. This filter evaluates the position information of the pixel intensity in the filter window and replaces the central pixel in the window with an estimated median value of $m(i, j)$. The filtering operation of the SM filter using filtering window size of $(2N+1)\times(2N+1)$ centered at $x(i, j)$ can be described as:

$$y(i, j) = m(i, j) = \mathrm{med}\{x(i - N, j - N), ..., x(i, j), ..., x(i + N, j + N)\} \qquad (2.2)$$

where $x(i, j)$ and $y(i, j)$ are the pixel values of position (i, j) in the image contaminated by the noise and the filtered image. Although the use of SM filters is effective in impulse noise filtering work, it also tends to get rid of fine line details and corners when filtering noise. In response to this problem several other variations of the median filter were later introduced. Examples are such as Max / Median (Max / Median (MM)) filters [27] and Multi Stage Median (MSM) filters [28] designed specifically for image detail preservation. Both filters work to maintain the details of the image better, but there are limitations in terms of noise removal.

Subsequently, various filters resulting from SM modifications were introduced. Among them are the Adaptive Median (AM) filter [29] and the Signal Adaptive Median (SAM) filter [30]. The median filter in this adaptive class will adjust the size of the filtering window based on the input state processed to balance the detail conservation and noise removal. However, just like the normal median filter, this type of adaptive median filter still tends to discard the details of the image as its output is also obtained based on the median filtering [31].

Other branches in median-based filters are weight-bearing filters such as Weighted Median (WM), Adaptive Weighted Median (AWM) filter [32] and filters Center Weighted Median (CWM). This type of filter will give more weight to several pixel values within the filtering window and the degree of control between noise removal and detail conservation can be controlled through adjustable weights.

Even though the median-based filters can reduce the effects of impulse noise degradation, unfortunately the conventional median filter is applied to all pixels in the image regardless of the status of a pixel whether it is noise-free or otherwise (for example, SM, WM and AM filters discussed before this). Such a negligent modus operandi will only cause noise-free pixels to be filtered together and result in deletion of fine details such as fine lines and corners of objects [33]. This problem contributes to the deterioration in the quality of the filtered image.

An effective solution to overcome the above mentioned disadvantages is to implement an impulse noise detection mechanism first before the noise filtering process is performed [34]. Therefore, only pixels that have been classified as noise will undergo the filtering process, while pixels classified as noise-free will continue to remain as they are. The integration of noise detection into the median filtering framework is also known as switching median filtering technique.

2.2.4 Conventional Switching Median Filter

For impulse noise filtering processes, median switching techniques or methods can be considered as the most recent. This switching procedure is based on local noise measurement or is referred to as the impulse noise detection performed by the switch unit (refer to Figure 2.6). Based on the analysis by the impulse detector, the switch unit will either choose to retain the original pixel of the image or replace it with the estimated pixel recovery value (i.e. filter output).

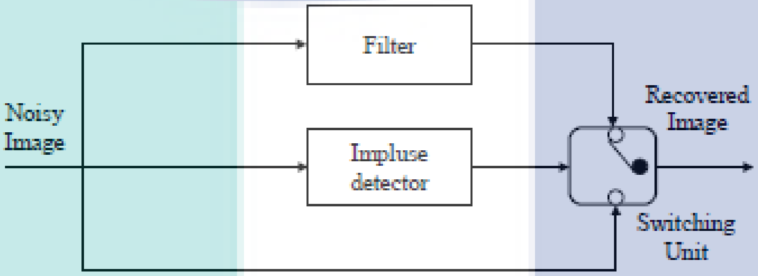


Figure 2.6: General framework of switching process

In general, the method of median filtering can be classified into 4 main groups namely standard switching, weighted switching, adaptive switching and hybrid switching. Among the earliest filters appearing among the median switching filters is the Switching Median I (SWM-I) filter proposed by Sun & Neuvo. In SWM-I, the noise detection mechanism compares the value of the absolute difference between the processed $x(i, j)$ pixel and the $m(i, j)$ median pixel in the filtration window with a predefined TSWM-1 threshold value. This impulse noise detection process is represented by:

$$M(i,j) = \begin{cases} 1 & :|x(i,j) - m(i,j)| > T_{\text{SWM-I}} \\ 0 & :|x(i,j) - m(i,j)| \leq T_{\text{SWM-I}} \end{cases} \quad (2.3)$$

where $M(i, j)$ is the generated noise mask. If the value of the absolute difference exceeds the threshold value (i.e. $M(i,j)$=1), then the processed pixel will be considered as the noise pixel and will need to be restored by the next filtering process. Instead, if $M(i, j)$ = 0, the pixel will be categorized as noise-free and retained. These image recovery terms by SWM-I filter are calculated as:

$$y(i,j) = M(i,j) \cdot m(i,j) + [1 - M(i,j)] \cdot x(i,j) \quad (2.4)$$

By detecting noisy pixels first before performing the filtering process, these SWM-I filters were found to produce better filtering image quality compared to conventional median filters with no concept of switching. However, noise detection based on the difference in intensity values of the metric photo used in the SWM-I was not able to distinguish between the impulses noises present along the fine line. This is because fine lines have significant differences in pixel intensity with impulse.

In an effort to rectify these weaknesses, a number of studies have been focused on developing noise detection methods that are able to detect the presence of high impulse noise. In this regard, the filtering method in the adaptive switching class has been identified as one of the approaches that can filter out high density impulse noise. Among the earliest methods in this class of adaptive median filters is the one proposed by Hwang & Haddad [3]. The basic idea of the proposed filter is to increase the size of the noise detection window $W(i, j)$ (i.e. by increasing the value of N) to a point where the criterion for the extension of the window has been met. Therefore, it will be able to ensure that the number of noise-free pixels in the detector window is always sufficient to make an accurate estimate of the final value of pixel recovery. In addition, such filters are also capable of adapting to a wide variety of noise density levels, whether low or high. Although capable of filtering high-noise, however, the main problem with these adaptive filters is the lack of clear guidelines for the process of expanding the window $W(i, j)$. This problem thus contributes to some other disadvantages such as over-smoothing of the image (over smoothing) and long processing time.

In addition to the standard filtering and adaptive median filtering classes, there are other variations of the median filter in the switching domain that have been introduced to produce good noise filtering quality; such as the method of weighted switching. Examples of filters in this class are the Switching Median II (SWM-II) filters [4]. This filtering mechanism employs a weighting approach that is appropriate to specific pixels within the analysis window in order to control the balance between noise removal and image detail conservation. Weighting refers to the repetition of a pixel several times in the original image data set. For example, the basic concept of SWM-II weighted median filtration is to repeatedly produce the center pixel in the filter window, depending on the set weight. Mathematically, the output for the SWM-II filter is calculated based on:

$$y(i, j) = \mathrm{med}\{x(i - N, j - N), \ldots, w\Delta x(i, j), \ldots, x(i + N, j + N)\} \qquad (2.5)$$

where the Δ operator represents a pixel-repeat operation of 'w' times. Based on this weighted concept, several types of filters in a more sophisticated level have also been tested and highlighted such as Directional Weighted Median (DWM) [8]. For example, the noise filtering process by the DWM filter is performed using a 5 × 5 $W(i, j)$ window along with a set of sensitive noise detection c coordinates to the sides in different orientations. Each of the coordinates in $W(i, j)$ is given a weight; where pixels closer to the center pixel will be given a larger w weight. This set of DWM noise detection coordinates is shown in Figure 2.7. Next the process of classification of noise pixels by DWM filters is done using the equation:

$$Mask(i,j) = \begin{cases} 1: \text{if } r(i,j) > \mathrm{T_{DWM}} \\ 0: \text{if } r(i,j) \le \mathrm{T_{DWM}} \end{cases} \tag{2.6}$$

where

$$r(i,j)=\min\left\{ \sum_{(k,l)\in S_c^0} w(s,t)\left|x(i,j) - x(i+k,j+l)\right| : 1 \le c \le 4 \right\} \tag{2.7}$$

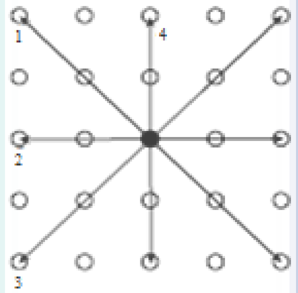


Figure 2.7: DWM filter impulse detection coordinate set.

Although this DWM filter is able to filter noise quite well, but the use of the *c* coordinate set will also cause the DWM filter to be too rigid and unable to detect noise pixels that have a small intensity difference value with the surrounding pixels.

In the meantime, the focus of research has tended towards the development of hybrid switching filters. Most of the existing filters belong to this filtering class. Filters in a hybrid class will often incorporate the characteristics of weighted switching and adaptive filtering methods into their framework. Examples of filters in this class are Tri-state Median (TSM) filters [11]. These TSM filters are formed by a combination of SM filters [35] and CWM [16], In summary, the TSM filter uses a set of two threshold values set in the noise detection stage and its output will be adapted to three possible conditions; replaced with the SM filter output) and the possibility of noise-free pixels (i.e. the pixel value will be replaced by the output from the CWM filter). Then, by making modifications to the median absolute difference (MAD) Crnojevic et al. has produced a filter called the Pixel-wise MAD (PWMAD) filter [36]. Next, Luo has come up with the technique of Efficient Detail-preserving Aproach (EDPA) which is based on alpha-cut mean statistical estimates to effectively filter impulse noise. Meanwhile, Luo has also proposed a filtering technique called Two-stage Efficient Algorithm (TEA) filter [13] in which he uses rank-ordered absolute difference (ROAD)) as well as fuzzy impulse noise detectors to remove noise in digital images. However, these filters are only capable of producing satisfactory performance at an impulse noise density level of less than 30%. On the other hand, there are other techniques such as Decision-based Algorithm (DBA) filter [37] and Open-Close Sequence (OCS) filter [38]. It has demonstrated the ability to filter out impulse noise at higher densities. However, the good performance shown by these DBA and OCS filters is constrained by the loss of fine details on the output image and / or the long processing time. Most recently, based on the idea by Chan et al. (2004) combining variational methods [39] with ACWM filters; Zhang has emerged with a technique called the Functional Minimization Effective Median (FMEM) filter [40]. This technique has two advantages over the previous method proposed by Chan et al. First, the FMEM filter is found to be more efficient because the concept of pixel recovery is simpler. Second, FMEM filters are also more flexible, where they can also be used on color images if adapted to the concept of vector filters [41], [42].

Comparison of performance between median filters in this switching domain can be referenced in Table 2.2. In general, the common problem faced by most of the filtering methods discussed is the inability to filter noise efficiently. Although there are filters that are able to provide good filtering results, it still has to compromise with the problem of missing details or fine details in the image. In addition, the performance of these filters will also decrease significantly as the noise density increases.

Table 2.2: Comparison of image recovery performance by filters in the switching median class.

Filter	**High Density Noise Filtering**	**Edge Preservation**	**Time Efficiency**
SWM-I	4	4	1
ACWM	2	2	4
SWM-II	4	3	1
DWM	2	1	3
TSM	3	2	2
EDPA	2	1	3
TEA	2	1	1
FMEM	1	2	4

Note: Scales [1] - [3] represent [1] Good, [2] Moderate, [3] Satisfactory and [4] Poor.

2.3 Remarks

In short, this chapter 2 discusses several studies on impulse noise filters in the median filtering category. These switching filters have been classified into four different classes, namely the standard switching median filter, the weighted median switch filter, the adaptive switching median filter and the hybrid switching median filter.

Through the observation and study of the various types of conventional switching median filters contained in this chapter, then there are some important formulations that can be stated here. First and foremost is that most conventional noise filters often do not consider the balance between good noise filtering and the complexity of the filtering process when a filter is designed. The objective of some of these conventional filters is to focus on producing good noise image quality, ignoring the complexity of the algorithm and processing time. In fact, in theory a good filter must be able to produce a good noise filter quality with minimal processing time. Second, the conventional filters are also considered less important in terms of preservation of detail and fine lines in the refined image. Both of these issues if properly researched and addressed can help to create a more efficient noise filter.

Based on the statements and observations given, this research has come up with new type of impulse noise filtering techniques. The framework and mechanism of these filters will be discussed in more detail in the next chapter.

Chapter 3

Methodology

3.1 Preliminary

Among the standard requirements in digital image processing is to filter the impulse noise that exists at random without damaging the texture of the image details. In general, linear filtering techniques are seen to be less effective in removing impulse noise in images compared to non-linear filters [43]. The statement has prompted this research to focus on non-linear filtering methods to filter out impulse noise on digital images.

In this research a new switching based median filter called Dual Sliding Statistics Switching Median Filter (DSSSM) has been proposed. The proposed technique is belongs to the class of adaptive switching median filter and it is designed for use of low level random value impulse noise filtering. In this chapter, the proposed DSSSM filtering mechanisms will be described in detail.

3.2 Dual Sliding Statistics Switching Median Filter

Dual sliding switching median (DSSSM) filter is an iterative nonlinear filter which consists of two processing stages. The first stage involves the detection of impulse noise and its location. A noise mask, acting as a classifier to separate the noise pixels from noise-free pixels is generated during this process. In the second stage, all noise-free pixels are left uncorrected while the other noise pixels will be subjected for further processing. At this level, the pixel restoration process is carried out recursively with the assistance of fuzzy based local information. The mechanism of the proposed DSSSM filter is discussed and explicated with more specific in the following subsections.

3.2.1 Stage 1: Impulse Noise Detection

In digital image, the noisy pixel can be characterized by a pixel with the intensity that varies greatly from those of its neighboring pixels. Basically, the intensities of these pixels are represented by a numerical integer. Based on this fact, the impulse detection can be realized by analysing the local image statistics within a window patch. In the beginning of the detection process, the proposed DSSSM filter employs a square local window W(i, j) with odd dimensions $(2N+1) \times (2N+1)$ and is centered at x(i, j). It is given as:

$$W(i,j) = \{x(i+k, j+l)\};\ \text{where } k,l \in (-N,...,0,...,N) \qquad (3.1)$$

All the pixel's elements within $W(i, j)$ are then stored in two separate arrays which represent the sorted statistics and non-sorted statistics, respectively. The process is continued by finding the median pixel *Pmed(i, j)* and central pixel *Pcenter(i, j)*. Both *Pmed(i, j)* and *Pcenter(i, j)* are defined by:

$$P_{med}(i,j) = med\{x(i+k, j+l)\} \qquad (3.2)$$

$$P_{center}(i,j)=x(i,j) \qquad (3.3)$$

Next, the median pixel *Pmed(i, j)* and central pixel *Pcenter(i, j)* are subtracted from all the pixels in $W(i, j)$. This modus operandi will produce two sets of absolute differences arrays, namely *dmed(i+k, j+l)* and *dcenter(i+k, j+l)*. Mathematically, these absolute differences arrays are computed as follows:

$$d_{med}(i+k, j+l)=|x(i+k, j+l)-P_{med}(i,j)|;\ \text{with } k,l \neq 0 \qquad (3.4)$$

$$d_{center}(i+k, j+l)=|x(i+k, j+l)-P_{center}(i,j)|;\ \text{with } k,l \neq 0 \qquad (3.5)$$

At this point, all the values computed in *dmed(i+k, j+l)* and *dcenter(i+k, j+l)* are rearranged in ascending order. After that, the median of absolute differences (i.e. *MADmed* and *MADcenter*) will be identified based on:

$$MAD_{med} = med\{d_{med}(i+k,\ j+l)\} \qquad (3.6)$$

$$MAD_{center} = med\{d_{center}(i+k,\ j+l)\} \qquad (3.7)$$

In order to make a distinction whether current processing pixel is a noise or not, the difference between *MADmed* and *MADcenter* will be first calculated. If the MAD difference is denoted as *diffMAD*, then alternatively *diffMAD* can be written as follows:

$$diffMAD = \left| MAD_{med} - MAD_{center} \right| \tag{3.8}$$

This *diffMAD* provides information about the likelihood of corruption for the current processing pixel. For example, if d*iffMAD* value is large then the current pixel is very likely being contaminated by impulse noise. On the other hand, in the case where *diffMAD* is small, the current pixel may be considered as a noise-free.

After *diffMAD* is counted, a binary noise mask *M(i, j)* will be formed to mark the locations of noise pixels and noise-free pixels. Thus, the process of generating noise mask can be grasped as:

$$M(i,\ j) = \begin{cases} 1, & diffMAD > T^{(t)}{}_{DSSSM} \\ 0, & diffMAD \le T^{(t)}{}_{DSSSM} \end{cases} \tag{3.9}$$

where *M(i, j)*=1 signifies the noise pixel, *M(i, j)*=0 represents the noise-free pixel and $T^{(t)}DSSSM$ actually is the threshold in the t-th iteration.

In step-by-step implementation, the proposed DSSSM algorithm is elucidated as follows:

Step 1: Select a two dimensional local window *W(i, j)* of size 3×3 from the noisy image. (The reason behind the selection of 3×3 window size is based on the fact that larger local window will blur the image's detail and edge).

Step 2: Put all elements within *W(i, j)* in two separate arrays, then identify the median pixel *Pmed(i, j)* and central pixel *Pcenter(i, j)* using Eq. (3.2) and Eq. (3.3), respectively.

Step 3: Compute the absolute difference luminance *dmed(i+k, j+l)* and *dcenter(i+k, j+l)* according to Eq. (3.4) and Eq. (3.5), respectively.

Step 4: Rearrange each value obtained in *dmed(i+k, j+l)* and *dcenter(i+k, j+l)* in ascending order. Then, calculate the median of absolute differences *MADmed* and *MADcenter* based on Eq. (3.6) and Eq. (3.7), respectively.

Step 5: Calculate the absolute MAD difference *diffMAD* based on Eq. (3.8).

Step 6: Compare the absolute *diffMAD* value found in Step 5 with the decision maker threshold T(t)DSSSM and generate the binary mask *M(i, j)* based on Eq. (3.9). (Repeat Step 2 to Step 6 until the entire pixels in the image have been processed)

3.2.2 Stage 2: Impulse Noise Detection

After the binary noise mask is created, the filtering action will replace the noise pixels marked with *M(i, j)*=1 with the estimated correction term. Yet again, the proposed DSSSM filter adopts a square sliding window *Wfilter(i, j)* with *(2Nfilter*+1*)* × *(2Nfilter*+1*)* dimensions. The so called filtering window is given as:

$$W_{filter}(i,\ j) = \{x(i+k,\ j+l)\};\ where\ k,l \in (-N_{filter},...,0,...,N_{filter}) \qquad (3.10)$$

For each detected noise pixel, the median pixel *m(i, j)* is selected from the corresponding region within *Wfilter(i, j)* using:

$$m(i,\ j) = med\,\{x(i+k,\ j+l)\} \qquad (3.11)$$

Then, the local information *Dlocal(i, j)* is extracted from *Wfilter(i, j)* according to:

$$D_{local}(i,\ j) = max\,|\,x(i+k,\ j+l) - x(i,\ j)\,|;\ with\ (i+k,\ j+l) \neq i,\ j \qquad (3.12)$$

As part of the filtering mechanism in the DSSSM filter, fuzzy reasoning is applied to the extracted local information. The fuzzy set will handle any uncertainties present in the local information due to the nonlinear nature of impulse noise corrupting an image. This adopted fuzzy set *f(i, j)* is defined by:

$$f(i,\ j) = \begin{cases} 0 & : D_{local}(i,\ j) < T_1 \\ \dfrac{D_{local}(i,\ j) - T_1}{T_2 - T_1} & : T_1 \leq D_{local}(i,\ j) < T \\ 1 & : D_{local}(i,\ j) \geq T_2 \end{cases} \qquad (3.13)$$

where T_1 and T_2 are two empirically predefined threshold with the value 10 and 30 respectively. Finally, the correction term $y_{DSSSM}(i, j)$ is used to replace the detected noise pixel. It will utilize the median pixel mi, j and fuzzy set *f(i, j)* obtained in two previous processes. The correction term is computed as follows:

$$y_{DSSSM}(i,\ j) = [1 - f(i,\ j)] \cdot x(i,\ j) + f(i,\ j) \cdot m(i,\ j) \qquad (3.14)$$

As to enhance the filter's ability in selecting a more accurate median pixel, this algorithm is implemented in recursive manner. An illustrative example of the impulse noise detection and filtering operation against an image with the corruption rate of 10% is shown in Figure 3.1.

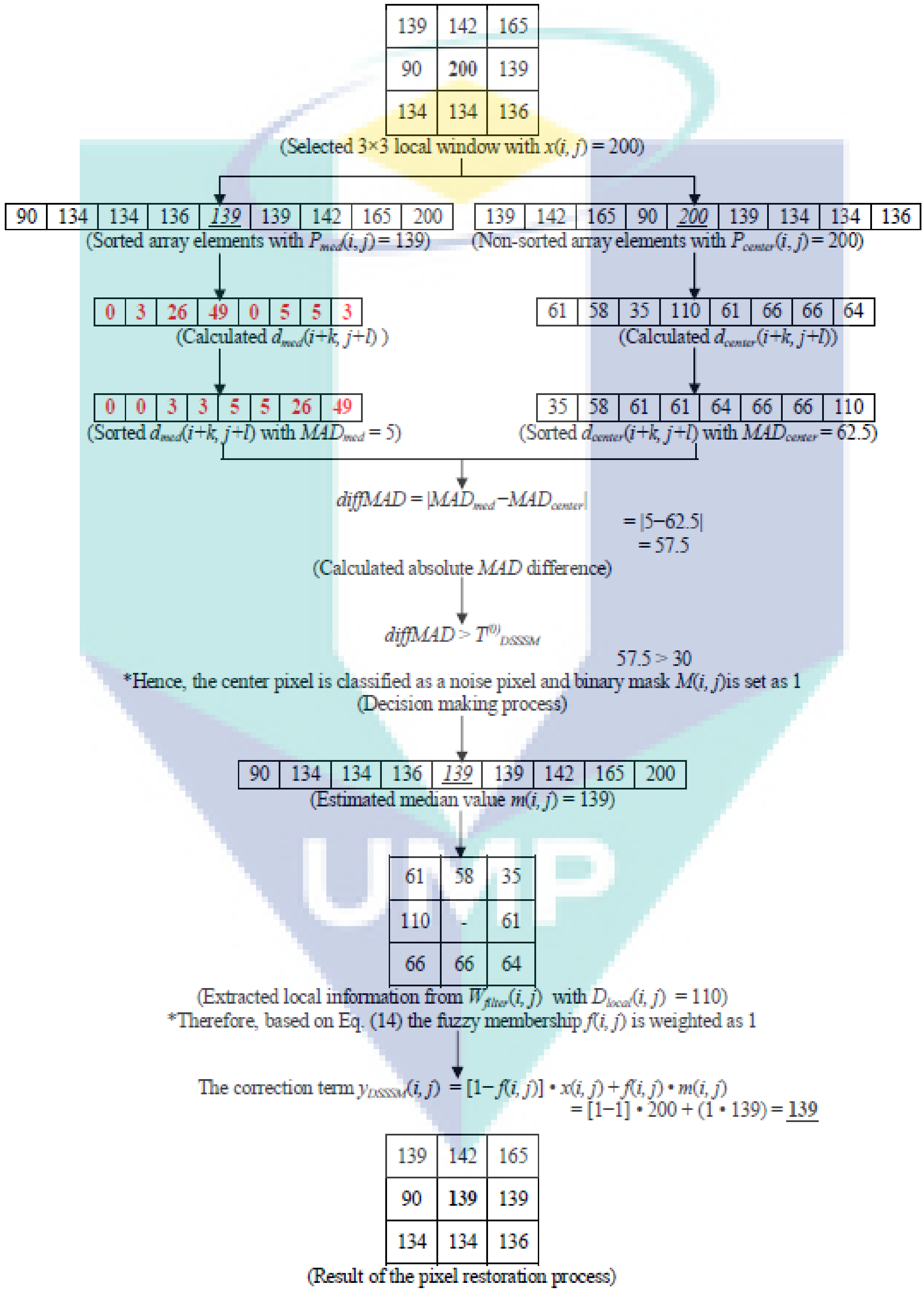


Figure 3.1: The illustrated example on the DSSSM noise filtering operation.

Chapter 4

Results and Discussions

4.1 Introduction

In this chapter, the performance of the proposed DSSSM filter will be compared to other related state-of-the-art impulse noise filters based on their simulation results. A total of 80 standard test images of size 512×512, obtained from diverse online sources were used for the simulations of each implemented filters. Each of the test images was corrupted with the impulse noise model described in (1), ranging from 5% to 30% with an increment of 5%. This set of standard test images contains various characteristics which are suitable to assess the robustness of the implemented filters.

For comparison, the following conventional impulse noise filters with their suggested tuning parameters were used to restore the contaminated test images. Among them are SWM-I (*Ti*=50), SWM-II (*Ti*=30, w=3), TSM (*T*=20, *w*=3), DWM (*wm*=2, T0=510, *Nmax*=[5, 10] and 5×5 window size), LUO (*LD*=1, *u*=3, *W1*=5, *W2*=30) and ACWM ([*δ0*, *δ1*, *δ2*, *δ3*]=[40, 25, 10, 5], *s*=0.6). Finally, the simulation results of the implemented impulse noise filters will be interpreted objectively and subjectively as to judge the filters effectiveness in removing random-valued impulse noise.

4.2 Qualitative Analysis

Providing visually pleasing output is imperative, since the image is eventually to be viewed by human eye. Therefore, the appearances of the resultant images are inspected visually to judge the filters efficiencies in reducing impulse noise effect and producing good image quality. Among the 80 test images, the simulation results for three commonly used grayscale test images Starfish, Boat and Goldhill are presented in Figure 4.1, Figure

4.2 and Figure 4.3, respectively. In each figure, portion (b) represents the noise corrupted images with 10%, 20% and 30% density of impulse noise.

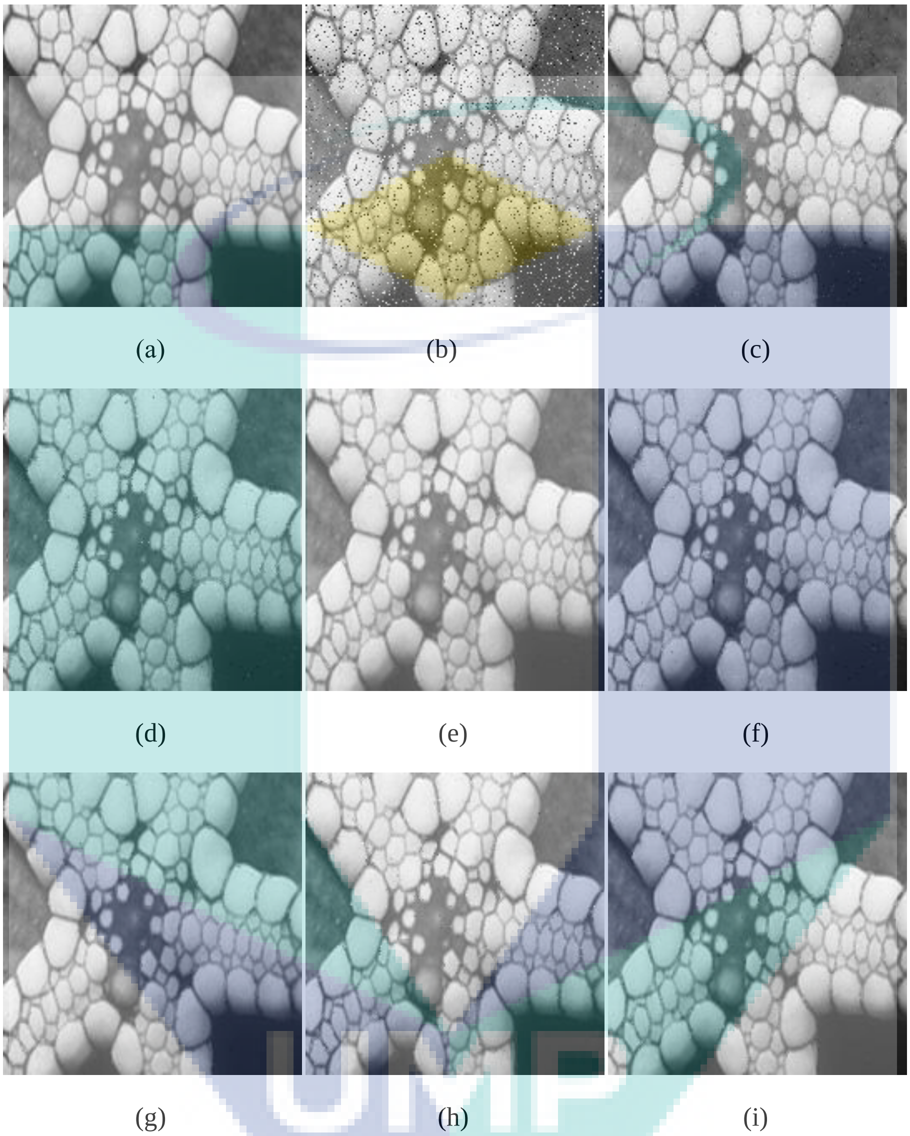
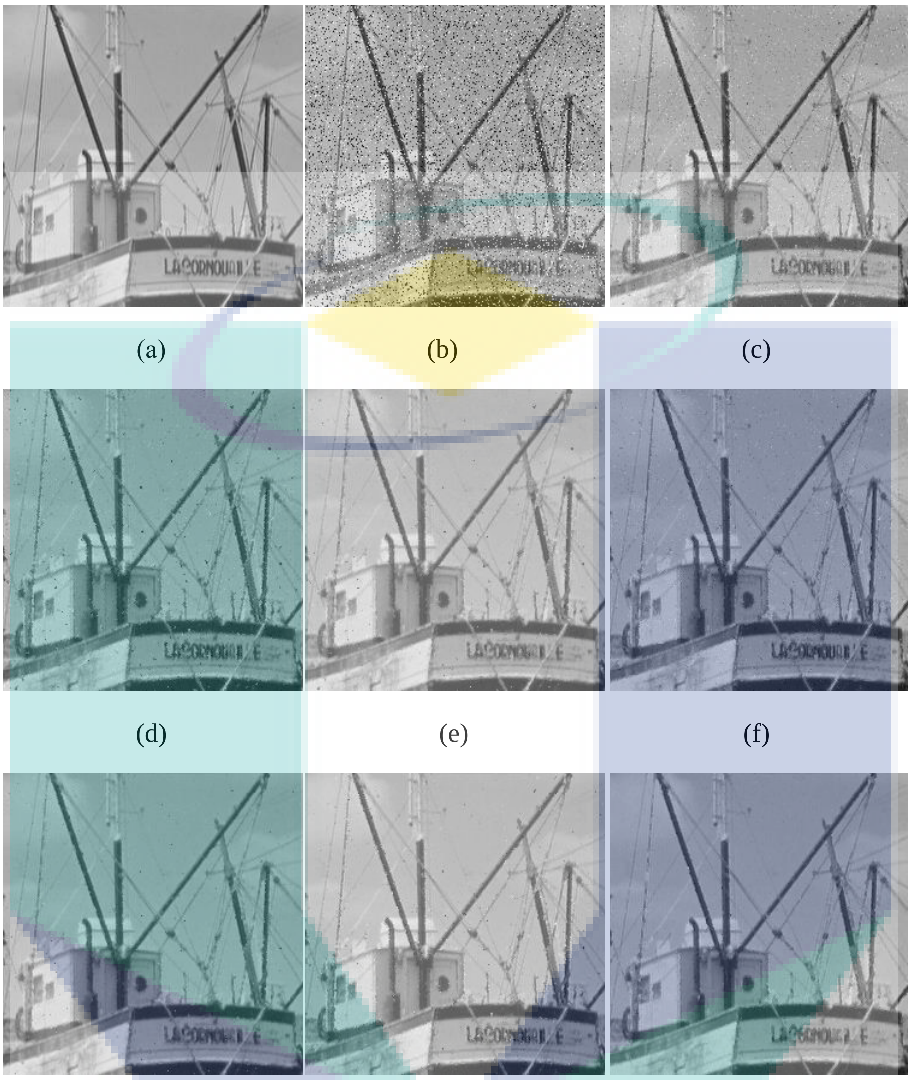


Figure 4.1: Simulation results on a portion of *Starfish* using; (a) original image, (b) noisy image with 10% density of impulse noise, (c) SWM-I, (d) SWM-II, (e) TSM, (f) DWM, (g) LUO, (h) ACWM and (i) DSSSM.

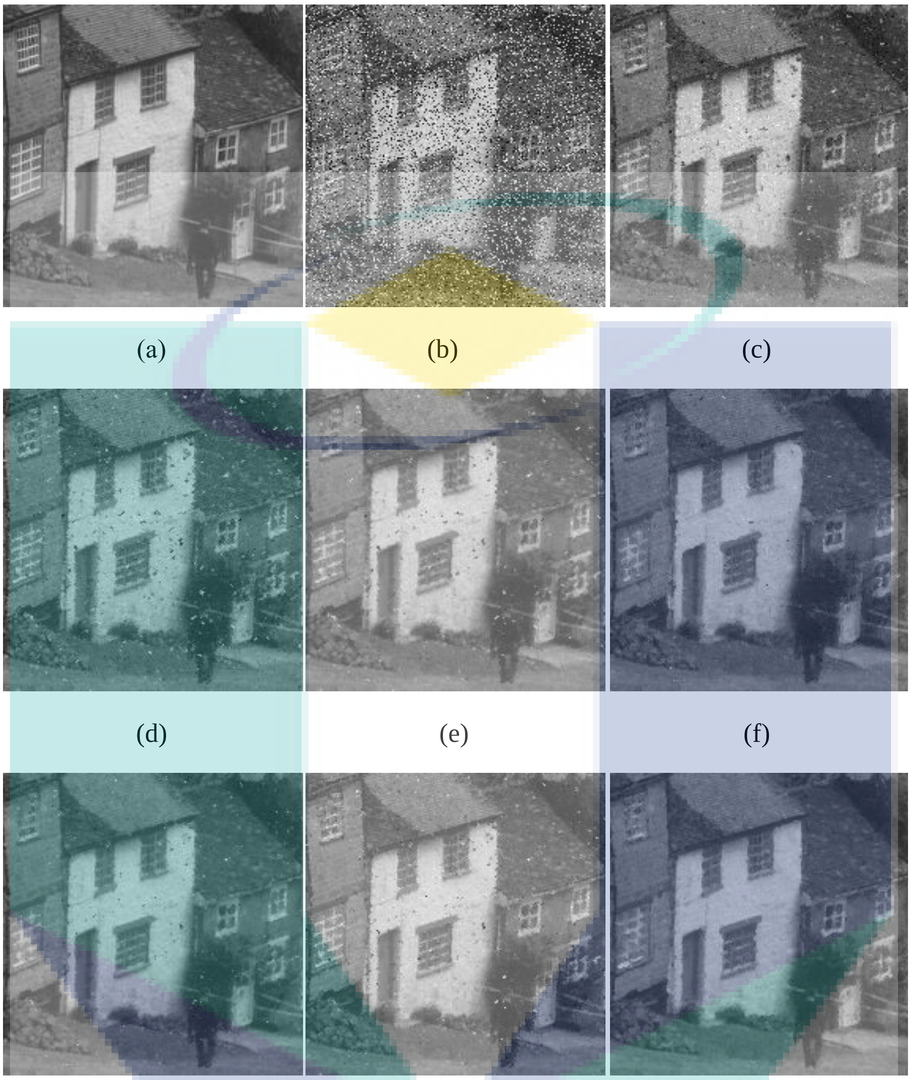


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.2: Simulation results on a portion of *Boat* using; (a) original image, (b) noisy image with 20% density of impulse noise, (c) SWM-I, (d) SWM-II, (e) TSM, (f) DWM, (g) LUO, (h) ACWM and (i) DSSSM.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 4.3: Simulation results on a portion of *Goldhill* using; (a) original image, (b) noisy image with 20% density of impulse noise, (c) SWM-I, (d) SWM-II, (e) TSM, (f) DWM, (g) LUO, (h) ACWM and (i) DSSSM.

As can be seen in Starfish, at 10% impulse noise density, the noise filtering performance of DSSSM filter is basically similar to those of the conventional noise filtering algorithms. All filters are found to be able of producing perceptible reconstructed image since the density of impulse noise is still low and not form any noise patches at this level.

However, in Boat which is contaminated with 20% of impulse noise (as shown in Figure 4.2), it can be visualized that the results produced by the conventional median filters are still influenced by the noise. It is able to notice that some small noise patches are remained intact on the resultant images. On contrary, the proposed DSSSM filtering algorithm can significantly remove the effect of noise added to the images and at the same time preserve the object shapes.

The similar results are obtained for the image named Goldhill; where the proposed DSSSM filtering algorithm outperforms the conventional SWM-I, SWM-II, TSM, DWM, LUO and ACWM algorithms by giving clearer filtered image even the density of noise in this image is higher (i.e. 30% of impulse noise). The proposed DSSSM filter has successfully reduced the noise particles and noise patches, consequently created less corrupted image. This observation indicates that the combination of the adaptive impulse noise detection concept with the fuzzy based local information significantly helps the proposed DSSMM filter to dexterously reduce the noise stain. Meanwhile, there is a great deal of noise contamination existed in the images produced by the conventional filtering techniques. The poor restoration results among these conventional filters can be attributed from their impulse noise filtering mechanisms which are less robust towards the contamination of random-valued impulse noise.

4.3 Quantitative Analysis

In addition to the visual inspection of the restored images, the quality of the restored images is also evaluated quantitatively using the peak signal-to-noise ratio (PSNR). Mathematically, the PSNR for a digital image of the dimension M×N is defined as:

.

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right)(dB) \tag{4.1}$$

For the above formulae, MSE stands for the mean-squared error and it is given as:

$$MSE = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\frac{\left[o_{i,j} - y_{i,j}\right]^2}{M \times N} \tag{4.2}$$

where *yi, j* is the filtered image and *oi, j* is the original noise-free image. Apart from the PSNR assessment, the mean of absolute error (MAE) has also been used in this analysis to characterize the filter’s detail preservation behavior, one which is defined by:

$$MAE = \frac{1}{M \times N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}| o_{i,j} - y_{i,j} | \tag{4.3}$$

The numerical results for Figures 2(c)−(i), Figures 3(c)−(i) and Figures 4(c)−(i) are tabulated in Table 5. From the functions, the larger PSNR and smaller MAE values show better restoration results. In all tables, the best results obtained are made bold.

As reported in Table 4.1, except for the 10% impulse noise of *Boat* and *Goldhill*, the proposed DSSSM filtering technique consistently yields the highest PSNR values compared to the other existing conventional filters. Even though the PSNR values of ACWM filter in Boat and *Goldhill* are slightly higher than DSSSM filter when *r* =10%, but in terms of visual evaluation in Figure 4(h) and 4(i), no significant difference between the two images are observed.

Table 4.1: Comparison of PSNR on Different Noise Level Restoration for 'Starfish', 'Boat' and 'Goldhill' (Test Image).

Images	Algorithms	PSNR(dB) 10%	20%	30%
Starfish	SWM-I	31.6219	27.8273	24.7745
	SWM-II	33.056	27.8818	23.8052
	TSM	35.0013	30.2872	26.1675
	DWM	32.6523	29.6524	27.9957
	LUO	35.1372	31.4958	28.0056
	ACWM	35.3272	30.6981	26.55
	DSSSM	**35.3753**	**32.7072**	**30.5841**
Boat	SWM-I	31.1459	27.7627	24.866
	SWM-II	32.4632	28.0734	24.1192
	TSM	33.6079	30.1461	26.3557
	DWM	32.5794	29.4737	27.6844
	LUO	34.0311	31.2224	27.9866
	ACWM	**34.4813**	30.5946	26.6566
	DSSSM	34.2323	**31.9472**	**30.0266**
Goldhill	SWM-I	31.3933	27.9749	25.0891
	SWM-II	33.1147	28.4592	24.4252
	TSM	34.1815	30.6761	26.7165
	DWM	33.3963	30.0995	29.0558
	LUO	34.8439	32.0345	28.6311
	ACWM	**35.0158**	31.1301	27.0137
	DSSSM	34.4235	**32.4756**	**30.8526**

Table 4.2: Comparison of MAE on Different Noise Level Restoration for 'Starfish', 'Boat' and 'Goldhill' (Test Image)

Images	Algorithms	PSNR(dB)		
		10%	20%	30%
Starfish	SWM-I	1.37569	3.01962	5.21482
	SWM-II	1.16153	2.82004	5.49509
	TSM	0.925304	2.02381	3.84945
	DWM	1.17076	2.3494	3.52177
	LUO	0.983086	1.88244	3.21782
	ACWM	**0.759693**	1.78849	3.48141
	DSSSM	0.826065	**1.58328**	**2.49641**
Boat	SWM-I	1.46273	3.05105	5.19042
	SWM-II	1.2075	2.7313	5.27398
	TSM	1.13034	2.14042	3.87268
	DWM	1.2042	2.42207	3.66792
	LUO	1.18715	2.03902	3.35503
	ACWM	**0.836605**	1.80618	3.43848
	DSSSM	0.955929	**1.71384**	**2.5858**
Goldhill	SWM-I	1.46043	3.07207	5.17048
	SWM-II	1.19814	2.73936	5.21354
	TSM	1.12705	2.1397	3.82618
	DWM	1.13602	2.33179	3.29423
	LUO	1.19761	2.03074	3.26606
	ACWM	**0.909317**	1.89498	3.5232
	DSSSM	1.01952	**1.77089**	**2.61645**

Meanwhile, the similar phenomenon is occurred in the analysis outlined in Table 4.2, where the proposed DSSSM filter without fail outclasses the other filters in comparison by producing the best MAE results for the cases of 20% and 30% impulse noise density. These findings strongly support the qualitative results in Section 4.1. On the contrary, at 10% of impulse noise density, it can be observed that ACWM gives the better MAE results as compared to our proposed filter. However the performances of these two algorithms in terms of the details preservation based on the visual perception are fairly similar (e.g. see Figure 4.1).

This study has further calculated the average PSNR for 80 tested images and the results are displayed in the graph shown in Figure 4.4. It could be obviously observed that, for such a mild corruption rate (i.e. noise density of 15% and above), the average PSNR curve of the proposed DSSSM filter has the highest curve as compared to the rest of the filters implemented. In the meantime, ACWM filter initially has a relatively higher average PSNR than the DSSSM filter; but it start to drop dramatically with the increased in noise density beginning from 15% impulse noise. Above 20% impulse noise density, the two worst performing filters are the SWM-I and SWM-II filters.

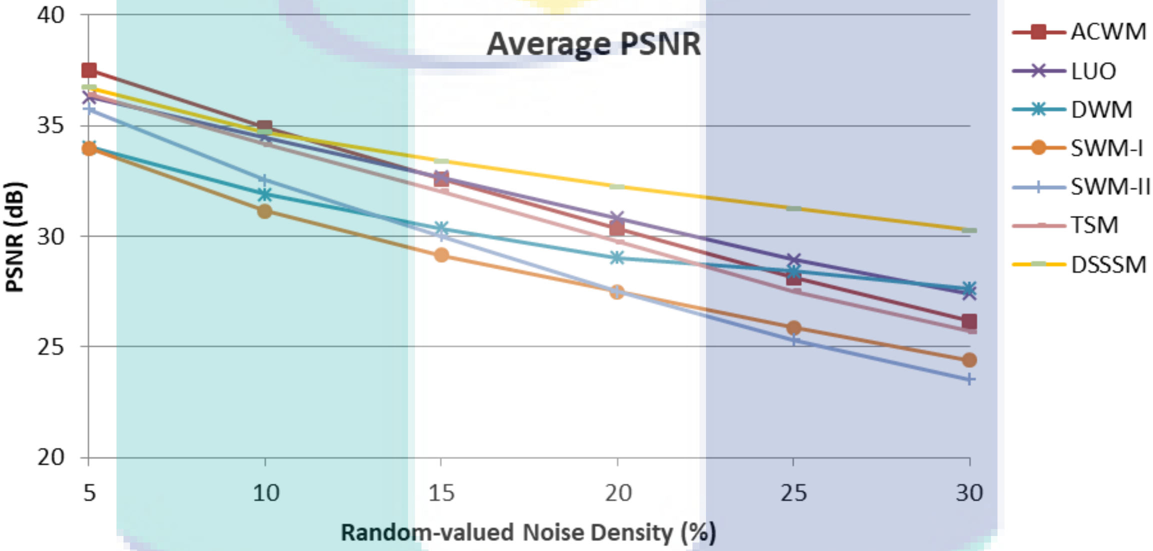


Figure 4.4: The graph of an average PSNR based on different noise level restorations for 80 standard grayscale test images.

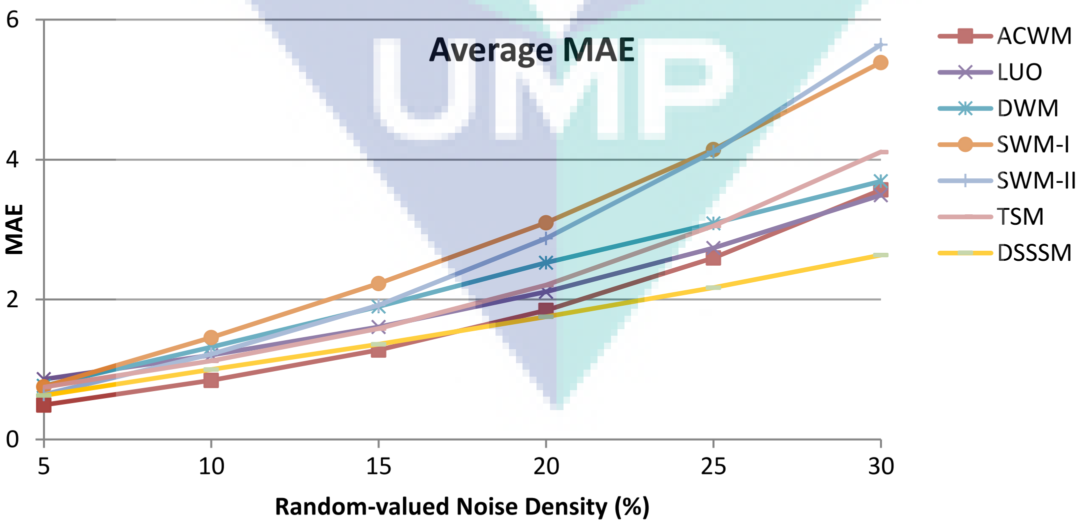


Figure 4.5: The graph of an average MAE based on different noise level restorations for 80 standard grayscale test images.

In addition, Figure 4.5 shows the extrapolation of the average MAE curves obtained using the various conventional filters in comparison and the proposed DSSSM filter. From the plot, it is noticed that all the filters do perform well at the extremely low density of noise (i.e. 5% impulse noise). Nonetheless, only the proposed DSSSM filter managed to survive at all level of noise densities with the MAE is slightly and gradually increased. Other filters, such as ACWM and LUO for instance, even though are able to produce relatively good details preservation results at the beginning but their MAE values have sharply increased; particularly when the noise level is above 20%. On the other hand, the SWM-I and SWM-II filters have completely failed to compete in this test. Once again, this finding shows that the DSSSM filter is not only able to eliminate noise efficiently but it can also preserve the original appearance and shape of an image very well.

4.4 Runtime Efficiency

We had also conducted a processing time analysis for each filter for the aim of performing its denoising task. All the algorithms were implemented in C-languange and the simulations were carried out using a personal computer with AMD Athlon II 2.1GHz processor, 2GB of RAM. The graph of average processing time in milliseconds for 80 standard test images after applying with the proposed DSSSM filter and other conventional filters is provided in Figure 4.6.

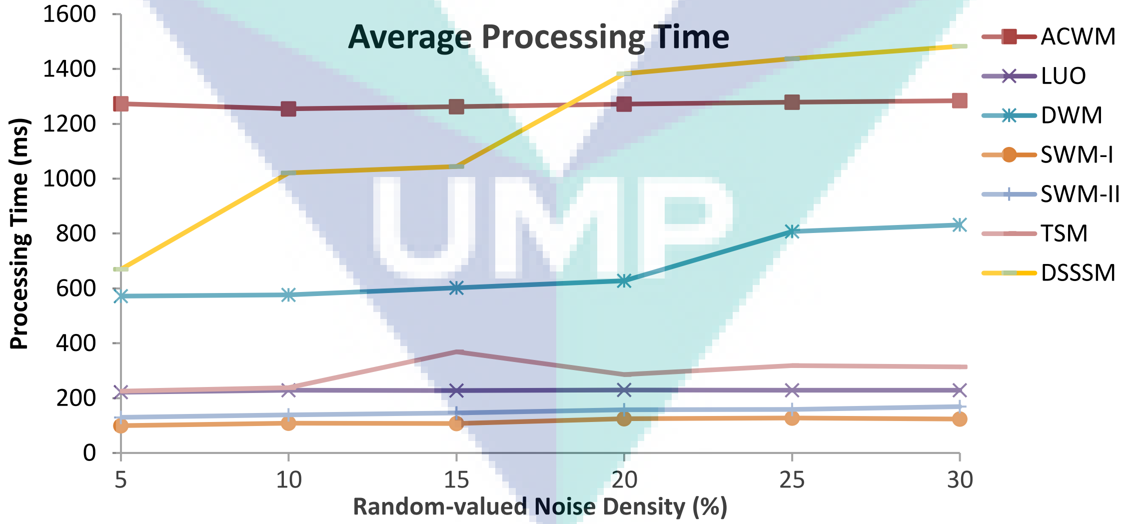


Figure 4.6: The graph of an average processing time in milliseconds (ms) versus impulse noise density (%) computed from a total of 80 standard grayscale test images

This figure shows that the processing time required by SWM-I, SWM-II, LUO and ACWM are almost constant, regardless to the noise density; with SWM-I filter has the lowest average processing time. Unfortunately, the filtering quality of these conventional filters are poor and perceptibly degraded. On the other hand, it is clearly seen that the ACWM filter consumes the highest processing time among all conventional filters. This is due to the filter's detection mechanism which is applied in four iterations. In the meantime, our DSSSM filter's processing time is seen to increase according to the level of noise and is found a bit higher than ACWM especially when the percentage of the impulse density is above 20%. The reason behind this trend is because the proposed DSSSM filter uses a set of two separate arrays to process the local image statistics and its filtering procedure is carried out iteratively. However, the slightly higher processing time is compensated by the better filtering results. As observed in the two previous subsections, only the DSSSM filter is able to produce a good and comprehensible filtered images at all level of noise. Thus, as far as the denoising performance is concerned, the proposed DSSSM can consequently be regarded as the best filter.

Chapter 5

Conclusion

Noise filtering is an art and science to improve the quality of digital images damaged by noise. Typically, the noise filtering process framework will involve several methods or image processing techniques implemented mathematically modeling. One of the noise models that has attracted the attention of most researchers today is the impulse noise of random value. This is because the presence of random-value impulse noise will not only tarnish the visual quality of an image and damage the original information contained in it; in fact, the filtering process is more challenging compared to fixed-value impulse noise (i.e. salt and pepper noise) because the intensity value of this type of noise pixel is almost the same as other pixels around it.

Throughout this study, an effective algorithm for the detection and suppression of random-valued impulse noise have been introduced. The proposed DSSSM filter is constructed by incorporating a robust impulse noise detection based on adaptive thresholding and recursive pixel restoration technique. Additionally, fuzzy reasoning set is embedded as part of the filtering mechanism in order to handle any imprecise local information. Extensive simulation results reveal that the DSSSM filter is able to reduce the random-valued impulse noise effect, while at the same time preserving the details and structures of fine images. Furthermore, its filtering performance is tremendously consistent all the time as compared to the number of well-known conventional techniques; and all these good results are achieved with a fairly efficient processing time.

References

1. W. Luo. Efficient removal of impulse noise from digital images. IEEE Trans. Consumer Electronics, 52 (2) (2006) 523-527.
2. R. C. Gonzalez and R. E. Woods. Digital image processing. New York: Addison-Wesley, (1992).
3. H. Hwang and R. A. Haddad. Adaptive Median Filters: New Algorithms and Results. IEEE Transactions on Image Processing, 4 (4) (1995) 499-502.
4. T. Sun and Y. Neuvo. Detail-Preserving Median Based Filters in Image Processing. Pattern Recognition Letters,15 (1994) 341–347.
5. T. Chen and H. R. Wu. Space variant median filters for the restoration of impulse noise corrupted images. IEEE Trans. on Circuits and Systems II, 48 (2001) 784–789.
6. S. Zhang and M. A. Karim. A new impulse detector for switching median filters. IEEE Signal Processing Letters, 9 (11) (2002) 360-363.
7. I. Aizenberg and C. Butakoff. Effective impulse detector based on rank-order criteria. IEEE Signal Process. Letters, 3 (11) (2004) 363-366.
8. Y. Dong and S. Xu. A new directional weighted median filter for removal of random-valued impulse noise. IEEE Signal Processing Letters, 14 (3) (2007) 193-196.
9. H. H. Chou and L. Y Hsu. A noise-ranking switching filter for images with general fixed impulse noises. Signal Processing, 6 (2015) 198-208.
10. T. Chen and H. R. Wu. Adaptive impulse detection using center-weighted median filter. IEEE Signal Processing Letters, 8 (1) (2001) 1-3.
11. Chen, K. K. Ma and L. H. Chen. Tri-state median filter for image denoising. IEEE Transactions on Image Processing, 8 (12) (1998) 1834-1838.
12. J. K. Sung, H. L. Yong. Center weighted median filters and their applications to image enhancement. IEEE Transactions on Circuits and Systems, 38 (9) (1991) 984-983.
13. W. Luo. An efficient algorithm for the removal of impulse noise from corrupted images. AEU-Int. J. Electron. Commun., 61 (2007) 551-555.
14. K. K. V. Toh and N. A. Mat Isa. Cluster-based adaptive fuzzy switching median filter for universal impulse noise reduction. IEEE Trans. Consumer Electronics, 56

(4) (2010) 2560-2568.

15. http://homepages.inf.ed.ac.uk/rbf/HIPR2/median.htm (accessed on April 2018).
16. Majid, A., Lee, CH., Mahmood, M.T. et al. Impulse noise filtering based on noise-free pixels using genetic programming. Knowl Inf Syst (2012) 32: 505.
17. Xu, Z., Wu, H.R., Yu, X. et al. Adaptive progressive filter to remove impulse noise in highly corrupted color images SIViP (2013) 7: 817.
18. Ilke Turkmen. A new method to remove randomvalued impulse noise in images, Int. J. Electron. Commun. (AEU) 67 (2013) 771-779.
19. M. S. Darus, S. N. Sulaiman, I. S. Isa et al. Modified hybrid median filter for removal of low density random-valued impulse noise in images. IEEE International Conference on Control System, Computing and Engineering, 25–27 November 2016.
20. M. H Suid and N. A. M. Isa. Scientific Research and Essays, 7(7), (2012) 813-823
21. Jahne, B., (1995) Digital image processing. Springer Verlag, Berlin German.
22. Tomasi, C. and Manduchi, R., (1998) Bilateral filtering for gray and color images. Proceeding of the IEEE International Conf. of Computer Vision, pp 836-846.
23. Comaniciu, D. and Meer, P., (2002) Mean shift: A robust approach toward feature space analysis. IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pp. 1-18.
24. Garnett, R., Timothy, H. and Charles, C. (2005) A universal noise removal algorithm with an impulse detector. IEEE Trans. on Image Processing, vol. 14, no. 11, pp. 1747-1754.
25. Fisher, R., Perkins, S., Walker, A. and Wolfart, E., (2004) Median Filter [Online]. [Accessed on 18 Febuari 2014]. Available from World Wide Web: http://www.homepages.inf.ed.ac.uk/rbf/HIPR2/median.htm
26. Wang, Z. and Zhang, D., (1999) Progressive switching median filter for the removal of impulse noise from highly corrupted images. IEEE Trans. Circuits and System- II: Analog and Digital Signal Processing, vol. 46, no. 1, pp. 78-80.
27. Arce, G. R. and McLoughlin, M. P., (1987) Theoretical analysis of max/median filters. IEEE Trans. Acoust ., Speech, Signal Processing, vol. 35, pp. 60-69.
28. Nieminen, A., Heinonen, P. and Neuvo, Y., (1987) A new class of detail preserving filters for image processing," IEEE Trans. Pattern Anal. Mach. Intell., vol. 9, pp. 74-

90.

29. Lin, H. M. and Willson, Jr. A. N., (1988) Median filters with adaptive length. IEEE Trans. Circuits System, vol. 35, pp. 675-690.

30. Bernstein, R., (1987) Adaptive nonlinear filters for simultaneous removal of different kinds of noise in images. IEEE Trans. Circuits Syst., vol. 34, pp. 1275-1291.

31. Ko, S. J. and Lee, Y. H., (1991) Center weighted median filters and their applications to image enhancement. IEEE Trans. Circuits Syst., vol. 38, no. 9, pp. 984–993.

32. Loupas, T., McDicken, W. N. and Allan, P. L., (1989) An adaptive weighted median filter for speckle suppression in medical ultrasonic images. IEEE Trans. Circuits Syst., vol. 36.

33. Luo, W., (2006) Efficient removal of impulse noise from digital images. IEEE Trans. Consumer Electronics, vol. 52, no. 2, pp. 523-527.

34. Ibrahim, H., Kong, N. S. P. and Ng, T. F., (2008) Simple adaptive median filter for the removal of impulse noise from highly corrupted images. IEEE Trans. Consumer Electronics, vol. 54, no. 4, pp. 1920-1927.

35. Astola, J. and Kuosmanen, P., (1997) Fundamentals of Nonlinear Digital Filtering.Boca Raton, FL; CRC Press, NewYork.

36. Crnojevic, V., Senk, V. and Trpovski, Z., (2004) Advanced impulse detection based on pixel-wise MAD. IEEE Signal Processing Letters, vol. 11, no. 7, pp. 589-592.

37. Srinivasan, K. S. and Ebenezer, D., (2007) A new fast and efficient decision-based algorithm for removal of high-density impulse noise. IEEE Signal Processing Letters, vol. 14, no. 3, pp. 189-192.

38. Deng, Z. F., Yin, Z. P. and Xiong, Y. L., (2007) High probability impulse noise-removing algorithm based on mathematical morphology. IEEE Signal Processing Letters, vol. 14, no. 1, pp. 31-34.

39. Nikolova, M., (2004) A variational approach to remove outliers and impulse noise. J. Math. Imaging Vision, vol. 20, pp. 99–120.

40. Zhang, J., (2010) An efficient median filter based method for removing random-valued impulse noise. Digital Signal Process., vol. 20, no. 4, pp. 1010-1018.

41. Lukac, R., Smolka, B., Plataniotis, K. N. and Venetsanopoulos, A. N., (2005a) A statistically-switched adaptive vector median filter, J. Intell. Robotic Syst., vol. 42, pp. 361-391.

42. Lukac, R., Smolka, B., Martin, K., Plataniotis, K. N. and Venetsanopoulos, A. N., (2005b) Vector filtering for color imaging. IEEE Signal Process. Mag., Special Issue on Color Image Processing, vol. 22, pp. 74–86.

43. Gallegos-Funes, F. J., Ponomaryov, V.I., Sadovnychiy, S., Nino-de-Rivera, L., (2002) Median M-type K-nearest neighbour (MM-KNN) filter to remove impulse noise from corrupted images. IEEE Electronics Letters, vol. 38, no. 15, pp. 786-787.

APPENDIX A - PUBLICATIONS

J Electr Eng Technol.2018; 13(3): 1383-1391
http://doi.org/10.5370/JEET.2018.13.3.1383

ISSN(Print) 1975-0102
ISSN(Online) 2093-7423

Dual Sliding Statistics Switching Median Filter for the Removal of Low Level Random-Valued Impulse Noise

M. H. Suid†, M. F. M. Jusof* and M. A. Ahmad*

Abstract – A new nonlinear filtering algorithm for effectively denoising images corrupted by the random-valued impulse noise, called dual sliding statistics switching median (DSSSM) filter is presented in this paper. The proposed DSSSM filter is made up of two subunits; i.e. Impulse noise detection and noise filtering. Initially, the impulse noise detection stage of DSSSM algorithm begins by processing the statistics of a localized detection window in sorted order and non-sorted order, simultaneously. Next, the median of absolute difference (MAD) obtained from both sorted statistics and non-sorted statistics will be further processed in order to classify any possible noise pixels. Subsequently, the filtering stage will replace the detected noise pixels with the estimated median value of the surrounding pixels. In addition, fuzzy based local information is used in the filtering stage to help the filter preserves the edges and details. Extensive simulations results conducted on gray scale images indicate that the DSSSM filter performs significantly better than a number of well-known impulse noise filters existing in literature in terms of noise suppression and detail preservation; with as much as 30% impulse noise corruption rate. Finally, this DSSSM filter is algorithmically simple and suitable to be implemented for electronic imaging products.

Keywords: Image processing, Random-valued impulse noise, Digital image, Nonlinear noise filtering.

1. Introduction

In the era of multimedia technology, the use of digital image-based visual information has gained a lot of attention due to its flexibility and this phenomenon is expected to continue growing. Medical imaging diagnosis and geographical analysis are among modern daily life applications which have adopted digital image processing technology. In general, these applications involve numerous image processing operations (e.g. image segmentation, edge detection, classification, etc.) of which are highly dependent on the quality of digital input images in order for them to work perfectly. Unfortunately, digital images are frequently subjected to the contamination of impulse noise that typically due to the interferences generated during transmission/acquisition or storage through electronic medium [1]. Therefore, it is imperative to remove the impulse noise effect before any subsequent image processing operations can be carried out as the occurrences of impulse noise can severely damage the information in the original image. One of the most effective approaches to cater for the occurrence of impulse noise and for the improvement of the quality of the acquired image is by using denoising-based algorithm. Accordingly, a large number of nonlinear filters have been widely exploited to remove the impulse noise as they are generally more superior to linear filtering techniques. For instance, standard median (SM) filter [2] and adaptive median (AM) filter [3] are two of the most basic nonlinear filtering techniques for suppressing impulse noise. Ironically, this SM is implemented unconditionally across the image while its variants (e.g. see AM) inherited this clumsy smoothing property; thus they tend to modify both noise and noise-free pixels simultaneously. Consequently, the detailed regions such as object edges and fine textures in image are smeared and appear blurry or jittered.

To get rid of the problem, various filters under switching scheme have been studied and experimented by a number of recently published works; such as switching median filter I and II (SWM-I and SWM-II) [4], multi-state median (MSM) filter [5], Laplacian switching median (LSM) filter [6], enhanced rank impulse detector (ERID) [7], directional weighted median (DWM) filter [8] and noise-ranking switching filter (NRSF) [9], etc. Basically, this filtering scheme divides its implementation into two stages; which are impulse noise detection stage and impulse noise filtering stage. Impulse noise detection algorithm is implemented prior to the filtering process in order to determine whether a pixel should be modified or left unchanged. With this kind of filtering properties, these techniques are shown to be more effective to preserve most of the image details compared to the conventional non-switching techniques.

In a different way, Chen and Wu [10] have come out with the technique based on the adaptive switching scheme called adaptive center-weighted median (ACWM) filter. Briefly, ACWM is a two phase iterative median filter

† Corresponding Author: Faculty of Electrical and Electronic Engineering, Universiti Malaysia Pahang, Malaysia. (mhelmi@ump.edu.my)
* Faculty of Electrical and Electronic Engineering, Universiti Malaysia Pahang, Malaysia. (mfalfazli@ump.edu.my)
Received: April 28, 2016; Accepted: January 22, 2018

which uses a fixed processing window size with adaptively filtering process. Apart from the switching scheme and the adaptive switching scheme, the hybrid switching scheme is another class of filters which has been groomed to yield good filtering results. Many researchers have embedded other order statistics (e.g. rank-order statistic, median of absolute deviations, etc.) and image processing techniques (e.g. mathematical morphology, directional or edge detection, etc.) into the hybrid switching scheme filters as part of its filtering mechanism. One of the techniques in this filtering scheme is the work done by Chen et al. [11]. In general, this tri-state median (TSM) filter is formed by a combination of SM and center weighted median (CWM) filter [12]. It uses a set of two predefined thresholds to determine whether the original pixel should be retained or replaced by the SM filtered output or the CWM filtered output. Meanwhile, a more sophisticated filtering technique has been presented by Luo in [13]. This filter incorporates the rank order absolute difference (ROAD) statistics with fuzzy impulse detection algorithm to classify and remove impulse noise from corrupted images. Noticeably, the restoration abilities of those aforementioned techniques are improved but at the cost of lost fine image details and increased complexity.

Of late, in accordance with the evolution in digital image acquisition technologies, the corruption rate of impulse noise in digital images has managed to be reduced to the level that may be regarded as low; i.e. less than 30% noise density [14-17]. Based on the aforementioned statements and observations; hence our aim in this paper is to develop an efficient filtering technique with a reasonable processing time, particularly for the range of low level impulse noise. Towards this, we introduce a new iterative and recursive filter known as dual sliding statistics switching median (DSSSM). This proposed filter is relatively fast and can remove the impulse noise dexterously without jeopardizing the details and textures inside the image.

The organization of this paper is as follows. Section 2 discusses on the impulse noise model. The design of the proposed filter is described in Section 3. Simulations and experimental results are presented in Section 4. Finally a brief conclusion is drawn in Section 5.

2. Impulse Noise Model

Theoretically, impulse noise contaminates an image with a random amplitude which could either fall within the image dynamic range (i.e. random-valued impulse noise) or out of the range (i.e. salt-and-pepper noise), and usually only certain percentage of pixels are affected. In this work, we tend to focus on the random-valued impulse noise and the model of this impulse noise is described for clarity. For detail, *let* $x(i, j)$ and $o(i, j)$ be the gray level of the noisy image and the original image at location (i, j), respectively. Then, the impulse noise model with noise density r can be defined as:

$$x(i,j) = \begin{cases} n(i,j) & \text{: with probability } r \\ o(i,j) & \text{: with probability } 1 - r \end{cases} \tag{1}$$

where $n(i, j)$ is the noise pixel value independent from $o(i, j)$. The image is said to be corrupted by the random-valued impulse noise when $n(i, j)$ uniformly distributed within the image dynamic range, i.e. $n(i, j)\in$[Nmin, Nmax]. For example, in an 8-bit gray scale image with 256 gray levels, the $n(i, j)$ may range from 0 (Nmin) to 255 (Nmax).

In practical, identifying this noise is more challenging compared to the salt-and-pepper noise because the intensity of noisy pixel is very similar to its surrounding [18-20].

3. Dual Sliding Statistics Switching Median Filter

Dual sliding switching median (DSSSM) filter is an iterative nonlinear filter which consists of two processing stages. The first stage involves the detection of impulse noise and its location. A noise mask, acting as a classifier to separate the noise pixels from noise-free pixels is generated during this process. In the second stage, all noise-free pixels are left uncorrected while the other noise pixels will be subjected for further processing. At this level, the pixel restoration process is carried out recursively with the assistance of fuzzy based local information. The mechanism of the proposed DSSSM filter is discussed and explicated with more specific in the following subsections.

3.1 Stage 1: Impulse noise detection

In digital image, the noisy pixel can be characterized by a pixel with the intensity that varies greatly from those of its neighboring pixels. Basically, the intensities of these pixels are represented by a numerical integer. Based on this fact, the impulse detection can be realized by analyzing the local image statistics within a window patch. In the beginning of the detection process, the proposed DSSSM filter employs a square local window $W(i, j)$ with odd dimensions $(2N+1)\times(2N+1)$ and is centered at $x(i, j)$. It is given as:

$$W(i,j) = \{x(i+k, j+l)\}; \text{ where } k,l \in (-N,...,0,...,N) \tag{2}$$

All the pixel's elements within $W(i, j)$ are then stored in two separate arrays which represent the sorted statistics and non-sorted statistics, respectively. The process is continued by finding the median pixel $Pmed(i, j)$ and central pixel $Pcenter(i, j)$. Both $Pmed(i, j)$ and $Pcenter(i, j)$ are defined by:

$$P_{med}(i,j) = \text{med}\{x(i+k, j+l)\} \tag{3}$$

$$P_{center}(i, j) = x(i, j) \tag{4}$$

Next, the median pixel $Pmed(i, j)$ and central pixel $Pcenter(i, j)$ are subtracted from all the pixels in $W(i, j)$. This modus operandi will produce two sets of absolute differences arrays, namely $dmed(i+k, j+l)$ and $dcenter(i+k, j+l)$. Mathematically, these absolute differences arrays are computed as follows:

$$\begin{aligned} &d_{med}(i+k, j+l) = \\ &\qquad |x(i+k, j+l) - P_{med}(i,j)|; \text{ with } k,l \neq 0 \end{aligned} \tag{5}$$

$$\begin{aligned} &d_{center}(i+k, j+l) = \\ &\qquad |x(i+k, j+l) - P_{center}(i,j)|; \text{ with } k,l \neq 0 \end{aligned} \tag{6}$$

At this point, all the values computed in $dmed(i+k, j+l)$ and $dcenter(i+k, j+l)$ are rearranged in ascending order. After that, the median of absolute differences (i.e. *MADmed* and *MADcenter*) will be identified based on:

$$MAD_{med} = \text{med}\{d_{med}(i+k, j+l)\} \tag{7}$$

$$MAD_{center} = \text{med}\{d_{center}(i+k, j+l)\} \tag{8}$$

In order to make a distinction whether current processing pixel is a noise or not, the difference between *MADmed* and *MADcenter* will be first calculated. If the MAD difference is denoted as *diffMAD*, then alternatively *diffMAD* can be written as follows:

$$diffMAD = |MAD_{med} - MAD_{center}| \tag{9}$$

This *diffMAD* provides information about the likelihood of corruption for the current processing pixel. For example, if d*iffMAD* value is large then the current pixel is very likely being contaminated by impulse noise. On the other hand, in the case where *diffMAD* is small, the current pixel may be considered as a noise-free.

After *diffMAD* is counted, a binary noise mask $M(i, j)$ will be formed to mark the locations of noise pixels and noise-free pixels. Thus, the process of generating noise mask can be grasped as:

$$M(i, j) = \begin{cases} 1, & diffMAD > T^{(t)}{}_{DSSSM} \\ 0, & diffMAD \leq T^{(t)}{}_{DSSSM} \end{cases} \tag{10}$$

where $M(i, j)=1$ signifies the noise pixel, $M(i, j)=0$ represents the noise-free pixel and $T^{(t)}DSSSM$ actually is the threshold in the t-th iteration.

3.2 Stage 2: Noise filtering

After the binary noise mask is created, the filtering action will replace the noise pixels marked with $M(i, j)=1$ with the estimated correction term. Yet again, the proposed DSSSM filter adopts a square sliding window $Wfilter(i, j)$ with $(2Nfilter+1) \times (2Nfilter+1)$ dimensions. The so called filtering window is given as:

$$\begin{aligned} &W_{filter}(i, j) = \{x(i+k, j+l)\}; \\ &\quad where\ k,l \in (-N_{filter}, \ldots, 0, \ldots, N_{filter}) \end{aligned} \tag{11}$$

For each detected noise pixel, the median pixel $m(i, j)$ is selected from the corresponding region within $Wfilter(i, j)$ using:

$$m(i, j) = med\{x(i+k, j+l)\} \tag{12}$$

Then, the local information $Dlocal(i, j)$ is extracted from $Wfilter(i, j)$ according to:

$$\begin{aligned} &D_{local}(i, j) = \max |x(i+k, j+l) - x(i, j)|; \\ &\text{with } (i+k, j+l) \neq i, j \end{aligned} \tag{13}$$

As part of the filtering mechanism in the DSSSM filter, fuzzy reasoning is applied to the extracted local information. The fuzzy set will handle any uncertainties present in the local information due to the nonlinear nature of impulse noise corrupting an image. This adopted fuzzy set $f(i, j)$ is defined by:

$$f(i, j) = \begin{cases} 0 & : D_{local}(i, j) < T_1 \\ \dfrac{D_{local}(i, j) - T_1}{T_2 - T_1} & : T_1 \leq D_{local}(i, j) < T_2 \\ 1 & : D_{local}(i, j) \geq T_2 \end{cases} \tag{14}$$

where T_1 and T_2 are two empirically predefined threshold with the value 10 and 30 respectively. Finally, the correction term $y_{DSSSM}(i, j)$ is used to replace the detected noise pixel. It will utilize the median pixel $m(i, j)$ and fuzzy set $f(i, j)$ obtained in two previous processes. The correction term is computed as follows:

$$y_{DSSSM}(i, j) = [1 - f(i, j)] \bullet x(i, j) + f(i, j) \bullet m(i, j) \tag{15}$$

As to enhance the filter's ability in selecting a more accurate median pixel, this algorithm is implemented in recursive manner. An illustrative example of the impulse noise detection and filtering operation against an image with the corruption rate of 10% is shown in Fig. 1.

3.3 The selection of the parameters

Generally, the selection of threshold set and number of iterations needed for every level of noise density are essential since it will influence the performance of the proposed filter. In this framework, we have fixed the number of iterations based on the impulse noise density. By taking into account the trade-off between good filtering

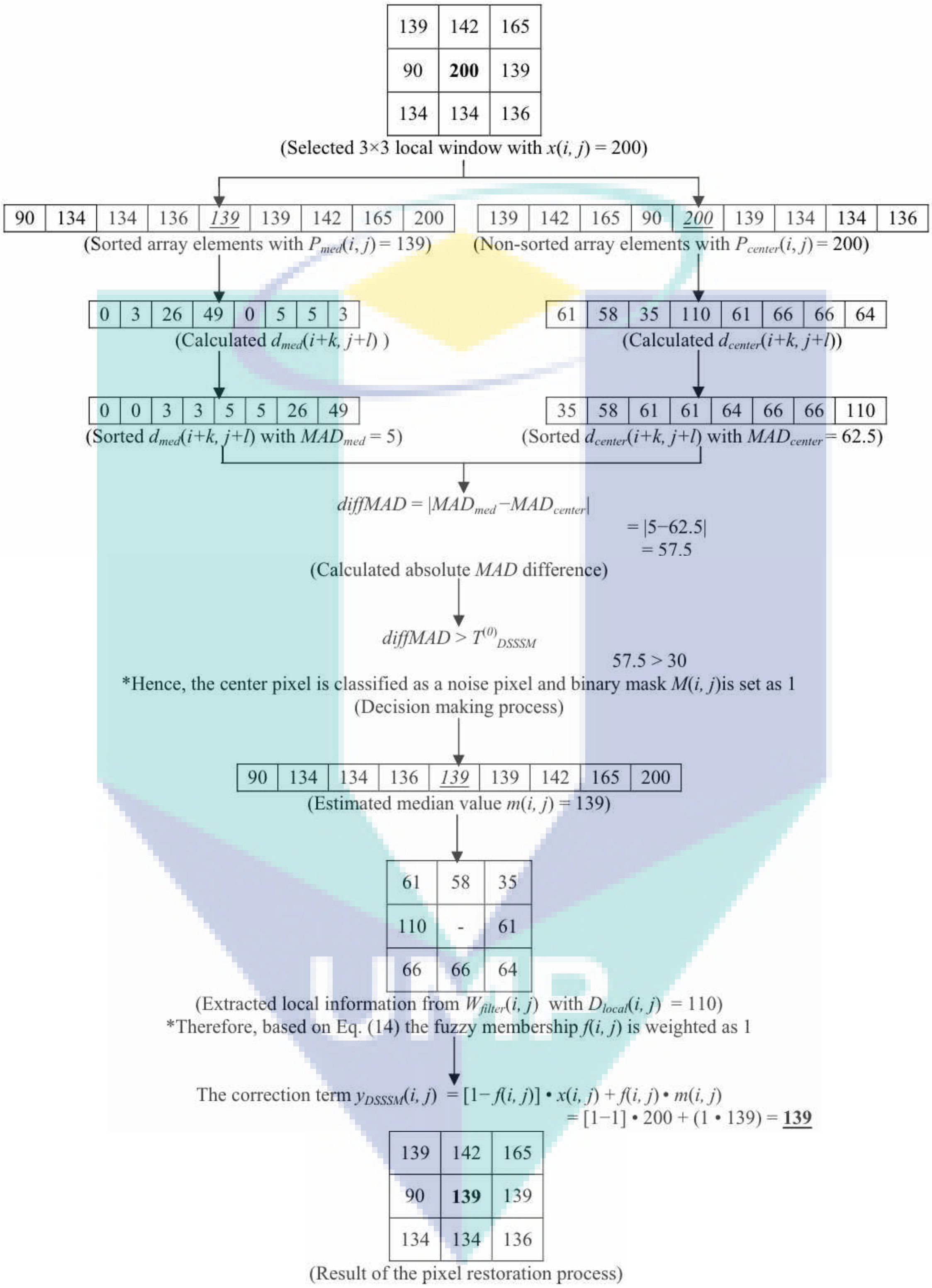


Fig. 1. The illustrated example on the DSSSM noise filtering operation

performance and efficient processing time, the suggested numbers of iterations for DSSSM filter are listed in Table 1.

From the results observed in Tables 2 to 4, the threshold sets are suggested as [$T^{(0)}=25$, $T^{(1)}=15$] for the 1% to 9% impulse noise cases, [$T^{(0)}=30$, $T^{(1)}=20$, $T^{(2)}=15$] for the 10% to 19% impulse noise cases and [$T^{(0)}=35$, $T^{(1)}=25$, $T^{(2)}=20$, $T^{(3)}=15$] for the 20% to 30% impulse noise cases. Noticeably, at each iteration the threshold $T^{(t)}DSSSM$ is applied in a decreasing manner. The reason in lowering $T^{(t)}DSSSM$ is to trace the remaining noise pixels which

Table 1. The suggested number of iterations for different noise densities

Number of Iterations	Impulse Noise Density
2	$r < 10\%$
3	$10\% \leq r < 20\%$
4	$20\% \leq r \leq 30\%$

Table 2. The noise filtering capabilities of DSSSM filter with different threshold set (5% noise).

Threshold (*T(0)*, *T(1)DSSSM*)	Average PSNR
30,20	35.9163
25,15	**36.0348**
20,10	35.3107

Table 3. The noise filtering capabilities of DSSSM filter with different threshold set (10%, 15% noise).

Threshold (*T(0)*,*T(1)*,*T(2)DSSSM*)	Average PSNR (10%)	Average PSNR (15%)
40,30,25	33.8338	32.3589
30,20,15	**34.1875**	**33.1333**
25,15,10	33.7779	32.7030

Table 4. The noise filtering capabilities of DSSSM filter with different threshold set (20%-30% noise).

Threshold (*T(0)*,*T(1)*, *T(2)*,*T(3) DSSSM*)	Average PSNR (20%)	Average PSNR (25%)	Average PSNR (30%)
45,35,20,25	31.094	30.111	29.128
35,25,20,15	31.927	31.059	30.166
30,20,15,10	31.580	30.733	29.770

have smaller *diffMAD*.

Meanwhile, in order to determine the optimum threshold set, five well-known test images consist of Cameraman, Pepper, Bridge, Jet and Goldhill with $r \in$ [5%, 10%, 15%, 20%, 25%, 30%] have been adopted to be tested in a series of simulations using different set of threshold. The average PSNR values for each noise level are shown in Table 2, 3 and 4.

4. Simulation Results and Discussion

In this section, the performance of the proposed DSSSM filter will be compared to other related state-of-the-art impulse noise filters based on their simulation results. A total of 80 standard test images of size 512×512, obtained from diverse online sources were used for the simulations of each implemented filters. They are selected because these test images contain fine image details and textures which are suitable to assess the strengths and weaknesses of the implemented impulse noise filters. Each of the test images was corrupted with the impulse noise model described in (1), ranging from 5% to 30% with an increment of 5%. This set of standard test images contains various characteristics which are suitable to assess the robustness of the implemented filters.

For comparison, the following conventional impulse noise filters with their suggested tuning parameters were used to restore the contaminated test images. Among them are SWM-I (*Ti*=50) [4], SWM-II (*Ti*=30, w=3) [4], TSM (*T*=20, *w*=3) [11], DWM (*wm*=2, T0=510, *Nmax*=[5, 11] and 5×5 window size) [8], LUO (*LD*=1, *u*=3, *W1*=5, *W2*=30) [13] and ACWM ([*δ0*, *δ1*, *δ2*, *δ3*]=[40, 25, 10, 5], *s*=0.6) [10]. Finally, the simulation results of the implemented impulse noise filters will be interpreted

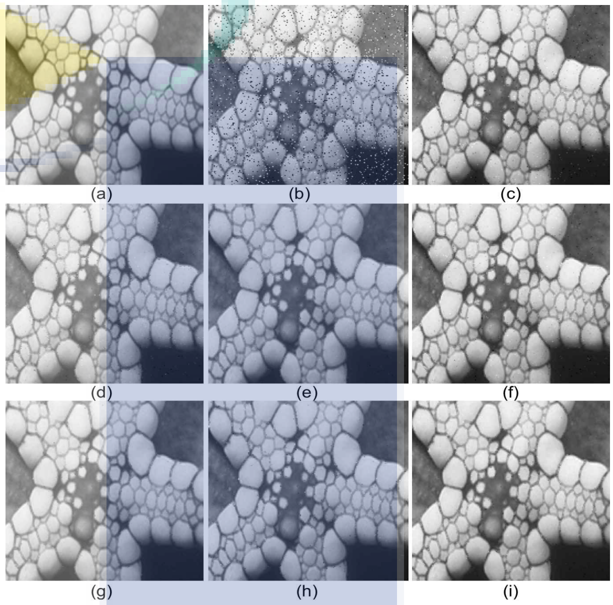


Fig. 2. Simulation results on a portion of *Starfish* using; (a) original image, (b) noisy image with 10% density of impulse noise, (c) SWM-I, (d) SWM-II, (e) TSM, (f) DWM, (g) LUO, (h) ACWM and (i) DSS

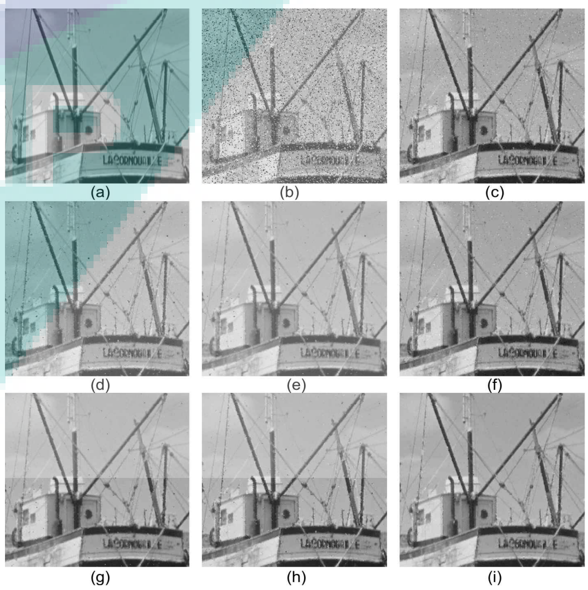


Fig. 3. Simulation results on a portion of *Boat* using; (a) original image, (b) noisy image with 20% density of impulse noise, (c) SWM-I, (d) SWM-II, (e) TSM, (f) DWM, (g) LUO, (h) ACWM and (i) DSSSM

objectively and subjectively as to judge the filters effectiveness in removing random-valued impulse noise.

4.1 Qualitative analysis

Providing visually pleasing output is imperative, since the image is eventually to be viewed by human eye. Therefore, the appearances of the resultant images are inspected visually to judge the filters efficiencies in reducing impulse noise effect and producing good image quality. Among the 80 test images, the simulation results for three commonly used grayscale test images Starfish, Boat and Goldhill are presented in Fig. 2, Fig. 3 and Fig. 4, respectively. In each figure, portion (b) represents the noise corrupted images with 10%, 20% and 30% density of noise.

As can be seen in Starfish, at 10% impulse noise density, the noise filtering performance of DSSSM filter is basically similar to those of the conventional noise filtering algorithms. All filters are found to be able of producing perceptible reconstructed image since the density of impulse noise is still low and not form any noise patches at this level.

However, in Boat which is contaminated with 20% of impulse noise (as shown in Fig. 3), it can be visualized that the results produced by the conventional median filters are still influenced by the noise. It is able to notice that some small noise patches are remained intact on the resultant images. On contrary, the proposed DSSSM filtering algorithm can significantly remove the effect of noise added to the images and at the same time preserve the object shapes.

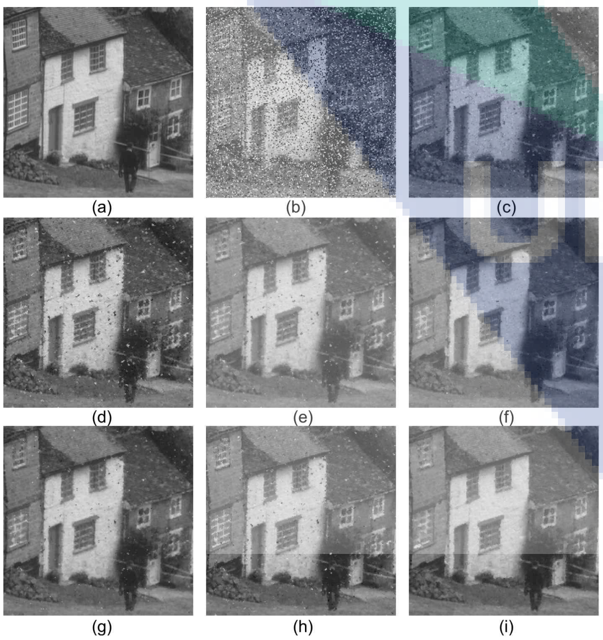


Fig. 4. Simulation results on a portion of *Goldhill* using; (a) original image, (b) noisy image with 30% density of impulse noise, (c) SWM-I, (d) SWM-II, (e) TSM, (f) DWM, (g) LUO, (h) ACWM and (i) DSSSM

The similar results are obtained for the image named Goldhill; where the proposed DSSSM filtering algorithm outperforms the conventional SWM-I, SWM-II, TSM, DWM, LUO and ACWM algorithms by giving clearer filtered image even the density of noise in this image is higher (i.e. 30% of impulse noise). The proposed DSSSM filter has successfully reduced the noise particles and noise patches, consequently created less corrupted image. This observation indicates that the combination of the adaptive impulse noise detection concept with the fuzzy based local information significantly helps the proposed DSSMM filter to dexterously reduce the noise stain. Meanwhile, there is a great deal of noise contamination existed in the images produced by the conventional filtering techniques. The poor restoration results among these conventional filters can be attributed from their impulse noise filtering mechanisms which are less robust towards the contamination of random-valued impulse noise.

4.2 Quantitative analysis

In addition to the visual inspection of the restored images, the quality of the restored images is also evaluated quantitatively using the peak signal-to-noise ratio (PSNR). Mathematically, the PSNR for a digital image of the dimension $M \times N$ is defined as:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right)(\mathrm{dB}) \tag{16}$$

For the above formulae, MSE stands for the mean-squared error and it is given as:

$$\mathrm{MSE} = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\frac{[o(i,j)-y(i,j)]^2}{M \times N} \tag{17}$$

where $y(i, j)$ is the filtered image and $o(i, j)$ is the original noise-free image. Apart from the PSNR assessment, the mean of absolute error (MAE) has also been used in this analysis to characterize the filter's detail preservation behavior, one which is defined by:

$$\mathrm{MAE} = \frac{1}{M \times N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}|o(i,j)-y(i,j)| \tag{18}$$

The numerical results for Figs. 2(c)-(i), Figs. 3(c)-(i) and Figs. 4(c)-(i) are tabulated in Table 5. From the functions, the larger PSNR and smaller MAE values show better restoration results. In all tables, the best results obtained are made bold.

As reported in Table 5, except for the 10% impulse noise

Table 5. Comparison of PSNR on Different Noise Level Restoration for ‘Starfish’, ‘Boat’ and ‘Goldhill’ (Test Image)

Images	Algorithms	10%	PSNR(dB) 20%	30%
Starfish	SWM-I	31.6219	27.8273	24.7745
	SWM-II	33.056	27.8818	23.8052
	TSM	35.0013	30.2872	26.1675
	DWM	32.6523	29.6524	27.9957
	LUO	35.1372	31.4958	28.0056
	ACWM	35.3272	30.6981	26.55
	DSSSM	**35.3753**	**32.7072**	**30.5841**
Boat	SWM-I	31.1459	27.7627	24.866
	SWM-II	32.4632	28.0734	24.1192
	TSM	33.6079	30.1461	26.3557
	DWM	32.5794	29.4737	27.6844
	LUO	34.0311	31.2224	27.9866
	ACWM	**34.4813**	30.5946	26.6566
	DSSSM	34.2323	**31.9472**	**30.0266**
Goldhill	SWM-I	31.3933	27.9749	25.0891
	SWM-II	33.1147	28.4592	24.4252
	TSM	34.1815	30.6761	26.7165
	DWM	33.3963	30.0995	29.0558
	LUO	34.8439	32.0345	28.6311
	ACWM	**35.0158**	31.1301	27.0137
	DSSSM	34.4235	**32.4756**	**30.8526**

Table 6. Comparison of MAE on Different Noise Level Restoration for ‘Starfish’, ‘Boat’ and ‘Goldhill’ (Test Image)

Images	Algorithms	10%	MAE 20%	30%
Starfish	SWM-I	1.37569	3.01962	5.21482
	SWM-II	1.16153	2.82004	5.49509
	TSM	0.925304	2.02381	3.84945
	DWM	1.17076	2.3494	3.52177
	LUO	0.983086	1.88244	3.21782
	ACWM	**0.759693**	1.78849	3.48141
	DSSSM	0.826065	**1.58328**	**2.49641**
Boat	SWM-I	1.46273	3.05105	5.19042
	SWM-II	1.2075	2.7313	5.27398
	TSM	1.13034	2.14042	3.87268
	DWM	1.2042	2.42207	3.66792
	LUO	1.18715	2.03902	3.35503
	ACWM	**0.836605**	1.80618	3.43848
	DSSSM	0.955929	**1.71384**	**2.5858**
Goldhill	SWM-I	1.46043	3.07207	5.17048
	SWM-II	1.19814	2.73936	5.21354
	TSM	1.12705	2.1397	3.82618
	DWM	1.13602	2.33179	3.29423
	LUO	1.19761	2.03074	3.26606
	ACWM	**0.909317**	1.89498	3.5232
	DSSSM	1.01952	**1.77089**	**2.61645**

of *Boat* and *Goldhill*, the proposed DSSSM filtering technique consistently yields the highest PSNR values compared to the other existing conventional filters. Even though the PSNR values of ACWM filter in Boat and *Goldhill* are slightly higher than DSSSM filter when r =10%, but in terms of visual evaluation in Fig. 4(h) and 4(i), no significant difference between the two images are observed.

Meanwhile, the similar phenomenon is occurred in the

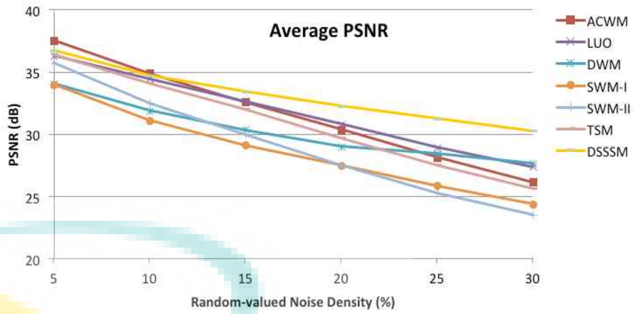


Fig. 5. The graph of an average PSNR based on different noise level restorations for 80 standard grayscale test images

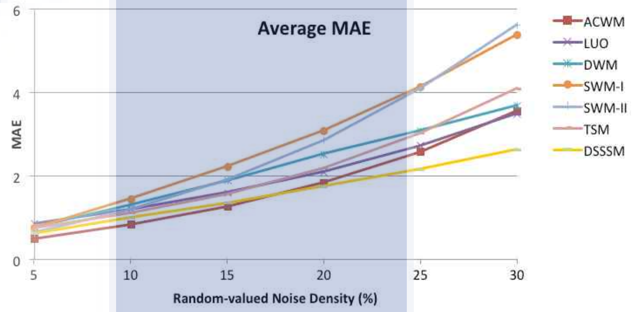


Fig. 6. The graph of an average MAE based on different noise level restorations for 80 standard grayscale test images

analysis outlined in Table 6, where the proposed DSSSM filter without fail outclasses the other filters in comparison by producing the best MAE results for the cases of 20% and 30% impulse noise density. These findings strongly support the qualitative results in Section 4.1. On the contrary, at 10% of impulse noise density, it can be observed that ACWM gives the better MAE results as compared to our proposed filter. However the performances of these two algorithms in terms of the details preservation based on the visual perception are fairly similar (e.g. see Fig. 2). This study has further calculated the average PSNR for 80 tested images and the results are displayed in the graph shown in Fig. 5. It could be obviously observed that, for such a mild corruption rate (i.e. noise density of 15% and above), the average PSNR curve of the proposed DSSSM filter has the highest curve as compared to the rest of the filters implemented.

In the meantime, ACWM filter initially has a relatively higher average PSNR than the DSSSM filter; but it start to drop dramatically with the increased in noise density beginning from 15% impulse noise. Above 20% impulse noise density, the two worst performing filters are the SWM-I and SWM-II filters.

In addition, Fig. 6 shows the extrapolation of the average MAE curves obtained using the various conventional filters in comparison and the proposed DSSSM filter. From the plot, it is noticed that all the filters do perform well at the extremely low density of noise (i.e. 5% impulse noise). Nonetheless, only the proposed DSSSM filter managed to survive at all level of noise densities with the MAE is slightly and gradually increased. Other filters, such as ACWM and LUO for instance, even though are able to produce relatively good details preservation results at the beginning but their MAE values have sharply increased; particularly when the noise level is above 20%. On the other hand, the SWM-I and SWM-II filters have completely failed to compete in this test. Once again, this finding shows that the DSSSM filter is not only able to eliminate noise efficiently but it can also preserve the original appearance and shape of an image very well.

5. Conclusion

Throughout this study, an effective algorithm for the detection and suppression of random-valued impulse noise has been introduced. The proposed DSSSM filter is constructed by incorporating a robust impulse noise detection based on adaptive thresholding and recursive pixel restoration technique. Additionally, fuzzy reasoning set is embedded as part of the filtering mechanism in order to handle any imprecise local information. Extensive simulation results reveal that the DSSSM filter is able to reduce the random-valued impulse noise effect, while at the same time preserving the details and structures of fine images. Furthermore, its filtering performance is tremendously consistent all the time as compared to the number of well-known conventional techniques; and all these good results are achieved with a fairly efficient processing time.

Acknowledgements

This work was supported by the Universiti Malaysia Pahang Internal Grants under grant (RDU1703151).

References

[1] W. Luo., "Efficient removal of impulse noise from digital images," *IEEE Trans. Consumer Electronics*, vol. 52, no. 2, pp. 523-527, 2006.

[2] R. C. Gonzalez and R. E. Woods., "Digital image processing," New York: Addison-Wesley, 1992.

[3] H. Hwang and R. A. Haddad., "Adaptive Median Filters: New Algorithms and Results," *IEEE Transactions on Image Processing*, vol. 4, no. 4, pp. 499-502, 1995.

[4] T. Sun and Y. Neuvo., "Detail-Preserving Median Based Filters in Image Processing," *Pattern Recognition Letters*, 15 (1994) pp. 341-347.

[5] T. Chen and H. R. Wu., "Space variant median filters for the restoration of impulse noise corrupted images," *IEEE Trans. on Circuits and Systems II*, 48, pp. 784-789, 2001.

[6] S. Zhang and M. A. Karim., "A new impulse detector for switching median filters," *IEEE Signal Processing Letters*, vol. 9, no. 11, pp. 360-363, 2002.

[7] I. Aizenberg and C. Butakoff., "Effective impulse detector based on rank-order criteria," *IEEE Signal Process. Letters*, vol. 3, no. 11, pp. 363-366, 2004.

[8] Y. Dong and S. Xu., "A new directional weighted median filter for removal of random-valued impulse noise," *IEEE Signal Processing Letters*, vol. 14, no. 3, pp. 193-196, 2007.

[9] H. H. Chou and L. Y Hsu., "A noise-ranking switching filter for images with general fixed impulse noises," *Signal Processing*, 6, pp. 198-208, 2015.

[10] T. Chen and H. R. Wu., "Adaptive impulse detection using centre-weighted median filter," *IEEE Signal Processing Letters*, vol. 8, no. 1, pp. 1-3 2001.

[11] Chen, K. K. Ma and L. H. Chen., "Tri-state median filter for image denoising," *IEEE Transactions on Image Processing*, vol. 8, no. 12, pp. 1834-1838, 1998.

[12] J. K. Sung, H. L. Yong., "Center weighted median filters and their applications to image enhancement," *IEEE Transactions on Circuits and Systems*, vol. 38, no. 9, pp. 984-983, 1991.

[13] W. Luo., "An efficient algorithm for the removal of impulse noise from corrupted images," *AEU-Int. J. Electron. Commun.*, 61, pp. 551-555, 2007.

[14] K. K. V. Toh and N. A. Mat Isa., "Cluster-based adaptive fuzzy switching median filter for universal impulse noise reduction," *IEEE Trans. Consumer Electronics*, vol. 56, no. 4, pp. 2560-2568, 2010.

[15] http://homepages.inf.ed.ac.uk/rbf/HIPR2/median.htm (accessed on April 2012).

[16] Majid, A., Lee, CH., Mahmood, M.T. et al. "Impulse noise filtering based on noise-free pixels using genetic programming," *Knowl Inf Syst* (2012) 32: p. 505.

[17] Xu, Z., Wu, H.R., Yu, X. et al. "Adaptive progressive filter to remove impulse noise in highly corrupted color images," SIViP (2013) 7: p. 817.

[18] Ilke Turkmen., "A new method to remove randomvalued impulse noise in images," *Int. J. Electron. Commun.* (*AEU*) 67 (2013) 771-779.

[19] M. S. Darus, S. N. Sulaiman, I. S. Isa et al. "Modified hybrid median filter for removal of low density random-valued impulse noise in images," *IEEE International Conference on Control System*, Computing and Engineering, pp. 25-27 November 2016.

[20] M. H Suid and N. A. M. Isa. Scientific Research and Essays, vol. 7, no. 7, pp. 813-823, 2012.

Mohd Helmi Suid received the B.Eng Hons in Electrical and Electronics Engineering from Universiti Teknikal Malaysia Melaka in 2007. In 2014, he received the M. Sc. from School of Electronics & Electrical Engineering, Universiti Sains Malaysia, Penang, Malaysia. He is currently, a lecturer in Faculty of Electrical and Electronics Engineering, Universiti Malaysia Pahang (UMP). His current research interests are digital image processing, vision for robotics application and computational intelligence.

Mohd Falfazli Bin Mat Jusof is a Lecturer in Faculty of Electrical & Electronics Engineering Universiti Malaysia Pahang. He was graduated from Ehime University, Japan for Bachelor (2008) and Master Degree (2010). His current research interests are IoT, application on embedded system, computer vision systems and physical computing.

Mohd Ashraf Ahmad received his first degree in B.Eng. Electrical Mechatronics and Master degree in M.Eng. Mechatronics and Automatic Control from University of Technology Malaysia (UTM) in 2006 and 2008, respectively. In 2015, he received a Ph.D in Informatics (Systems Science) from Kyoto University. Currently, he is a lecturer in Faculty of Electrical and Electronics Engineering, Universiti Malaysia Pahang (UMP). His current research interests are model-free control, control of mechatronic systems, nonlinear system identification and vibration control.

MATEC Web of Conferences **214**, 01003 (2018) https://doi.org/10.1051/matecconf/201821401003
ICIPCE 2018

Universal Impulse Noise Suppression Using Extended Efficient Nonparametric Switching Median Filter

M. H. Suid[1] ,M. A. Ahmad[1],M. I. F. M. Hanif[2],M. Z. Tumari[3] and M. S. Saealal[3]

[1]*Faculty of Electrical & Electronic Engineering, Universiti Malaysia Pahang,26600 Pekan, Pahang, Malaysia.*
[2]*SEGi University, No. 9 Jalan Teknologi, Taman Sains Selangor, Kota Damansara, PJU 5, 47810 Petaing Jaya, Selangor.*
[3]*Faculty of Engineering Technology, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.*

Abstract. This paper presents a filtering algorithm called extended efficient nonparametric switching median (EENPSM) filter. The proposed filter is composed of a nonparametric easy to implement impulse noise detector and a recursive pixel restoration technique. Initially, the impulse detector classifies any possible impulsive noise pixels. Subsequently, the filtering phase replaces the detected noise pixels. In addition, the filtering phase employs fuzzy reasoning to deal with uncertainties present in local information. Contrary to the existing conventional filters that only focus on a particular impulse noise model, the EENPSM filter is capable of filtering all kinds of impulse noise (i.e. the random-valued and/or fixed-valued impulse noise models). Extensive qualitative and quantitative evaluations have shown that the EENPSM method performs better than some of the existing methods by giving better filtering performance.

Keywords: Image processing, impulse noise, digital image, noise filtering, nonparametric switching median filter.

1 INTRODUCTION

With the usage of multimedia material becoming more widespread from day to day, visual information from high quality digital images plays an important role in many daily life applications. Unfortunately, digital images are frequently subjected to the contamination of impulse noise due to the interferences generated during transmission, acquisition in noisy environment [1]. Luckily, due to the advancement in digital technologies the level of noise density in digital images has dropped significantly to the low contamination rate. Still, even at low densities, the occurrence of impulse noise can severely damage the information in the original image. Therefore, it is essential to remove impulse noise effect before carrying any subsequent image processing task (e.g. segmentation, object recognition, data compression etc.). Mostly, these subsequent processing steps are largely affected by the quality of the filtered image [2].

For this purpose, many filters have been proposed. The median filter for instance, is a well-known nonlinear filter for suppressing impulsive noise due to its effectiveness and high computational efficiency [3],[4]. Despite its effectiveness in smoothing noise, the median filter tends to blur fine details and often destroys edges due to its clumsy filtering property that treats all the pixels equally without considering whether or not it is noise-free pixel.

The aforementioned drawbacks have led to the development of various switching-based filters, e.g. the switching median (SWM) filters along with the centre weighted switching median (CWSWM) filter [5], the Laplacian switching median (LSM) filter [6], progressive switching median (PSM) filter [7] and the multi-state median (MSM) filter [8], etc. Basically, this class of filtering scheme works based on the impulse detection mechanism which uses a fixed size filtering window and predefined threshold value to differentiate between noise and noise-free pixels. With the noise detector, these filters are shown to be more effective in terms of the detail and edge preservation compared to the uniformly applied conventional median filters. However, one disadvantage is that the switching rule is typically based on a fixed threshold for locally obtained statistics. This approach in certain circumstances tends to yield problem of pixel's misclassification and fails to replace the noise pixels.

Of late, works in [9] has come out with a more flexible switching-based filter called efficient nonparametric switching median (ENPSM) filter, for detail-preserving restoration. The ENPSM filter is customary based on the combination of local variance threshold in the impulse noise detection module and recursive restoration technique in the pixel restoration module. Even though this method performs well, yet it only touched on the filtering of the random-valued (RV) impulse noise corrupted cases.

Thus, in this paper we take one step further by focusing on the detection and suppression of any type of impulse noise models. By using the same existing recursive filtering technique in the ENPSM filter, the proposed technique called extended efficient nonparametric switching median (EENPSM) filter is equipped with histogram-based impulse noise detector which is specifically designed for accurate fixed-valued (FV) impulse noise detection.

2 IMPULSE NOISE MODEL

In theory, impulse noise contamination amplitude could fall either within the image dynamic range (i.e. RV impulse) or out of the range (i.e. FV impulse) and usually only certain percentages of pixels are altered. For more detail, let $x(i, j)$ and $o(i, j)$ be the gray level of the noisy image and the original image at location (i, j), respectively. Then, the impulse noise model with noise density ρ can be defined as:

$$x_{(i,j)} = \begin{cases} n_{(i,j)}: & \text{with probability } \rho \\ o_{(i,j)}: & \text{with probability } 1-\rho \end{cases} \tag{1}$$

where $n(i, j)$ is the noise pixel value. The image is likely to be contaminated by the RV impulse noise when $n(i, j)$ is uniformly distributed within the image dynamic range, i.e. $n(i, j) \in [Nmin, Nmax]$. On the other hand, the simplest impulse noise model is the FV impulse noise; where $n(i, j)$ is assumed to take the maximal and minimal intensities, i.e. $n(i, j) \in (Nmin, Nmax)$.

Although many researchers have paid special attention in obtaining a good filter for on a particular impulse noise model, this proposed technique takes one step further by focusing on the removal of RV and FV impulse noise all at once. In practical, identifying this scenario is more challenging compared to smoothing the RV noise or FV noise alone since it is closer to the real-world situations.

3 EFFICIENT NONPARAMETRIC SWITCHING MEDIAN FILTER

In this section, the conventional ENPSM filtering algorithm proposed in [9] will be briefly reviewed. Given a noisy image and initial filtering window $W(i, j)$ of size $(2N+1)\times(2N+1)$, the RV noise detection stages of ENPSM are described as follows:

Step 1: Sort all elements within $W(i, j)$ in ascending order and find the median pixel $m(i, j)$.

Step 2: Compute the absolute luminance differences $d(i\pm k, j\pm l)$ between $m(i, j)$ and all pixels in $W(i, j)$;

$$d(i\pm k, j\pm l) = \left|x(i\pm k, j\pm l) - m(i,j)\right|; \quad \text{for} -N \le k,l \le N$$

Step 3: Rearrange each value obtained in $d(i\pm k, j\pm l)$ and set the predefined threshold T_{ENPSM} as;

$$T_{ENPSM} = \text{med}\left\{d(i\pm k, j\pm l) \mid : -N \le k,l \le N\right\}.$$

Step 4: Mark the locations of 'noise' pixels (i.e. $d(i\pm k, j\pm l) > T_{ENPSM}$) and 'noise-free'(i.e. $d(i\pm k, j\pm l) \le T_{ENPSM}$) pixels in the binary noise map $M(i, j)$. Slide $W(i, j)$ to the next pixel and repeat Step 1 to Step 4 until the process is completed for the entire image.

This first part of this detection process is sufficient to handle impulse noise density (up to 20%), largely for RV impulse noise. However, it is still not appropriate in the ENPSM to detect and replace the FV impulse noise accurately, particularly when the noise density is moderate or high.

4 EXTENDED EFFICIENT NONPARAMETRIC SWITCHING MEDIAN FILTER

The proposed filter is a modification of the original ENPSM filter, which operates on the same sliding window spatial filter that targets each pixel in a filtered image sequentially. The difference of the proposed filter with the former ones is that the proposed EENPSM filter modifies the existing ENPSM by adding one more process, in case when the case of FV contamination happens.

4.1 Noise detection stage

Taking the FV impulse noise into account, the noisy image histogram will be utilized by the proposed EENPSM filter. It is known that the peak intensities at the ends normally represent the FV noise intensities; e.g, see [10]. By employing the local maximum, the two fixed-valued impulsive intensities can be found by traversing the noisy image histogram from both ends and directed towards the centre of the histogram simultaneously. Once the local maximums denoted as L_{Min} and L_{Max}, are found then the search will be stopped immediately. The detected local maximums represent the two fixed-valued FV impulse noise intensities. Furthermore, FV noise model is constructed based on the assumption that noise pixels will assume the two extreme values in the image dynamic range. Under some real-world situations, these FV noise pixels can be replaced by close approximations of their actual noise intensities [11-13]. For example, pixels with the intensities value of 0 are possible to be replaced with 1 or 2, and intensity value 255 will be replaced with 254 or 253 in an image stored as an 8-bit integer. Hence, distortion will be hardly detected by common FV impulse noise filters. At the end of the detection stage, a two-dimensional binary noise detection map $M(i, j)$ is formed based on:

$$M(i,j) = \begin{cases} 1, & \left[d(i\pm k, j\pm l) > T_{ENPSM}\right] \\ & \cap\left[x(i,j) = L_{Min} \cap L_{Max}\right] \\ 0, & d(i\pm k, j\pm l) \le T_{ENPSM} \end{cases} \tag{2}$$

where $M(i, j)$ in Step 4 is modified to produce a new version of $M(i, j)$ as shown in equation (2). Logic '1s' in the equation shows the positions of noisy pixels and logic '0s' intended for those non-noisy ones.

4.2 Noise filtering stage

After binary noise mask $M(i, j)$ is formed, those pixels marked with $M(i, j) = 1$ is then will be swapped by the estimated median value. Otherwise, the filtering action is skipped when $M(i, j) = 0$ and the pixels will be left unprocessed. Once again, the proposed EENPSM algorithm uses a square filtering window $W_{filter}(i, j)$ with odd $(2N+1)\times(2N+1)$ dimensions and it is given as:

$$W_{filter}(i,j) = \left\{ x(i \pm k, j \pm l), \ldots, x(i,j), \ldots, x(i \pm k, j \pm l) \right\}; \quad \text{where} - N_{filter} \le k, l \le N_{filter} \tag{3}$$

For every '1s' pixel detected, the estimated median value is counted using only 'noise-free' pixels in the current filtering window. The calculation process is carried out using:

$$m_{est}(i,j) = \text{ med}\left\{ x(i \pm k, j \pm l), .., x(i,j), .., x(i \pm k, j \pm l) \right\}; \quad \text{with } M(i \pm k, j \pm l) = 0 \tag{4}$$

This criterion of choosing only 'noise-free' pixels is imposed to avoid selecting a 'noise' pixel as the estimated median pixel. Finally, the correction term to restore a detected noisy pixel is a linear combination between the current processing pixel and the estimated median pixel. The restoration term is given here as;

$$y_{EENPSM}(i,j) = \left[1 - M(i,j)\right] \cdot x(i,j) + M(i,j) \cdot m_{est}(i,j) \tag{5}$$

5 SIMULATION RESULTS AND DISCUSSIONS

In this section, the practicability of the proposed EENPSM filter will be compared to the original ENPSM filter based on their simulation results. Three examples are provided to verify and justify the ideas described in Section 4. In this experiment, the original 512×512, 8 bits gray scale images *Flower*, *Pens* and *Yacht* were used in the simulation of the implemented filters. Those images are commonly used in image processing research and studies. Each of them was corrupted with fixed-valued and random-valued impulse noise ranging from 10% to 30%. Both qualitative and quantitative assessments are employed to assess the performances of the proposed EENPSM and the original ENPSM filter. The quantitative assessment used here is the peak signal to noise ratio (PSNR) which is defined as:

$$PSNR = 10 \log_{10}\left(\frac{255^2}{MSE}\right) dB \tag{6}$$

where MSE is the mean-squared error given as:

$$MSE = \left(\frac{1}{M \times N}\right) \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left[o(i,j) - y(i,j)\right]^2 \tag{7}$$

For the above mentioned formula, $M\times N$ is the image size with 'M' rows and 'N' columns, $y(i, j)$ is the filtered image and $o(i, j)$ is the original noise-free image.

As can be seen in Figure 1, at 10% impulse noise density, the noise filtering performance of EENPSM filter is basically similar to the original ENPSM filtering algorithm. Both filters are found to be able of producing visible restored image at this noise level since the density of impulse noise is still low and not yet form any noise patches at this level

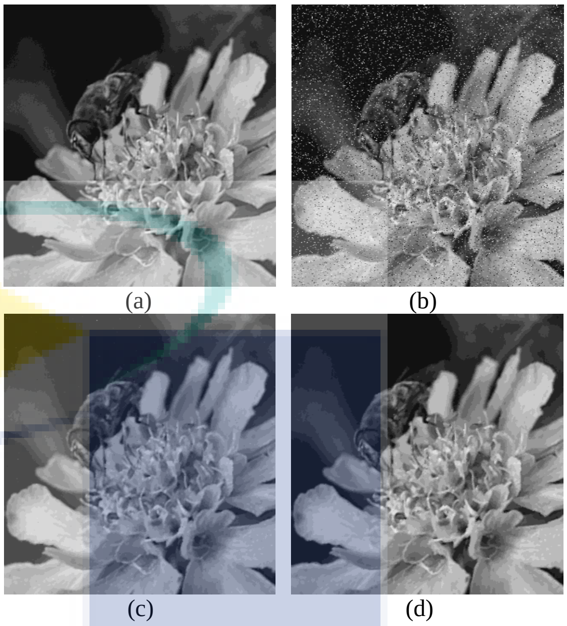


Figure 1. Simulation results on a portion of *Flower* with 10% density of impulse noise using; (a) original image, (b) noisy image, (c) ENPSM and (d) EENPSM.

However, in the *Pens* image (see Figure 2), we may notice that the proposed EENPSM filter gives better and clearer filtering result compared to the original ENPSM algorithm. The noise particles and effects are significantly reduced and at the same time the image details are well preserved. In contrast, we may be able to notice that some small noise spots are remained intact on the resultant images produced by the ENPSM filter. It is found that the original ENPSM filter has problem in the case of FV noise blotches (a place in image where a large number of fixed-valued impulse pixels may connect).

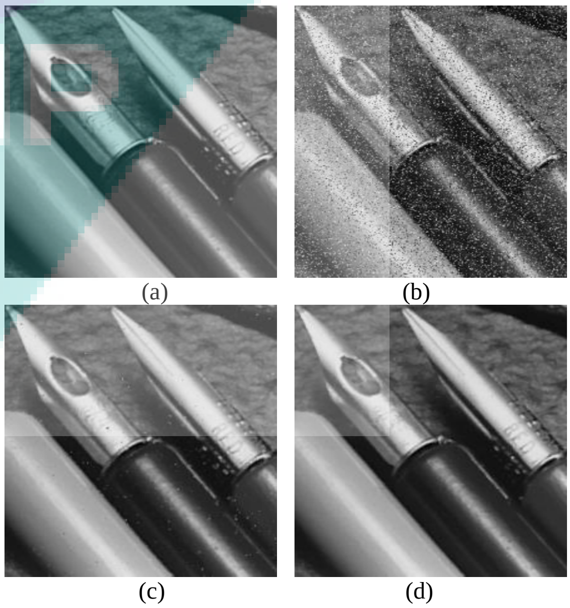


Figure 2. Simulation results on a portion of *Pens* with 20% density of impulse noise using; (a) original image, (b) noisy image, (c) ENPSM and (d) EENPSM.

The similar phenomenons are obtained for the *Yacht* test image (shown in Figure 3), where the proposed EENPSM filtering algorithm consistantly outperforms the ENPSM filter by giving clearer image; even the density of noise in this image is increasing. This is due to the ability of the proposed algorithm to distinguish the FV noise pixels more dexterously as compared to the previous filtering version.

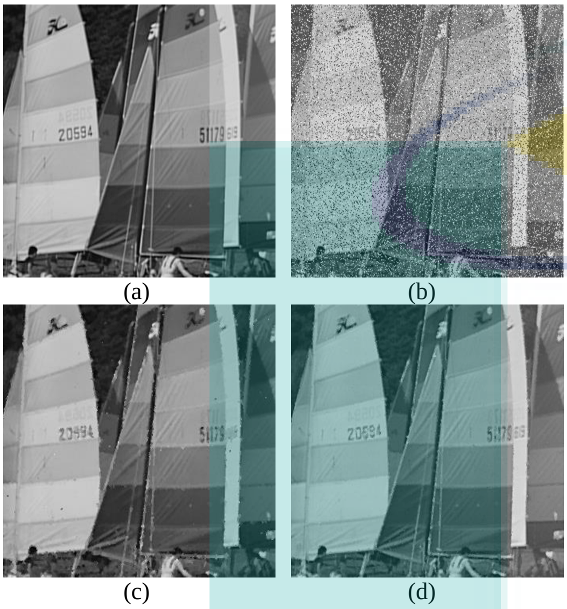


Figure 3. Simulation results on a portion of *Yacht* with 30% density of impulse noise using; (a) original image, (b) noisy image, (c) ENPSM and (d) EENPSM.

Table 1. Comparison of PSNR on Different Noise Level Restoration for *Flower* (Test Image)

Algorithms	PSNR (dB)		
	10%	20%	30%
ENPSM Proposed EENPSM	33.9590 **34.5308**	31.9943 **33.2594**	30.6433 **31.6185**

Table 2. Comparison of PSNR on Different Noise Level Restoration for *Pens* (Test Image)

Algorithms	PSNR (dB)		
	10%	20%	30%
ENPSM Proposed EENPSM	36.1737 **37.5011**	33.5591 **34.8123**	30.6339 **32.845**

Table 3. Comparison of PSNR on Different Noise Level Restoration for *Yacht* (Test Image)

Algorithms	PSNR (dB)		
	10%	20%	30%
ENPSM Proposed EENPSM	31.5160 **33.3688**	29.1585 **30.3931**	26.6862 **28.6231**

Meanwhile, the numerical results for the used standard test images (i.e *Flower*, *Pens* and *Yacht*) are presented in Tables 1-3, respectively. Overall, it is made known that the proposed EENPSM unfailingly outperforms the previous algorithm at all level of universal impuse contamination case. It is evident that EENPSM filtering performance is tremendously consistent. In contrast, inconsistentce performances have been shown by the ENPSM filter with their PSNR values to have decreased intensely especially at the 30% noise level. Indirectly this result also shows that the conventional methods were unable to cater for the occurrences of noise in a proper manner especially when it comes to the fixed-valued contamination cases.

6 CONCLUSIONS

In this article, an extended version of ENPSM filter namely EENPSM for effective universal impulse noise restoration is presented. The variance thresholding and recursive pixel restoration techniques that involved in the design of the filter make it able to suppress both random-valued and fixed-valued impulse noise effectively, at the same time preserving fine image edges and textures. Furthermore, this filter does not require any special tuning of parameter since its predefined threshold is established based on nonparametric framework. The simulation results indicate that a better noise filtering performance is achieved. In addition, the fuzzy reasoning could be embedded as part of its filtering mechanism, which permits us to exploit the effectiveness of fuzzy paradigm in handling imprecise local information. Overall, it is a feasible approach for removing the effects of low level universal impulse noise in digital images.

Acknowledgement

This work was partially supported by the Research and Innovation Department, University Malaysia Pahang and the Ministry of Higher Education under research grant RDU1703151.

References

1. X. Lan, Z. Zuo, Random-valued impulse noise removal by the adaptive switching median detects and detail preserving regularization, (2013).
2. K. K. V. Toh, H. Ibrahim and M. N. Mahyuddin. Cluster-based Adaptive Fuzzy Switching Median Filter for Universal Impulse Noise Reduction. *IEEE trans. Consumer Electron.*, **56(4)**, 2560-2568 (2010).
3. J. Astola, P. Kuosmanen, *Fundamentals of Nonlinear Digital Filtering*, CRC, Boca Raton, FL, (1997).
4. R.C. Gonzalez, R.E. Woods, Digital Image Processing, Prentice Hall, New Jersey, (2002).
5. T. Sun and Y. Neuvo. Detail-Preserving Median Based Filters in Image Processing. *Pattern Recognition Letters*,**15** (1994) 341–347.

6. S. Zhang and M. A. Karim. A new impulse detector for switching median filters. *IEEE Signal Processing Letters*, **9 (11)**, 360-363, (2002).
7. Z. Wang and D. Zhang. Progressive switching median filter for the removal of impulse noise from highly corrupted images. *IEEE Trans. on Circuits and Systems II*, **46 (1)**,78-80, (1999).
8. T. Chen and H. R. Wu. Adaptive impulse detection using centre-weighted median filter. *IEEE Signal Processing Letters*, **8 (1),** 1-3 (2001).
9. M. H. Suid and N. A. M. Isa. Random valued impulse noise reduction in digital image using efficient nonparametric switching median filter. *Scientific Research and Essays* vol. **7(7)**, 813-823, (2012).
10. K. K. V. Toh and N. A. M. Isa "Noise Adaptive Fuzzy Switching Median Filter for Salt-and-Pepper Noise Reduction", *IEEE Trans. Consumer Electron*., vol. **17(3)**, 281–284, (2010).
11. N. I. Petrovic and V. Crnojevic, " Universal impulse noise filter based on genetic programming," *IEEE Trans. Image Process.*, vol. **(17)(7)**, 1109-1120, (2008).
12. M. Ismaeil, K. Pritamdas, K. J. K. Devi and S. Goyal, "Performance analysis of new adaptive decision based median filter on FPGA for impulsive noise filtering," *1st International Conference on Electronics, Materials Engineering and Nano-Technology (IEMENTech)*, Kolkata, 2017, pp. 1-5, (2017).
13. M. S. Darus, S. N. Sulaiman, I. S. Isa, Z. Hussain, N. M. Tahir and N. A. M. Isa, "Modified hybrid median filter for removal of low density random-valued impulse noise in images," *6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Batu Ferringhi, 2016, pp. 528-533, (2016).

Proc. of the 2019 IEEE 6th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA 2019)
27-29 August 2019, Kuala Lumpur, Malaysia

Identification of Continuous-time Hammerstein System using Sine Cosine Algorithm

E. F. Junis[#1], J. J. Jui[#1], M. H. Suid[#1], M. A. Ahmad[#1]

[#1]*Faculty of Electrical and Electronics Engineering, University Malaysia Pahang*
University Malaysia Pahang, 26600, Pekan, Pahang, Malaysia

[1]erafareha@gmail.com, [1]julakha.ump@gmail.com, [1]mhelmi@ump.edu.my, [1]mashraf@ump.edu.my

***Abstract*— This paper presents the development of identification of continuous-time Hammerstein systems based on Sine Cosine Algorithm (SCA). Here, the structure of the nonlinear is assumed to be known, and the structure of the linear subsystem which is the system order is assumed to be available. The SCA based method is then used to estimate the parameters in both the linear and nonlinear parts based on the given input and output data. Two numerical examples are given to illustrate the effectiveness of the SCA based algorithm. A continuous-time Infinite Impulse Response (IIR) filter is considered in the linear part, while the nonlinear functions, such as quadratic and hyperbolic are considered in the nonlinear part. The analysis of the numerical results is observed in terms of the parameter identification error, the convergence curve of the objective function, the output response in the time domain and the linear system response in the frequency domain. The results show that the potential of SCA based algorithm in giving an accurate parameter estimation of the Hammerstein models, especially for low noise level.**

Keywords*— Hammerstein system; sine cosine algorithm.

I. INTRODUCTION

Generally, as the complexity of problems has been increasing, to create physical system by mathematical analysis approach can be quite difficult. However, with the advance of technology, system identification has given a great improvement where a mathematically equivalent to a given physical system model can be developed. We can describe system identification as an approach to identify or analyze the mathematical model of a system from measurements of the system input and output. System identification has a great significance for any systems whose inputs and outputs can be measured, for instance, in industrial process, control system and more.

To ease the difficulty of modelling, there are several nonlinear system identification methods that have been invented which are Hammerstein system, Wiener system and Hammerstein-Wiener system. Hammerstein system is a system model that consists of a nonlinear gain followed by linear dynamic element, while Wiener system consists of a linear subsystem and static nonlinear model, lastly Hammerstein-Wiener system that contains a linear system inserted between two or more nonlinear functions in series. Among these three methods, Hammerstein system is famous for its simple model structure and it has been widely used for nonlinear system identification. On top of that, the effectiveness has been proven by several real applications such as fuel cells [1] and actuator [2]. Some examples that have been used on the linear part of the system identification are, Moving Average, Autoregressive (AR) [3], ARMAX [4], Finite Impulse Response (FIR) [5] and Infinite Impulse Response (IIR) [5] models, while on the nonlinear part Volterra [6] and Bilinear [7] models are used. Hammerstein model is quite widely used as a system modelling as being discussed in the literature, such as designing controller using Hammerstein Single Input Single Output (SISO) model in discrete time [8], control of discrete-time Hammerstein systems based on the passivity theorem [9], detecting variation of actuator characteristics via estimation techniques [10], discrete fractional order Hammerstein system [11], designing robust controller for an uncertain discrete-time nonlinear function [12], as well as identification of discrete multivariate Hammerstein system by kernel regression [13]. On the other hand, various method for identification of Hammerstein system have been studied broadly such as continuous linearity [14], subspace method [15], the least square method [16], non-parametric identification method [17]. Besides, another method that using evolutional computation, such as cuckoo method [18][19], bacteria foraging [20], genetic algorithm [20] and particle swarm optimization [21][22].

Based on the literature that has been discussed before, we can observe that most author applied discrete-time Hammerstein system identification in their paper and lack of literature applying continuous time Hammerstein system identification. In addition, it can be seen that this is a great opportunity to applied another multi agent-based optimization which is Sine Cosine Algorithm (SCA) for continuous-time Hammerstein system identification due to some drawback encountered by single agent-based optimization which are high probability to trap in local minima as well as lack of information sharing that results to low accuracy of system identification. SCA is a multi-agent based optimization which its updating equation are established based on mathematical trigonometric Sine and Cosine function [23]. Besides, the simplicity of this Sine Cosine Algorithm was also part of the motivation which it is able to optimize real problems with unknown searching space effectively regardless of its simple optimization algorithm. SCA has been applied in various field of engineering and technology such as solving the unit commitment problem related to thermoelectric generation units in electrical power system [24], peak load forecasting

[25], trajectory problem for space shuttle vehicle [26] as well as the elimination of lower order 5th harmonic in cascaded five level inverters [27] , power maximization in wind farm [28].

In this paper, we propose Sine Cosine Algorithm (SCA) for system identification of Hammerstein model. Two numerical examples are used to test the effectiveness of proposed identification. Next, the performance is analyzed based on parameter identification error, time domain response and frequency response of linear system in continuous time signal.

II. PROBLEM SETTING

The continuous-time Hammerstein model is considered as shown in **Fig. 1**. This model is consisted of a nonlinear function f and a linear dynamical system H where it's differential operator

$(s : \frac{d}{dt})$ can be defined by:

$$H(s) = \frac{B(s)}{A(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + ... + b_0}{s^n + a_{n-1} s^{n-1} + ... + a_0}, \quad (1)$$

the nonlinear function f, is given by,

$$f(u(t)) = \sum_{d=1}^{D} c_i u^d(t), \quad (2)$$

if the nonlinearity is other than polynomial type, then it can be written as,

$$f(u(t)) = \sum_{d=1}^{D} c_i \varphi[u(t)], \quad (3)$$

where $\varphi[.]$ is a nonlinear function, the input is $u(t)$, while $\hat{u}(t)$ is the output signal after the nonlinear function, which is denoted by $\hat{u}(t) = f(u(t))$, and the output signal of $y(t)$ is labeled as $\tilde{y}(t)$ when it is injected by the noise signal $v(t)$. Finally, we can describe the function of $\tilde{y}(t)$ as,

$$\tilde{y}(t) = H(s) f(u(t)) + v(t). \quad (4)$$

Then, we address this identification problem for continuous-time Hammerstein system by assuming:

- The symbols m , n and d are known,
- Coefficients a_i (i= 0,1…, n-1) and b_i (i=0, 1,…,m), c_i (i=0,1,2…,m) are positive real numbers,
- $f(0) = 0$,
- $G(s)$ is minimum phase and stable,
- $b_m = 1$, such that $G(s)$ and $f(u(t))$ can be obtained solely.

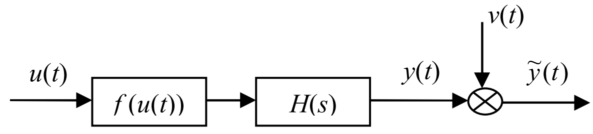


Fig. 1. Hammerstein model in continuous time

The following objective function is introduced to assess the identified model,

$$J(\hat{H}, \hat{f}) = \sum_{k=0}^{N} (\tilde{y}(kt_s) - \hat{y}(kt_s))^2 \quad (5)$$

where t_s is the sampling rate for $(u(t), \tilde{y}(t))$ $(t = 0, t_s, 2t_s, ..., Nt_s)$ and (N=0,1,2,…). The symbols $\hat{H}$ and $\hat{f}$ are the estimated linear system and nonlinear function of H and f, respectively, and $\hat{y} = \hat{H}(s)\hat{f}(u(t))$.

Next, we can describe the identification approach as follows,

Problem 2.1 The input-output data $(u(t), \tilde{y}(t))$ $(t = 0, t_s, 2t_s, ..., Nt_s)$ are given for the continuous-time Hammerstein system in **Fig. 1**. Our aim is to find $\hat{H}$ and $\hat{f}$ such that $J(\hat{H}, \hat{f})$ is minimized.

III. IDENTIFICATION APPROACH USING SINE COSINE ALGORITHM

This section discusses about the proposed method to solve problem in **Problem 2.1**. Then, the conventional Sine Cosine Algorithm is shortly explained and lastly the identification will be presented based on Sine Cosine Algorithm. Lastly, the application of SCA for identification of Hammerstein system is presented.

A. Review of sine cosine algorithm

Consider the optimization problem

$$\min_{x \in R^n} g(\boldsymbol{x}),$$

where the objective function is g: $R^n \to R$ and $\boldsymbol{x} \in R^n$ stand for the design variable.

Sine Cosine Algorithm begins with an initialization of a position of each search agents. The search agents are then moved based on the update equations given by,

$$x_i^{t+1} = x_i^t + r_1 \times \sin(r_2) \times \left| r_3 \, p_i^t - x_i^t \right|, r_4 < 0.5, \quad (6)$$

$$x_i^{t+1} = x_i^t + r_1 \times \cos(r_2) \times \left| r_3 \, p_i^t - x_i^t \right|, r_4 \geq 0.5, \quad (7)$$

where x_i^{t+1} is the location of the current solution in i-th dimension at t-th iteration, r_1 dictates the next position's region and can be described mathematically as,

$$r_1(t) = a - t\frac{a}{T}, \tag{8}$$

where a is a tunable constant, t is the current iteration, and T is the maximum number of iteration. Meanwhile, r_2 implies how long the step size should be to approach or to avoid the destination. Then, coefficient r_3 is denoted as a random weight to stochastically emphasize or deemphasize the effect of destination from each position. Finally, the coefficent r_4 switches between the Sine and Cosine functions in the probability of 50% as stated in (6) and (7).

B. Identification Approach

This part shows how to apply SCA algorithm for identification problem in section *A*. Using Sine Cosine algorithm, the optimization problem in (5) is modified to the objective function,

$$g(\boldsymbol{\theta}) = \sum_{k=0}^{N} (\tilde{y}(kt_s) - \hat{y}(kt_s))^2 \tag{9}$$

for the design variable

$$\theta = [\hat{b}_0, \hat{b}_1, ..., \hat{b}_{m-1}, \hat{a}_0, \hat{a}_{n-1}, ..., \hat{c}_0, \hat{c}_1, ..., \hat{c}_m]. \tag{10}$$

For a fixed $\boldsymbol{\theta}$, it is necessary to compute the value of $g(\boldsymbol{\theta})$, this is simplified to compute $\hat{y}(kt_s)$, which is shown as follow. Firstly, the input signal $u(t)$ is generated with zero-order hold, where $u(t)$ $(t = 0, t_s, 2t_s, ..., Nt_s)$. Next, we calculate

$$\hat{y}(t) = \frac{b_m s^m + \hat{b}_{m-1} s^{m-1} + ... + \hat{b}_0}{s^n + \hat{a}_{n-1} s^{n-1} + ... + \hat{a}_0} \hat{f}(u(t)) \tag{11}$$

which is also in the continuous-time signal. Then, $\hat{y}(t)$ is sampled to $\hat{y}(kt_s)$ at a fix sampling rate $k = 0, 1..., N$.

Finally, a solution to **Problem 2.1** is obtained using SCA based method. The procedure can be summarized as followed:

Step 1: Identify the maximum iterations *T*, of SCA. Determine the number of agent and generate the starting value for $\boldsymbol{x}(0)$.

Step 2: For the objective function $g(\boldsymbol{\theta})$, perform the SCA in section 3.

Step 3: Obtain the best solution x_i^{t+1}.

Step 4: When *T* is reached, we obtain $\boldsymbol{x}^* = \boldsymbol{x}^T$.

Then, $\boldsymbol{\theta}^* = [x_1^*, x_2^*, x_3^*, ..., x_n^*]$ is a solution for **Problem 2.1**.

IV. NUMERICAL EXAMPLE

This section discussed about the effectiveness of the proposed identification approach by demonstrating it by some numerical examples.

A. Example 1

Consider the system:

$$H(s) = \frac{B(s)}{A(s)} \tag{12}$$

$A(s) = s^6 + 10.0000\,s^5 + 54.7700\,s^4 + 156.8000\,s^3 + 87.0843\,s^2 + 25.2810s + 4.0197.$

$B(s) = s^3$

$$f(u(t)) = 125(u(t) + 0.5u^2(t) + 0.25u^3(t)). \tag{13}$$

For input $u(t)$, a varying amplitude of Pseudo Random Binary Sequence (PRBS) signal is utilized, while $v(t)$ is a white noise with a zero mean and variance $\sigma_v^2 = 0.01$.The amplitude of input signal is varied in a range [-1, 1]. The signal $\tilde{y}(t)$ is sampled at $t_s = 1\times10^{-3}$. The parameter identification error is then calculated to show how close the obtained identified parameters with the true parameters, as given by

$$\xi = \left\| \left[\frac{\theta_1 - \bar{\theta}_1}{\bar{\theta}_1}, ..., \frac{\theta_p - \bar{\theta}_p}{\bar{\theta}_p} \right]^T \right\|_2 \tag{14}$$

where $\bar{\theta} \in R$ is a vector and $\bar{\theta}_i$ is the i-th component of the vector $\bar{\theta}$. We can see the result more clearly based on the nonlinear function *f* graph and the bode plot of the transfer function *H* with difference noise variances, which are 0.01, 0.25, and 1.0, as shown in **Fig. 2** and **Fig. 3**, respectively. The blue color line indicates the actual graph for both **Fig. 2** and **Fig. 3** while red, black and pink color lines are for the noise variances $\sigma_v^2 = 0.01$, $\sigma_v^2 = 0.25$ and $\sigma_v^2 = 1.0$ respectively. For the nonlinear graph in **Fig. 2**, the nonlinear graph result shows that SCA can approximate the actual graph of the nonlinear function. But, as the noise variance increases, the difference between the results and actual graph is getting bigger, particularly for noise variance, $\sigma_v^2 = 1.0$. While, for the bode plot response shown in **Fig. 3**, it is shown that, the graph results are almost identical with the actual bode plot response. Based on this result, SCA can closely determine the linear subsystem in the Hammerstein model. Besides, we can observe the result based on the value of parameter identification which is relatively small for each noise variance $\sigma_v^2 =$ 0.01, 0.25 and 1.0 which is 0.8551, 0.8622 and 1.2436 respectively.

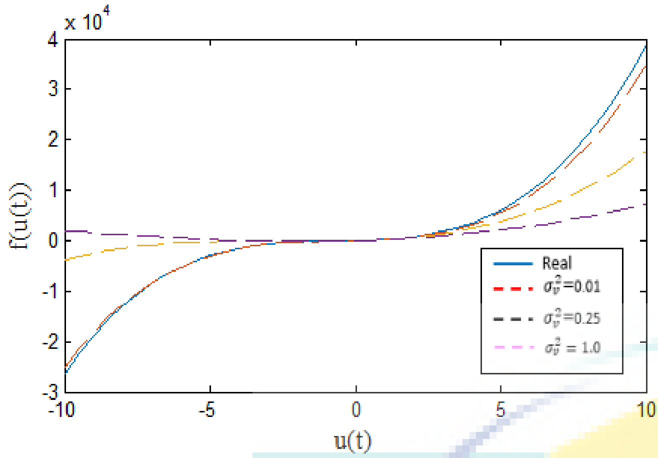


Fig. 2. Nonlinear function plot for Example 1.

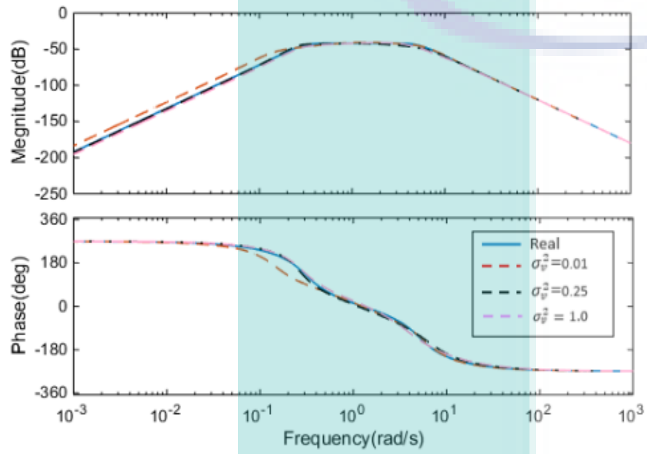
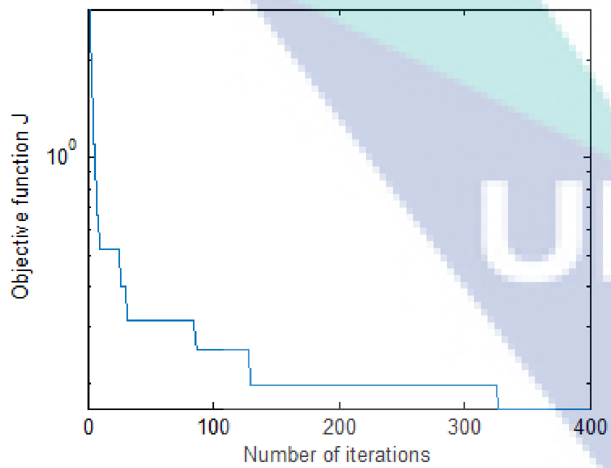
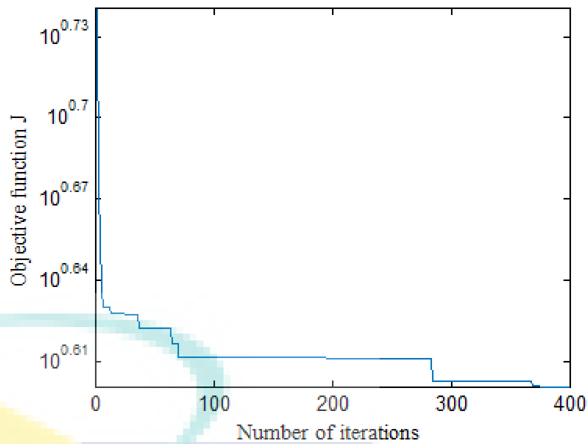


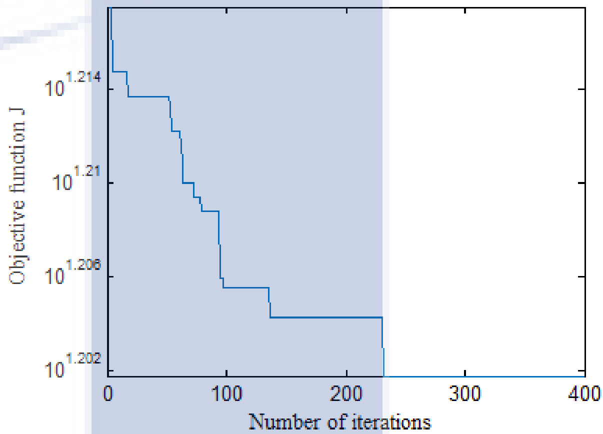
Fig. 3. Bode plot of linear system for Example 1



a. Convergence Curve with $\sigma_v^2 = 0.01$



b. Convergence Curve with $\sigma_v^2 = 0.25$



c. Convergence Curve with $\sigma_v^2 = 1.0$

Fig. 4. Convergence curve for Example 1 with different noise

B. *Example 2*

In the seconde example, a 4th order transfer function with complex poles is considered for linear part. Meanwhile a tangent hyperbolic function is used for nonlinear part. Both linear and nonlinear subsystems are given by

$$H(s) = \frac{B(s)}{A(s)} \tag{15}$$

$A(s) = s^4 + 5s^3 + 408s^2 + 416s + 1600$,

$B(s) = s + 0.25$,

$$f(u(t)) = 250\tanh(1.5u(t)) \,. \tag{16}$$

In this example, same input $u(t)$, white noise $v(t)$, t_s and N and also the color line indicator are used as in Example 1. The nonlinear function f graph and bode plot of H with same noise variances are shown in **Fig. 5** and **Fig. 6,** respectively. It shows that the SCA can almost identify the actual nonlinear graph for noise variance, $\sigma_v^2 = 0.01$, but the result start to degrade as the noise level increases, particularly for $\sigma_v^2 = 1.0$.

As for the bode plot response, for noise level $\sigma_v^2 = 0.01$, SCA can almost identify the actual bode plot response for Example 2, but bode plot response is starting to differ at the noise level 0.25 and 1.0 especially at the low frequency's response region. However, for the parameter identification error, the statistical values of the mean, best, worst and standard deviation are relatively small for $\sigma_v^2 = 0.01$, 0.25 and 1.0, which are 0.1312, 0.8965 and 1.6102 respectively.

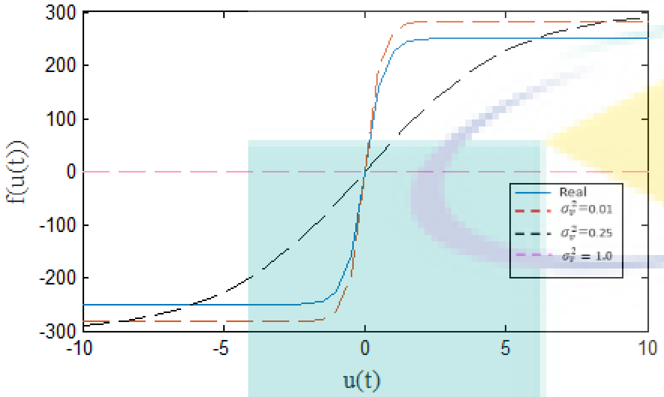


Fig. 5. Nonlinear function plot for Example 2

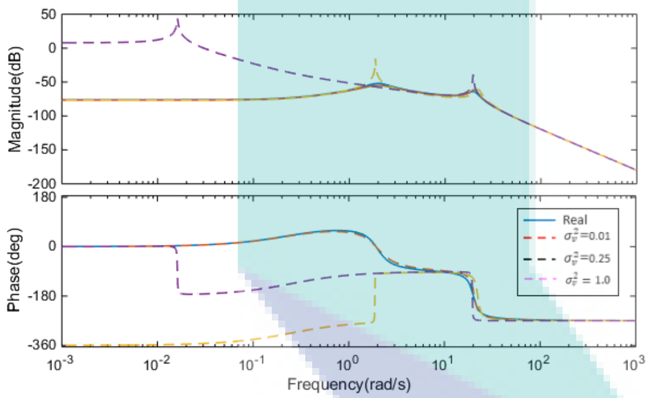
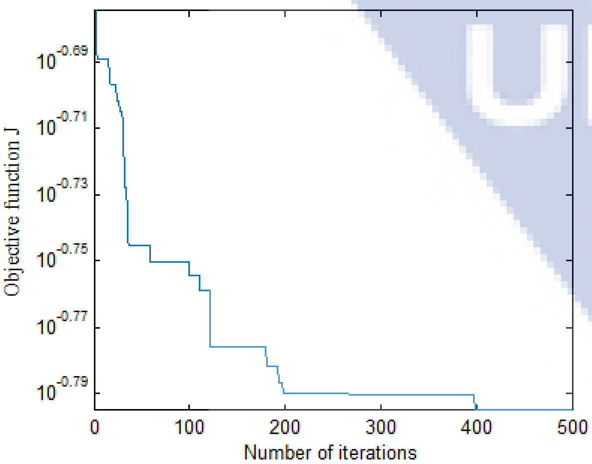
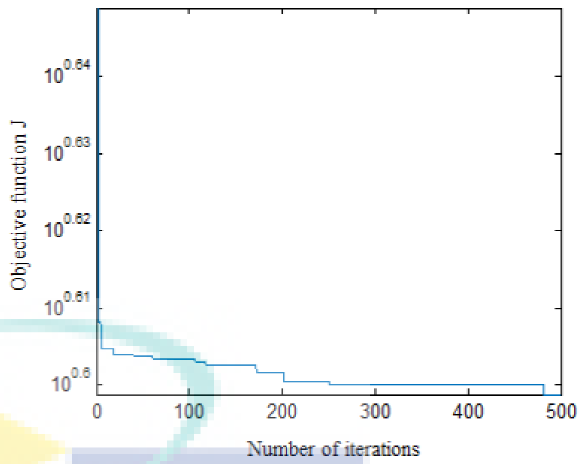


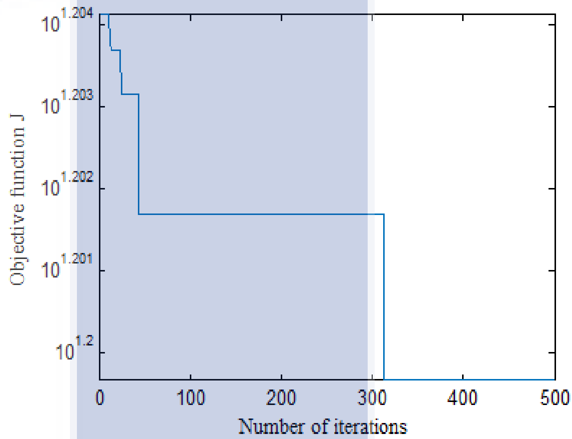
Fig. 6. Bode plot of linear system for example 2



a. Convergence Curve with $\sigma_v^2 = 0.01$



b. Convergence Curve with $\sigma_v^2 = 0.25$



c. Convergence Curve with $\sigma_v^2 = 1.0$

Fig. 7. Convergence curve for Example 2 with difference noise variance.

V. CONCLUSION

In this paper, the identification of continuous-time Hammerstein systems using Sine Cosine Algorithm has been proposed. Based on the two examples, it shows a good potential of SCA-based method in identifying the continuous-time Hammerstein model especially in low noise level. In particular, the SCA-based method is able to provide acceptable values of parameter identification error for both examples especially at noise variance of 0.01. In the future, it is good to improve the SCA-based method such that is can handle the identification of continuous-time Hammerstein systems in the presence of high level of noise.

ACKNOWLEDGMENT

This work was fully supported by Internal Research Grant UMP RDU1703151 from Research Management Centre University Malaysia Pahang.

REFERENCES

[1] The W. Wu and D. Jhao, "Identification and Control of a fuel cell system using wavelet network-based," vol. 45, no. 15. IFAC, 2012.

[2] A. M. Ousaid and M. G. M. Boukallel, "Adaptive Control Based on an Extended Kalman Filter for Hammerstein System -Application to Piezoelectric Actuators," vol. 43, no. 18. IFAC, 2015.

[3] O. Ait, A. Achuthan, K. Janoyan, and P. Marzocca, "Recursive wind speed forecasting based on Hammerstein Auto-Regressive model," Appl. Energy, vol. 145, pp. 191–197, 2015.

[4] Q. Jin, Y. Xing, X. Du, Y. Niu, and W. Cai, "Expectation – maximization algorithm based identification of Hammerstein nonlinear ARMAX systems with Gaussian mixture noises," 2018 37th Chinese Control Conf., pp. 1746–1751, 2018.

[5] L. Gan and E. Abd-elrady, "Adaptive Predistortion of IIR Hammerstein Systems Using the Nonlinear Filtered-x LMS Algorithm," 2008 6th Int. Symp. Commun. Syst. Networks Digit. Signal Process., vol. 2, no. 1, pp. 702–705, 2008.

[6] M. Mosleh and M. Otadi, "Least squares approximation method for the solution of Hammerstein – Volterra delay integral equations," Appl. Math. Comput., vol. 258, pp. 105–110, 2015.

[7] A. Ramos, J. L. M. De Carvalho, and P. Lopes, "Control Engineering Practice Identification of a Benchmark Wiener – Hammerstein : A bilinear and Hammerstein – Bilinear model approach," vol. 20, pp. 1156–1164, 2012.

[8] S. Qingbo and D. Yuanming, "A Model Reference Adaptive Controller Design for Discrete Hammerstein Systems," 2007 Chinese Control Conf., pp. 743–747, 2007.

[9] K. H. Chan, I. Chemistry, J. Kruttschnit, and I. Road, "Contr.01," vol. 61, no. 2, 2004.

[10] Z. Sun, Y. Cheng, N. Xi, C. Li, S. Bi, J. Chen, and D. Guo, "Discrete Nonlinear Contraction Theory based Adaptive Control Strategy for a Class of Hammerstein Systems with Saturated Hysteresis."

[11] D. V Ivanov, "Identification Discrete Fractional Order Hammerstein Systems," 2015 Int. Sib. Conf. Control Commun., no. 1, pp. 1–4, 2015.

[12] A. Znidi, K. Dehri, and A. S. Nouri, "Indirect Adaptive Sliding mode control for Uncertain Discrete Hammerstein Systems Using Inverse Nonlinear Function," 2017 6th Int. Conf. Syst. Control, pp. 634–639, 2017.

[13] A. Krzysak, "Hammerstein system by kernel regression estimate ," no. 1.

[14] T. P. Characteristics and J. Vörös, "Identification of Hammerstein Systems With," vol. 52, no. 12, pp. 865–869, 2005.

[15] K. Jalaleddini and R. E. Kearney, "An identification algorithm for Hammerstein systems using subspace method," Proc. 2011 Am. Control Conf., pp. 4793–4797, 2011.

[16] I. Goethals, K. Pelckmans, J. A. K. Suykens, and B. De Moor, "Identification of MIMO Hammerstein Models using Least Squares Support Vector machines," Automatica, vol. 41, no. 7, pp. 1263–1272, 2005.

[17] R. S. Risuleo, G. Bottegal, and H. Hjalmarsson, "A nonparametric kernel-based approach to Hammerstein system identification," Automatica, vol. 85, pp. 234–247, 2017.

[18] A. Gotmare, R. Patidar, and N. V. George, "Nonlinear system identification using a cuckoo search optimized adaptive Hammerstein model," Expert Syst. Appl., vol. 42, no. 5, pp. 2538–2546, 2015.

[19] A. H. Gandomi, X. S. Yang, and A. H. Alavi, "Erratum: Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems (Engineering with Computers DOI:10.1007/s00366-011-0241-y)," Eng. Comput., vol. 29, no. 2, p. 245, 2013.

[20] P. S. Pal, A. Ghosh, S. Choudhury, D. Debapriya, R. Kar, D. Mandal, and S. P. Ghoshal, "Identification of Hammerstein model using bacteria foraging optimization algorithm," Int. Conf. Commun. Signal Process. ICCSP 2016, pp. 1609–1613, 2016.

[21] S. J. Nanda, G. Panda, and B. Majhi, "Improved identification of Hammerstein plants using new CPSO and IPSO algorithms," Expert Syst. Appl., vol. 37, no. 10, pp. 6818–6831, 2010.

[22] L. J. Liu and H. M. Li, "Intelligence behavior in SE," Proc. - Work. Intell. Inf. Technol. Appl. IITA 2007, pp. 82–86, 2007.

[23] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems," Knowledge-Based Syst., vol. 96, pp. 120–133, 2016.

[24] L. M. De Oliveira, F. F. Panoeiro, I. C. S. Junior, L. W. Oliveira, and I. Member, "Application of the Sine Cosine Optimization Algorithm for Thermal Unit Commitment," 2018 Simp. Bras. Sist. Eletr., no. i, pp. 1–6.

[25] S. Hamdan, S. Binkhatim, A. Jarndal, and I. Alsyouf, "On the Performance of Artificial Neural Network with Sine-Cosine Algorithm in Forecasting Electricity Load Demand," 2017.

[26] A. Banerjee and M. Nabi, "Re-Entry Trajectory Optimization for Space Shuttle using Sine-Cosine Algorithm," pp. 73–77.

[27] N. Sahu, "Selective Harmonic Elimination in Five Level," 2017 IEEE Int. Conf. Power, Control. Signals Instrum. Eng., pp. 385–388, 2017.

[28] M.H. Suid, M.Z. Mohd Tumari, M.A. Ahmad, "A modified sine cosine algorithm for improving wind plant energy production," International Journal of Electrical Engineering and Computer Science, Vol. 16(1), 2019, pp. 101-106.