Ontological Intelligent Agent for Impulse Noise Removal

Chang-Shing Lee[1,*], Mei-Hui Wang[2], and Chin-Yuan Hsu[3]

[1] Department of Computer Science and Information Engineering, National University of Tainan, Tainan 700, Taiwan
leecs@mail.nutn.edu.tw

[2] Department of Computer Science and Information Engineering, National University of Tainan, Tainan 700, Taiwan
mh.alice.wang@gmail.com

[3] Advance e-Commerce Institute, Institute for Information Industry
Tainan 700, Taiwan
hsucy@iii.org.tw

Received 27 February 2007; Revised 26 March 2007; Accepted 30 March 2007

Abstract. This paper presents an ontological intelligent agent to remove impulse noise from highly corrupted images. It contains an image noise ontology to represent the image noise knowledge for the agent, a fuzzy inference mechanism for noise detection and removal, and an intelligent learning process for automatically generating the fuzzy numbers of the agent. The working environment for the intelligent agent is defined and the image noise ontology referred by the fuzzy inference mechanism is utilized to perform the task of noise removal. Then, using orthogonal array and factor analysis, a genetic algorithm is applied to the intelligent learning process. Finally, the fuzzy numbers of the image noise ontology are adjusted via the intelligent learning process to increase the performance of the intelligent agent. Experimental results show that the proposed approach can achieve better results than the state-of-the-art filters based on the criteria of Mean-Absolute-Error and Mean-Square-Error. Besides, on the subjective evaluation of those filtered images, the proposed approach can also generate a higher quality of global restorations.

Keywords: Ontology, Intelligent Agent, Fuzzy Inference, Genetic Learning, Image Processing

1 Introduction

The ontology is a computational model of some portions of the world. It is a collection of key concepts and their inter-relationships collectively providing an abstract view of an application domain [1]. In addition, ontology is no longer a mere research topic, but its relevance has been recognized in several practical fields. Recently, the research on the ontology has been spread widely to be critical components in the knowledge management, Semantic Web, business-to-business applications, and several other application areas [2]. For example, in [1] and [3], Lee *et al.* proposed a fuzzy ontology application to news summarization and a genetic fuzzy agent using the ontology model for meeting scheduling system, respectively. Alani *et al.* [4] proposed an automatic ontology-based knowledge extraction from Web documents and used it to generate personalized biographies. Burstein [5] presented a dynamic invocation of semantic web services using unfamiliar ontologies. Hameed *et al.* [6] presented an approach to acquire knowledge and construct multiple experts' ontologies in a uniform way. Navigli *et al.* [7] presented an OntoLearn system for automated ontology learning to extract relevant domain terms from a corpus of text, relate them to appropriate concepts in a general-purpose ontology, and detect taxonomic and other semantic relations among the concepts.

An agent is a physical or virtual entity that is capable of acting in an environment and communicating directly with other agents. The concept of the action is based on the fact that the agents carry out actions, which are going to modify the agents' environment, and the future decision making [1][8]. There are many various definitions for an agent. For example, an agent possesses skills and can offer service, it is capable of perceiving its environment, it is driven by a set of tendencies, and so on [8]. Furthermore, an intelligent agent is more powerful than an agent because of the reasoning and learning capabilities [9]. Delen and Pratt designed and developed an intelligent decision support system for manufacturing [10]. Hamdi presented a multi-agent customization system based on

[*] Correspondence author

machine learning mechanism for web mining [11]. Lee *et al.* proposed a genetic fuzzy agent for meeting scheduling system [3]. Wang applied the concept of agent-based control for networked systems in control theory to traffic and transportation management [12]. Yan *et al.* proposed a perceptron-based medical decision support system for the diagnosis of heart diseases [13]. Grossklags and Schmidt studied how software agents affect the market behavior of human traders [14].

An image may be corrupted during the process of image transmission, so that a number of approaches have been developed for removing impulse noise from corrupted image. For example, the median filter [15] performs well for low corrupted image, but it does not work efficiently when noise rate is above 50%. Eng *et al.* [16] presented a novel switching-based median filter to remove impulse noise and preserve signal details. Abreu *et al.* [17] proposed an efficient nonlinear algorithm, called signal-dependent rank ordered mean (SD-ROM) filter, to suppress impulse noise from highly corrupted images while preserving details and features. Russo [18] presented the hybrid neuro-fuzzy filters for highly corrupted impulse noise removal. Wang *et al.* [19] presented a histogram-based fuzzy filter (HFF) to the restoration of noise-corrupted images, which is particularly effective at removing highly impulsive noise while preserving image details. Lukac [20] proposed an adaptive vector median filter for impulse noise suppression and outliers rejection in multi-channel images. Pok *et al.* [21] proposed a decision-based, signal-adaptive median filtering algorithm for removal of impulse noise. Chang *et al.* [22] proposed a classifier-augmented median filter for impulse noise removal from images. Liu [23] proposed a fuzzy neural network filter for impulse noise removal from images. Lin *et al.* [24] proposed a multi-channel filtering by gradient information for impulse noise removal from images. Ng and Ma [25] proposed a switching median filter (SMF) to effectively remove noise from the extremely corrupted image. Lee *et al.* proposed a genetic-based fuzzy image filter (GFIF) [27], a multi-dimension genetic fuzzy filter (MGFF) [28], and an ontology-based genetic fuzzy filter (OGFF) [26] to remove impulse noise from highly corrupted images and preserve the quality of fine details and textures. W. Luo [29] proposed an efficient impulse detection algorithm (EIDA) to remove impulse noise.

In this paper, we propose an ontological intelligent agent (*OIA*) based on the GFIF and OGFF filters to remove impulse noise from highly corrupted images. The proposed agent consists of an image noise ontology to represent the image noise knowledge for the intelligent agent, a fuzzy inference mechanism for noise detection and removal, and an intelligent learning process to automatically generate the fuzzy numbers stored in the ontology for the agent. Through the constructed noise ontology and the technique of noise detection, the *OIA* can automatically generate the optimal parameters for the fuzzy inference mechanism and use the orthogonal array to help the learning process of the OGFF. From the experimental results, the proposed ontological intelligent agent not only performs better than our previously proposed works [27][28] and other works [18][25][29], but also largely outmatches the state-of-the-art methods in the literature. The reminder of this paper is organized as follows. In Section 2, we describe the intelligent agent based on the image noise ontology, including the ontology structure, the image noise ontology construction for ontological intelligent agent, and the ontological intelligent agent. Section 3 describes the intelligent genetic learning process for the ontological intelligent agent. The experimental results are shown in Section 4. Finally, conclusions are drawn in Section 5.

2 The Intelligent Agent based on Image Noise Ontology

This paper proposes an ontological intelligent agent for image processing. First, the structure and construction of the image noise ontology for the intelligent agent are briefly described in Section 2.1 and 2.2, respectively. Then, we present the intelligent agent in Section 2.3.

2.1 Ontology Structure

Fig. 1 shows the domain ontology structure adopted in this paper, including the *domain layer*, the *category layer*, the *sub-category layer*, and the *class layer*. The domain ontology contains three types of relationships, namely the "*generalization*," the "*aggregation*," and the "*association*." The "*generalization*" relation denotes a semantic relationship between the domain name and the categories in the *category layer*. The "*aggregation*" relation denotes a semantic relation among the *category layer*, the *sub-category layer*, and the *class layer*. The relationship between two concepts in the *class layer* is the "*association*." The *domain layer* represents the domain name of an ontology, and comprises various categories defined by domain experts. The *category layer* defines several categories, namely "category *1*, category *2*, …, category *k*." Each concept in the *class layer*, contains a concept name C_i, an attribute set $\{A_{C_i 1},\cdots,A_{C_i q_i}\}$ and an operation set $\{O_{C_i 1},\cdots,O_{C_i q_i}\}$ for an application domain [1].

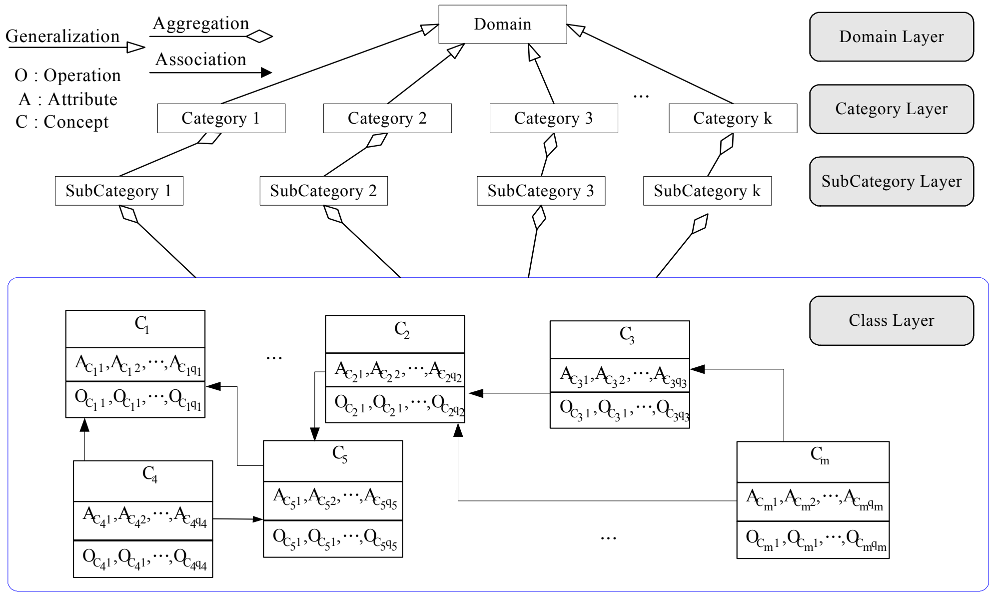


Fig. 1. Structure of domain ontology

2.2 Image Noise Ontology Construction for Ontological Intelligent Agent

In this section, we developed the image noise ontology for the *OIA*, based on the Knowledge-Engineering methodology [30]. The steps of the developing the image noise ontology are listed as below.

- Step.1 Determine the domain and scope of the ontology.
- Step.2 Consider reusing existing ontologies.
- Step.3 Enumerate important terms in the ontology.
- Step.4 Define the classes and the class hierarchy.
- Step.5 Define the properties of classes-slots.
- Step.6 Define the facets of the slots.
- Step.7 Create instances.

Fig. 2 shows the image noise ontology for the *OIA*. This image noise ontology has five layers, called the *domain layer*, *category layer*, *sub-category layer*, *fuzzy variable layer* and *linguistic term layer*. The domain name is the “image noise ontology.” The *category layer* has several categories such as the “Additive Impulse Noise”, “Uniform Mixed Noise”, and so on. The next layer, *sub-category layer*, also has some sub-categories such as “Salt & Peppers Noise”, “Additive Long-tailed Noise”, “Additive Middle-tailed Noise”, and so on. In the *Fuzzy Variable Layer* and *Linguistic Term Layer*, each concept contains a concept name, an attribute set and an operation set for an application domain. For example, a *Fuzzy Luminance* concept has concept name, “Fuzzy Luminance,” and its attributes are “Linguistic Term *1*,” “Linguistic Term *2*,” …, and “Linguistic Term *n*.”

In this paper, we adopted the trapezoidal function to be the membership function. The trapezoidal membership function $f_A(x:a,b,c,d)$ of fuzzy set A is represented as $A = [a, b, c, d]$ and shown in (1). Fig. 3 illustrates an example for the *luminance* fuzzy variable with five linguistic terms. The membership degree is usually a value in the range [0, 1], where “1” denotes a full membership, and “0” denotes no membership. There are five fuzzy numbers “*VDK*,” “*DK*,” “*MD*,” “*BR*,” and “*VBR*” to denote the linguistic terms “*Very Dark*,” “*Dark*,” “*Medium*,” “*Bright*,” and “*Very Bright*,” respectively, for the gray level image.

$$f_A(x:a,b,c,d)=\begin{cases} 0, & x<a \\ (x-a)/(b-a), & a\le x<b \\ 1, & b\le x<c \\ (d-x)/(d-c), & c\le x<d \\ 0, & x\ge d \end{cases} \quad (1)$$

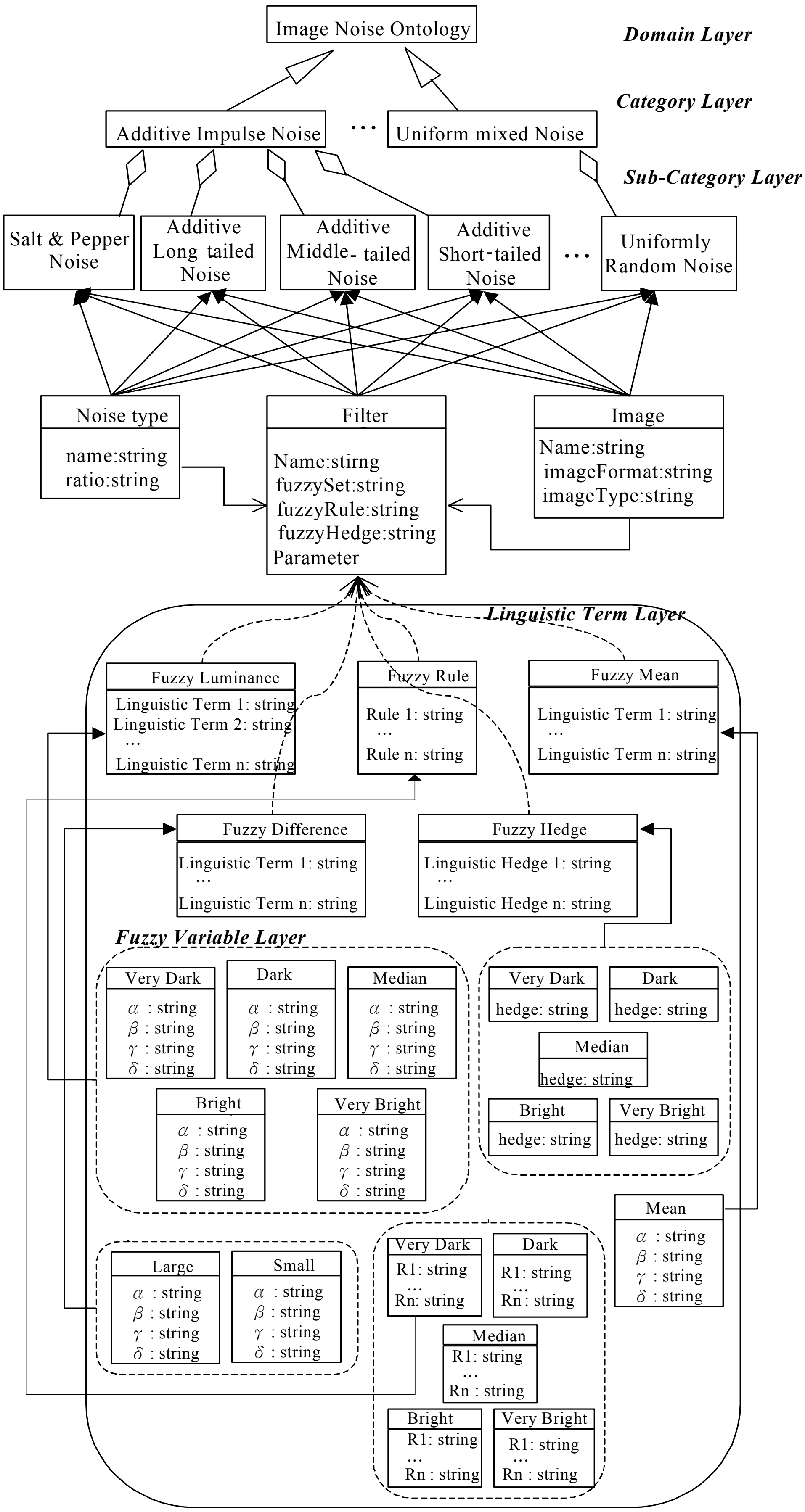


Fig. 2. Image noise ontology

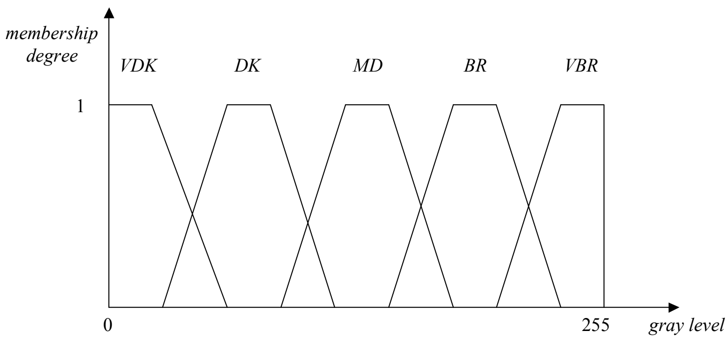


Fig. 3. The *luminance* fuzzy variable with five linguistic terms

2.3 Ontological Intelligent Agent

This section describes the ontological intelligent agent (*OIA*) for image processing. First, we define the working environment for the intelligent agent as follows. The signal space ζ of the *OIA* is a set of digital signals together with five operators, namely the *arithmetic sum*, the *arithmetic difference*, the *bounded sum*, the *bounded difference* and the *scalar multiplication* [31]. The gray level of each signal is represented by an *n*-bits value so it is ranged in $[0, 2^{n-1}]$. The *bounded sum* associates two signals $x, y \in \zeta$ with a signal $x \oplus y \in \zeta$ where the operator $\oplus$ is the *bounded sum* of two signals *x* and *y*. The *bounded difference* associates two signals $x, y \in \zeta$ with a signal $x \Theta y \in \zeta$, where the operator Θ is the *bounded difference* between two signals *x* and *y*. The *scalar multiplication* associates a signal $x \in \zeta$ and a scalar $\lambda \in [0, 1]$ with a signal $\lambda x \in \zeta$ calculated by the arithmetic multiplication of signal *x* and scalar value λ .

[*Definition 1*] Fuzzy Signal Space for *OIA*

A fuzzy signal space is a signal space whose partitions are determined by fuzzy numbers stored in the ontology. The partitions of the fuzzy signal space, including *fuzzy normal subspace*, *fuzzy negative subspace*, *fuzzy positive subspace* and *fuzzy undecided subspace*, are determined using the following four fuzzy numbers:

1. The fuzzy number *OIA_normal* of fuzzy normal subspace is defined as follows:
   $OIA\_normal = [\alpha_{nor}, \beta_{nor}, \gamma_{nor}, \delta_{nor}]$
2. The fuzzy number *OIA_negative* of fuzzy negative subspace is defined as follows:
   $OIA\_negative = [\alpha_{neg}, \beta_{neg}, \gamma_{neg}, \delta_{neg}]$
3. The fuzzy number *OIA_positive* of fuzzy positive subspace is defined as follows:
   $OIA\_positive = [\alpha_{pos}, \beta_{pos}, \gamma_{pos}, \delta_{pos}]$
4. The fuzzy number *OIA_undecided* of fuzzy undecided subspace is defined as follows:
   $OIA\_undecided = [\alpha_{und}, \beta_{und}, \gamma_{und}, \delta_{und}]$

Fig. 4 shows the structure of the *OIA*. First, a noise-free image is input to learn the best parameters of the image noise ontology during the intelligent learning process. Meanwhile, this intelligent learning process also retrieves the information stored in the image noise ontology to perform the learning process, and then restores the learning parameters back to the image noise ontology. The structure of the fuzzy inference mechanism consists of three layers, including the *fuzzy linguistic layer*, the *fuzzy term layer*, and the *rule layer* [27]. There are three kinds of nodes in this inference mechansim, including the fuzzy linguistic nodes in the *fuzzy linguistic layer*, the fuzzy term nodes in the *fuzzy term layer*, and the rule nodes in the *rule layer*. A fuzzy linguistic node represents a fuzzy variable. A fuzzy term node represents the mapping degree of the fuzzy variable. A rule node represents a rule and decides the final firing strength of that rule during inferring. The fuzzy linguistic nodes just directly transmit input values to the next layer. Each fuzzy variable of the *fuzzy linguistic layer* appearing in the premise part is represented with a condition node. Each of the outputs of the condition node is connected to the rule nodes in the *rule layer* to constitute a condition specified in some rules. The *fuzzy term layer* performs the first inference step to compute matching degrees. The links in the *rule layer* are used to perform precondition matching of fuzzy logical rules. The fuzzy inference rules are listed in Table 1. The *fuzzy mean process* performs the fuzzy mean of input variables *F_mean*. There are five computation functions including $f_{comp}(\cdot)$, $f_{-}(\cdot)$, $f_{x_1}(\cdot)$, $f_{x_2}(\cdot)$ and $f_{+}(\cdot)$, and two membership functions including the f_{large} and the f_{small} utilized in *fuzzy decision process* [27].

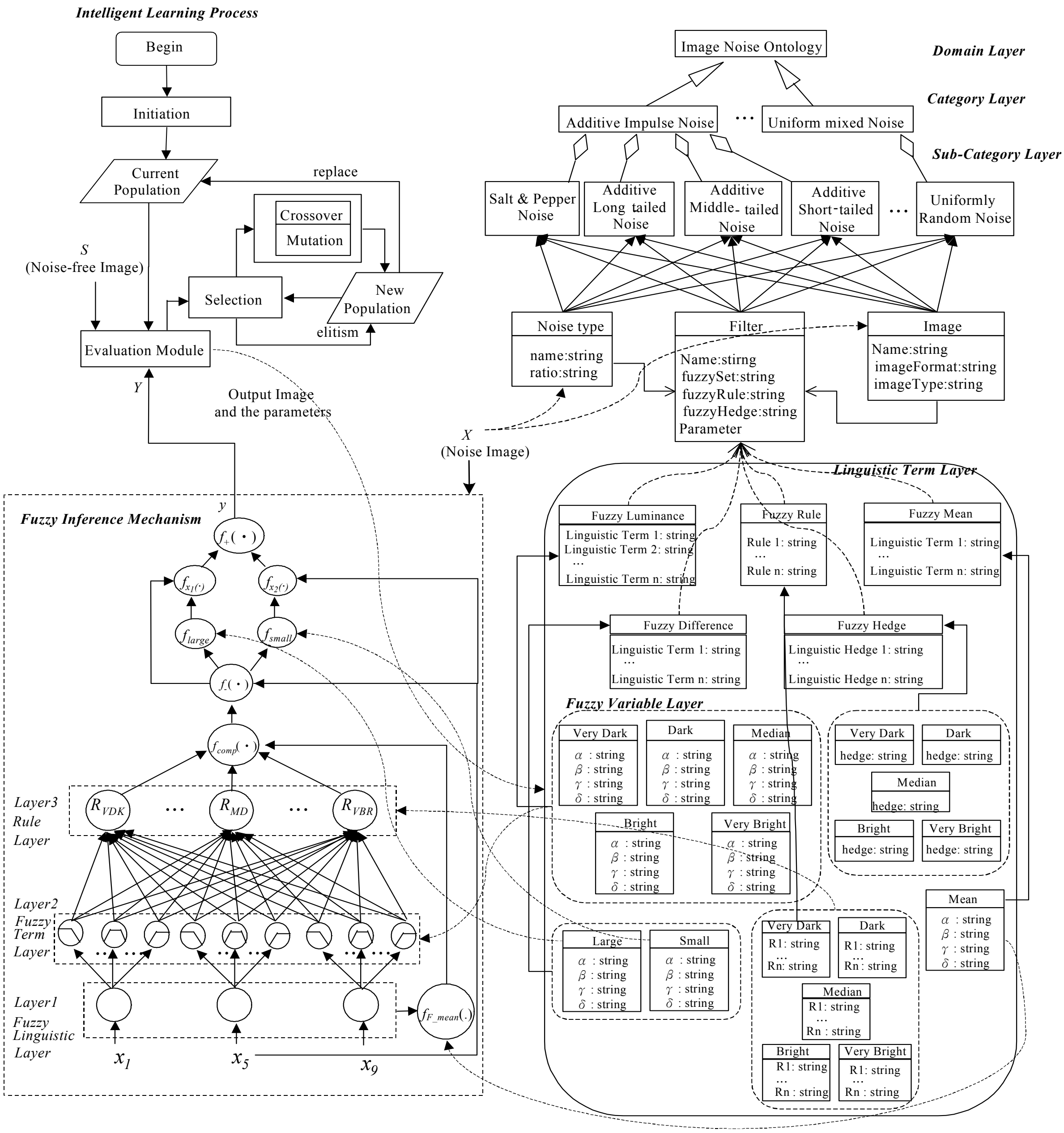


Fig. 4. Structure of *Ontological Intelligent Agent*

Table 1. Fuzzy inference ruels

Rule VDK:	IF (x_1 is *VDK* with weight w_{11}) AND (x_2 is *VDK* with weight w_{21}) AND … AND (x_9 is *VDK* with weight w_{91}) THEN μ_1^3 is *VDK*
Rule DK:	IF (x_1 is *DK* with weight w_{12}) AND (x_2 is *DK* with weight w_{22}) AND … AND (x_9 is *DK* with weight w_{92}) THEN μ_2^3 is *DK*
Rule MD:	IF (x_1 is *MD* with weight w_{13}) AND (x_2 is *MD* with weight w_{23}) AND … AND (x_9 is *MD* with weight w_{93}) THEN μ_3^3 is *MD*
Rule BR:	IF (x_1 is *BR* with weight w_{14}) AND (x_2 is *BR* with weight w_{24}) AND … AND (x_9 is *BR* with weight w_{94}) THEN μ_4^3 is *BR*
Rule VBR:	IF (x_1 is *VBR* with weight w_{15}) AND (x_2 is *VBR* with weight w_{25}) AND … AND (x_9 is *VBR* with weight w_{95}) THEN μ_5^3 is *VBR*

The final output y of *fuzzy decision process* is the computing result of $f_+(\cdot)$. The membership functions, f_{large} and f_{small}, define the detailed preserving process of the mechanism. In fact, the function $f_-(\cdot)$ can be interpreted as a measure function of the correction process. If this measure is large, a full correction is allowed. If this measure is small, on the contrary, the correction is further reduced in order to better preserve the quality of fine details and textures. Next, the Protege 2000 [32], developed by Stanford University to build the Web Ontology Language (OWL), is used to build the noise ontology.

3 Intelligent Genetic Learning Process for *OIA*

This section introduces the intelligent genetic learning process for the *OIA*. The learning approach proposed by Ho *et al.* [33] is used to learn the parameters of fuzzy numbers and fuzzy rules. Fig. 5 shows the main parameters of the image fuzzy numbers [26]. The four components of the fuzzy numbers to be encoded are the membership functions of the fuzzy variables (CS_{a1} and CS_{a2}), the fuzzy rules (CS_{b1}) and the linguistic modifiers (CS_{b2}) of the linguistic terms: (1) the CS_{a1} denotes the parameters of fuzzy numbers and is represented by 20 genes; (2) the CS_{a2} denotes the parameters of fuzzy numbers *F_mean*; (3) the CS_{b1} denotes a group of 9-gene which directly encodes the binary weights { w_{ij} } where i = 1, …, 9 and j = 1, …, 5; (4) the CS_{b2} denotes the linguistic modifiers of the linguistic terms and is represented by 5 genes. For example, the set of binary weight {{0, 1, 0, 0, 1, 0, 1, 0, 0}, …, {0, 0, 1, 0, 0, 0, 0, 1, 0}, …, {1, 0, 0, 1, 0, 1, 0, 0, 1}} identifies the pattern defined by the set of indexes {{2, 5, 7}, …, {3, 8}, …, {1, 4, 6, 9}}, and the corresponding fuzzy rules are denoted as follows.

Rule *VDK*:

IF ((x_2 is *VDK*) AND (x_5 is *VDK*) AND (x_7 is *VDK*)) THEN μ_1^3 is *VDK*

⋮

Rule *MD*:

IF ((x_3 is *MD*) AND (x_8 is *MD*)) THEN μ_3^3 is *MD*

⋮

Rule *VBR*:

IF ((x_1 is *VBR*) AND (x_4 is *VBR*) AND (x_6 is *VBR*) AND (x_9 is *VBR*)) THEN μ_5^3 is *VBR*

Two of the most well-known modifiers are the erosion linguistic modifier "very" ($\Omega = 2$) and dilation linguistic modifier "more-or-less" (Ω = 0.5). The functions of these two modifiers are presented using $\mu^{very}(x) = (\mu(x))^2$ and $\mu^{more-or-less}(x) = (\mu(x))^{0.5}$, respectively. The initial population for the gene pool is composed of four groups with the same number $CS_{a1} + CS_{a2}$ part and $CS_{b1} + CS_{b2}$ part. The first group contains the original $CS_{a1} + CS_{a2}$ part and original $CS_{b1} + CS_{b2}$ part with the binary weight (w_{ij} = 1, i = 1,…,9, and j = 1,…, 5) and the unit modifier (Ω =1). The second group includes the original $CS_{a1} + CS_{a2}$ part and randomized $CS_{b1} + CS_{b2}$ part. The third group has randomized $CS_{a1} + CS_{a2}$ part and original $CS_{b1} + CS_{b2}$ part. The fourth group consists of randomized $CS_{a1} + CS_{a2}$ part and randomized $CS_{b1} + CS_{b2}$ part. Table 2 illustrates an example of the orthogonal arrays. Generally, level 0 and level 1 of a factor represent the selected genes from parents 0 and parents 1, respectively. After proper tabulation of experimental results, the summarized data are analyzed to determine the relative effects of various factors.

α_{VDK}	β_{VDK}	γ_{VDK}	δ_{VDK}	…	α_{VBR}	β_{VBR}	γ_{VBR}	δ_{VBR}	λ_1	λ_2	λ_3

(a) CS_{a1} (b) CS_{a2}

w_{11}	…	w_{91}	…	w_{15}	…	w_{95}	Ω_{VDK}		Ω_{VBR}

(c) CS_{b1} (d) CS_{b2}

Fig. 5. (a) Encoding of the parameters of fuzzy numbers, (b) Encoding of the parameters of fuzzy number *F_mean* and fuzzy numbers for luminance difference, (c) Encoding of fuzzy rules, and (d) Encoding of the linguistic modifiers of the linguistic terms.

Table 2. Orthogonal Array [26]

Experiment Number	Factor							Evaluation Value
	1	2	3	4	5	6	7	
1	0	0	0	0	0	0	0	y_1
2	0	0	0	1	1	1	1	y_2
3	0	1	1	0	0	1	1	y_3
4	0	1	1	1	1	0	0	y_4
5	1	0	1	0	1	0	1	y_5
6	1	0	1	1	0	1	0	y_6
7	1	1	0	0	1	1	0	y_7
8	1	1	0	1	0	0	1	y_8

The crossover structure is briefly described as follows. The parameters of CS_{a1} are divided into five groups, *VDK*, *DK*, *MD*, *BR*, and *VBR*. If the input factors are the parameters of the fuzzy numbers, then the output of the main factor is CS_{a1}. The parameters of CS_{b1} are also divided into five groups, *VDK*, *DK*, *MD*, *BR*, and *VBR*. If the input factors are the parameters of the rule sets, then the output of the main factor is CS_{b1}. Otherwise, if the input factors are the parameters of the hedge sets, then the output of the main factor for this layer is CS_{b2}.

The genetic learning process initiates the population by the encoding schema and restrictions, and then records the initiation population as the current population. The chromosome in the current population is evaluated by the fitness function. If the evaluation module does not satisfy the fitness function, then the intelligent learning process will execute the selection process. In addition, the elitism is also used in the learning process. In the beginning of the selection, the best two chromosomes in the current population are selected to new population without crossover and mutation. After the elitism, we produce the other population by mean of reproduction, crossover and mutation operators. The individuals having the best fitness have more chances to be reproduced. The one-point crossover method is adopted and the crossover point is randomly placed. The mutation process is applied to each offspring after the crossover. We use the Mean-Absolute-Error (MAE) to measure the fitness of each individual between the processed image and the original noise-free image. The MAE is calculated by (2), where $N \times M$ denotes the number of the processed pixels, $y(n,m)$ denotes the pixel of the processed image, and $s(n,m)$ denotes the pixel of the original noise-free image. The learning process is stopped when an assigned number of generations have been evolved or when a satisfactory value of fitness has been obtained. The Mean-Square-Error (MSE) is another criteria to evaluate the performance of the proposed approach, which is calculated by (3), where $N \times M$, $y(n,m)$, and $s(n,m)$ denote the same meanings as (2).

$$MAE = \frac{\sum_{n=1}^{N}\sum_{m=1}^{M}\left|y(n,m) - s(n,m)\right|}{N \times M} \tag{2}$$

$$MSE = \frac{\sum_{n=1}^{N}\sum_{m=1}^{M}\left(y(n,m) - s(n,m)\right)^2}{N \times M} \tag{3}$$

4 Experimental Results

The ontological intelligent agent (*OIA*) for impulse noise removal described in this paper was implemented with the Visual Basic (Version 6.0) programming language. The experimental environment was constructed to test the performance of the proposed approach. Fig. 6 shows the user interface of the *OIA*, which is highlighted the functions with opening the desired image, making noise rate, selecting the filter method, and showing the values based on the criteria of peak signal to noise ratio (PSNR), MSE, MAE, and the time complexity after implementation. Fig. 7 shows the screenshot of the *OIA* when the agent is learning and tuning the best membership functions of the different fuzzy variables for the gray image "Barbara." Fig. 8 shows the similar screenshot of the *OIA* with Fig. 7 but it is for the color image "Lena." The experiments involved in comparing with our previous works [26][27][28] and other famous works [18][25][29] based on the criteria of MAE, MSE, and the time complexity. The experimental images are corrupted with 50%, 60%, 70%, and 80% additive identical independent distribution (i.i.d.) impulse noise, and the impulses take on positive and negative values with an equal. To decide the parameter sets of the experiments, both gray and color 256×256 images "Lena" are utilized as the sample gray and color images, respectively. A small population of 20 individuals is chosen and the parameters of the genetic learning are set as follows: crossover probability equals 0.9, mutation rate equals 0.3, and 50 generations.

In our experiments, four images, including two gray images and two color images, are adapted to implement the performance of the proposed approach. Fig. 9(a)-(d) show the original test images for the gray image "Barbara," the gray image "Bridge," the color image "Butterfly," and the color image "Lena," respectively. Fig. 10(a), 14(a), 18(a), and 22(a) show the corresponding noisy image of Fig. 9(a)-(d) with 50% noise rate. The first experiment is to compare the results of denoising the corrupted gray image "Barbara" by using different types of the filters. Fig. 10(b)-(f) show the implemented results by using the EIDA filter, the Russo filter, the SMF filter, the GFIF filter, and the *OIA*, respectively, when the noisy image is corrupted with 50% noise rate. Fig. 10 indicates that the performance of the *OIA* is better than other filters. The second experiment is to evaluate the performance of different types of the filters using the MAE criteria. Fig. 11 shows the results of MAE bar chart of each filter for the noisy gray image "Barbara" corrupted with 50%, 60%, 70%, and 80% noise rate. The third experiment is to observe the *OIA*'s performance from the MSE criteria. Fig. 12 shows the results of MSE bar chart of each filter for the noisy gray image "Barbara" corrupted with 50%, 60%, 70%, and 80% noise rate. The results depicted in Fig. 11 and Fig. 12 demonstrate that the MAE and MSE values of the *OIA* are the smallest than other filters under the identical corrupted noise rate condition. The fourth experiment is to carry out the efficiency for different types of the filters through the time complexity. Fig. 13 shows the time complexity for the gray image "Barbara." Compared with other filters on the executing time, the results depicted in Fig. 13 demonstrate that the cost of implementing an *OIA* is not the most efficient but it is the most effective when the impulsive noise corruption noise rate is above 50%. Therefore, it is necessary to make a balance between efficiency and effectiveness of an image processing. The difference between the *OIA* and the OGFF is in the method of acquiring the parameters of the genetic learning process. The *OIA* automatically generates the optimal parameters for the fuzzy inference mechanism and uses the orthogonal array to help the learning process of the OGFF, but the OGFF manually does this. In addition, the difference between the *OIA* and the GFIF is briefly summarized as follows: (1) The *OIA* utilized the image noise ontology to represent the image noise knowledge for the agent, (2) the noise detection technique is able to choose the best fitted filter parameters, and (3) the evolution computation of the GFIF is able to be modified by using the orthogonal array to acquire the best filter parameters. Therefore, the performance of the *OIA* is better than the GFIF filter.

Finally, we also do the same experiments for the gray image "Bridge," the color image "Butterfly," and the color image "Lena," which are shown as Fig. 9(b)-(d), respectively. The results of denoising corrupted image using different filters for Fig. 9(b)-(d) are shown in Fig. 14, Fig. 18, and Fig. 22, respectively. The results of MAE bar chart of each filter for Fig. 9(b)-(d) are shown in Fig. 15, Fig. 19, and Fig. 23, respectively. The results of MSE bar chart of each filter for Fig. 9(b)-(d) are shown in Fig. 16, Fig. 20 and Fig. 24, respectively. The results of time complexity bar chart of each filter for Fig. 9(b)-(d) are shown in Fig. 17, Fig. 21, and Fig. 25, respectively.

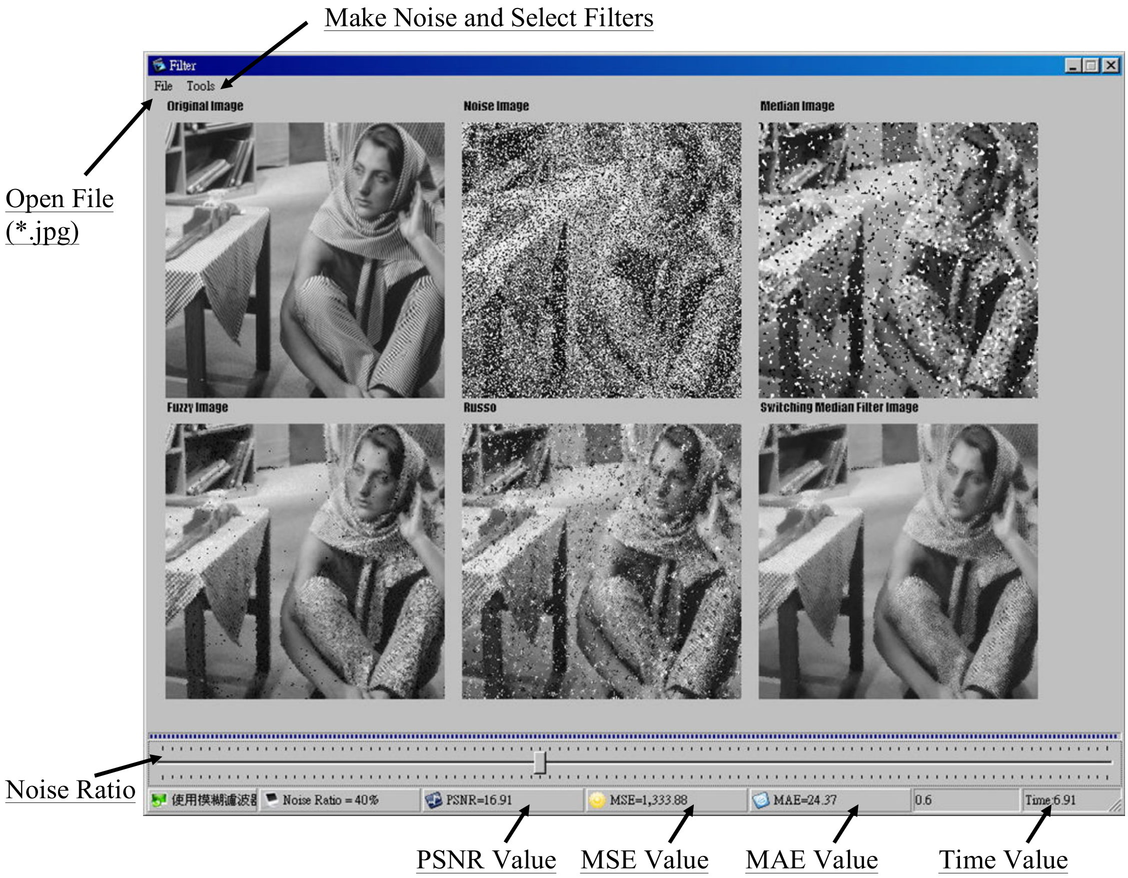


Fig. 6. User Interface of the *OIA*

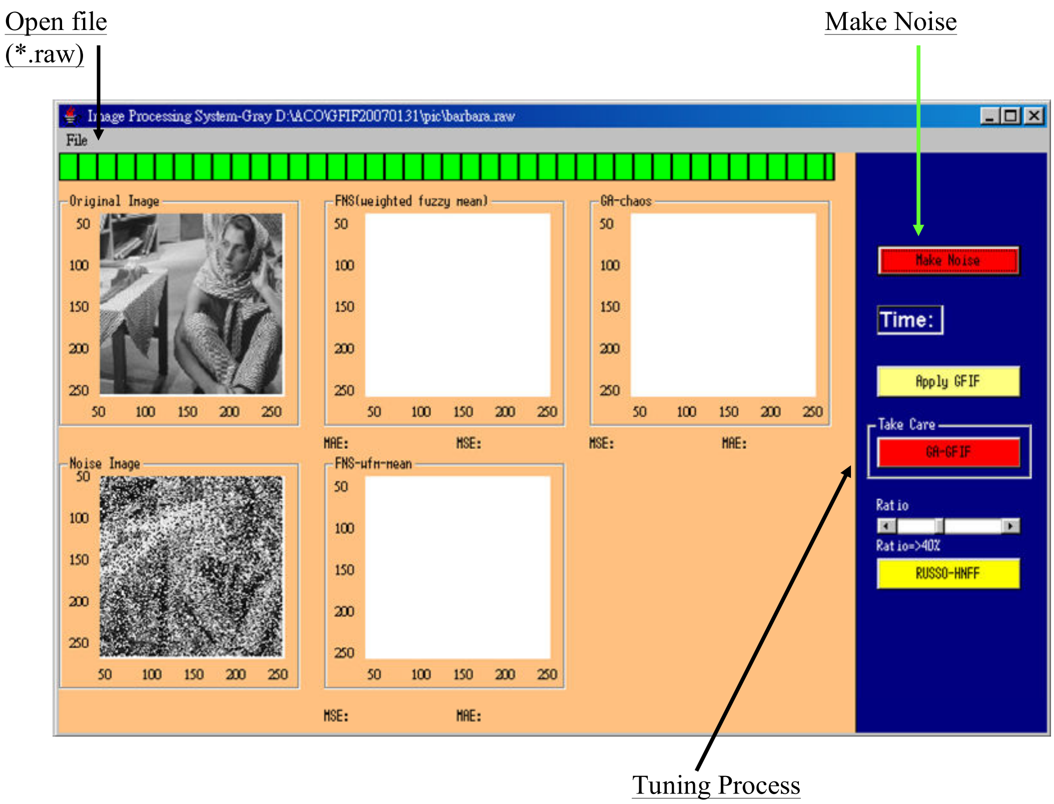


Fig. 7. Learn and Tune the best membership for the gray image “Barbara”

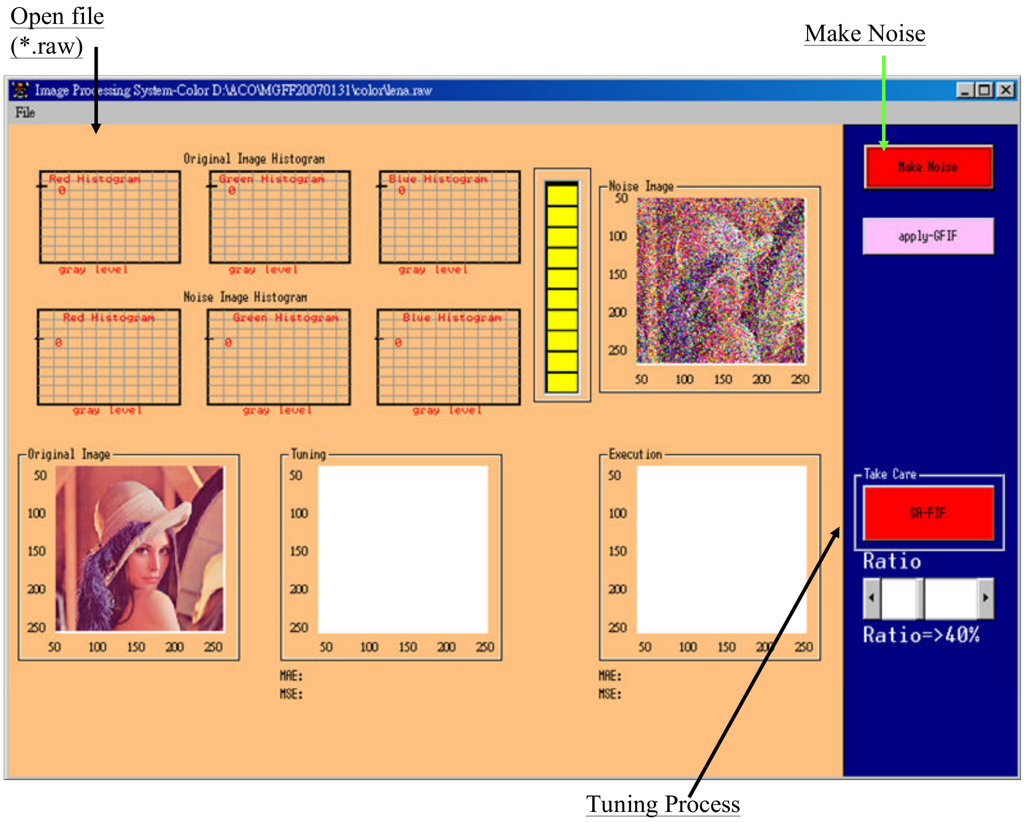
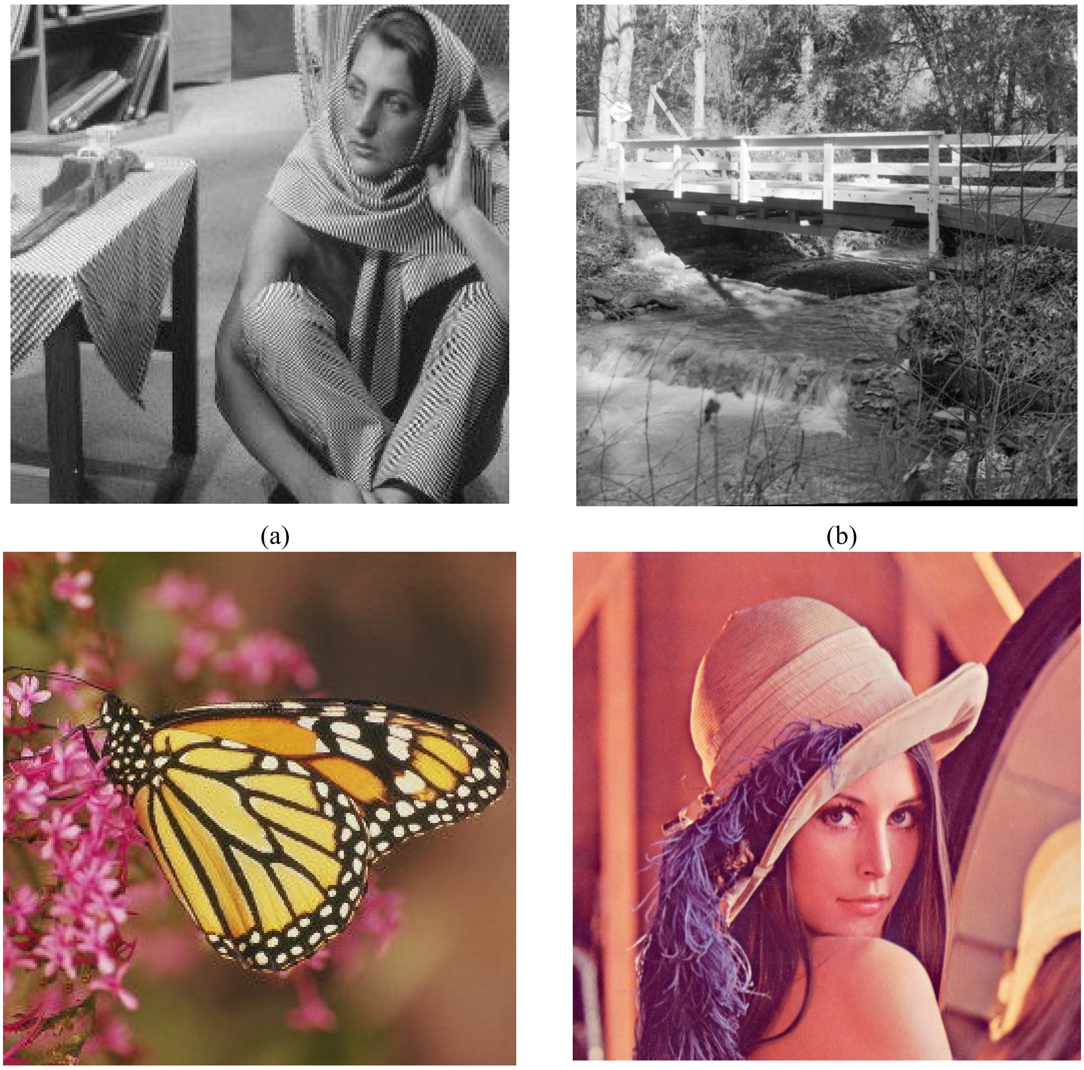


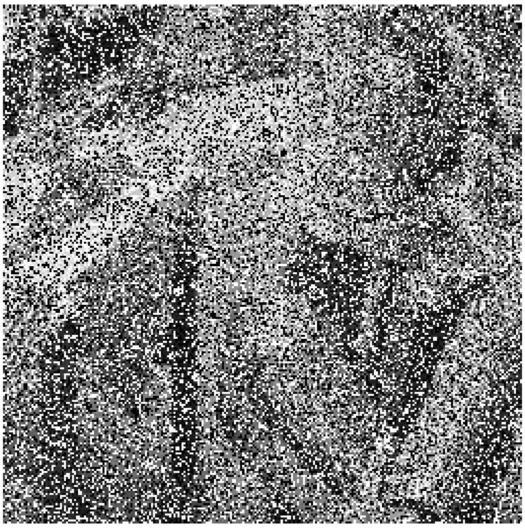
Fig. 8. Learn and Tune the best membership for the color image "Lena"



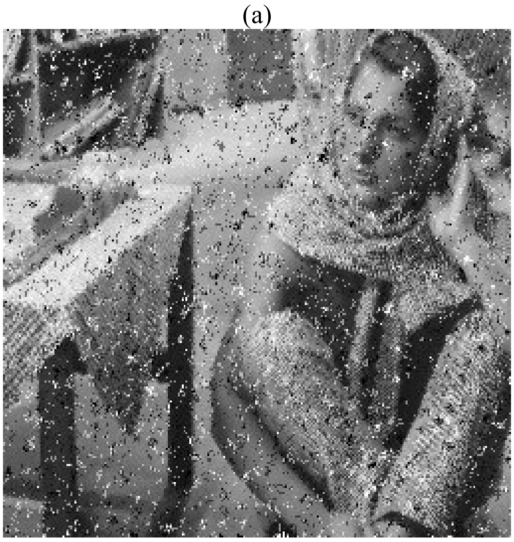
(a) (b)

(c) (d)

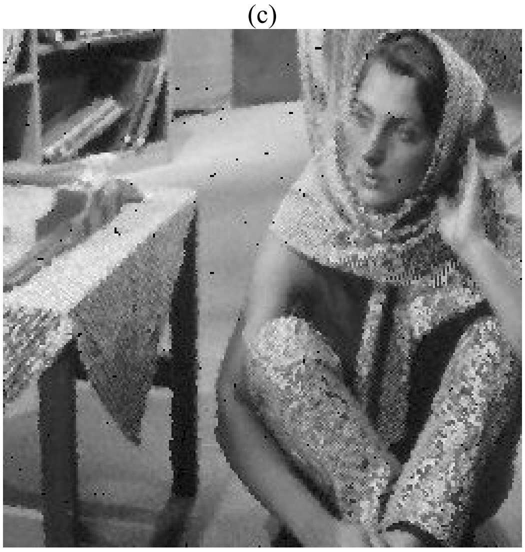
Fig. 9. Original test images for the (a) gray image “Barbara,” (b) gray image “Bridge,” (c) color image “Butterfly,” and (d) color image “Lena”


(a)


(b)


(c)


(d)


(e)

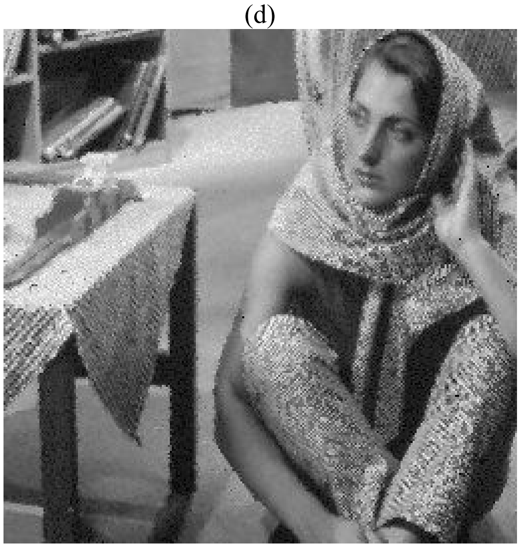

(f)

Fig. 10. (a) Noisy gray image "Barbara" corrupted with 50% noise rate, and the results of denoising corrupted image using the (b) EIDA filter, (c) Russo filter, (d) SMF filter, (e) GFIF filter, and (f) *OIA*

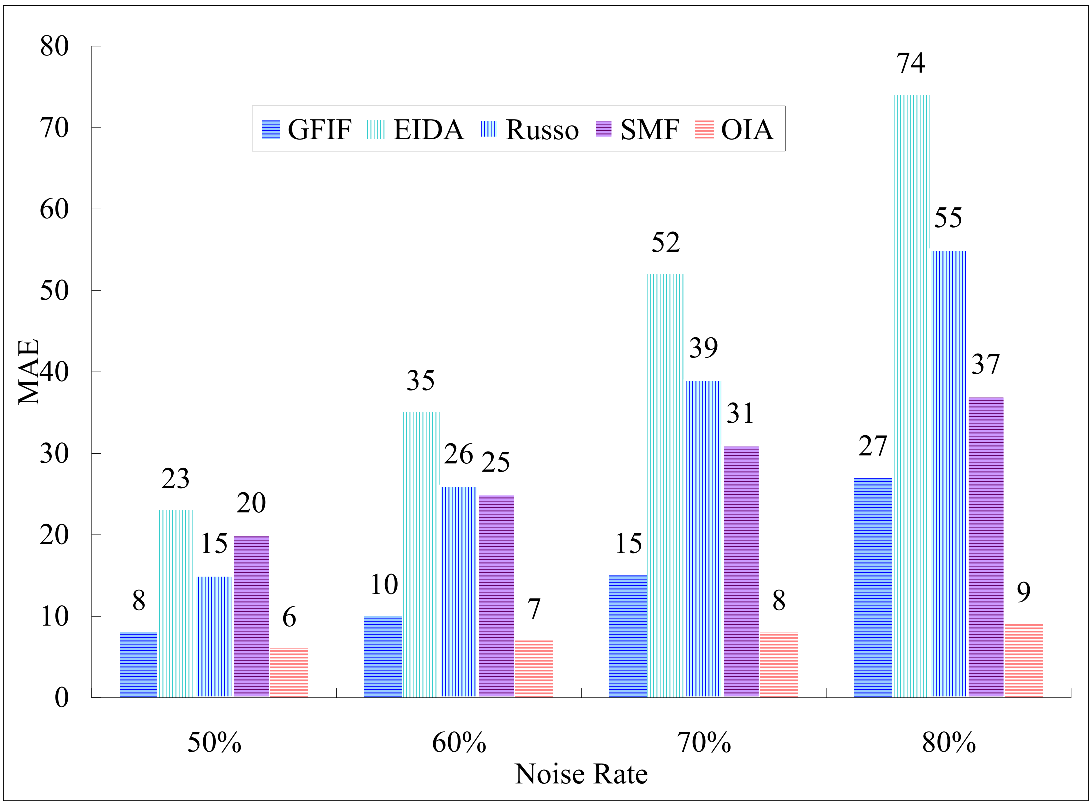


Fig. 11. Results of MAE bar chart of each filter for the noisy gray image "Barbara" corrupted with 50%, 60%, 70%, and 80% noise rate

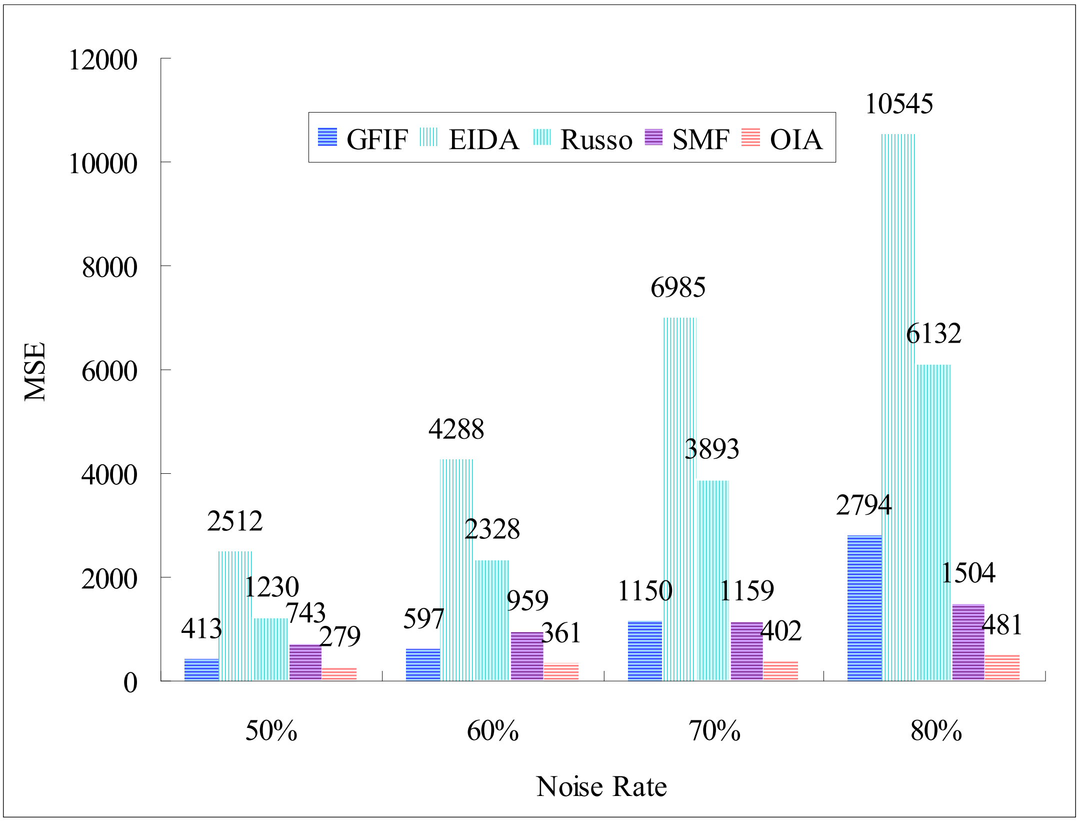


Fig. 12. Results of MSE bar chart of each filter for the noisy gray image "Barbara" corrupted with 50%, 60%, 70%, and 80% noise rate

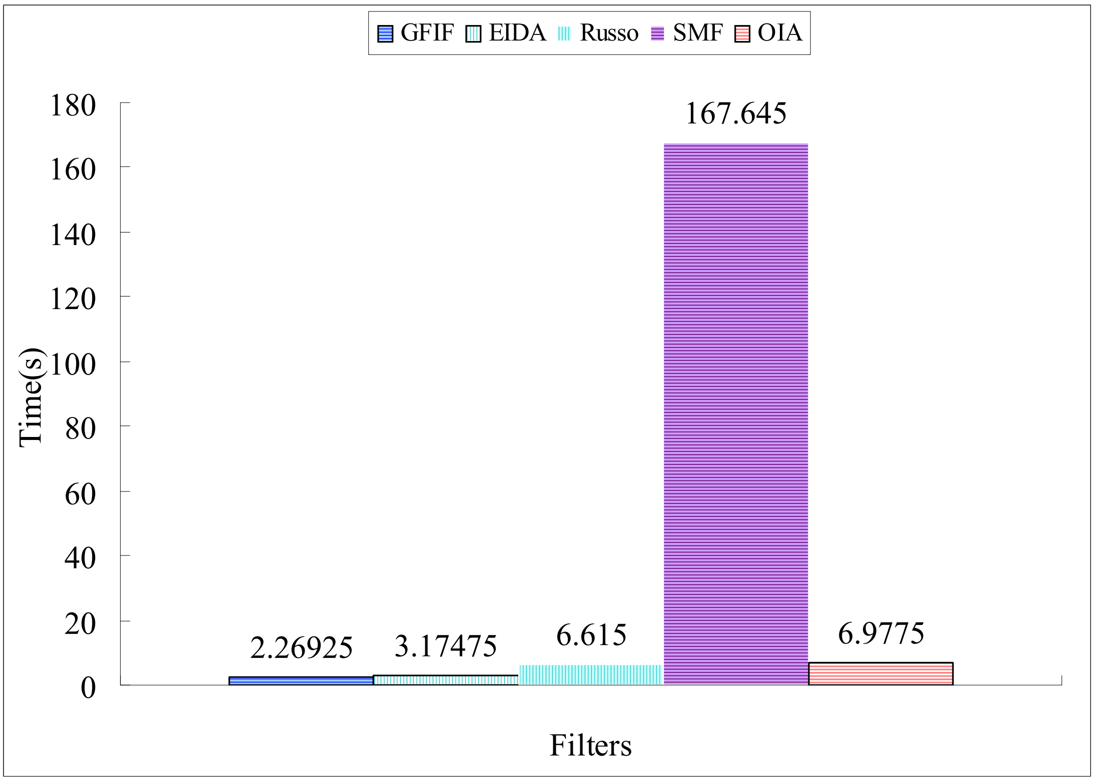


Fig. 13. Results of time complexity bar chart of each filter for the noisy gray image "Barbara"

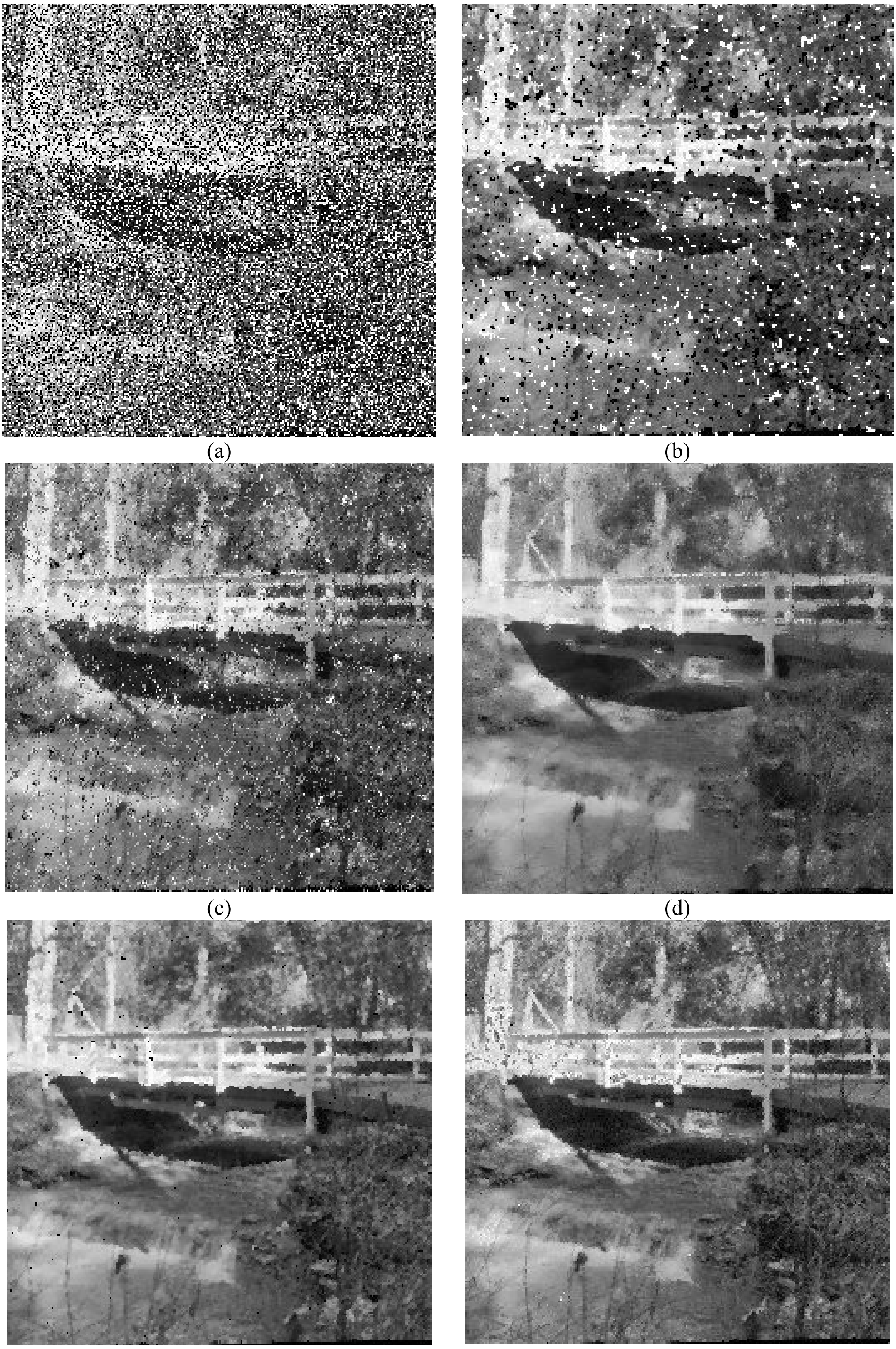


Fig. 14. (a) Noisy gray image “Bridge” corrupted with 50% noise rate, and the results of denoising corrupted image using the (b) EIDA filter, (c) Russo filter, (d) SMF filter, (e) GFIF filter, and (f) *OIA*

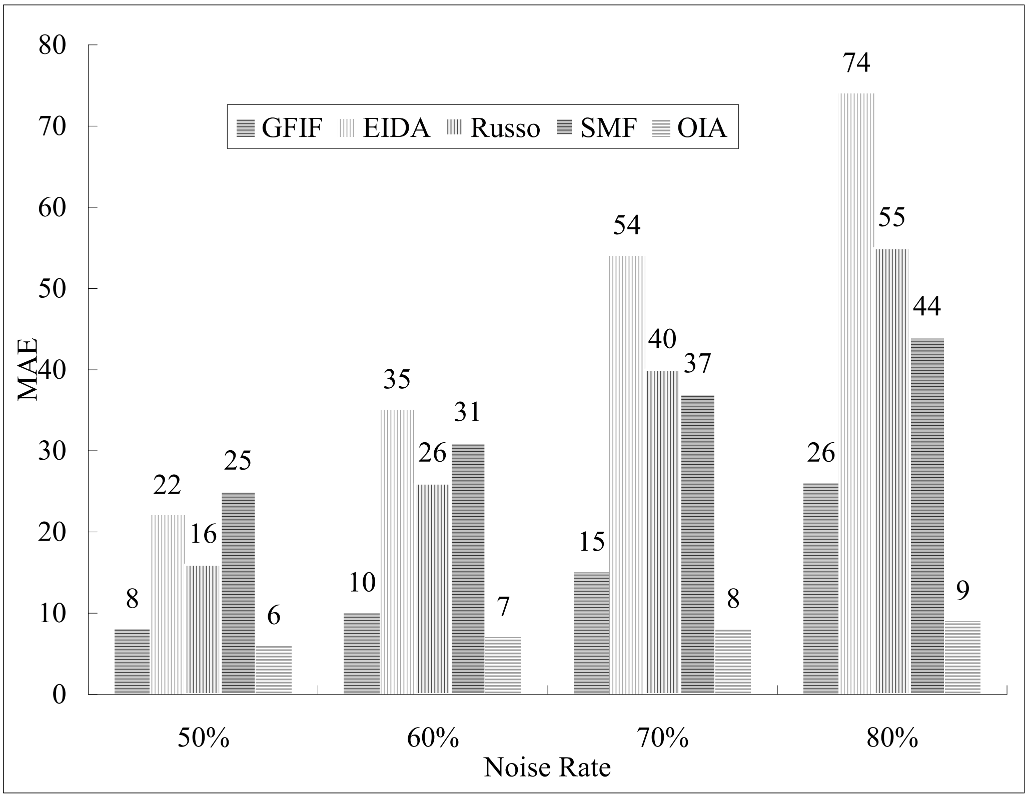


Fig. 15. Results of MAE bar chart of each filter for the noisy gray image "Bridge" corrupted with 50%, 60%, 70%, and 80% noise rate

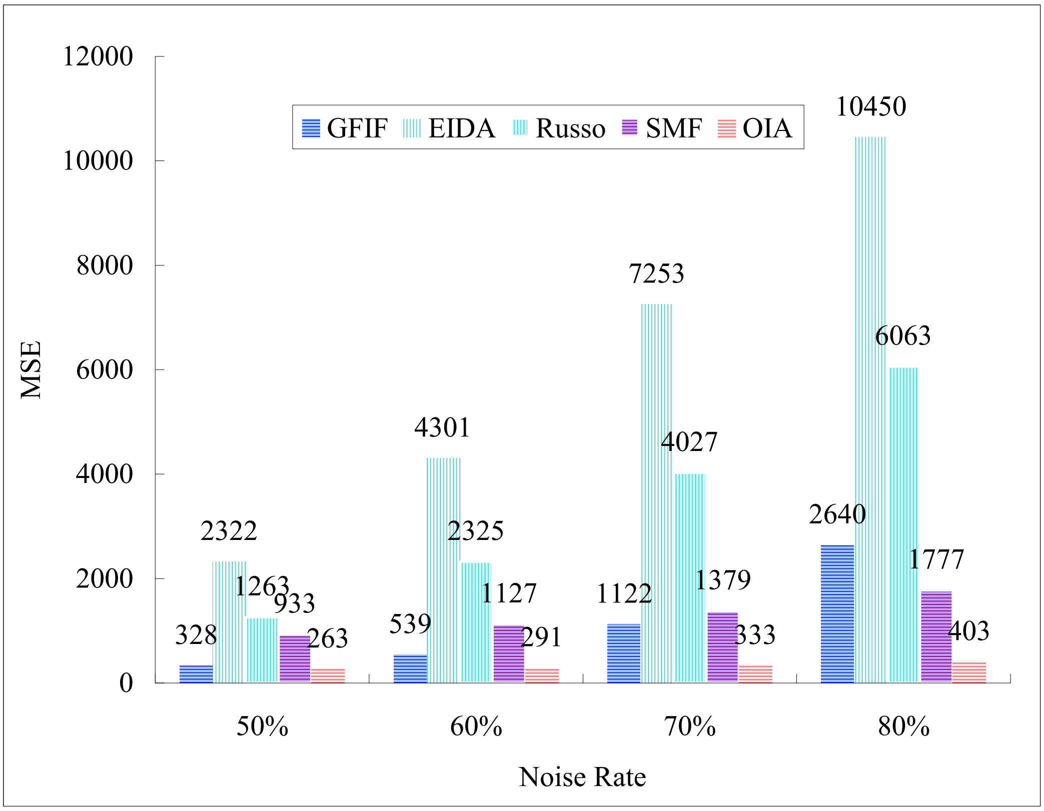


Fig. 16. Results of MSE bar chart of each filter for the noisy gray image "Bridge" corrupted with 50%, 60%, 70%, and 80% noise rate

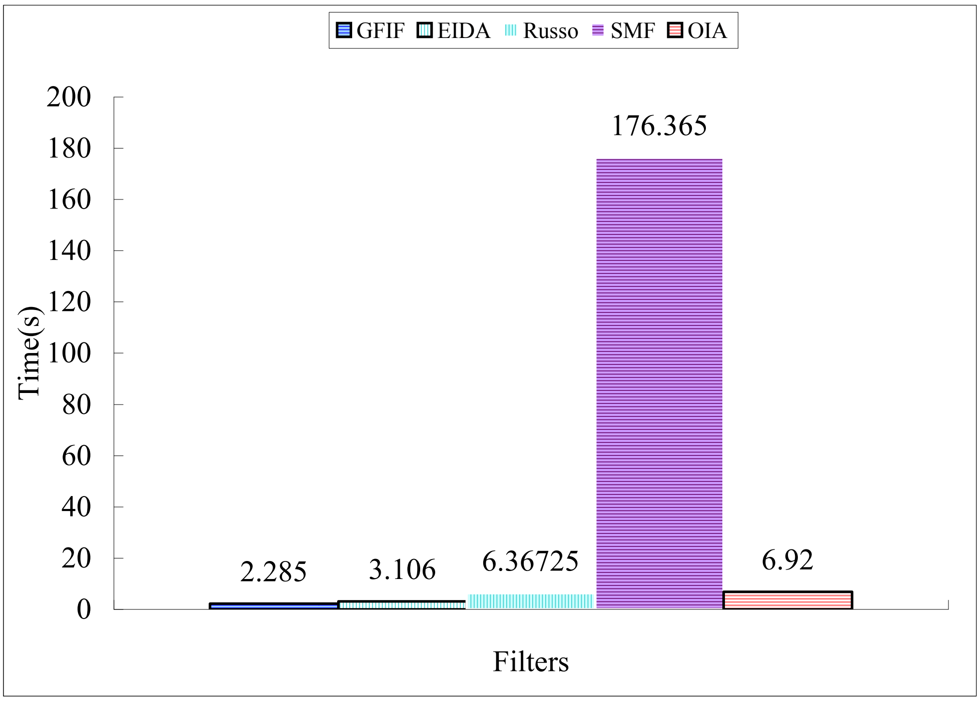


Fig. 17. Results of time complexity bar chart of each filter for the noisy gray image “Bridge”

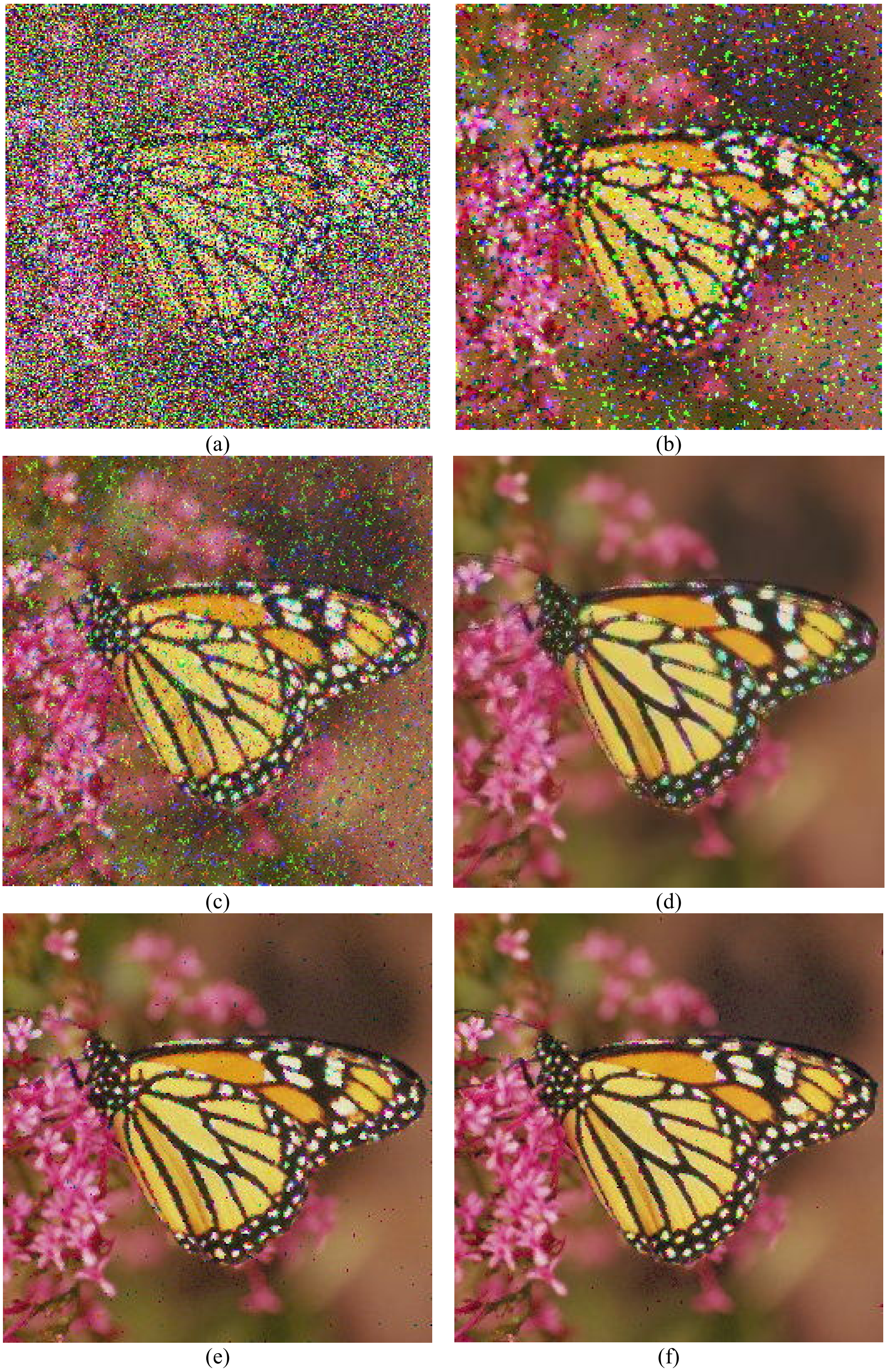


Fig. 18. (a) Noisy color image "Butterfly" corrupted with 50% noise rate, and the results of denoising corrupted image using the (b) EIDA filter, (c) Russo filter, (d) SMF filter, (e) MGFF filter, and (f) *OIA*

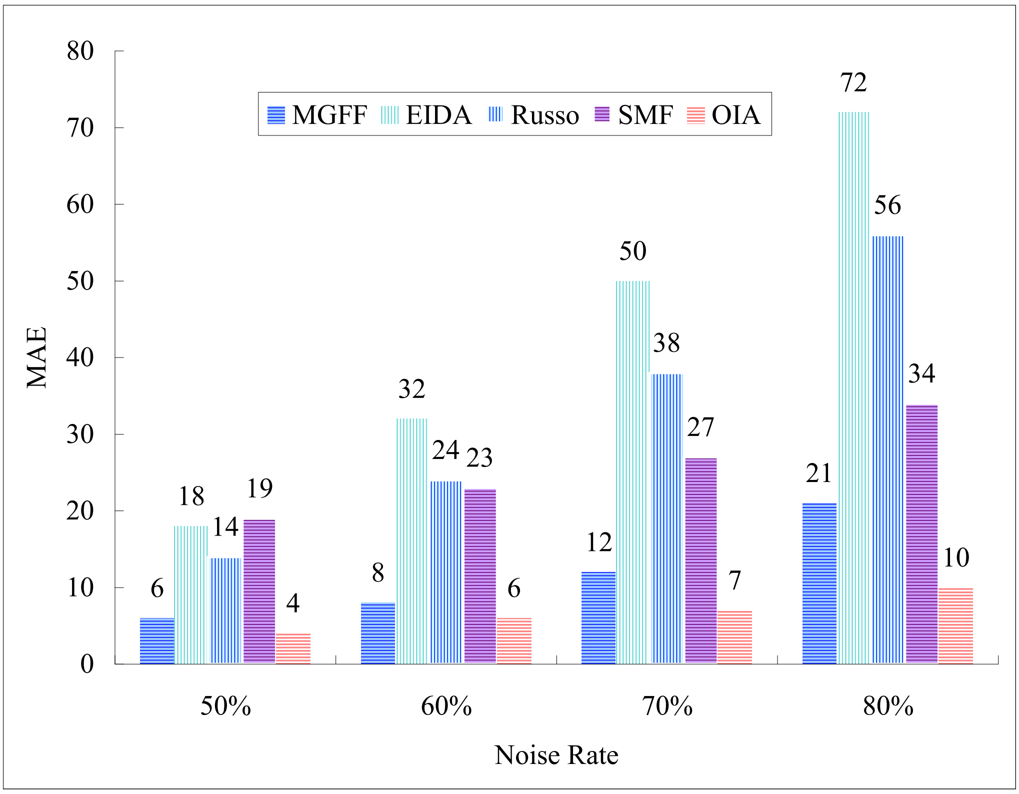


Fig. 19. Results of MAE bar chart of each filter for the noisy color image "Butterfly" corrupted with 50%, 60%, 70%, and 80% noise rate

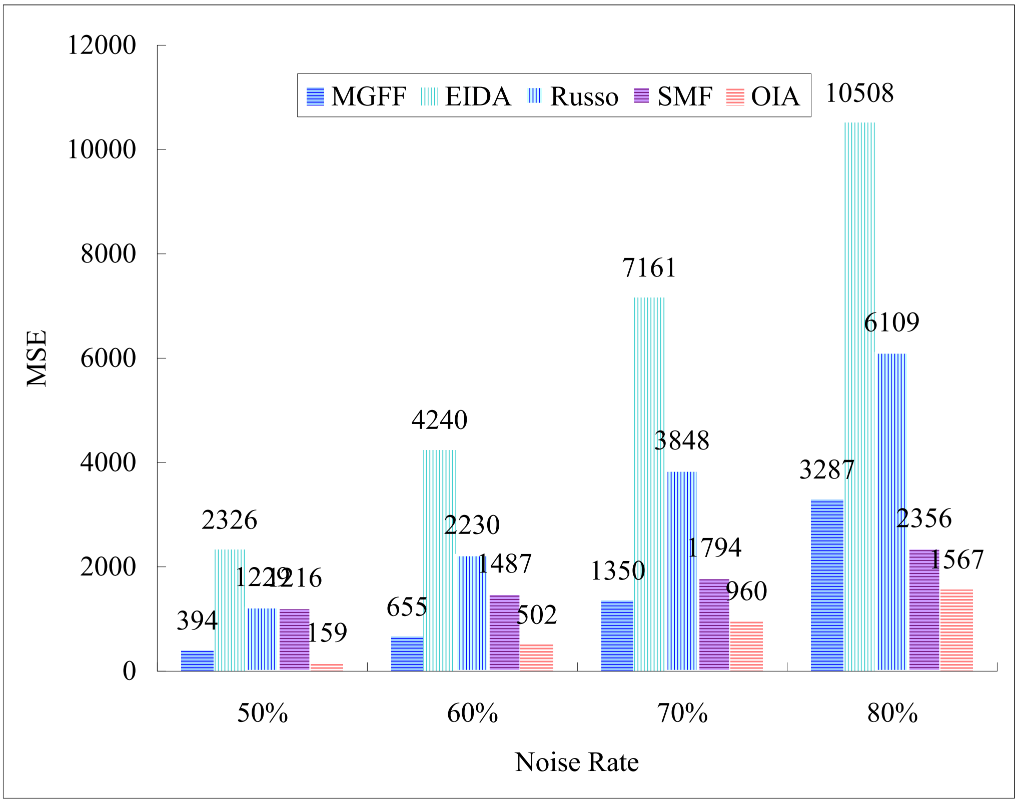


Fig. 20. Results of MSE bar chart of each filter for the noisy color image "Butterfly" corrupted with 50%, 60%, 70%, and 80% noise rate

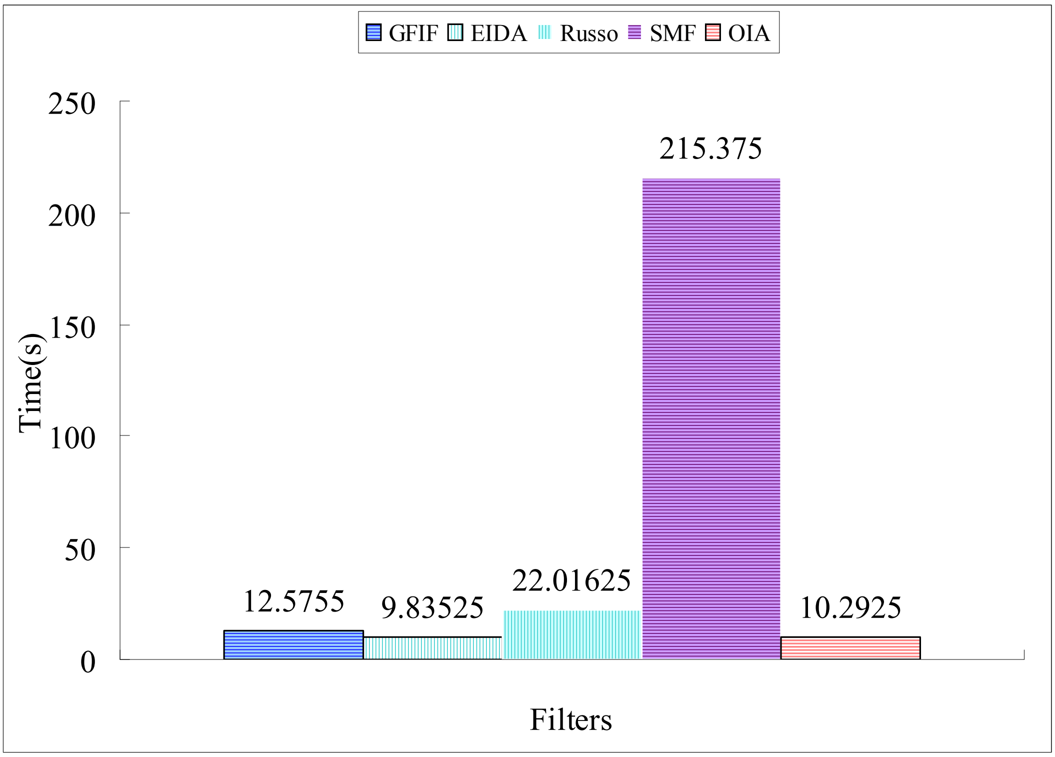
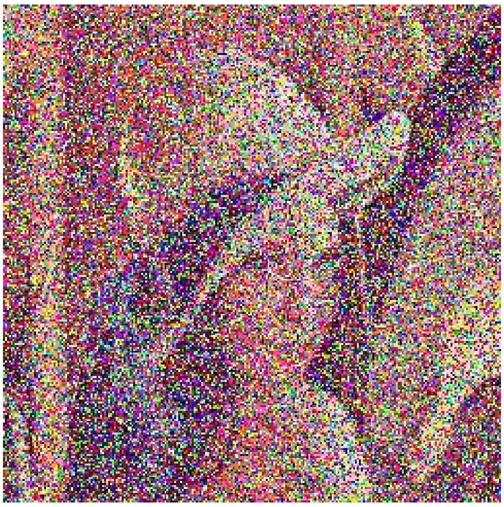
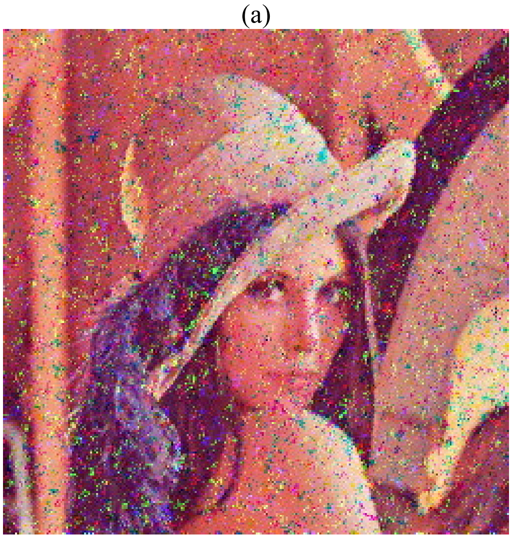


Fig. 21. Results of time complexity bar chart of each filter for the noisy color image "Butterfly"

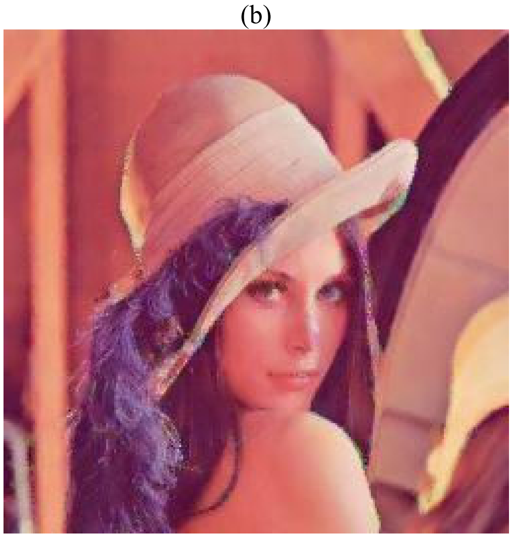


(a)

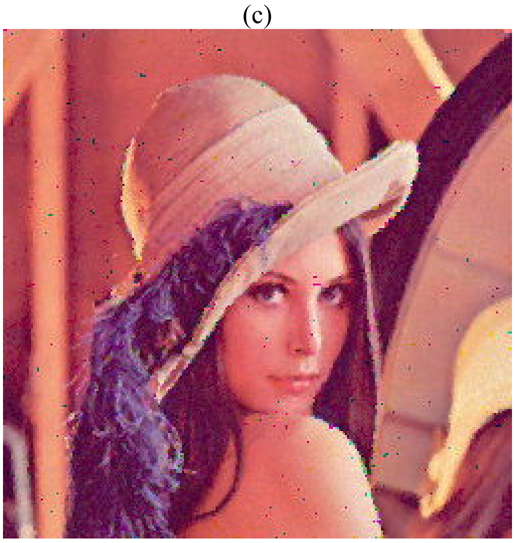
(b)



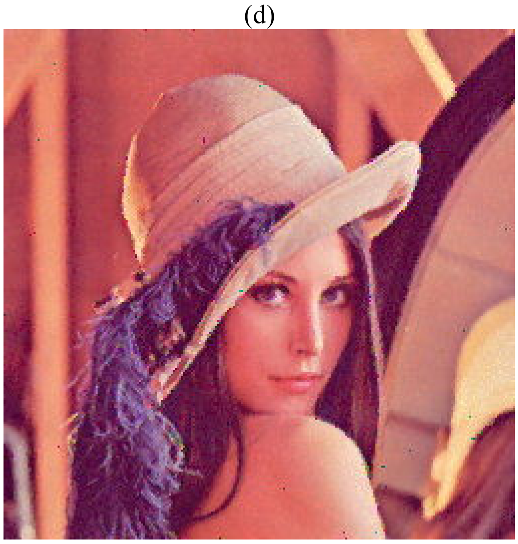
(c)



(d)



(e)



(f)

Fig. 22. (a) Noisy color image "Lena" corrupted with 50% noise rate, and the results of denoising corrupted image using the (b) EIDA filter, (c) Russo filter, (d) SMF filter, (e) MGFF filter, and (f) *OIA*

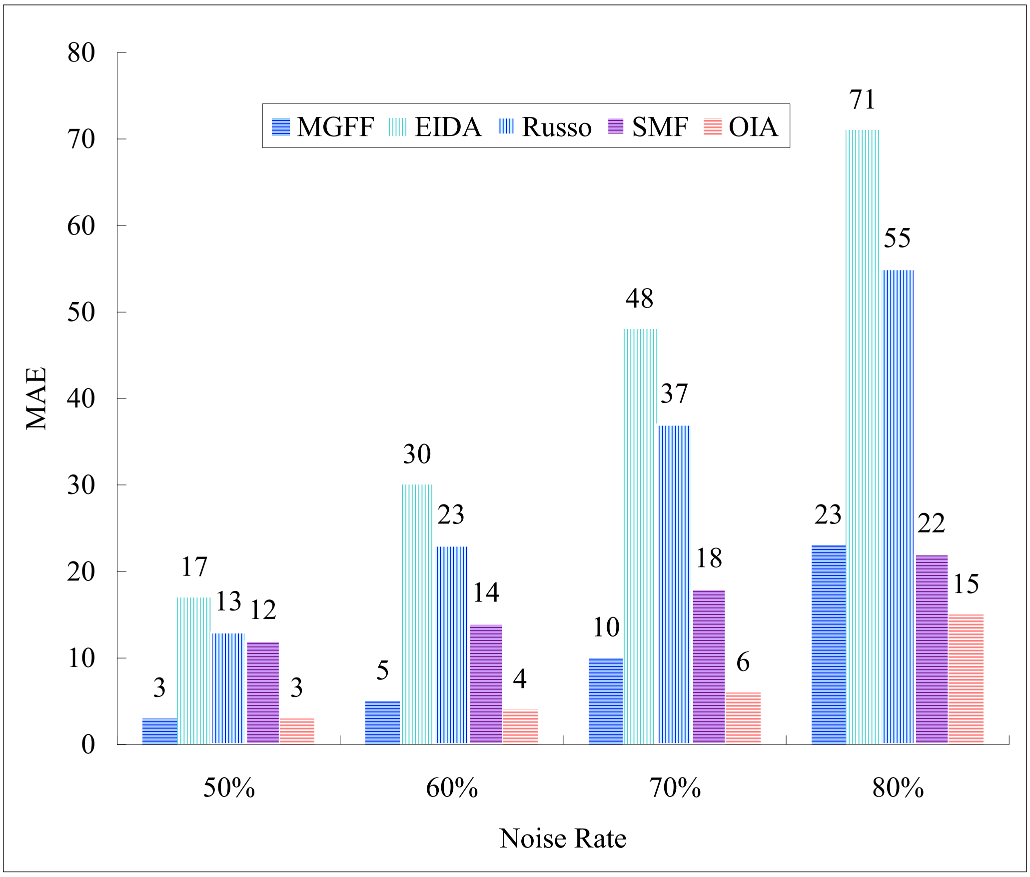


Fig. 23. Results of MAE bar chart of each filter for the noisy color image "Lena" corrupted with 50%, 60%, 70%, and 80% noise rate

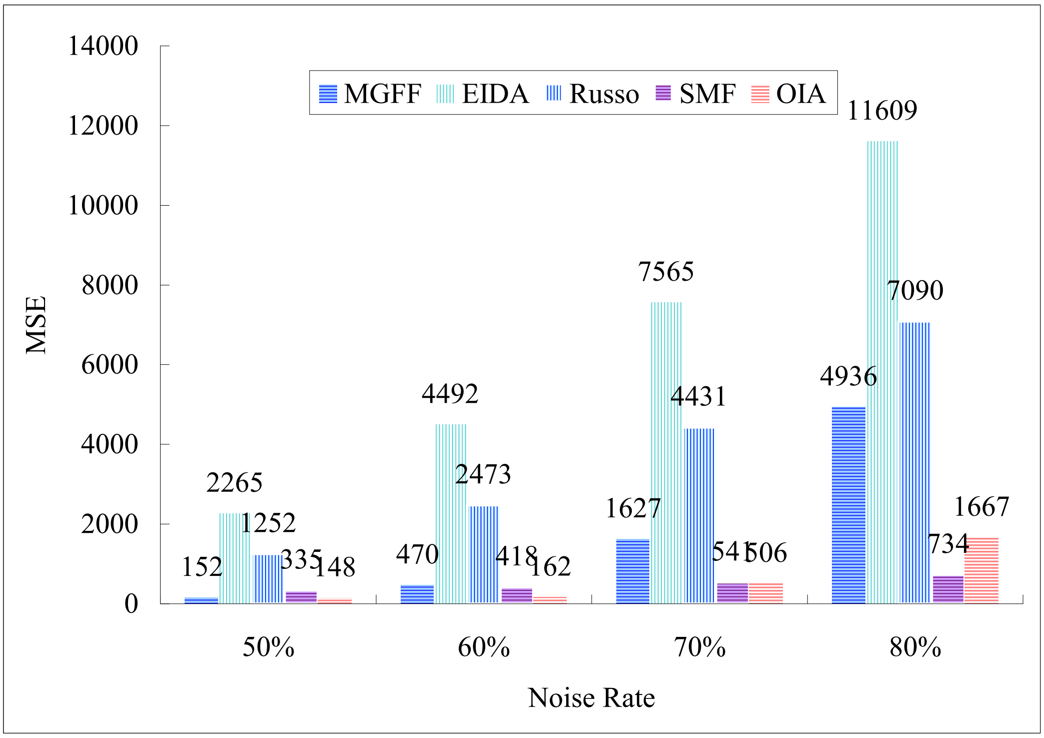


Fig. 24. Results of MSE bar chart of each filter for the noisy color image "Lena" corrupted with 50%, 60%, 70%, and 80% noise rate

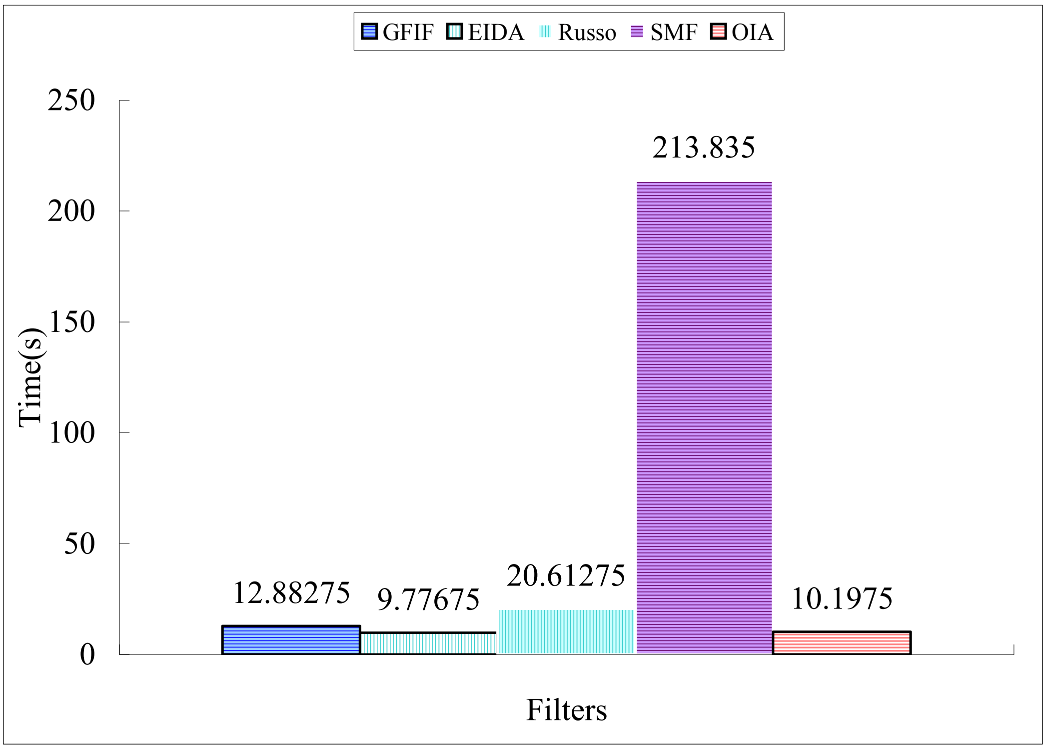


Fig. 25. Results of time complexity bar chart of each filter for the noisy color image "Lena"

5 Conclusions

In this paper, an ontological intelligent agent (*OIA*) is proposed to remove impulse noise from highly corrupted images. The fuzzy signal space for the working environment of the intelligent agent is defined and presented in the paper. The *OIA* consists of an image noisy ontology, a fuzzy inference mechanism, and an intelligent learning process. The fuzzy inference mechanism executes the fuzzy inference not only to remove impulse noise but also to keep the image structure. The image noise ontology stored the fuzzy numbers of the *OIA* for impulse noise removal is also presented in this paper. An intelligent learning process is used to adjust the fuzzy numbers of the image noise ontology. The proposed approach can achieve more effective results than other approaches such as the EIDA filter, the Russo filter, the SMF filter, the GFIF filter, and the MGFF filter. However, there are still some problems need further study in the future. For example, improving the learning process to reduce the time complexity of the *OIA* is one of our future works.

Acknowledgement

The authors may express their acknowledgement to the anonymous reviewers for their comments that improved the quality of this paper, and to the financial support of the project numbers: NSC 95-2221-E-024 -009-MY2 and NSC 95-2221-E-024-010.

References

[1] C. S. Lee, Z. W. Jian, and L. K. Huang, "A Fuzzy Ontology and its Application to News Summarization," *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 35, No. 5, pp. 859-880, Oct. 2005.

[2] C. Brewster, K. O'Hara, S. Fuller, Y. Wilks, E. Franconi, M. A. Musen. J. Ellman, and S. B. Shum, "Knowledge Representation with Ontologies: the Present and Future," *IEEE Intelligent Systems*, Vol. 19, No. 1, pp. 72-81, Jan./Feb. 2004.

[3] C. S. Lee, C. C Jiang, and T. C. Hsieh, "A Genetic Fuzzy Agent Using Ontology Model for Meeting Scheduling System," *Information Sciences*, Vol. 176, No. 9, pp. 1131-1155, 2005.

[4] H. Alani, S. Kim, D. E. Millard, M. J. Weal, W. Hall, P. H. Lewis, and N. R. Shadbolt, "Automatic Ontology-Based Knowledge Extraction from Web Documents," *IEEE Intelligent Systems*, Vol. 18, No. 1, pp. 14-21, Jan./Feb. 2003.

[5] M. H, Burstein, "Dynamic Invocation of Semantic Web Services that Use Unfamiliar Ontologies," *IEEE Intelligent Systems*, Vol. 19, No. 4, pp. 67-73, Jul./Aug. 2004.

[6] A. Hameed, D. Sleeman, and A. Preece, "Detecting Mismatches Among Experts' Ontologies Acquired Through Knowledge Elicitation," *Knowledge-Based Systems*, Vol. 15, No. 5, pp. 265-273, Jun. 2002.

[7] R. Navigli, P. Velardi, and A. Gangemi, "Ontology Learning and its Application to Automated Terminology Translation," *IEEE Intlligent Systems*, Vol. 18, No. 1, pp. 22-31, Jan./Feb. 2003.

[8] J. Ferber, *Multi-Agent Systems*, Addison-Wesley, New York, 1999.

[9] H. Chen and J. Yen, "Toward Intelligent Meeting Agents," *IEEE Computer*, Vol. 29, No. 8, pp. 62-70, 1996.

[10] D. Delen and B. Pratt, "An Integrated and Intelligent DSS for Manufacturing System," *Expert Systems with Applications*, Vol. 30, No. 2, pp. 325-336, 2006.

[11] M. S. Hamdi, "MASACAD: A Multiagent-based Approach to Information Customization," *IEEE Intelligent System*, Vol. 21, No. 1, pp. 60-67, 2006.

[12] F. Y. Wang, "Agent-based Control for Networked Traffic Management Systems," *IEEE Intelligent System*, Vol. 20, No. 5, pp. 92-96, 2005.

[13] H. Yan, Y. Jiang, J. Zheng, C. Peng, and Q. Li," A Multiplayer Perceptron-based Medical Decision Support System for Heart Disease Diagnosis," *Expert Systems with Applications*, Vol. 30, No. 3, pp. 272-281, 2006.

[14] J. Grossklags and C. Schmidt, "Software Agents and Market in Efficiency: A Human Trader Experiment," *IEEE Transactions on Systems, Man, and Cybernetics-PartC: Applications and Reviews*, Vol. 36, No. 1, pp. 1-13, 2006.

[15] K. Arakawa, "Median Filter Based on Fuzzy Rules and its Application to Image Restoration," *Fuzzy Sets and Systems*, Vol. 77, No. 1, pp. 3-13, 1996.

[16] H. L. Eng and K. K. Ma, "Noise Adaptive Soft-switching Median Filter," *IEEE Trans. on Image Processing*, Vol. 10, No. 2, pp. 242-251, Feb. 2001.

[17] E. Abreu and S.K. Mitra, "A Signal-Dependent Rank Ordered Mean (SD-ROM) Filter", *Proceedings of IEEE International Conference on Acoustics*, *Speech and Signal Processing*, ICASSP-95, Detroit, pp. 2371-2374, 1995.

[18] F. Russo, "Noise Removal From Image Data Using Recursive Neurofuzzy Filters," *IEEE Trans. on Instrumentation and Measurement*, Vol. 49, No. 2, pp.307-314, 2000.

[19] J. H. Wang, W. J. Liu and L. D. Lin, "Histogram-Based Fuzzy Filter for Image Restoration," *IEEE Trans. On Systems, Man and Cybernetics, Part B*, Vol.32, No. 2, pp.230-238, 2002.

[20] R. Lukac, "Adaptive Vector Median Filtering," *Pattern Recognition Letters*, Vol. 24, No. 12, pp.1889-1899, 2003.

[21] G. Pok, J. C. Liu and A. S. Nair, "Selective Removal of Impulse Noise Based on Homogeneity Level Information," *IEEE Trans. on Image Processing*, Vol. 12, No. 1, pp.85-92, 2003.

[22] J. Y. Chang and J. L. Chen, "Classified-Augmented Median Filters for Image Restoration," *IEEE Trans. on Instrumentation and Measurement*, Vol. 53, No. 2, pp.351-356, 2004.

[23] P. Liu, "Representation of Digital Image by Fuzzy Neural Network," *Fuzzy Sets and Systems*, Vol. 130, No. 1, pp.109-123, 2002.

[24] R. S. Lin and Y. C. Hsueh, "Multichannel Filtering by Gradient Information," *Signal Processing* Vol. 80, No. 2, pp. 279-293, 2000.

[25] P. E. Ng and K. K. Ma, "A Switching Median Filter with Boundary Discriminative Noise Detection for Extremely Corrupted Images," *IEEE Trans. on Image Processing*, Vol. 15, No. 6, pp. 1506-1516, 2006.

[26] C. S. Lee and C-Y Hsu, "Ontology-based Genetic Fuzzy Filter for Image Processing," *North American Fuzzy Information Processing Society* (*NAFIPS*), Montreal, Quebec, Canada, 2006.

[27] C. S. Lee, S. M. Guo, and C. Y. Hsu, "Genetic-Based Fuzzy Image Filter and Its Application to Image Processing," *IEEE Trans. on Systems, Man and Cybernetics Part B*, Vol. 35, No. 4, pp. 694-711, Aug. 2005.

[28] S. M. Guo, C. S. Lee, and C. Y. Hsu, "An Intelligent Image Agent Based on Soft-Computing Techniques for Color Image Processing," *Expert Systems with Applications*, Vol. 28, No. 3, pp. 483-494, 2005.

[29] W. Luo, "A New Efficient Impulse Detection Algorithm for the Removal of Impulse Noise," *IEICE Trans. On Fundamentals of Electronics, Communications and Computer Sciences*, Vol. 88, No. 10, pp. 2579-2586, 2005.

[30] J. Cantais, D. Dominguez, V. Gigante, L. Laera L, and V. Tamma, "An Example of Food Ontology for Diabetes Control," *Proceedings of the International Semantic Web Conference 2005 workshop on Ontology Patterns for the Semantic Web*, Galway, Ireland, 2005.

[31] C. S. Lee and Y. H. Kuo, "Adaptive Fuzzy Edge Detector for Image Enhancement," *IEEE International Conference on Fuzzy Systems (WCCI'98, IEEE-FUZZ'98*)," Alaska USA, 1998.

[32] M. Horridge, H. Knublauch, A. Rector, R. Stevens, and C. Wroe, *A Practical Guide to Building OWL Ontologies Using the Protege-OWL Plugin and CO-ODE Tools,* Edition 1.0.

[33] S. Y. Ho, C. C. Liu, and S. Liu, "Design of an Optimal Nearest Neighbor Classifier Using an Intelligent Genetic Algorithm," *Pattern Recog. Lett*., Vol. 23, No. 13, pp. 1495-1503, Nov. 2002.