Computational and Applied Mathematics (2022) 41:192
https://doi.org/10.1007/s40314-022-01893-4

Interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft matrices and their application to performance-based value assignment to noise-removal filters

Tuğçe Aydın[1] · **Serdar Enginoğlu**[1]

Received: 15 September 2021 / Revised: 24 January 2022 / Accepted: 2 May 2022 /
Published online: 31 May 2022

Abstract
Recently, the concept of interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft sets (d-sets) has successfully modelled decision-making problems, where the parameters and alternatives have interval-valued intuitionistic fuzzy values. In the present study, to be able to transfer a large number of data in such problems to a computer environment and to process them therein, we define the concept of interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft matrices (d-matrices). Moreover, we introduce operations, such as union, intersection, and AND/OR/ANDNOT/ORNOT-products, on this concept and study some of their basic properties. We then configure the state-of-the-art soft decision-making (SDM) method constructed by d-sets to render it operable in d-matrices space. Furthermore, we apply it to a performance-based value assignment (PVA) to the seven noise removal filters to compare their ranking orders. Thereafter, we conduct a comparative analysis of the configured method with five state-of-the-art SDM methods. Finally, we discuss d-matrices for future research.

Keywords Fuzzy sets · Soft sets · d-sets · d-matrices · Soft decision making

Mathematics Subject Classification 03F55 · 03E72

Communicated by Anibal Tavares de Azevedo.

✉ Tuğçe Aydın
aydinttugce@gmail.com

Serdar Enginoğlu
serdarenginoglu@gmail.com

[1] Department of Mathematics, Faculty of Arts and Sciences, Çanakkale Onsekiz Mart University, Çanakkale, Turkey

1 Introduction

Many mathematical tools have been proposed to overcome problems containing uncertainties in the real world. Fuzzy sets (Zadeh 1965) and soft sets (Molodtsov 1999) are among the known mathematical tools. In addition to these, intuitionistic fuzzy sets (Atanassov 1986) and interval-valued intuitionistic fuzzy sets (*ivif*-sets) (Atanassov 2020; Atanassov and Gargov 1989), being the generalisations of the concept of fuzzy sets, have been propounded. Afterwards, various hybrid versions of these concepts, such as fuzzy soft sets (Maji et al. 2001), fuzzy parameterized soft sets (Çağman et al. 2011a), fuzzy parameterized fuzzy soft sets (Çağman et al. 2010), intuitionistic fuzzy parameterized soft sets (Deli and Çağman 2015), interval-valued intuitionistic fuzzy parameterized soft sets (Deli and Karataş 2016), intuitionistic fuzzy parameterized intuitionistic fuzzy soft sets (Karaaslan 2016), and fuzzy parameterized intuitionistic fuzzy soft sets (Sulukan et al. 2019) have been introduced. So far, the researchers have conducted numerous theoretical and applied studies on these concepts in various fields, such as algebra (Çıtak and Çağman 2015; Senapati and Shum 2019; Sezgin 2016; Sezgin et al. 2019; Ullah et al. 2018), topology (Atmaca 2017; Aydın and Enginoğlu 2021b; Enginoğlu et al. 2015; Riaz and Hashmi 2017; Şenel 2016; Thomas and John 2016), analysis (Molodtsov 2004; Riaz et al. 2018; Şenel 2018), and decision making (Çağman and Enginoğlu 2010b; Çağman et al. 2011b; Garg and Arora 2020; Kumar and Garg 2018; Liu and Jiang 2020; Maji et al. 2002; Memiş and Enginoğlu 2019; Mishra and Rani 2018; Petchimuthu et al. 2020; Xue et al. 2021).

However, when a problem containing uncertainties incorporates a large number of data, the aforesaid set concepts display some time- and complexity-related disadvantages. To cope with these difficulties, Çağman and Enginoğlu (2010a) have defined the concept of soft matrices allowing data in such problems to be transferred to and processed in a computer environment and suggested the soft max-min method. Then, Çağman and Enginoğlu (2012) have presented the concept of fuzzy soft matrices and constructed a soft decision-making (SDM) method. Enginoğlu and Çağman (2020) have propounded the concept of fuzzy parameterized fuzzy soft matrices (*fpfs*-matrices). Moreover, they have proposed an SDM method called Prevalence Effect Method (PEM) and applied it to a performance-based value assignment (PVA) problem, so that they can order image-denoising filters in terms of noise-removal performance. Afterwards, Enginoğlu et al. (2019a) have offered a novel SDM method constructed with *fpfs*-matrices and PEM, and applied it to the problem of monolithic columns classification.

Lately, the concept of *fpfs*-matrices has stood out among others due to its modelling success in decision-making problems, where the alternatives and parameters have fuzzy membership degrees. Therefore, many SDM methods, constructed by its substructures, have been configured in (Aydın and Enginoğlu 2019, 2020; Enginoğlu and Memiş 2018b; Enginoğlu and Öngel 2020; Enginoğlu et al. 2021a, b) to operate them in *fpfs*-matrices space, faithfully to the original. Some of the configured methods have been applied to PVA problems, and successful results have been obtained (Aydın and Enginoğlu 2019, 2020; Enginoğlu and Öngel 2020). Besides, Enginoğlu and Memiş (2018a, c) and Enginoğlu et al. (2018a, b) have focussed on mathematical simplifications and improvements of some of the configured methods. Memiş et al. (2019) have developed a classification algorithm based on normalised Hamming pseudo-similarity of *fpfs*-matrices. Further, Memiş et al. (2021b) have proposed a classification algorithm based on the Euclidean pseudo-similarity of *fpfs*-matrices.

Afterwards, the concept of intuitionistic fuzzy parameterized intuitionistic fuzzy soft matrices (*ifpifs*-matrices) (Enginoğlu and Arslan 2020) has been introduced to model uncertainties in which the alternatives and parameters have intuitionistic fuzzy values. Furthermore, using this concept, a new SDM method has been proposed and applied to a hypothetical problem concerning the determination of eligible candidates in a recruitment scenario and a real-life problem of image processing. Arslan et al. (2021) have then generalised 24 SDM methods operating in *fpfs*-matrices space via this concept. Besides, they have suggested five test scenarios to compare the performances of the generalised SDM methods and applied the SDM methods successful in these test scenarios to a PVA problem. In addition, Memiş et al. (2021a) have offered a classifier based on the similarity of *ifpifs*-matrices and applied this classifier to machine learning.

Recently, to be able to model some problems mathematically in which parameters and alternatives contain serious uncertainties, Aydın and Enginoğlu (2021a) have defined the concept of interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft sets (d-sets), which can be regarded as the general form of the concepts of interval-valued intuitionistic fuzzy parameterized soft sets (Deli and Karataş 2016) and interval-valued intuitionistic fuzzy soft sets (Jiang et al. 2010; Min 2008). They then have proposed an SDM method using d-sets and applied it to two decision-making problems concerning the eligibility of candidates for two vacant positions in an online job advertisement and PVA to the known filters used in image denoising. The applications have shown that d-sets can be successfully applied to problems containing further uncertainties. Thus, in decision-making problems where the parameters and alternatives contain multiple measurement results, the ambiguity as to which value to assign to a parameter or an alternative has been clarified. The primary motivation of the present study is to develop effective SDM methods by improving d-sets' skills in modelling such problems. The second one is to propound a novel mathematical tool to enable data in similar problems, containing both a large number of data and multiple intuitionistic fuzzy measurement results, to be transferred to a computer environment. Thus, it will be possible to use the concept of d-sets effectively.

In the current study, we focus on the concept of *ivif*-sets, more meaningful and convenient than the others, to minimise data loss when modelling the problem of which value to assign to a parameter or an alternative with multiple fuzzy or intuitionistic fuzzy measurement results. For example, in Section 5, the results of Based on Pixel Density Filter (BPDF) (Erkan and Gökrem 2018) for 20 traditional test images at noise density 10% are as follows:

$$\begin{array}{lllll}\mu_1=0.9848, & \mu_2=0.9911, & \mu_3=0.9743, & \mu_4=0.9795, & \mu_5=0.9735,\\ \mu_6=0.9747, & \mu_7=0.9795, & \mu_8=0.9885, & \mu_9=0.9761, & \mu_{10}=0.9801,\\ \mu_{11}=0.9753, & \mu_{12}=0.9938, & \mu_{13}=0.9705, & \mu_{14}=0.9707, & \mu_{15}=0.9726,\\ \mu_{16}=0.9808, & \mu_{17}=0.9791, & \mu_{18}=0.9909, & \mu_{19}=0.9657, & \mu_{20}=0.9830\end{array}$$

We can regard these results as the multiple membership degrees of BPDF herein. Thus, we can obtain the multiple non-membership degrees of BPDF corresponding to these multiple membership degrees using $\nu_i = 1 - \mu_i$, for $i \in \{1, 2, \dots, 20\}$. Namely,

$$\begin{array}{lllll}\nu_1=0.0152, & \nu_2=0.0089, & \nu_3=0.0257, & \nu_4=0.0205, & \nu_5=0.0265,\\ \nu_6=0.0253, & \nu_7=0.0205, & \nu_8=0.0115, & \nu_9=0.0239, & \nu_{10}=0.0199,\\ \nu_{11}=0.0247, & \nu_{12}=0.0062, & \nu_{13}=0.0295, & \nu_{14}=0.0293, & \nu_{15}=0.0274,\\ \nu_{16}=0.0192, & \nu_{17}=0.0209, & \nu_{18}=0.0091, & \nu_{19}=0.0343, & \nu_{20}=0.0170\end{array}$$

We can calculate the membership and non-membership degrees of BPDF in three different ways by availing of the aforesaid values as follows:

1. Using $\mu(\text{BPDF}) = \frac{1}{20}\sum_{i=1}^{20}\mu_i$, we obtain the degree of BPDF's membership to a fuzzy set as $\mu(\text{BPDF}) = 0.9792$.

2. By utilising $\mu(\text{BPDF})= \min\limits_{i\in I_{20}} \mu_i$ and $\nu(\text{BPDF})= 1 - \max\limits_{i\in I_{20}} \nu_i$, we obtain the degrees of BPDF's membership and non-membership to an intuitionistic fuzzy set as $\mu(\text{BPDF})= 0.9657$ and $\nu(\text{BPDF})= 0.0062$, respectively.
3. By employing $\mu(\text{BPDF})= \left[\frac{\min\limits_{i\in I_{20}} \mu_i}{\max\limits_{i\in I_{20}} \mu_i+\max\limits_{i\in I_{20}} \nu_i}, \frac{\max\limits_{i\in I_{20}} \mu_i}{\max\limits_{i\in I_{20}} \mu_i+\max\limits_{i\in I_{20}} \nu_i}\right]$ and $\nu(\text{BPDF})= \left[\frac{\min\limits_{i\in I_{20}} \nu_i}{\max\limits_{i\in I_{20}} \mu_i+\max\limits_{i\in I_{20}} \nu_i}, \frac{\max\limits_{i\in I_{20}} \nu_i}{\max\limits_{i\in I_{20}} \mu_i+\max\limits_{i\in I_{20}} \nu_i}\right]$, we obtain the degrees of BPDF's membership and non-membership to an *ivif*-set as $\mu(\text{BPDF})= [0.9392, 0.9666]$ and $\nu(\text{BPDF})= [0.0060, 0.0334]$, respectively.

The first case shows that BPDF's noise-removal performance at noise density 10% accounts for approximately 98%. The second signifies that BPDF exhibits a success rate of around 97% and a failure rate of 1% in noise removal. The last one indicates that the noise-removal success of BPDF ranges from 94% to 97% and its failure from 1 to 3%. These comments manifest that membership and non-membership degrees assigned to an alternative in *ivif*-sets offer more information than fuzzy sets and intuitionistic fuzzy sets do. Hence, we can summarise the significant advantages and contributions of the present study as follows:

– The concept of interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft matrices (d-matrices) has an important advantage to prevent errors arising from manual calculations in SDM methods constructed by d-sets. This concept makes it possible to obtain fast and reliable results.
– The concept of d-matrices allows to process a large number of data and multiple measurement results by transferring them to a computer environment.
– The concept of d-matrices utilises *ivif*-values containing more information compared to fuzzy or intuitionistic fuzzy values to determine membership and non-membership degrees of parameters and alternatives.
– The pre-processing step of the configured method presents an approach related to the conversion of multiple intuitionistic fuzzy measurement results to *ivif*-values.

On the other hand, the running time of the configured method can be slightly longer than those of the others. This relatively minor drawback results from computations while converting multiple intuitionistic fuzzy measurement results to *ivif*-values. For instance, for d-matrix $[b_{ij}]$ and *ifpifs*-matrix $[c_{ij}]$ in Sects. 5 and 6 , the data concerning the average running time of the methods (in second), using MATLAB R2021a and a laptop with 2.5 GHz i5-2450M CPU and 8 GB RAM, in 1000 runs are as follows:

The configured method: 0.0063, iMBR01: 0.0011, iMRB02(I_9): 0.0009, iCCE10: 0.0002, iCCE11: 0.0004, and iPEM: 0.0028

Section 2 of the present study provides some of the basic definitions to be employed in the paper's next sections. Section 3 defines the concept of d-matrices and investigates some of its basic properties. Section 4 configures a state-of-the-art SDM method constructed with d-sets to operate it in d-matrices space. Section 5 applies it to a real-life problem concerning PVA to the known image-denoising filters using the Structural Similarity (SSIM) results of these filters for the images provided in two different databases. Furthermore, the section comments on the ranking orders of the filters. Section 6 provides a comparative analysis of the ranking performances of the configured method and those of the five methods by applying five state-of-the-art SDM methods constructed with *ifpifs*-matrices to the same problem. Finally, d-matrices are discussed for further research. This study is a part of the first author's PhD dissertation (Aydın 2020).

2 Preliminaries

This section first presents several the known definitions and propositions. Throughout this paper, let $Int([0,1])$ be the set of all closed classical subintervals of [0, 1].

Definition 1 Let $\gamma_1, \gamma_2 \in Int([0,1])$. For $\gamma_1 := [\gamma_1^-, \gamma_1^+]$ and $\gamma_2 := [\gamma_2^-, \gamma_2^+]$,

i. if $\gamma_2^- \leq \gamma_1^-$ and $\gamma_1^+ \leq \gamma_2^+$, then γ_1 is called a classical subinterval of γ_2 and is denoted by $\gamma_1 \subseteq \gamma_2$.
ii. if $\gamma_1^- \leq \gamma_2^-$ and $\gamma_1^+ \leq \gamma_2^+$, then γ_1 is called a subinterval of γ_2 and is denoted by $\gamma_1 \tilde{\subseteq} \gamma_2$.
iii. if $\gamma_1^- = \gamma_2^-$ and $\gamma_1^+ = \gamma_2^+$, then γ_1 and γ_2 are called equal intervals and is denoted by $\gamma_1 = \gamma_2$.

Proposition 1 *Let* $\gamma_1, \gamma_2 \in Int([0,1])$. *Then,* $\gamma_1 \tilde{\leq} \gamma_2 \Leftrightarrow \gamma_1 \tilde{\subseteq} \gamma_2$. *Here, "*$\tilde{\leq}$*" is a partially ordered relation over* $Int([0,1])$.

In the present paper, the smallest upper bound and greatest lower bound of the elements of the set $Int([0,1])$ are obtained from the partially ordered relation "$\tilde{\leq}$".

Definition 2 Let $\gamma, \gamma_1, \gamma_2 \in Int(\mathbb{R})$ and $c \in \mathbb{R}^+$ such that $\gamma := [\gamma^-, \gamma^+]$, $\gamma_1 := [\gamma_1^-, \gamma_1^+]$, and $\gamma_2 := [\gamma_2^-, \gamma_2^+]$. Then,

i. $\gamma_1 + \gamma_2 := [\gamma_1^- + \gamma_2^-, \gamma_1^+ + \gamma_2^+]$
ii. $\gamma_1 - \gamma_2 := [\gamma_1^- - \gamma_2^+, \gamma_1^+ - \gamma_2^-]$
iii. $\gamma_1 \cdot \gamma_2 := [\min\{\gamma_1^-\gamma_2^-, \gamma_1^-\gamma_2^+, \gamma_1^+\gamma_2^-, \gamma_1^+\gamma_2^+\}, \max\{\gamma_1^-\gamma_2^-, \gamma_1^-\gamma_2^+, \gamma_1^+\gamma_2^-, \gamma_1^+\gamma_2^+\}]$
iv. $c \cdot \gamma := [c \cdot \gamma^-, c \cdot \gamma^+]$

Proposition 2 *Let* $\gamma_1, \gamma_2 \in Int([0,1])$ *such that* $\gamma_1 := [\gamma_1^-, \gamma_1^+]$ *and* $\gamma_2 := [\gamma_2^-, \gamma_2^+]$. *Then,*

i. $\sup\{\gamma_1, \gamma_2\} = [\max\{\gamma_1^-, \gamma_2^-\}, \max\{\gamma_1^+, \gamma_2^+\}]$
ii. $\inf\{\gamma_1, \gamma_2\} = [\min\{\gamma_1^-, \gamma_2^-\}, \min\{\gamma_1^+, \gamma_2^+\}]$

Second, this section presents some of the basic definitions to be used in the paper's next sections.

Definition 3 (Atanassov and Gargov 1989) Let E be a universal set and κ be a function from E to $Int([0,1]) \times Int([0,1])$. Then, the set $\{(x, \kappa(x)) : x \in E\}$, being the graphic of κ, is called an interval-valued intuitionistic fuzzy set (*ivif*-set) over E.

Here, for all $x \in E$, $\kappa(x) := (\alpha(x), \beta(x))$, $\alpha(x) := [\alpha^-(x), \alpha^+(x)]$, and $\beta(x) := [\beta^-(x), \beta^+(x)]$ such that $\alpha^+(x) + \beta^+(x) \leq 1$. Moreover, α and β are called membership function and non-membership function in an *ivif*-set, respectively.

From now on, the set of all the *ivif*-sets over E is denoted by $IVIF(E)$. In $IVIF(E)$, since the graph(κ) and κ generate each other uniquely, the notations are interchangeable. Therefore, as long as it causes no confusion, we denote an *ivif*-set graph(κ) by κ. Moreover, we use the notation ${}^{\alpha(x)}_{\beta(x)}x$ instead of $(x, \alpha(x), \beta(x))$, for brevity. Thus, we represent an *ivif*-set over E with $\kappa := \left\{ {}^{\alpha(x)}_{\beta(x)}x : x \in E \right\}$.

Note 1 Since $[k, k] := k$, we use ${}^{k}_{t}x$ instead of ${}^{[k,k]}_{[t,t]}x$, for all $k, t \in [0,1]$. Moreover, we do not display the elements ${}^{0}_{1}x$ in an *ivif*-set.

Definition 4 (Aydın and Enginoğlu 2021a) Let U be a universal set, E be a parameter set, $\kappa \in IVIF(E)$, and f be a function from κ to $IVIF(U)$. Then, the set $\left\{\left({}^{\alpha(x)}_{\beta(x)}x, f\left({}^{\alpha(x)}_{\beta(x)}x\right)\right) : x \in E\right\}$, being the graphic of f, is called an interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft set (d-set) parameterized via E over U (or briefly over U).

Note 2 We do not display the elements $\left({}^0_1x, 0_U\right)$ in a d-set. Here, 0_U is the empty *ivif*-set over U.

Hereinafter, the set of all the d-sets over U is denoted by $D_E(U)$. In $D_E(U)$, since the graph(f) and f generate each other uniquely, the notations are interchangeable. Therefore, as long as it causes no confusion, we denote a d-set graph(f) by f.

Example 1 Let $E = \{x_1, x_2, x_3, x_4\}$ be a parameter set and $U = \{u_1, u_2, u_3, u_4, u_5\}$ be a universal set. Then,

$$f = \left\{\left({}^{[0.1,0.4]}_{[0.4,0.5]}x_1, \left\{{}^{[0.4,0.6]}_{[0.2,0.3]}u_1, {}^{[0.7,0.8]}_{[0,0.1]}u_2, {}^{[0.1,0.4]}_{[0,0.2]}u_4\right\}\right), \left({}^0_1x_2, \left\{{}^{[0,0.5]}_{[0.1,0.2]}u_3, {}^{[0.3,0.5]}_{[0.2,0.3]}u_5\right\}\right),\right.$$
$$\left.\left({}^0_1x_3, 1_U\right), \left({}^{[0.2,0.5]}_{[0.1,0.2]}x_4, \left\{{}^{[0.3,0.4]}_{[0.5,0.6]}u_2, {}^{[0,0.2]}_{[0.5,0.6]}u_4, {}^{[0.3,0.7]}_{[0.1,0.2]}u_5\right\}\right)\right\}$$

is a d-set over U. Here, $1_U := \left\{{}^1_0u : u \in U\right\}$.

3 Interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft matrices

This section first defines the concept of d-matrices and introduces some of its basic properties. The primary purpose of the present section is to enable a large number of data containing multiple measurement results to be transferred to a computer environment with the help of this concept. The second one is to develop effective SDM methods by improving d-sets' skills in modelling such cases. To do so, this section focuses on making a theoretical contribution to the concept of soft matrices and defining product operations over d-matrices to use in SDM methods based on group decision making for the subsequent studies. From now on, let E be a parameter set and U be a universal set.

Definition 5 Let $f \in D_E(U)$. Then, $[a_{ij}]$ is called the d-matrix of f and is defined by

$$[a_{ij}] = \begin{bmatrix} a_{01} & a_{02} & a_{03} & \dots & a_{0n} & \dots \\ a_{11} & a_{12} & a_{13} & \dots & a_{1n} & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \end{bmatrix}$$

such that for $i \in \{0, 1, 2, \cdots\}$ and $j \in \{1, 2, \cdots\}$,

$$a_{ij} := \begin{cases} {}^{\alpha(x_j)}_{\beta(x_j)}, & i = 0 \\ f\left({}^{\alpha(x_j)}_{\beta(x_j)}x_j\right)(u_i), & i \neq 0 \end{cases}$$

Moreover, if $|U| = m - 1$ and $|E| = n$, then $[a_{ij}]$ is an $m \times n$ d-matrix. We represent the entry of a d-matrix $[a_{ij}]$ with $a_{ij} := {}^{\alpha_{ij}}_{\beta_{ij}}$. It must be noted that for all i and j, $\alpha_{ij} := [\alpha^-_{ij}, \alpha^+_{ij}]$ and $\beta_{ij} := [\beta^-_{ij}, \beta^+_{ij}]$ such that $\alpha^+_{ij} + \beta^+_{ij} \leq 1$. In this paper, to avoid any confusion, as needed, the membership and non-membership degrees of a_{ij}, i.e. α_{ij} and β_{ij}, will also be represented by α^a_{ij} and β^a_{ij}, respectively. Besides, the set of all the d-matrices parameterized via E over U is denoted by $D_E[U]$ and $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$.

The entries of a d-matrix $[a_{ij}]_{m\times n}$ consist of *ivif*-values. The entries of row with zero indexed of its contain membership and non-membership degrees of each parameter. For example, the entry a_{01} indicates the membership and non-membership degrees of the first parameter. Moreover, the entries of the other rows of its involve the membership and non-membership degrees of an alternative corresponding to each parameter. For instance, the entry a_{32} signifies the membership and non-membership degrees of the third alternative corresponding to the second parameter.

Example 2 The d-matrix of f provided in Example 1 is as follows:

$$[a_{ij}] = \begin{bmatrix} {}^{[0.1,0.4]}_{[0.4,0.5]} & {}^{0}_{1} & {}^{0}_{1} & {}^{[0.2,0.5]}_{[0.1,0.2]} \\ {}^{[0.4,0.6]}_{[0.2,0.3]} & {}^{0}_{1} & {}^{1}_{0} & {}^{0}_{1} \\ {}^{[0.7,0.8]}_{[0,0.1]} & {}^{0}_{1} & {}^{1}_{0} & {}^{[0.3,0.4]}_{[0.5,0.6]} \\ {}^{0}_{1} & {}^{[0,0.5]}_{[0.1,0.2]} & {}^{1}_{0} & {}^{0}_{1} \\ {}^{[0.1,0.4]}_{[0,0.2]} & {}^{0}_{1} & {}^{1}_{0} & {}^{[0,0.2]}_{[0.5,0.6]} \\ {}^{0}_{1} & {}^{[0.3,0.5]}_{[0.2,0.3]} & {}^{1}_{0} & {}^{[0.3,0.7]}_{[0.1,0.2]} \end{bmatrix}$$

Definition 6 Let $[a_{ij}] \in D_E[U]$. For all i and j, and for $\lambda, \varepsilon \in Int([0, 1])$, if $\alpha_{ij} = \lambda$ and $\beta_{ij} = \varepsilon$, then $[a_{ij}]$ is called (λ, ε)-d-matrix and is denoted by $[{}^{\lambda}_{\varepsilon}]$. Here, $[{}^{0}_{1}]$ is called empty d-matrix and $[{}^{1}_{0}]$ is called universal d-matrix.

Definition 7 Let $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$, $I_E := \{j : x_j \in E\}$, and $R \subseteq I_E$. If

$$\alpha^c_{ij} = \begin{cases} \alpha^a_{ij}, & j \in R \\ \alpha^b_{ij}, & j \in I_E \setminus R \end{cases} \quad \text{and} \quad \beta^c_{ij} = \begin{cases} \beta^a_{ij}, & j \in R \\ \beta^b_{ij}, & j \in I_E \setminus R \end{cases}$$

then $[c_{ij}]$ is called Rb-restriction of $[a_{ij}]$ and is denoted by $\left[(a_{Rb})_{ij}\right]$.

Briefly, if $[b_{ij}] = [{}^{0}_{1}]$, then $[(a_R)_{ij}]$ can be used instead of $\left[\left(a_{R^0_1}\right)_{ij}\right]$ and called R-restriction of $[a_{ij}]$. It is clear that

$$(a_R)_{ij} = \begin{cases} {}^{\alpha_{ij}}_{\beta_{ij}}, & j \in R \\ {}^{0}_{1}, & j \in I_E \setminus R \end{cases}$$

Example 3 For $R = \{1, 3, 4\}$ and $S = \{1, 3\}$, R_0^1-restriction and S-restriction of $[a_{ij}]$ provided in Example 2 are as follows:

$$\left[\left(a_{R_0^1}\right)_{ij}\right] = \begin{bmatrix} {[0.1,0.4] \atop [0.4,0.5]} & {1 \atop 0} & {0 \atop 1} & {[0.2,0.5] \atop [0.1,0.2]} \\ {[0.4,0.6] \atop [0.2,0.3]} & {1 \atop 0} & {1 \atop 0} & {0 \atop 1} \\ {[0.7,0.8] \atop [0,0.1]} & {1 \atop 0} & {1 \atop 0} & {[0.3,0.4] \atop [0.5,0.6]} \\ {0 \atop 1} & {1 \atop 0} & {1 \atop 0} & {0 \atop 1} \\ {[0.1,0.4] \atop [0,0.2]} & {1 \atop 0} & {1 \atop 0} & {[0,0.2] \atop [0.5,0.6]} \\ {0 \atop 1} & {1 \atop 0} & {1 \atop 0} & {[0.3,0.7] \atop [0.1,0.2]} \end{bmatrix} \text{ and } \left[(a_S)_{ij}\right] = \begin{bmatrix} {[0.1,0.4] \atop [0.4,0.5]} & {0 \atop 1} & {0 \atop 1} & {0 \atop 1} \\ {[0.4,0.6] \atop [0.2,0.3]} & {0 \atop 1} & {1 \atop 0} & {0 \atop 1} \\ {[0.7,0.8] \atop [0,0.1]} & {0 \atop 1} & {1 \atop 0} & {0 \atop 1} \\ {0 \atop 1} & {0 \atop 1} & {1 \atop 0} & {0 \atop 1} \\ {[0.1,0.4] \atop [0,0.2]} & {0 \atop 1} & {1 \atop 0} & {0 \atop 1} \\ {0 \atop 1} & {0 \atop 1} & {1 \atop 0} & {0 \atop 1} \end{bmatrix}$$

Definition 8 Let $[a_{ij}], [b_{ij}] \in D_E[U]$. For all i and j, if $\alpha^a_{ij} \tilde{\leq} \alpha^b_{ij}$ and $\beta^b_{ij} \tilde{\leq} \beta^a_{ij}$, then $[a_{ij}]$ is called a submatrix of $[b_{ij}]$ and is denoted by $[a_{ij}] \tilde{\subseteq} [b_{ij}]$.

Definition 9 Let $[a_{ij}], [b_{ij}] \in D_E[U]$. For all i and j, if $\alpha^a_{ij} = \alpha^b_{ij}$ and $\beta^a_{ij} = \beta^b_{ij}$, then $[a_{ij}]$ and $[b_{ij}]$ are called equal d-matrices and is denoted by $[a_{ij}] = [b_{ij}]$.

Proposition 3 *Let* $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$. *Then,*

i. $[a_{ij}] \tilde{\subseteq} \left[{1 \atop 0}\right]$
ii. $\left[{0 \atop 1}\right] \tilde{\subseteq} [a_{ij}]$
iii. $[a_{ij}] \tilde{\subseteq} [a_{ij}]$
iv. $([a_{ij}] = [b_{ij}] \wedge [b_{ij}] = [c_{ij}]) \Rightarrow [a_{ij}] = [c_{ij}]$
v. $([a_{ij}] \tilde{\subseteq} [b_{ij}] \wedge [b_{ij}] \tilde{\subseteq} [a_{ij}]) \Leftrightarrow [a_{ij}] = [b_{ij}]$
vi. $([a_{ij}] \tilde{\subseteq} [b_{ij}] \wedge [b_{ij}] \tilde{\subseteq} [c_{ij}]) \Rightarrow [a_{ij}] \tilde{\subseteq} [c_{ij}]$

Definition 10 Let $[a_{ij}], [b_{ij}] \in D_E[U]$. If $[a_{ij}] \tilde{\subseteq} [b_{ij}]$ and $[a_{ij}] \neq [b_{ij}]$, then $[a_{ij}]$ is called a proper submatrix of $[b_{ij}]$ and is denoted by $[a_{ij}] \tilde{\subsetneq} [b_{ij}]$.

Definition 11 Let $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$. For all i and j, if $\alpha^c_{ij} = \sup\{\alpha^a_{ij}, \alpha^b_{ij}\}$ and $\beta^c_{ij} = \inf\{\beta^a_{ij}, \beta^b_{ij}\}$, then $[c_{ij}]$ is called union of $[a_{ij}]$ and $[b_{ij}]$ and is denoted by $[a_{ij}] \tilde{\cup} [b_{ij}]$.

Definition 12 Let $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$. For all i and j, if $\alpha^c_{ij} = \inf\{\alpha^a_{ij}, \alpha^b_{ij}\}$ and $\beta^c_{ij} = \sup\{\beta^a_{ij}, \beta^b_{ij}\}$, then $[c_{ij}]$ is called intersection of $[a_{ij}]$ and $[b_{ij}]$ and is denoted by $[a_{ij}] \tilde{\cap} [b_{ij}]$.

Proposition 4 *Let* $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$. *Then,*

i. $[a_{ij}] \tilde{\cup} [a_{ij}] = [a_{ij}]$ *and* $[a_{ij}] \tilde{\cap} [a_{ij}] = [a_{ij}]$
ii. $[a_{ij}] \tilde{\cup} \left[{0 \atop 1}\right] = [a_{ij}]$ *and* $[a_{ij}] \tilde{\cap} \left[{1 \atop 0}\right] = [a_{ij}]$
iii. $[a_{ij}] \tilde{\cup} \left[{1 \atop 0}\right] = \left[{1 \atop 0}\right]$ *and* $[a_{ij}] \tilde{\cap} \left[{0 \atop 1}\right] = \left[{0 \atop 1}\right]$
iv. $[a_{ij}] \tilde{\cup} [b_{ij}] = [b_{ij}] \tilde{\cup} [a_{ij}]$ *and* $[a_{ij}] \tilde{\cap} [b_{ij}] = [b_{ij}] \tilde{\cap} [a_{ij}]$
v. $([a_{ij}] \tilde{\cup} [b_{ij}]) \tilde{\cup} [c_{ij}] = [a_{ij}] \tilde{\cup} ([b_{ij}] \tilde{\cup} [c_{ij}])$ *and* $([a_{ij}] \tilde{\cap} [b_{ij}]) \tilde{\cap} [c_{ij}] = [a_{ij}] \tilde{\cap} ([b_{ij}] \tilde{\cap} [c_{ij}])$
vi. $[a_{ij}] \tilde{\cup} ([b_{ij}] \tilde{\cap} [c_{ij}]) = ([a_{ij}] \tilde{\cup} [b_{ij}]) \tilde{\cap} ([a_{ij}] \tilde{\cup} [c_{ij}])$
$[a_{ij}] \tilde{\cap} ([b_{ij}] \tilde{\cup} [c_{ij}]) = ([a_{ij}] \tilde{\cap} [b_{ij}]) \tilde{\cup} ([a_{ij}] \tilde{\cap} [c_{ij}])$
vii. $[a_{ij}] \tilde{\subseteq} [b_{ij}] \Rightarrow [a_{ij}] \tilde{\cup} [b_{ij}] = [b_{ij}]$ *and* $[a_{ij}] \tilde{\subseteq} [b_{ij}] \Rightarrow [a_{ij}] \tilde{\cap} [b_{ij}] = [a_{ij}]$

Proof *vi.* Let $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$. Then,

$$\begin{aligned}
[a_{ij}]\tilde{\cup}([b_{ij}]\tilde{\cap}[c_{ij}]) &= [a_{ij}]\tilde{\cup}\begin{bmatrix}\inf\left\{\alpha^b_{ij},\alpha^c_{ij}\right\}\\ \sup\left\{\beta^b_{ij},\beta^c_{ij}\right\}\end{bmatrix}\\
&= \begin{bmatrix}\sup\left\{\alpha^a_{ij},\inf\left\{\alpha^b_{ij},\alpha^c_{ij}\right\}\right\}\\ \inf\left\{\beta^a_{ij},\sup\left\{\beta^b_{ij},\beta^c_{ij}\right\}\right\}\end{bmatrix}\\
&= \begin{bmatrix}\inf\left\{\sup\left\{\alpha^a_{ij},\alpha^b_{ij}\right\},\sup\left\{\alpha^a_{ij},\alpha^c_{ij}\right\}\right\}\\ \sup\left\{\inf\left\{\beta^a_{ij},\beta^b_{ij}\right\},\inf\left\{\beta^a_{ij},\beta^c_{ij}\right\}\right\}\end{bmatrix}\\
&= \begin{bmatrix}\sup\left\{\alpha^a_{ij},\alpha^b_{ij}\right\}\\ \inf\left\{\beta^a_{ij},\beta^b_{ij}\right\}\end{bmatrix}\tilde{\cap}\begin{bmatrix}\sup\left\{\alpha^a_{ij},\alpha^c_{ij}\right\}\\ \inf\left\{\beta^a_{ij},\beta^c_{ij}\right\}\end{bmatrix}\\
&= ([a_{ij}]\tilde{\cup}[b_{ij}])\tilde{\cap}([a_{ij}]\tilde{\cup}[c_{ij}])
\end{aligned}$$

□

Example 4 Let $E = \{x_1, x_2, x_3\}$ and $U = \{u_1, u_2\}$. Assume that two d-matrices $[a_{ij}]$ and $[b_{ij}]$ are as follows:

$$[a_{ij}] = \begin{bmatrix}
\begin{smallmatrix}[0.2,0.4]\\ [0,0.6]\end{smallmatrix} & \begin{smallmatrix}0.3\\ 0.4\end{smallmatrix} & \begin{smallmatrix}[0.3,0.4]\\ [0.1,0.2]\end{smallmatrix}\\
\begin{smallmatrix}0\\ 1\end{smallmatrix} & \begin{smallmatrix}[0,0.3]\\ [0.4,0.6]\end{smallmatrix} & \begin{smallmatrix}0.5\\ [0,0.4]\end{smallmatrix}\\
\begin{smallmatrix}[0.5,0.7]\\ [0,0.3]\end{smallmatrix} & \begin{smallmatrix}0.2\\ 0.7\end{smallmatrix} & \begin{smallmatrix}[0.5,0.6]\\ [0.1,0.3]\end{smallmatrix}
\end{bmatrix} \quad\text{and}\quad [b_{ij}] = \begin{bmatrix}
\begin{smallmatrix}[0.1,0.3]\\ [0.1,0.2]\end{smallmatrix} & \begin{smallmatrix}[0.2,0.4]\\ [0.3,0.5]\end{smallmatrix} & \begin{smallmatrix}[0.2,0.8]\\ [0,0.1]\end{smallmatrix}\\
\begin{smallmatrix}[0.3,0.5]\\ [0.1,0.2]\end{smallmatrix} & \begin{smallmatrix}[0.1,0.3]\\ [0.1,0.2]\end{smallmatrix} & \begin{smallmatrix}0.6\\ 0.1\end{smallmatrix}\\
\begin{smallmatrix}[0.4,0.8]\\ [0.1,0.2]\end{smallmatrix} & \begin{smallmatrix}0\\ 1\end{smallmatrix} & \begin{smallmatrix}[0,0.1]\\ [0,0.4]\end{smallmatrix}
\end{bmatrix}$$

Then,

$$[a_{ij}]\tilde{\cup}[b_{ij}] = \begin{bmatrix}
\begin{smallmatrix}[0.2,0.4]\\ [0,0.2]\end{smallmatrix} & \begin{smallmatrix}[0.3,0.4]\\ [0.3,0.4]\end{smallmatrix} & \begin{smallmatrix}[0.3,0.8]\\ [0,0.1]\end{smallmatrix}\\
\begin{smallmatrix}[0.3,0.5]\\ [0.1,0.2]\end{smallmatrix} & \begin{smallmatrix}[0.1,0.3]\\ [0.1,0.2]\end{smallmatrix} & \begin{smallmatrix}0.6\\ [0,0.1]\end{smallmatrix}\\
\begin{smallmatrix}[0.5,0.8]\\ [0,0.2]\end{smallmatrix} & \begin{smallmatrix}0.2\\ 0.7\end{smallmatrix} & \begin{smallmatrix}[0.5,0.6]\\ [0,0.3]\end{smallmatrix}
\end{bmatrix} \quad\text{and}\quad [a_{ij}]\tilde{\cap}[b_{ij}] = \begin{bmatrix}
\begin{smallmatrix}[0.1,0.3]\\ [0.1,0.6]\end{smallmatrix} & \begin{smallmatrix}[0.2,0.3]\\ [0.4,0.5]\end{smallmatrix} & \begin{smallmatrix}[0.2,0.4]\\ [0.1,0.2]\end{smallmatrix}\\
\begin{smallmatrix}0\\ 1\end{smallmatrix} & \begin{smallmatrix}[0,0.3]\\ [0.4,0.6]\end{smallmatrix} & \begin{smallmatrix}0.5\\ [0.1,0.4]\end{smallmatrix}\\
\begin{smallmatrix}[0.4,0.7]\\ [0.1,0.3]\end{smallmatrix} & \begin{smallmatrix}0\\ 1\end{smallmatrix} & \begin{smallmatrix}[0,0.1]\\ [0.1,0.4]\end{smallmatrix}
\end{bmatrix}$$

Definition 13 Let $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$. For all i and j, if $\alpha^c_{ij} = \inf\{\alpha^a_{ij}, \beta^b_{ij}\}$ and $\beta^c_{ij} = \sup\{\beta^a_{ij}, \alpha^b_{ij}\}$, then $[c_{ij}]$ is called difference between $[a_{ij}]$ and $[b_{ij}]$ and is denoted by $[a_{ij}]\tilde{\setminus}[b_{ij}]$.

Proposition 5 *Let* $[a_{ij}] \in D_E[U]$. *Then,*

i. $[a_{ij}]\tilde{\setminus}\begin{bmatrix}0\\1\end{bmatrix} = [a_{ij}]$
ii. $[a_{ij}]\tilde{\setminus}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0\\1\end{bmatrix}$
iii. $\begin{bmatrix}0\\1\end{bmatrix}\tilde{\setminus}[a_{ij}] = \begin{bmatrix}0\\1\end{bmatrix}$

Note 3 The difference operation does not provide associative and commutative properties.

Definition 14 Let $[a_{ij}], [b_{ij}] \in D_E[U]$. For all i and j, if $\alpha^b_{ij} = \beta^a_{ij}$ and $\beta^b_{ij} = \alpha^a_{ij}$, then $[b_{ij}]$ is complement of $[a_{ij}]$ and is denoted by $[a_{ij}]^{\tilde{c}}$ or $[a^{\tilde{c}}_{ij}]$. It is clear that, $[a_{ij}]^{\tilde{c}} = \begin{bmatrix}1\\0\end{bmatrix}\tilde{\setminus}[a_{ij}]$.

Proposition 6 *Let* $[a_{ij}], [b_{ij}] \in D_E[U]$. *Then,*

i. $([a_{ij}]^{\tilde{c}})^{\tilde{c}} = [a_{ij}]$
ii. $\begin{bmatrix}0\\1\end{bmatrix}^{\tilde{c}} = \begin{bmatrix}1\\0\end{bmatrix}$
iii. $[a_{ij}]\tilde{\setminus}[b_{ij}] = [a_{ij}]\tilde{\cap}[b_{ij}]^{\tilde{c}}$
iv. $[a_{ij}]\tilde{\subseteq}[b_{ij}] \Rightarrow [b_{ij}]^{\tilde{c}}\tilde{\subseteq}[a_{ij}]^{\tilde{c}}$

Proposition 7 *Let* $[a_{ij}], [b_{ij}] \in D_E[U]$. *Then, the following De Morgan's laws are valid:*

i. $([a_{ij}]\tilde{\cup}[b_{ij}])^{\tilde{c}} = [a_{ij}]^{\tilde{c}}\tilde{\cap}[b_{ij}]^{\tilde{c}}$
ii. $([a_{ij}]\tilde{\cap}[b_{ij}])^{\tilde{c}} = [a_{ij}]^{\tilde{c}}\tilde{\cup}[b_{ij}]^{\tilde{c}}$

Proof *i.* Let $[a_{ij}], [b_{ij}] \in D_E[U]$. Then,

$$([a_{ij}]\tilde{\cup}[b_{ij}])^{\tilde{c}} = \begin{bmatrix}\sup\{\alpha^a_{ij},\alpha^b_{ij}\}\\ \inf\{\beta^a_{ij},\beta^b_{ij}\}\end{bmatrix}^{\tilde{c}} = \begin{bmatrix}\inf\{\beta^a_{ij},\beta^b_{ij}\}\\ \sup\{\alpha^a_{ij},\alpha^b_{ij}\}\end{bmatrix} = \begin{bmatrix}\beta^a_{ij}\\ \alpha^a_{ij}\end{bmatrix}\tilde{\cap}\begin{bmatrix}\beta^b_{ij}\\ \alpha^b_{ij}\end{bmatrix} = \left[a_{ij}\right]^{\tilde{c}}\tilde{\cap}\left[b_{ij}\right]^{\tilde{c}}$$

□

Definition 15 Let $[a_{ij}], [b_{ij}], [c_{ij}] \in D_E[U]$. For all i and j, if

$$\alpha^c_{ij} = \sup\left\{\inf\{\alpha^a_{ij}, \beta^b_{ij}\}, \inf\{\alpha^b_{ij}, \beta^a_{ij}\}\right\} \quad \text{and} \quad \beta^c_{ij} = \inf\left\{\sup\{\beta^a_{ij}, \alpha^b_{ij}\}, \sup\{\beta^b_{ij}, \alpha^a_{ij}\}\right\}$$

then $[c_{ij}]$ is called symmetric difference between $[a_{ij}]$ and $[b_{ij}]$ and is denoted by $[a_{ij}]\tilde{\triangle}[b_{ij}]$.

Proposition 8 *Let* $[a_{ij}], [b_{ij}] \in D_E[U]$. *Then,*

i. $[a_{ij}]\tilde{\triangle}\begin{bmatrix}0\\1\end{bmatrix} = [a_{ij}]$
ii. $[a_{ij}]\tilde{\triangle}\begin{bmatrix}1\\0\end{bmatrix} = [a_{ij}]^{\tilde{c}}$
iii. $[a_{ij}]\tilde{\triangle}[b_{ij}] = [b_{ij}]\tilde{\triangle}[a_{ij}]$

Note 4 The symmetric difference operation does not provide associative property.

Example 5 For $[a_{ij}]$ and $[b_{ij}]$ in Example 4, $[a_{ij}]\tilde{\setminus}[b_{ij}]$ and $[a_{ij}]\tilde{\triangle}[b_{ij}]$ are as follows:

$$[a_{ij}]\tilde{\setminus}[b_{ij}] = \begin{bmatrix} \begin{matrix}[0.1,0.2]\\ [0.1,0.6]\end{matrix} & \begin{matrix}0.3\\ 0.4\end{matrix} & \begin{matrix}[0,0.1]\\ [0.2,0.8]\end{matrix} \\ \begin{matrix}0\\ 1\end{matrix} & \begin{matrix}[0,0.2]\\ [0.4,0.6]\end{matrix} & \begin{matrix}0.1\\ 0.6\end{matrix} \\ \begin{matrix}[0.1,0.2]\\ [0.4,0.8]\end{matrix} & \begin{matrix}0.2\\ 0.7\end{matrix} & \begin{matrix}[0,0.4]\\ [0.1,0.3]\end{matrix} \end{bmatrix} \quad \text{and} \quad [a_{ij}]\tilde{\triangle}[b_{ij}] = \begin{bmatrix} \begin{matrix}[0.1,0.3]\\ [0.1,0.4]\end{matrix} & \begin{matrix}[0.3,0.4]\\ [0.3,0.4]\end{matrix} & \begin{matrix}[0.1,0.2]\\ [0.2,0.4]\end{matrix} \\ \begin{matrix}[0.3,0.5]\\ [0.1,0.2]\end{matrix} & \begin{matrix}[0.1,0.3]\\ [0.1,0.3]\end{matrix} & \begin{matrix}[0.1,0.4]\\ 0.5\end{matrix} \\ \begin{matrix}[0.1,0.3]\\ [0.4,0.7]\end{matrix} & \begin{matrix}0.2\\ 0.7\end{matrix} & \begin{matrix}[0,0.4]\\ [0.1,0.3]\end{matrix} \end{bmatrix}$$

Definition 16 Let $[a_{ij}], [b_{ij}] \in D_E[U]$. If $[a_{ij}]\tilde{\cap}[b_{ij}] = \begin{bmatrix}0\\1\end{bmatrix}$, then $[a_{ij}]$ and $[b_{ij}]$ are called disjoint.

Definition 17 Let $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$, $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$, and $[c_{ip}]_{m\times n_1 n_2} \in D_{E_1\times E_2}[U]$ such that $p = n_2(j-1)+k$. For all i and p, if $\alpha^c_{ip} = \inf\{\alpha^a_{ij}, \alpha^b_{ik}\}$ and $\beta^c_{ip} = \sup\{\beta^a_{ij}, \beta^b_{ik}\}$, then $[c_{ip}]$ is called AND-product of $[a_{ij}]$ and $[b_{ik}]$ and is denoted by $[a_{ij}]\wedge[b_{ik}]$.

Definition 18 Let $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$, $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$, and $[c_{ip}]_{m\times n_1 n_2} \in D_{E_1\times E_2}[U]$ such that $p = n_2(j-1)+k$. For all i and p, if $\alpha^c_{ip} = \sup\{\alpha^a_{ij}, \alpha^b_{ik}\}$ and $\beta^c_{ip} = \inf\{\beta^a_{ij}, \beta^b_{ik}\}$, then $[c_{ip}]$ is called OR-product of $[a_{ij}]$ and $[b_{ik}]$ and is denoted by $[a_{ij}]\vee[b_{ik}]$.

Definition 19 Let $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$, $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$, and $[c_{ip}]_{m\times n_1 n_2} \in D_{E_1\times E_2}[U]$ such that $p = n_2(j-1)+k$. For all i and p, if $\alpha^c_{ip} = \inf\{\alpha^a_{ij}, \beta^b_{ik}\}$ and $\beta^c_{ip} = \sup\{\beta^a_{ij}, \alpha^b_{ik}\}$, then $[c_{ip}]$ is called ANDNOT-product of $[a_{ij}]$ and $[b_{ik}]$ and is denoted by $[a_{ij}]\overline{\wedge}[b_{ik}]$.

Definition 20 Let $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$, $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$, and $[c_{ip}]_{m\times n_1 n_2} \in D_{E_1\times E_2}[U]$ such that $p = n_2(j-1)+k$. For all i and p, if $\alpha^c_{ip} = \sup\{\alpha^a_{ij}, \beta^b_{ik}\}$ and $\beta^c_{ip} = \inf\{\beta^a_{ij}, \alpha^b_{ik}\}$, then $[c_{ip}]$ is called ORNOT-product of $[a_{ij}]$ and $[b_{ik}]$ and is denoted by $[a_{ij}]\underline{\vee}[b_{ik}]$.

Example 6 For $[a_{ij}]$ and $[b_{ik}]$ in Example 4, $[a_{ij}]\overline{\wedge}[b_{ik}]$ is as follows:

$$[a_{ij}]\overline{\wedge}[b_{ik}] = \begin{bmatrix} \substack{[0.1,0.2]\\ [0.1,0.6]} & \substack{[0.2,0.4]\\ [0.2,0.6]} & \substack{[0,0.1]\\ [0.2,0.8]} & \substack{[0.1,0.2]\\ 0.4} & \substack{0.3\\ 0.4} & \substack{[0,0.1]\\ [0.4,0.8]} & \substack{[0.1,0.2]\\ [0.1,0.3]} & \substack{[0.3,0.4]\\ [0.2,0.4]} & \substack{[0,0.1]\\ [0.2,0.8]} \\ \substack{0\\ 1} & \substack{0\\ 1} & \substack{0\\ 1} & \substack{[0,0.2]\\ [0.4,0.6]} & \substack{[0,0.2]\\ [0.4,0.6]} & \substack{[0,0.1]\\ 0.6} & \substack{[0.1,0.2]\\ [0.3,0.5]} & \substack{[0.1,0.2]\\ [0.1,0.4]} & \substack{0.1\\ 0.6} \\ \substack{[0.1,0.2]\\ [0.4,0.8]} & \substack{[0.5,0.7]\\ [0,0.3]} & \substack{[0,0.4]\\ [0,0.3]} & \substack{[0.1,0.2]\\ [0.7,0.8]} & \substack{0.2\\ 0.7} & \substack{[0,0.2]\\ 0.7} & \substack{[0.1,0.2]\\ [0.4,0.8]} & \substack{[0.5,0.6]\\ [0.1,0.3]} & \substack{[0,0.4]\\ [0.1,0.3]} \end{bmatrix}$$

Proposition 9 *Let* $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$, $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$, *and* $[c_{il}]_{m\times n_3} \in D_{E_3}[U]$. *Then,*

i. $([a_{ij}] \wedge [b_{ik}]) \wedge [c_{il}] = [a_{ij}] \wedge ([b_{ik}] \wedge [c_{il}])$
ii. $([a_{ij}] \vee [b_{ik}]) \vee [c_{il}] = [a_{ij}] \vee ([b_{ik}] \vee [c_{il}])$

Proof *i.* Let $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$, $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$, $[c_{il}]_{m\times n_3} \in D_{E_3}[U]$, $[a_{ij}] \wedge [b_{ik}] = [d_{ip}]$, $[b_{ik}] \wedge [c_{il}] = [e_{ir}]$, $([a_{ij}] \wedge [b_{ik}]) \wedge [c_{il}] = [f_{is}]$, and $[a_{ij}] \wedge ([b_{ik}] \wedge [c_{il}]) = [h_{it}]$. Therefore, $[d_{ip}]_{m\times n_1 n_2} \in D_{E_1\times E_2}[U]$, $[e_{ir}]_{m\times n_2 n_3} \in D_{E_2\times E_3}[U]$, and $[f_{is}]_{m\times n_1 n_2 n_3}$, $[h_{it}]_{m\times n_1 n_2 n_3} \in D_{E_1\times E_2\times E_3}[U]$. Because of Definition 17, since $p = n_2(j-1)+k$ and $s = n_3(p-1)+l$, then

$$s = n_3 n_2(j-1) + n_3(k-1) + l$$

Similarly, because of Definition 17, since $r = n_3(k-1)+l$ and $t = n_2 n_3(j-1)+r$, then

$$t = n_2 n_3(j-1) + n_3(k-1) + l$$

Moreover, for all i, s, and t, since

$$\alpha^f_{is} = \inf\{\inf\{\alpha^a_{ij}, \alpha^b_{ik}\}, \alpha^c_{il}\} \quad \text{and} \quad \beta^f_{is} = \sup\{\sup\{\beta^a_{ij}, \beta^b_{ik}\}, \beta^c_{il}\}$$

and

$$\alpha^h_{it} = \inf\{\alpha^a_{ij}, \inf\{\alpha^b_{ik}, \alpha^c_{il}\}\} \quad \text{and} \quad \beta^h_{it} = \sup\{\beta^a_{ij}, \sup\{\beta^b_{ik}, \beta^c_{il}\}\}$$

then $\alpha^f_{is} = \alpha^h_{it}$ and $\beta^f_{is} = \beta^h_{it}$. Thus, $([a_{ij}] \wedge [b_{ik}]) \wedge [c_{il}] = [a_{ij}] \wedge ([b_{ik}] \wedge [c_{il}])$. □

Proposition 10 *Let* $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$ *and* $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$. *Then, the following De Morgan's laws are valid:*

i. $([a_{ij}] \vee [b_{ik}])^{\tilde{c}} = [a_{ij}]^{\tilde{c}} \wedge [b_{ik}]^{\tilde{c}}$
ii. $([a_{ij}] \wedge [b_{ik}])^{\tilde{c}} = [a_{ij}]^{\tilde{c}} \vee [b_{ik}]^{\tilde{c}}$
iii. $([a_{ij}] \underline{\vee} [b_{ik}])^{\tilde{c}} = [a_{ij}]^{\tilde{c}} \overline{\wedge} [b_{ik}]^{\tilde{c}}$

iv. $([a_{ij}]\,\overline{\wedge}\,[b_{ik}])^{\tilde{c}} = [a_{ij}]^{\tilde{c}} \,\underline{\vee}\, [b_{ik}]^{\tilde{c}}$

Proof *iv.* Let $[a_{ij}]_{m\times n_1} \in D_{E_1}[U]$ and $[b_{ik}]_{m\times n_2} \in D_{E_2}[U]$. Then,

$$([a_{ij}]\overline{\wedge}[b_{ik}])^{\tilde{c}} = \begin{bmatrix} \inf\{\alpha^a_{ij},\beta^b_{ik}\} \\ \sup\{\beta^a_{ij},\alpha^b_{ik}\} \end{bmatrix}^{\tilde{c}} = \begin{bmatrix} \sup\{\beta^a_{ij},\alpha^b_{ik}\} \\ \inf\{\alpha^a_{ij},\beta^b_{ik}\} \end{bmatrix} = \left[a_{ij}\right]^{\tilde{c}} \underline{\vee} \left[b_{ij}\right]^{\tilde{c}}$$

□

Note 5 The aforesaid products of d-matrices do not provide distributive property upon each other and commutative property. Moreover, ANDNOT-product and ORNOT-product do not provide associative property.

4 The configured soft decision-making method

This section first configures the SDM method (Aydın and Enginoğlu 2021a) to operate it in d-matrices space. Thus, we can employ this method in the presence of decision-making problems. The configured method is used to model a problem containing parameters and alternatives with multiple intuitionistic fuzzy values. This method consists of a pre-processing step and the main process steps. In the pre-processing, the multiple intuitionistic fuzzy values are inputted for each parameter and the alternatives corresponding to the parameters. In the first step of the main process, a d-matrix is constructed using the membership function, the non-membership function, and the multiple intuitionistic fuzzy values. In the second, a column matrix with the *ivif*-values is obtained by weighting the non-zero-indexed rows of the d-matrix with the zero-indexed one. In the third step, a score matrix is attained with the difference between membership and non-membership values in each entry of this matrix. Fourthly, an interval-valued fuzzy decision set over a set of alternatives is produced by normalising the score values and translating them to a closed classical subinterval of [0, 1]. In the final step, the optimal alternatives are selected through the linear ordering relation (Xu and Yager 2006). Henceforth, $I_n = \{1, 2, 3, \ldots, n\}$ and $I^*_n = \{0, 1, 2, \ldots, n\}$.

Algorithm Steps of the Configured Method

Input Step. Input the values μ^{ij}_t and ν^{ij}_t such that $i \in I^*_{m-1}$, $j \in I_n$, and $t \in I_s$

Main Steps

Step 1. Construct a d-matrix $[a_{ij}]_{m\times n}$ defined by $a_{ij} := \begin{smallmatrix}\alpha^a_{ij}\\ \beta^a_{ij}\end{smallmatrix}$

Here, $\pi^{ij}_t = 1 - \mu^{ij}_t - \nu^{ij}_t$, $I = \left\{p : \mu^{ij}_p = \max\limits_t \mu^{ij}_t\right\}$, $J = \left\{r : \nu^{ij}_r = \max\limits_t \nu^{ij}_t\right\}$, $i \in I^*_{m-1}$, $j \in I_n$, and $t \in I_s$ such that

$$\alpha^a_{ij} := \left[\frac{\min\limits_t \mu^{ij}_t}{\max\limits_t \mu^{ij}_t + \max\limits_t \nu^{ij}_t + \min\left\{\min\limits_{p\in I} \pi^{ij}_p, \min\limits_{r\in J} \pi^{ij}_r\right\}}, \frac{\max\limits_t \mu^{ij}_t}{\max\limits_t \mu^{ij}_t + \max\limits_t \nu^{ij}_t + \min\left\{\min\limits_{p\in I} \pi^{ij}_p, \min\limits_{r\in J} \pi^{ij}_r\right\}}\right]$$

and

$$\beta^a_{ij} := \left[\frac{\min\limits_t \nu^{ij}_t}{\max\limits_t \mu^{ij}_t + \max\limits_t \nu^{ij}_t + \min\left\{\min\limits_{p\in I} \pi^{ij}_p, \min\limits_{r\in J} \pi^{ij}_r\right\}}, \frac{\max\limits_t \nu^{ij}_t}{\max\limits_t \mu^{ij}_t + \max\limits_t \nu^{ij}_t + \min\left\{\min\limits_{p\in I} \pi^{ij}_p, \min\limits_{r\in J} \pi^{ij}_r\right\}}\right]$$

Step 2. Obtain the *ivif*-valued column matrix $\begin{bmatrix}\alpha_{i1}\\ \beta_{i1}\end{bmatrix}_{(m-1)\times 1}$ defined by

$$\alpha_{i1} := \frac{1}{\lambda}\sum_{j=1}^{n}\alpha_{0j}^{a}\alpha_{ij}^{a} \quad \text{and} \quad \beta_{i1} := \frac{1}{\lambda}\sum_{j=1}^{n}\beta_{0j}^{a}\beta_{ij}^{a}$$

such that $i \in I_{m-1}$
Here,

$$\lambda := \frac{1}{2}\sum_{j=1}^{n}\left(1 + \frac{(\alpha_{0j}^{a})^{-} + (\alpha_{0j}^{a})^{+}}{2} - \frac{(\beta_{0j}^{a})^{-} + (\beta_{0j}^{a})^{+}}{2}\right)$$

Step 3. Obtain the score matrix $[s_{i1}]_{(m-1)\times 1}$ defined by $s_{i1} := \alpha_{i1} - \beta_{i1}$ such that $i \in I_{m-1}$
Step 4. Obtain the decision set $\{^{d(u_k)}u_k | u_k \in U\}$ such that

$$d(u_k) = \begin{cases} \left[\frac{s_{k1}^{-}+|\min_i s_{i1}^{-}|}{\max_i s_{i1}^{+}+|\min_i s_{i1}^{-}|}, \frac{s_{k1}^{+}+|\min_i s_{i1}^{-}|}{\max_i s_{i1}^{+}+|\min_i s_{i1}^{-}|}\right], & \max_i s_{i1}^{+} + |\min_i s_{i1}^{-}| \neq 0 \\ [1, 1], & \max_i s_{i1}^{+} + |\min_i s_{i1}^{-}| = 0 \end{cases}$$

Step 5. Select the optimal elements among the alternatives via linear ordering relation (Xu and Yager 2006)

$$[\gamma_1^-, \gamma_1^+] \leq_{XY} [\gamma_2^-, \gamma_2^+]$$
$$\Leftrightarrow [(\gamma_1^- + \gamma_1^+ < \gamma_2^- + \gamma_2^+) \vee (\gamma_1^- + \gamma_1^+ = \gamma_2^- + \gamma_2^+ \wedge \gamma_1^- - \gamma_1^+ \leq \gamma_2^- - \gamma_2^+)]$$

Here, $\alpha_{0j}^{a} = [(\alpha_{0j}^{a})^{-}, (\alpha_{0j}^{a})^{+}]$, $\beta_{0j}^{a} = [(\beta_{0j}^{a})^{-}, (\beta_{0j}^{a})^{+}]$, and $s_{i1} = [s_{i1}^{-}, s_{i1}^{+}]$.

5 An application of the configured method to performance-based value assignment problem

In this section, we apply the configured method to the PVA problem for seven known filters used in image denoising, namely Based on Pixel Density Filter (BPDF) (Erkan and Gökrem 2018), Modified Decision-Based Unsymmetric Trimmed Median Filter (MDBUTMF) (Esakkirajan et al. 2011), Decision-Based Algorithm (DBAIN) (Srinivasan and Ebenezer 2007), Noise Adaptive Fuzzy Switching Median Filter (NAFSMF) (Toh and Isa 2010), Different Applied Median Filter (DAMF) (Erkan et al. 2018), Adaptive Weighted Mean Filter (AWMF) (Tang et al. 2016), and Adaptive Riesz Mean Filter (ARmF) (Enginoğlu et al. 2019b). Hereinafter, let $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7\}$ be an alternative set such that $u_1 =$ "BPDF", $u_2 =$ "MDBUTMF", $u_3 =$ "DBAIN", $u_4 =$ "NAFSMF", $u_5 =$ "DAMF", $u_6 =$ "AWMF", and $u_7 =$ "ARmF". Moreover, let $E = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$ be a parameter set determined by a decision-maker such that $x_1 =$ "*noise density* 10%", $x_2 =$ "*noise density* 20%", $x_3 =$ "*noise density* 30%", $x_4 =$ "*noise density* 40%", $x_5 =$ "*noise density* 50%", $x_6 =$ "*noise density* 60%", $x_7 =$ "*noise density* 70%", $x_8 =$ "*noise density* 80%", and $x_9 =$ "*noise density* 90%".

First, we consider 20 traditional test images, i.e. "Lena", "Cameraman", "Barbara", "Baboon", "Peppers", "Living Room", "Lake", "Plane", "Hill", "Pirate", "Boat", "House", "Bridge", "Elaine", "Flintstones", "Flower", "Parrot", "Dark-Haired Woman", "Blonde Woman", and "Einstein". To this end, we present the noise-removal performance values of the aforesaid filters by Structural Similarity (SSIM) (Wang et al. 2004) for the images at noise densities ranging from 10% to 90%, in Tables 1, 2, 3, and 4, respectively. Moreover, we obtain the results herein by MATLAB R2021a. When the SSIM values provided in the tables are examined, it is observed that ARmF absolutely performs better than the other filters at all the noise densities and for all the images. However, it is non-obvious which one is the second and third etc. Our motivation is to overcome this problem.

For the problem, let (μ_t^{ij}) be ordered-vigintuple such that μ_t^{ij} corresponds to the SSIM results in Tables 1, 2, 3, and 4 obtained by t^{th} image for i^{th} filter at j^{th} noise density. Here, since $\nu_t^{ij} = 1 - \mu_t^{ij}$ and $\pi_t^{ij} = 0$ such that $i \in I_7$, $j \in I_9$, and $t \in I_{20}$, then for d-matrix $[a_{ij}]$,

$$\alpha_{ij}^a := \left[\frac{\min\limits_t \mu_t^{ij}}{\max\limits_t \mu_t^{ij} + \max\limits_t\{1-\mu_t^{ij}\}}, \frac{\max\limits_t \mu_t^{ij}}{\max\limits_t \mu_t^{ij} + \max\limits_t\{1-\mu_t^{ij}\}}\right]$$

and

$$\beta_{ij}^a := \left[\frac{\min\limits_t\{1-\mu_t^{ij}\}}{\max\limits_t \mu_t^{ij} + \max\limits_t\{1-\mu_t^{ij}\}}, \frac{\max\limits_t\{1-\mu_t^{ij}\}}{\max\limits_t \mu_t^{ij} + \max\limits_t\{1-\mu_t^{ij}\}}\right]$$

For example, the ordered-vigintuple

$$(\mu_t^{54}) = (0.9488, 0.9759, 0.9013, 0.9356, 0.9110, 0.9152, 0.9285, 0.9648, 0.9181, 0.9332, 0.9123, 0.9861, 0.8953, 0.8961, 0.9173, 0.9513, 0.9563, 0.9743, 0.9053, 0.9445)$$

indicates SSIM results of DAMF for 20 traditional test images at noise density 40%. Since

$$\alpha_{54}^a = \left[\frac{\min\limits_t \mu_t^{54}}{\max\limits_t \mu_t^{54} + \max\limits_t\{1-\mu_t^{54}\}}, \frac{\max\limits_t \mu_t^{54}}{\max\limits_t \mu_t^{54} + \max\limits_t\{1-\mu_t^{54}\}}\right] = \left[\frac{0.8953}{0.9861+0.1047}, \frac{0.9861}{0.9861+0.1047}\right] = [0.8207, 0.9040]$$

and

$$\beta_{54}^a = \left[\frac{\min\limits_t\{1-\mu_t^{54}\}}{\max\limits_t \mu_t^{54} + \max\limits_t\{1-\mu_t^{54}\}}, \frac{\max\limits_t\{1-\mu_t^{54}\}}{\max\limits_t \mu_t^{54} + \max\limits_t\{1-\mu_t^{54}\}}\right] = \left[\frac{0.0139}{0.9861+0.1047}, \frac{0.1047}{0.9861+0.1047}\right] = [0.0127, 0.0960]$$

then $a_{54} = \frac{[0.8207,0.9040]}{[0.0127,0.0960]}$. Here, $[0.8207, 0.9040]$ signifies that the success of DAMF on image denoising at noise density 40% ranges from approximately 82% to 90%. Moreover, $[0.0127, 0.0960]$ means that the rate of DAMF's failure in image denoising at the same noise density occurs approximately between 1% and 9%. Similarly, the all rows of the d-matrix $[a_{ij}]$ but the zero-indexed row can be obtained. Besides, suppose that the noise-removal performances of the filters are more significant in high noise densities, in which noisy pixels outnumber uncorrupted pixels, then performance-based success would be more important in the presence of high noise densities than of the others. For example, let

$$[a_{0j}] = \begin{bmatrix} [0,0.01] & [0,0.05] & [0,0.1] & [0.05,0.35] & [0.2,0.45] & [0.25,0.5] & [0.8,0.85] & [0.85,0.9] & [0.9,0.95] \\ [0.9,0.95] & [0.85,0.9] & [0.8,0.85] & [0.25,0.5] & [0.2,0.45] & [0.05,0.35] & [0,0.1] & [0,0.05] & [0,0.01] \end{bmatrix}$$

Table 1 SSIM results of the filters for Lena, cameraman, Barbara, baboon, and peppers images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Lena	**BPDF**	0.9848	0.9657	0.9411	0.9087	0.8689	0.8120	0.7247	0.5683	0.3063
	MDBUTMF	0.9865	0.9479	0.8498	0.8155	0.8655	0.8898	0.8668	0.7830	0.4010
	DBAIN	0.9885	0.9741	0.9555	0.9291	0.8989	0.8560	0.7942	0.7139	0.5979
	NAFSMF	0.9839	0.9669	0.9485	0.9279	0.9080	0.8821	0.8511	0.8040	0.6862
	DAMF	0.9902	0.9789	0.9653	0.9488	0.9310	0.9085	0.8796	0.8396	0.7657
	AWMF	0.9822	0.9740	0.9636	0.9497	0.9349	0.9134	0.8852	0.8447	0.7737
	ARmF	**0.9910**	**0.9810**	**0.9697**	**0.9554**	**0.9398**	**0.9176**	**0.8885**	**0.8471**	**0.7752**
Cameraman	**BPDF**	0.9911	0.9782	0.9608	0.9344	0.8966	0.8453	0.7726	0.6722	0.5105
	MDBUTMF	0.9911	0.9412	0.8160	0.7818	0.8653	0.9174	0.9052	0.8265	0.4667
	DBAIN	0.9948	0.9867	0.9758	0.9586	0.9332	0.8977	0.8452	0.7805	0.6917
	NAFSMF	0.9798	0.9643	0.9500	0.9340	0.9177	0.8988	0.8727	0.8325	0.7207
	DAMF	0.9961	0.9908	0.9844	0.9759	0.9652	0.9512	0.9321	0.9012	0.8347
	AWMF	0.9883	0.9849	0.9813	0.9759	0.9681	0.9563	0.9371	0.9059	0.8401
	ARmF	**0.9970**	**0.9933**	**0.9890**	**0.9828**	**0.9743**	**0.9614**	**0.9416**	**0.9093**	**0.8426**
Barbara	**BPDF**	0.9743	0.9427	0.9046	0.8606	0.8024	0.7289	0.6258	0.4597	0.2316
	MDBUTMF	0.9741	0.9228	0.8235	0.7757	0.7962	0.7914	0.7477	0.6573	0.3884
	DBAIN	0.9769	0.9502	0.9174	0.8762	0.8279	0.7662	0.6880	0.5882	0.4589
	NAFSMF	0.9749	0.9472	0.9174	0.8843	0.8483	0.8039	0.7533	0.6896	0.5729
	DAMF	0.9815	0.9588	0.9327	0.9013	0.8675	0.8261	0.7786	0.7176	0.6308
	AWMF	0.9718	0.9540	0.9331	0.9065	0.8762	0.8366	0.7879	0.7250	0.6382
	ARmF	**0.9841**	**0.9654**	**0.9438**	**0.9172**	**0.8861**	**0.8450**	**0.7949**	**0.7291**	**0.6394**

Table 1 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Baboon	**BPDF**	0.9795	0.9516	0.9112	0.8556	0.7812	0.6841	0.5622	0.4080	0.1377
	MDBUTMF	0.9727	0.9321	0.8655	0.8228	0.8126	0.7869	0.7317	0.6333	0.3625
	DBAIN	0.9844	0.9644	0.9352	0.8933	0.8373	0.7605	0.6587	0.5422	0.4161
	NAFSMF	0.9612	0.9216	0.8767	0.8305	0.7800	0.7211	0.6540	0.5777	0.4671
	DAMF	0.9884	0.9748	0.9572	0.9356	0.9086	0.8738	0.8237	0.7466	0.6037
	AWMF	0.9720	0.9616	0.9487	0.9343	0.9135	0.8824	0.8331	0.7550	0.6108
	ARmF	**0.9915**	**0.9818**	**0.9689**	**0.9523**	**0.9294**	**0.8960**	**0.8442**	**0.7630**	**0.6150**
Peppers	**BPDF**	0.9735	0.9460	0.9158	0.8798	0.8363	0.7780	0.7001	0.5584	0.2194
	MDBUTMF	0.9794	0.9331	0.8263	0.7884	0.8321	0.8484	0.8206	0.7382	0.4131
	DBAIN	0.9742	0.9508	0.9239	0.8909	0.8535	0.8034	0.7387	0.6565	0.5402
	NAFSMF	0.9772	0.9551	0.9328	0.9068	0.8810	0.8512	0.8154	0.7665	0.6470
	DAMF	0.9804	0.9594	0.9372	0.9110	0.8835	0.8515	0.8152	0.7707	0.7018
	AWMF	0.9609	0.9560	0.9410	0.9204	0.8952	0.8633	0.8256	0.7789	0.7096
	ARmF	**0.9826**	**0.9640**	**0.9439**	**0.9205**	**0.8939**	**0.8618**	**0.8241**	**0.7779**	**0.7096**

Bold values indicate the best scores

Table 2 SSIM results of the filters for living room, lake, plane, hill, pirate, boat, and house images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Living Room	**BPDF**	0.9747	0.9432	0.9056	0.8569	0.7962	0.7153	0.6012	0.4372	0.2337
	MDBUTMF	0.9764	0.9338	0.8567	0.8137	0.8251	0.8066	0.7621	0.6682	0.3744
	DBAIN	0.9802	0.9557	0.9251	0.8857	0.8368	0.7693	0.6888	0.5838	0.4565
	NAFSMF	0.9704	0.9382	0.9047	0.8687	0.8301	0.7839	0.7329	0.6678	0.5472
	DAMF	0.9846	0.9654	0.9422	0.9152	0.8824	0.8443	0.7976	0.7325	0.6295
	AWMF	0.9693	0.9539	0.9358	0.9144	0.8879	0.8523	0.8062	0.7394	0.6356
	ARmF	**0.9856**	**0.9699**	**0.9514**	**0.9294**	**0.9018**	**0.8653**	**0.8171**	**0.7483**	**0.6418**
Lake	**BPDF**	0.9795	0.9526	0.9218	0.8796	0.8253	0.7468	0.6464	0.4839	0.2226
	MDBUTMF	0.9802	0.9275	0.8097	0.7749	0.8177	0.8374	0.8066	0.7192	0.4084
	DBAIN	0.9768	0.9565	0.9315	0.8988	0.8561	0.7984	0.7228	0.6267	0.5053
	NAFSMF	0.9754	0.9489	0.9210	0.8925	0.8588	0.8229	0.7805	0.7221	0.6021
	DAMF	0.9856	0.9690	0.9499	0.9285	0.9020	0.8689	0.8293	0.7737	0.6842
	AWMF	0.9742	0.9620	0.9474	0.9297	0.9067	0.8758	0.8361	0.7799	0.6904
	ARmF	**0.9867**	**0.9716**	**0.9553**	**0.9361**	**0.9113**	**0.8793**	**0.8391**	**0.7828**	**0.6926**
Plane	**BPDF**	0.9885	0.9733	0.9533	0.9220	0.8797	0.8194	0.7309	0.5631	0.1894
	MDBUTMF	0.9884	0.9317	0.7907	0.7539	0.8392	0.8978	0.8833	0.7857	0.3518
	DBAIN	0.9885	0.9781	0.9642	0.9423	0.9124	0.8706	0.8139	0.7343	0.6268
	NAFSMF	0.9845	0.9685	0.9524	0.9334	0.9136	0.8892	0.8596	0.8175	0.7019
	DAMF	0.9938	0.9861	0.9769	0.9648	0.9505	0.9331	0.9086	0.8714	0.7987
	AWMF	0.9850	0.9796	0.9733	0.9645	0.9532	0.9376	0.9133	0.8760	0.8055
	ARmF	**0.9947**	**0.9887**	**0.9816**	**0.9719**	**0.9599**	**0.9433**	**0.9182**	**0.8795**	**0.8080**
Hill	**BPDF**	0.9761	0.9480	0.9129	0.8676	0.8062	0.7275	0.6232	0.4954	0.3573
	MDBUTMF	0.9781	0.9340	0.8335	0.7938	0.8193	0.8220	0.7827	0.6976	0.3921
	DBAIN	0.9801	0.9578	0.9287	0.8912	0.8410	0.7784	0.6997	0.6036	0.4833
	NAFSMF	0.9733	0.9451	0.9148	0.8824	0.8463	0.8064	0.7585	0.7010	0.5843

Table 2 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
	DAMF	0.9841	0.9656	0.9438	0.9181	0.8875	0.8515	0.8075	0.7495	0.6571
	AWMF	0.9724	0.9576	0.9409	0.9195	0.8929	0.8593	0.8152	0.7562	0.6632
	ARmF	**0.9860**	**0.9703**	**0.9526**	**0.9310**	**0.9038**	**0.8690**	**0.8240**	**0.7626**	**0.6672**
Pirate	**BPDF**	0.9801	0.9549	0.9232	0.8817	0.8266	0.7506	0.6494	0.4797	0.2741
	MDBUTMF	0.9813	0.9381	0.8418	0.8072	0.8363	0.8430	0.8096	0.7178	0.4185
	DBAIN	0.9832	0.9637	0.9387	0.9062	0.8605	0.8017	0.7286	0.6247	0.5002
	NAFSMF	0.9766	0.9511	0.9248	0.8970	0.8635	0.8251	0.7844	0.7227	0.6093
	DAMF	0.9875	0.9722	0.9542	0.9332	0.9063	0.8744	0.8362	0.7784	0.6853
	AWMF	0.9753	0.9624	0.9489	0.9322	0.9088	0.8790	0.8417	0.7834	0.6913
	ARmF	**0.9884**	**0.9750**	**0.9600**	**0.9424**	**0.9181**	**0.8875**	**0.8487**	**0.7886**	**0.6939**
Boat	**BPDF**	0.9753	0.9456	0.9085	0.8608	0.8010	0.7245	0.6155	0.4697	0.2851
	MDBUTMF	0.9783	0.9353	0.8450	0.8064	0.8268	0.8243	0.7833	0.6906	0.3796
	DBAIN	0.9767	0.9532	0.9239	0.8844	0.8396	0.7785	0.6968	0.5992	0.4825
	NAFSMF	0.9723	0.9422	0.9115	0.8766	0.8414	0.8005	0.7528	0.6898	0.5778
	DAMF	0.9833	0.9634	0.9407	0.9123	0.8829	0.8463	0.8011	0.7419	0.6514
	AWMF	0.9706	0.9555	0.9375	0.9146	0.8887	0.8543	0.8091	0.7483	0.6571
	ARmF	**0.9842**	**0.9664**	**0.9467**	**0.9223**	**0.8957**	**0.8606**	**0.8142**	**0.7529**	**0.6610**
House	**BPDF**	0.9938	0.9858	0.9730	0.9550	0.9241	0.8835	0.8113	0.7002	0.4932
	MDBUTMF	0.9950	0.9491	0.8178	0.7831	0.8833	0.9449	0.9425	0.8641	0.4270
	DBAIN	0.9969	0.9920	0.9832	0.9703	0.9522	0.9238	0.8777	0.8142	0.7234
	NAFSMF	0.9914	0.9831	0.9733	0.9643	0.9535	0.9405	0.9210	0.8918	0.7827
	DAMF	0.9982	0.9955	0.9912	0.9861	0.9796	0.9709	0.9577	0.9376	0.8852
	AWMF	0.9933	0.9924	0.9905	0.9878	0.9834	0.9760	0.9630	0.9426	0.8948
	ARmF	**0.9987**	**0.9970**	**0.9946**	**0.9913**	**0.9863**	**0.9786**	**0.9652**	**0.9446**	**0.8962**

Bold values indicate the best scores

Table 3 SSIM results of the filters for bridge, Elaine, flintstones, flower, parrot, dark-haired woman, and blonde woman images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Bridge	**BPDF**	0.9705	0.9335	0.8856	0.8269	0.7503	0.6452	0.5159	0.3648	0.1815
	MDBUTMF	0.9699	0.9236	0.8433	0.7994	0.7855	0.7572	0.6950	0.6000	0.3651
	DBAIN	0.9728	0.9424	0.9047	0.8552	0.7917	0.7104	0.6060	0.4880	0.3518
	NAFSMF	0.9631	0.9222	0.8788	0.8337	0.7818	0.7237	0.6544	0.5766	0.4578
	DAMF	0.9798	0.9560	0.9276	0.8953	0.8563	0.8072	0.7465	0.6667	0.5415
	AWMF	0.9638	0.9440	0.9209	0.8948	0.8611	0.8148	0.7551	0.6736	0.5469
	ARmF	**0.9823**	**0.9621**	**0.9385**	**0.9113**	**0.8762**	**0.8285**	**0.7663**	**0.6819**	**0.5515**
Elaine	**BPDF**	0.9707	0.9405	0.9052	0.8649	0.8149	0.7517	0.6628	0.4927	0.2911
	MDBUTMF	0.9774	0.9324	0.8347	0.7965	0.8224	0.8295	0.7925	0.6973	0.3492
	DBAIN	0.9746	0.9483	0.9173	0.8800	0.8358	0.7832	0.7157	0.6292	0.5121
	NAFSMF	0.9774	0.9542	0.9295	0.9025	0.8730	0.8404	0.8010	0.7470	0.6310
	DAMF	0.9774	0.9534	0.9270	0.8961	0.8620	0.8230	0.7784	0.7248	0.6584
	AWMF	0.9684	0.9514	0.9296	0.9021	0.8696	0.8313	0.7857	0.7310	0.6640
	ARmF	**0.9773**	**0.9532**	**0.9272**	**0.8971**	**0.8630**	**0.8239**	**0.7791**	**0.7270**	**0.6631**
Flintstones	**BPDF**	0.9726	0.9417	0.9021	0.8550	0.7912	0.7099	0.5908	0.4125	0.1259
	MDBUTMF	0.9764	0.9304	0.8315	0.7932	0.8169	0.8128	0.7671	0.6735	0.3965
	DBAIN	0.9769	0.9533	0.9210	0.8793	0.8239	0.7487	0.6490	0.5308	0.3807
	NAFSMF	0.9659	0.9333	0.8983	0.8631	0.8220	0.7743	0.7165	0.6464	0.5215
	DAMF	0.9840	0.9658	0.9430	0.9173	0.8865	0.8464	0.7980	0.7268	0.6061
	AWMF	0.9551	0.9502	0.9364	0.9167	0.8908	0.8541	0.8058	0.7334	0.6118
	ARmF	**0.9847**	**0.9688**	**0.9491**	**0.9267**	**0.8987**	**0.8608**	**0.8112**	**0.7381**	**0.6154**
Flower	**BPDF**	0.9808	0.9618	0.9346	0.8998	0.8446	0.7718	0.6634	0.4970	0.2249
	MDBUTMF	0.9820	0.9486	0.8681	0.8407	0.8732	0.8832	0.8523	0.7679	0.4292
	DBAIN	0.9854	0.9722	0.9517	0.9259	0.8841	0.8330	0.7579	0.6588	0.5230

Table 3 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
	NAFSMF	0.9763	0.9568	0.9363	0.9143	0.8883	0.8600	0.8218	0.7682	0.6492
	DAMF	0.9878	0.9786	0.9662	0.9513	0.9321	0.9089	0.8772	0.8290	0.7404
	AWMF	0.9752	0.9684	0.9594	0.9488	0.9333	0.9126	0.8820	0.8340	0.7459
	ARmF	**0.9877**	**0.9796**	**0.9696**	**0.9577**	**0.9411**	**0.9195**	**0.8876**	**0.8384**	**0.7489**
Parrot	**BPDF**	0.9791	0.9663	0.9490	0.9270	0.8992	0.8580	0.7955	0.6816	0.3541
	MDBUTMF	0.9771	0.9334	0.8242	0.7958	0.8655	0.9042	0.8911	0.8123	0.4008
	DBAIN	0.9840	0.9741	0.9607	0.9440	0.9209	0.8900	0.8467	0.7871	0.6951
	NAFSMF	0.9785	0.9653	0.9519	0.9380	0.9209	0.9030	0.8774	0.8418	0.7331
	DAMF	0.9839	0.9763	0.9666	0.9563	0.9423	0.9270	0.9064	0.8775	0.8226
	AWMF	0.9779	0.9727	0.9655	0.9572	0.9457	0.9316	0.9112	0.8828	0.8309
	ARmF	**0.9851**	**0.9786**	**0.9706**	**0.9621**	**0.9499**	**0.9351**	**0.9141**	**0.8848**	**0.8320**
Dark-Haired Woman	**BPDF**	0.9909	0.9802	0.9665	0.9471	0.9200	0.8789	0.8100	0.6828	0.4483
	MDBUTMF	0.9923	0.9395	0.7833	0.7576	0.8620	0.9294	0.9272	0.8566	0.4772
	DBAIN	0.9925	0.9850	0.9754	0.9614	0.9414	0.9133	0.8715	0.8065	0.7056
	NAFSMF	0.9906	0.9815	0.9723	0.9622	0.9513	0.9361	0.9192	0.8891	0.7756
	DAMF	0.9950	0.9891	0.9826	0.9743	0.9647	0.9525	0.9362	0.9134	0.8664
	AWMF	0.9910	0.9870	0.9823	0.9761	0.9678	0.9565	0.9404	0.9177	0.8744
	ARmF	**0.9956**	**0.9909**	**0.9854**	**0.9787**	**0.9701**	**0.9585**	**0.9420**	**0.9189**	**0.8753**
Blonde Woman	**BPDF**	0.9657	0.9385	0.9055	0.8664	0.8191	0.7561	0.6624	0.5003	0.2184
	MDBUTMF	0.9642	0.9236	0.8294	0.7952	0.8214	0.8258	0.7936	0.7017	0.3539
	DBAIN	0.9666	0.9449	0.9184	0.8856	0.8441	0.7938	0.7259	0.6470	0.5432
	NAFSMF	0.9606	0.9366	0.9104	0.8833	0.8526	0.8184	0.7805	0.7259	0.6113
	DAMF	0.9700	0.9518	0.9301	0.9053	0.8764	0.8424	0.8015	0.7505	0.6753
	AWMF	0.9579	0.9450	0.9273	0.9061	0.8802	0.8476	0.8069	0.7554	0.6814
	ARmF	**0.9718**	**0.9551**	**0.9355**	**0.9132**	**0.8864**	**0.8531**	**0.8114**	**0.7582**	**0.6825**

Bold values indicate the best scores

Table 4 SSIM results of the filters for Einstein image

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Einstein	**BPDF**	0.9830	0.9614	0.9361	0.9051	0.8640	0.8085	0.7315	0.5892	0.3465
	MDBUTMF	0.9833	0.9418	0.8476	0.8127	0.8528	0.8677	0.8393	0.7561	0.4127
	DBAIN	0.9867	0.9706	0.9500	0.9236	0.8881	0.8449	0.7839	0.7102	0.6142
	NAFSMF	0.9801	0.9591	0.9364	0.9132	0.8878	0.8591	0.8231	0.7732	0.6698
	DAMF	0.9894	0.9765	0.9619	0.9445	0.9244	0.8989	0.8666	0.8208	0.7472
	AWMF	0.9798	0.9701	0.9588	0.9450	0.9280	0.9043	0.8724	0.8259	0.7531
	ARmF	**0.9911**	**0.9805**	**0.9687**	**0.9543**	**0.9367**	**0.9121**	**0.8788**	**0.8305**	**0.7551**

Bold values indicate the best scores

Thus, the d-matrix $[a_{ij}]$, modelling the SSIM values provided in Tables 1, 2, 3, and 4, is as follows:

$$[a_{ij}] = \begin{bmatrix}
\substack{[0,0.01]\\{[0.9,0.95]}} & \substack{[0,0.05]\\{[0.85,0.9]}} & \substack{[0,0.1]\\{[0.8,0.85]}} & \substack{[0.05,0.35]\\{[0.25,0.5]}} & \substack{[0.2,0.45]\\{[0.2,0.45]}} & \substack{[0.25,0.5]\\{[0.05,0.35]}} & \substack{[0.8,0.85]\\{[0,0.1]}} & \substack{[0.85,0.9]\\{[0,0.05]}} & \substack{[0.9,0.95]\\{[0,0.01]}} \\
\substack{[0.9392,0.9666]\\{[0.0060,0.0334]}} & \substack{[0.8872,0.9368]\\{[0.0135,0.0632]}} & \substack{[0.8145,0.8948]\\{[0.0248,0.1052]}} & \substack{[0.7330,0.8465]\\{[0.0399,0.1535]}} & \substack{[0.6392,0.7873]\\{[0.0646,0.2127]}} & \substack{[0.5210,0.7135]\\{[0.0940,0.2865]}} & \substack{[0.3982,0.6263]\\{[0.1456,0.3737]}} & \substack{[0.2732,0.5243]\\{[0.2245,0.4757]}} & \substack{[0.0909,0.3687]\\{[0.3535,0.6313]}} \\
\substack{[0.9355,0.9653]\\{[0.0049,0.0347]}} & \substack{[0.8991,0.9248]\\{[0.0496,0.0752]}} & \substack{[0.7221,0.8002]\\{[0.1216,0.1998]}} & \substack{[0.6937,0.7736]\\{[0.1465,0.2264]}} & \substack{[0.7155,0.8046]\\{[0.1063,0.1954]}} & \substack{[0.6376,0.7956]\\{[0.0464,0.2044]}} & \substack{[0.5572,0.7555]\\{[0.0461,0.2445]}} & \substack{[0.4747,0.6836]\\{[0.1075,0.3164]}} & \substack{[0.3096,0.4230]\\{[0.4635,0.5770]}} \\
\substack{[0.9383,0.9676]\\{[0.0030,0.0324]}} & \substack{[0.8978,0.9451]\\{[0.0076,0.0549]}} & \substack{[0.8388,0.9116]\\{[0.0156,0.0884]}} & \substack{[0.7669,0.8702]\\{[0.0266,0.1298]}} & \substack{[0.6822,0.8205]\\{[0.0412,0.1795]}} & \substack{[0.5855,0.7614]\\{[0.0628,0.2386]}} & \substack{[0.4766,0.6902]\\{[0.0962,0.3098]}} & \substack{[0.3680,0.6139]\\{[0.1401,0.3861]}} & \substack{[0.2565,0.5274]\\{[0.2017,0.4726]}} \\
\substack{[0.9319,0.9618]\\{[0.0084,0.0382]}} & \substack{[0.8682,0.9261]\\{[0.0159,0.0739]}} & \substack{[0.7994,0.8875]\\{[0.0243,0.1125]}} & \substack{[0.7325,0.8505]\\{[0.0314,0.1495]}} & \substack{[0.6648,0.8125]\\{[0.0397,0.1875]}} & \substack{[0.5913,0.7713]\\{[0.0488,0.2287]}} & \substack{[0.5162,0.7269]\\{[0.0623,0.2731]}} & \substack{[0.4384,0.6781]\\{[0.0823,0.3219]}} & \substack{[0.3455,0.5908]\\{[0.1640,0.4092]}} \\
\substack{[0.9433,0.9708]\\{[0.0018,0.0292]}} & \substack{[0.9119,0.9538]\\{[0.0043,0.0462]}} & \substack{[0.8710,0.9314]\\{[0.0082,0.0686]}} & \substack{[0.8207,0.9040]\\{[0.0127,0.0960]}} & \substack{[0.7623,0.8721]\\{[0.0182,0.1279]}} & \substack{[0.6936,0.8343]\\{[0.0250,0.1657]}} & \substack{[0.6163,0.7907]\\{[0.0349,0.2093]}} & \substack{[0.5247,0.7378]\\{[0.0491,0.2622]}} & \substack{[0.4030,0.6588]\\{[0.0854,0.3412]}} \\
\substack{[0.9200,0.9568]\\{[0.0065,0.0432]}} & \substack{[0.9003,0.9465]\\{[0.0073,0.0535]}} & \substack{[0.8610,0.9261]\\{[0.0089,0.0739]}} & \substack{[0.8187,0.9038]\\{[0.0112,0.0962]}} & \substack{[0.7673,0.8762]\\{[0.0148,0.1238]}} & \substack{[0.7018,0.8405]\\{[0.0207,0.1595]}} & \substack{[0.6252,0.7973]\\{[0.0307,0.2027]}} & \substack{[0.5308,0.7428]\\{[0.0452,0.2572]}} & \substack{[0.4058,0.6638]\\{[0.0781,0.3362]}} \\
\substack{[0.9463,0.9725]\\{[0.0013,0.0275]}} & \substack{[0.9131,0.9551]\\{[0.0028,0.0449]}} & \substack{[0.8687,0.9318]\\{[0.0051,0.0682]}} & \substack{[0.8199,0.9060]\\{[0.0079,0.0940]}} & \substack{[0.7682,0.8780]\\{[0.0122,0.1220]}} & \substack{[0.7135,0.8475]\\{[0.0186,0.1525]}} & \substack{[0.6392,0.8051]\\{[0.0290,0.1949]}} & \substack{[0.5401,0.7481]\\{[0.0439,0.2519]}} & \substack{[0.4101,0.6665]\\{[0.0772,0.3335]}}
\end{bmatrix}$$

Second, we apply the configured method to $[a_{ij}]$. Moreover, we obtain the results herein by MATLAB R2021a.

Step 2. The column matrix $\begin{bmatrix}\alpha_{i1}\\ \beta_{i1}\end{bmatrix}$ is as follows:

$$\begin{bmatrix}\alpha_{i1}\\ \beta_{i1}\end{bmatrix} = \begin{bmatrix}
\substack{[0.2061,0.5573]\\{[0.0143,0.1151]}} & \substack{[0.3256,0.6280]\\{[0.0454,0.1309]}} & \substack{[0.2769,0.6317]\\{[0.0088,0.0977]}} & \substack{[0.3142,0.6629]\\{[0.0131,0.1078]}} & \substack{[0.3708,0.7197]\\{[0.0044,0.0730]}} & \substack{[0.3747,0.7238]\\{[0.0058,0.0774]}} & \substack{[0.3805,0.7283]\\{[0.0029,0.0700]}}
\end{bmatrix}^T$$

To exemplify, α_{11} and β_{11} are calculated as follows:

$$\begin{aligned}
\alpha_{11} &= \tfrac{1}{\lambda}\textstyle\sum_{j=1}^{9}\alpha_{0j}^{a}\alpha_{1j}^{a} \\
&= \tfrac{1}{4.5}\left(\alpha_{01}^{a}\alpha_{11}^{a}+\alpha_{02}^{a}\alpha_{12}^{a}+\alpha_{03}^{a}\alpha_{13}^{a}+\alpha_{04}^{a}\alpha_{14}^{a}+\alpha_{05}^{a}\alpha_{15}^{a}+\alpha_{06}^{a}\alpha_{16}^{a}+\alpha_{07}^{a}\alpha_{17}^{a}+\alpha_{08}^{a}\alpha_{18}^{a}+\alpha_{09}^{a}\alpha_{19}^{a}\right) \\
&= \tfrac{1}{4.5}\left([0, 0.01]\cdot[0.9392, 0.9666]+[0, 0.05]\cdot[0.8872, 0.9368]\right. \\
&\quad +[0, 0.1]\cdot[0.8145, 0.8948]+[0.05, 0.35]\cdot[0.7330, 0.8465]+[0.2, 0.45]\cdot[0.6392, 0.7873] \\
&\quad +[0.25, 0.5]\cdot[0.5210, 0.7135]+[0.8, 0.85]\cdot[0.3982, 0.6263]+[0.85, 0.9]\cdot[0.2732, 0.5243] \\
&\quad \left.+[0.9, 0.95]\cdot[0.0909, 0.3687]\right) \\
&= [0.2061, 0.5573]
\end{aligned}$$

and

$$\begin{aligned}\beta_{11} &= \tfrac{1}{\lambda}\sum_{j=1}^{9}\beta_{0j}^{a}\beta_{1j}^{a}\\ &= \tfrac{1}{4.5}\left(\beta_{01}^{a}\beta_{11}^{a}+\beta_{02}^{a}\beta_{12}^{a}+\beta_{03}^{a}\beta_{13}^{a}+\beta_{04}^{a}\beta_{14}^{a}+\beta_{05}^{a}\beta_{15}^{a}+\beta_{06}^{a}\beta_{16}^{a}+\beta_{07}^{a}\beta_{17}^{a}+\beta_{08}^{a}\beta_{18}^{a}+\beta_{09}^{a}\beta_{19}^{a}\right)\\ &= \tfrac{1}{4.5}\left([0.9,0.95]\cdot[0.0060,0.0334]+[0.85,0.9]\cdot[0.0135,0.0632]+[0.8,0.85]\cdot[0.0248,0.1052]\right.\\ &\quad +[0.25,0.5]\cdot[0.0399,0.1535]+[0.2,0.45]\cdot[0.0646,0.2127]+[0.05,0.35]\cdot[0.0940,0.2865]\\ &\quad \left.+[0,0.1]\cdot[0.1456,0.3737]+[0,0.05]\cdot[0.2245,0.4757]+[0,0.01]\cdot[0.3535,0.6313]\right)\\ &= [0.0143,0.1151]\end{aligned}$$

such that

$$\begin{aligned}\lambda &= \tfrac{1}{2}\sum_{j=1}^{9}\left(1+\tfrac{(\alpha_{0j}^{a})^{-}+(\alpha_{0j}^{a})^{+}}{2}-\tfrac{(\beta_{0j}^{a})^{-}+(\beta_{0j}^{a})^{+}}{2}\right)\\ &= \tfrac{1}{2}\left(\left(1+\tfrac{(\alpha_{01}^{a})^{-}+(\alpha_{01}^{a})^{+}}{2}-\tfrac{(\beta_{01}^{a})^{-}+(\beta_{01}^{a})^{+}}{2}\right)+\left(1+\tfrac{(\alpha_{02}^{a})^{-}+(\alpha_{02}^{a})^{+}}{2}-\tfrac{(\beta_{02}^{a})^{-}+(\beta_{02}^{a})^{+}}{2}\right)\right.\\ &\quad +\left(1+\tfrac{(\alpha_{03}^{a})^{-}+(\alpha_{03}^{a})^{+}}{2}-\tfrac{(\beta_{03}^{a})^{-}+(\beta_{03}^{a})^{+}}{2}\right)+\left(1+\tfrac{(\alpha_{04}^{a})^{-}+(\alpha_{04}^{a})^{+}}{2}-\tfrac{(\beta_{04}^{a})^{-}+(\beta_{04}^{a})^{+}}{2}\right)\\ &\quad +\left(1+\tfrac{(\alpha_{05}^{a})^{-}+(\alpha_{05}^{a})^{+}}{2}-\tfrac{(\beta_{05}^{a})^{-}+(\beta_{05}^{a})^{+}}{2}\right)+\left(1+\tfrac{(\alpha_{06}^{a})^{-}+(\alpha_{06}^{a})^{+}}{2}-\tfrac{(\beta_{06}^{a})^{-}+(\beta_{06}^{a})^{+}}{2}\right)\\ &\quad +\left(1+\tfrac{(\alpha_{07}^{a})^{-}+(\alpha_{07}^{a})^{+}}{2}-\tfrac{(\beta_{07}^{a})^{-}+(\beta_{07}^{a})^{+}}{2}\right)+\left(1+\tfrac{(\alpha_{08}^{a})^{-}+(\alpha_{08}^{a})^{+}}{2}-\tfrac{(\beta_{08}^{a})^{-}+(\beta_{08}^{a})^{+}}{2}\right)\\ &\quad \left.+\left(1+\tfrac{(\alpha_{09}^{a})^{-}+(\alpha_{09}^{a})^{+}}{2}-\tfrac{(\beta_{09}^{a})^{-}+(\beta_{09}^{a})^{+}}{2}\right)\right)\\ &= \tfrac{1}{2}\left[\left(1+\tfrac{0+0.01}{2}-\tfrac{0.9+0.95}{2}\right)+\left(1+\tfrac{0+0.05}{2}-\tfrac{0.85+0.9}{2}\right)+\left(1+\tfrac{0+0.1}{2}-\tfrac{0.8+0.85}{2}\right)\right.\\ &\quad +\left(1+\tfrac{0.05+0.35}{2}-\tfrac{0.25+0.5}{2}\right)+\left(1+\tfrac{0.2+0.45}{2}-\tfrac{0.2+0.45}{2}\right)+\left(1+\tfrac{0.25+0.5}{2}-\tfrac{0.05+0.35}{2}\right)\\ &\quad \left.+\left(1+\tfrac{0.8+0.85}{2}-\tfrac{0+0.1}{2}\right)+\left(1+\tfrac{0.85+0.9}{2}-\tfrac{0+0.05}{2}\right)+\left(1+\tfrac{0.9+0.95}{2}-\tfrac{0+0.01}{2}\right)\right]\\ &= 4.5\end{aligned}$$

Step 3. The score matrix is as follows:

$$\begin{aligned}[s_{i1}] = [&[0.0909,0.5430]\ \ [0.1946,0.5826]\ \ [0.1792,0.6229]\ \ [0.2064,0.6498]\\ &[0.2977,0.7152]\ \ [0.2974,0.7181]\ \ [0.3105,0.7254]]^{T}\end{aligned}$$

Here,

$$s_{11}=\alpha_{11}-\beta_{11}=[0.2061,0.5573]-[0.0143,0.1151]=[0.0909,0.5430]$$

Step 4. The decision set is as follows:

$$\begin{aligned}\{&{}^{[0.2228,0.7765]}\text{BPDF},{}^{[0.3498,0.8251]}\text{ MDBUTMF},{}^{[0.3309,0.8744]}\text{ DBAIN},\\ &{}^{[0.3642,0.9074]}\text{NAFSMF},\ {}^{[0.4761,0.9875]}\text{DAMF},{}^{[0.4757,0.9910]}\text{ AWMF},{}^{[0.4917,1]}\text{ ARmF}\}\end{aligned}$$

Here,

$$\begin{aligned}d(u_1) &= \left[\frac{s_{11}^{-}+|\min_i s_{i1}^{-}|}{\max_i s_{i1}^{+}+|\min_i s_{i1}^{-}|},\frac{s_{11}^{+}+|\min_i s_{i1}^{-}|}{\max_i s_{i1}^{+}+|\min_i s_{i1}^{-}|}\right]\\ &= \left[\frac{0.0909+|0.0909|}{0.7254+|0.0909|},\frac{0.5430+|0.0909|}{0.7254+|0.0909|}\right]\\ &= [0.2228,0.7765]\end{aligned}$$

Step 5. The ranking order

$$\text{BPDF}\prec\text{MDBUTMF}\prec\text{DBAIN}\prec\text{NAFSMF}\prec\text{DAMF}\prec\text{AWMF}\prec\text{ARmF}$$

is valid. Therefore, the performance ranking of the filters shows that ARmF outperforms the other filters.

Thirdly, we consider 40 test images in the TESTIMAGES database (Asuni and Giachetti 2014), i.e. “Almonds”, “Apples”, “Balloons”, “Bananas”, “Billiard Balls 1”, “Billiard Balls 2”, “Building”, “Cards 1”, “Cards 2”, “Carrots”, “Chairs”, “Clips”, “Coins”, “Cushions”, “Duck”, “Fence”, “Flowers”, “Garden Table”, “Guitar Bridge”, “Guitar Fret”, “Guitar Head”, “Keyboard 1”, “Keyboard 2”, “Lion”, “Multimeter”, “Pencils 1”, “Pencils 2”, “Pillar”, “Plastic”, “Roof”, “Scarf”, “Screws”, “Snails”, “Socks”, “Sweets”, “Tomatoes 1”, “Tomatoes 2”, “Tools 1”, “Tools 2”, and “Wood Game”. To this end, we present the results of the aforesaid filters by SSIM for the images at noise densities ranging from 10% to 90%, in Tables 5, 6, 7, 8, 9, 10, and 11, respectively. Moreover, we obtain the results herein by MATLAB R2021a.

For the problem, let (μ_t^{ij}) be ordered-quadragintuple such that μ_t^{ij} corresponds to the SSIM results in Tables 5, 6, 7, 8, 9, 10, and 11, obtained by t^{th} image for i^{th} filter at j^{th} noise density. Here, since $\nu_t^{ij} = 1 - \mu_t^{ij}$ and $\pi_t^{ij} = 0$ such that $i \in I_7$, $j \in I_9$, and $t \in I_{40}$, then for d-matrix $[b_{ij}]$,

$$\alpha_{ij}^b := \left[\frac{\min\limits_t \mu_t^{ij}}{\max\limits_t \mu_t^{ij} + \max\limits_t \{1 - \mu_t^{ij}\}}, \frac{\max\limits_t \mu_t^{ij}}{\max\limits_t \mu_t^{ij} + \max\limits_t \{1 - \mu_t^{ij}\}} \right]$$

and

$$\beta_{ij}^b := \left[\frac{\min\limits_t \{1 - \mu_t^{ij}\}}{\max\limits_t \mu_t^{ij} + \max\limits_t \{1 - \mu_t^{ij}\}}, \frac{\max\limits_t \{1 - \mu_t^{ij}\}}{\max\limits_t \mu_t^{ij} + \max\limits_t \{1 - \mu_t^{ij}\}} \right]$$

For example, the ordered-quadragintuple

$$(\mu_t^{11}) = (0.9815, 0.9931, 0.9935, 0.9873, 0.9953, 0.9901, 0.9821, 0.9814, 0.9894, 0.9866, 0.9970, 0.9869, 0.9782, 0.9937, 0.9956, 0.9840, 0.9841, 0.9751, 0.9788, 0.9874, 0.9776, 0.9845, 0.9782, 0.9900, 0.9760, 0.9824, 0.9822, 0.9861, 0.9735, 0.9884, 0.9816, 0.9832, 0.9913, 0.9688, 0.9895, 0.9924, 0.9951, 0.9824, 0.9844, 0.9915)$$

indicates SSIM results of BPDF for 40 test images at noise density 10%. Since

$$\begin{aligned} \alpha_{11}^b &= \left[\frac{\min\limits_t \mu_t^{11}}{\max\limits_t \mu_t^{11} + \max\limits_t \{1 - \mu_t^{11}\}}, \frac{\max\limits_t \mu_t^{11}}{\max\limits_t \mu_t^{11} + \max\limits_t \{1 - \mu_t^{11}\}} \right] \\ &= \left[\frac{0.9688}{0.9970 + 0.0312}, \frac{0.9970}{0.9970 + 0.0312} \right] = [0.9422, 0.9696] \end{aligned}$$

and

$$\begin{aligned} \beta_{11}^b &= \left[\frac{\min\limits_t \{1 - \mu_t^{11}\}}{\max\limits_t \mu_t^{11} + \max\limits_t \{1 - \mu_t^{11}\}}, \frac{\max\limits_t \{1 - \mu_t^{11}\}}{\max\limits_t \mu_t^{11} + \max\limits_t \{1 - \mu_t^{11}\}} \right] \\ &= \left[\frac{0.003}{0.9970 + 0.0312}, \frac{0.0312}{0.9970 + 0.0312} \right] = [0.0029, 0.0304] \end{aligned}$$

then $b_{11} = \frac{[0.9422, 0.9696]}{[0.0029, 0.0304]}$. Here, $[0.9422, 0.9696]$ denotes that the success of BPDF on image denoising (i.e. correcting corrupted pixels) at noise density 10% occurs approximately between 94% and 96%. Moreover, $[0.0029, 0.0304]$ means that the rate of BPDF's failure in image denoising at the same noise density ranges from approximately 0% to 3%. Similarly, the all rows of the d-matrix $[b_{ij}]$ but the zero-indexed row can be obtained. Besides, suppose

Table 5 SSIM results of the filters for almonds, apples, balloons, and bananas images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Almonds	**BPDF**	0.9815	0.9565	0.9220	0.8734	0.8094	0.7182	0.5918	0.4091	0.1692
	MDBUTMF	0.9695	0.9275	0.8414	0.8066	0.8317	0.8327	0.7933	0.7099	0.4522
	DBAIN	0.9866	0.9693	0.9445	0.9095	0.8597	0.7956	0.7000	0.5838	0.4357
	NAFSMF	0.9726	0.9444	0.9143	0.8836	0.8486	0.8106	0.7620	0.6975	0.5728
	DAMF	0.9879	0.9751	0.9591	0.9396	0.9154	0.8849	0.8421	0.7800	0.6606
	AWMF	0.9756	0.9657	0.9543	0.9402	0.9216	0.8947	0.8536	0.7923	0.6785
	ARmF	**0.9908**	**0.9809**	**0.9684**	**0.9529**	**0.9329**	**0.9047**	**0.8619**	**0.7985**	**0.6818**
Apples	**BPDF**	0.9931	0.9836	0.9693	0.9492	0.9185	0.8710	0.7892	0.6355	0.3176
	MDBUTMF	0.9861	0.9234	0.7633	0.7325	0.8508	0.9245	0.9204	0.8269	0.3825
	DBAIN	0.9958	0.9898	0.9811	0.9687	0.9497	0.9204	0.8780	0.8158	0.7084
	NAFSMF	0.9867	0.9765	0.9673	0.9586	0.9489	0.9359	0.9182	0.8838	0.7731
	DAMF	0.9968	0.9927	0.9876	0.9815	0.9736	0.9630	0.9489	0.9254	0.8722
	AWMF	0.9924	0.9896	0.9863	0.9819	0.9758	0.9664	0.9525	0.9302	0.8837
	ARmF	**0.9973**	**0.9942**	**0.9905**	**0.9858**	**0.9791**	**0.9692**	**0.9549**	**0.9321**	**0.8848**
Balloons	**BPDF**	0.9935	0.9835	0.9666	0.9425	0.9019	0.8361	0.7327	0.5323	0.1423
	MDBUTMF	0.9958	0.9561	0.8386	0.8148	0.8931	0.9454	0.9353	0.8493	0.4394
	DBAIN	0.9962	0.9906	0.9805	0.9663	0.9419	0.9036	0.8415	0.7437	0.5973
	NAFSMF	0.9905	0.9822	0.9737	0.9654	0.9545	0.9412	0.9192	0.8828	0.7618
	DAMF	0.9981	0.9950	0.9899	0.9838	0.9752	0.9639	0.9475	0.9197	0.8547
	AWMF	0.9921	0.9906	0.9881	0.9845	0.9781	0.9685	0.9526	0.9250	0.8626
	ARmF	**0.9982**	**0.9959**	**0.9927**	**0.9886**	**0.9816**	**0.9715**	**0.9553**	**0.9274**	**0.8645**

Table 5 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Bananas	**BPDF**	0.9873	0.9726	0.9557	0.9320	0.9039	0.8535	0.7943	0.6622	0.3133
	MDBUTMF	0.9857	0.9160	0.7270	0.6892	0.8178	0.9111	0.9088	0.8294	0.4196
	DBAIN	0.9880	0.9769	0.9624	0.9470	0.9258	0.8958	0.8568	0.7971	0.7084
	NAFSMF	0.9825	0.9688	0.9556	0.9431	0.9299	0.9158	0.8953	0.8654	0.7579
	DAMF	0.9854	0.9739	0.9619	0.9488	0.9338	0.9166	0.8967	0.8738	0.8363
	AWMF	0.9843	0.9781	0.9690	0.9579	0.9445	0.9282	0.9085	0.8849	0.8506
	ARmF	**0.9910**	**0.9821**	**0.9725**	**0.9614**	**0.9481**	**0.9316**	**0.9114**	**0.8870**	**0.8517**

Bold values indicate the best scores

Table 6 SSIM results of the filters for billiard balls 1, billiard balls 2, building, cards 1, cards 2, carrots, and chairs images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Billiard Balls 1	**BPDF**	0.9953	0.9888	0.9772	0.9610	0.9371	0.8949	0.8352	0.7112	0.3091
	MDBUTMF	0.9915	0.9344	0.7737	0.7401	0.8625	0.9434	0.9460	0.8702	0.4693
	DBAIN	0.9967	0.9927	0.9860	0.9759	0.9603	0.9332	0.8920	0.8256	0.7076
	NAFSMF	0.9886	0.9816	0.9758	0.9692	0.9624	0.9524	0.9385	0.9074	0.7891
	DAMF	0.9969	0.9942	0.9905	0.9856	0.9797	0.9716	0.9600	0.9403	0.8897
	AWMF	0.9929	0.9921	0.9904	0.9872	0.9828	0.9757	0.9645	0.9452	0.9008
	ARmF	**0.9981**	**0.9962**	**0.9938**	**0.9901**	**0.9854**	**0.9779**	**0.9665**	**0.9469**	**0.9021**
Billiard Balls 2	**BPDF**	0.9901	0.9760	0.9579	0.9332	0.8955	0.8441	0.7720	0.6300	0.4104
	MDBUTMF	0.9881	0.9367	0.8030	0.7758	0.8606	0.9127	0.9048	0.8360	0.4991
	DBAIN	0.9926	0.9833	0.9713	0.9534	0.9280	0.8912	0.8395	0.7597	0.6365
	NAFSMF	0.9841	0.9707	0.9582	0.9456	0.9309	0.9140	0.8911	0.8550	0.7407
	DAMF	0.9938	0.9869	0.9792	0.9696	0.9574	0.9424	0.9228	0.8914	0.8290
	AWMF	0.9861	0.9826	0.9779	0.9710	0.9613	0.9479	0.9287	0.8975	0.8382
	ARmF	**0.9952**	**0.9903**	**0.9844**	**0.9770**	**0.9666**	**0.9526**	**0.9327**	**0.9005**	**0.8401**
Building	**BPDF**	0.9821	0.9653	0.9343	0.8978	0.8437	0.7647	0.6525	0.4683	0.2105
	MDBUTMF	0.9720	0.9262	0.8090	0.7786	0.8392	0.8696	0.8481	0.7770	0.4812
	DBAIN	0.9898	0.9770	0.9597	0.9323	0.8968	0.8406	0.7693	0.6602	0.5044
	NAFSMF	0.9779	0.9583	0.9374	0.9175	0.8943	0.8680	0.8352	0.7835	0.6595
	DAMF	0.9870	0.9775	0.9651	0.9508	0.9334	0.9116	0.8814	0.8380	0.7525
	AWMF	0.9785	0.9733	0.9658	0.9569	0.9439	0.9239	0.8953	0.8508	0.7668
	ARmF	**0.9922**	**0.9860**	**0.9775**	**0.9674**	**0.9535**	**0.9327**	**0.9027**	**0.8568**	**0.7705**
Cards 1	**BPDF**	0.9814	0.9533	0.9169	0.8682	0.8009	0.7122	0.5946	0.4137	0.1308
	MDBUTMF	0.9755	0.9232	0.8063	0.7603	0.8059	0.8219	0.7821	0.6873	0.3697
	DBAIN	0.9837	0.9645	0.9367	0.9006	0.8492	0.7793	0.6883	0.5739	0.4320
	NAFSMF	0.9719	0.9450	0.9151	0.8833	0.8475	0.8042	0.7542	0.6902	0.5644

Table 6 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
	DAMF	0.9884	0.9742	0.9556	0.9342	0.9076	0.8727	0.8274	0.7634	0.6506
	AWMF	0.9729	0.9613	0.9465	0.9310	0.9081	0.8767	0.8330	0.7687	0.6576
	ARmF	**0.9890**	**0.9771**	**0.9618**	**0.9448**	**0.9207**	**0.8876**	**0.8420**	**0.7756**	**0.6608**
Cards 2	**BPDF**	0.9894	0.9743	0.9514	0.9200	0.8769	0.8163	0.7329	0.5823	0.1861
	MDBUTMF	0.9869	0.9108	0.7237	0.6831	0.8081	0.8924	0.8794	0.7829	0.3547
	DBAIN	0.9908	0.9799	0.9638	0.9413	0.9114	0.8682	0.8109	0.7327	0.6220
	NAFSMF	0.9763	0.9603	0.9452	0.9312	0.9151	0.8946	0.8656	0.8251	0.7005
	DAMF	0.9931	0.9856	0.9751	0.9634	0.9486	0.9284	0.9017	0.8641	0.7891
	AWMF	0.9826	0.9763	0.9687	0.9596	0.9475	0.9293	0.9039	0.8659	0.7952
	ARmF	**0.9930**	**0.9861**	**0.9774**	**0.9674**	**0.9543**	**0.9352**	**0.9087**	**0.8697**	**0.7978**
Carrots	**BPDF**	0.9866	0.9674	0.9435	0.9105	0.8632	0.7871	0.6817	0.4619	0.0861
	MDBUTMF	0.9870	0.9321	0.7968	0.7604	0.8451	0.8939	0.8763	0.7746	0.3649
	DBAIN	0.9905	0.9779	0.9617	0.9390	0.9047	0.8563	0.7905	0.6857	0.5468
	NAFSMF	0.9839	0.9666	0.9496	0.9321	0.9108	0.8863	0.8580	0.8120	0.6964
	DAMF	0.9929	0.9842	0.9740	0.9620	0.9460	0.9260	0.9001	0.8601	0.7808
	AWMF	0.9843	0.9783	0.9713	0.9632	0.9503	0.9327	0.9070	0.8673	0.7903
	ARmF	**0.9941**	**0.9874**	**0.9799**	**0.9707**	**0.9574**	**0.9388**	**0.9122**	**0.8712**	**0.7927**
Chairs	**BPDF**	0.9970	0.9921	0.9831	0.9714	0.9490	0.9164	0.8609	0.7419	0.2090
	MDBUTMF	0.9972	0.9473	0.8106	0.7771	0.8912	0.9641	0.9684	0.8949	0.4788
	DBAIN	0.9981	0.9954	0.9908	0.9839	0.9710	0.9543	0.9220	0.8690	0.7811
	NAFSMF	0.9941	0.9904	0.9868	0.9841	0.9797	0.9740	0.9615	0.9359	0.8321
	DAMF	0.9989	0.9972	0.9948	0.9917	0.9877	0.9826	0.9758	0.9625	0.9322
	AWMF	0.9953	0.9950	0.9939	0.9919	0.9889	0.9843	0.9777	0.9648	0.9389
	ARmF	**0.9988**	**0.9974**	**0.9958**	**0.9936**	**0.9902**	**0.9855**	**0.9788**	**0.9657**	**0.9396**

Bold values indicate the best scores

Table 7 SSIM results of the filters for clips, coins, cushions, duck, fence, flowers, and garden table images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Clips	**BPDF**	0.9869	0.9621	0.9215	0.8606	0.7745	0.6553	0.4958	0.3162	0.1369
	MDBUTMF	0.9879	0.9603	0.9024	0.8761	0.8857	0.8747	0.8321	0.7515	0.5460
	DBAIN	0.9891	0.9746	0.9511	0.9126	0.8557	0.7733	0.6486	0.4860	0.3059
	NAFSMF	0.9781	0.9565	0.9313	0.9025	0.8699	0.8305	0.7773	0.7012	0.5657
	DAMF	0.9946	0.9864	0.9742	0.9575	0.9359	0.9074	0.8653	0.7970	0.6546
	AWMF	0.9763	0.9709	0.9639	0.9532	0.9377	0.9130	0.8719	0.8041	0.6617
	ARmF	**0.9943**	**0.9878**	**0.9788**	**0.9662**	**0.9490**	**0.9227**	**0.8803**	**0.8111**	**0.6668**
Coins	**BPDF**	0.9782	0.9497	0.9136	0.8693	0.8091	0.7281	0.6254	0.4583	0.1698
	MDBUTMF	0.9740	0.9251	0.8243	0.7835	0.8160	0.8207	0.7830	0.6999	0.4357
	DBAIN	0.9825	0.9616	0.9351	0.8996	0.8500	0.7863	0.7017	0.5930	0.4745
	NAFSMF	0.9689	0.9381	0.9068	0.8739	0.8377	0.7967	0.7477	0.6854	0.5658
	DAMF	0.9850	0.9688	0.9506	0.9282	0.9012	0.8671	0.8235	0.7596	0.6515
	AWMF	0.9691	0.9575	0.9437	0.9263	0.9033	0.8721	0.8285	0.7650	0.6577
	ARmF	**0.9867**	**0.9732**	**0.9582**	**0.9392**	**0.9153**	**0.8825**	**0.8370**	**0.7709**	**0.6609**
Cushions	**BPDF**	0.9937	0.9846	0.9714	0.9522	0.9216	0.8789	0.8096	0.6474	0.2109
	MDBUTMF	0.9939	0.9372	0.7809	0.7465	0.8601	0.9421	0.9414	0.8705	0.4778
	DBAIN	0.9958	0.9902	0.9820	0.9704	0.9500	0.9224	0.8832	0.8117	0.6952
	NAFSMF	0.9887	0.9804	0.9729	0.9655	0.9563	0.9454	0.9282	0.8992	0.7862
	DAMF	0.9964	0.9927	0.9877	0.9815	0.9739	0.9643	0.9509	0.9304	0.8838
	AWMF	0.9921	0.9900	0.9870	0.9829	0.9769	0.9687	0.9555	0.9355	0.8927
	ARmF	**0.9971**	**0.9942**	**0.9907**	**0.9860**	**0.9797**	**0.9710**	**0.9576**	**0.9372**	**0.8939**
Duck	**BPDF**	0.9956	0.9891	0.9788	0.9649	0.9443	0.9120	0.8454	0.6864	0.3034
	MDBUTMF	0.9955	0.9497	0.8097	0.7855	0.8878	0.9555	0.9531	0.8754	0.4343
	DBAIN	0.9973	0.9931	0.9871	0.9773	0.9629	0.9391	0.9001	0.8344	0.7105
	NAFSMF	0.9953	0.9908	0.9866	0.9815	0.9754	0.9666	0.9512	0.9232	0.8085

Table 7 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
	DAMF	0.9983	0.9957	0.9922	0.9878	0.9820	0.9746	0.9632	0.9462	0.9047
	AWMF	0.9944	0.9930	0.9911	0.9882	0.9835	0.9768	0.9661	0.9497	0.9109
	ARmF	**0.9983**	**0.9963**	**0.9938**	**0.9905**	**0.9856**	**0.9787**	**0.9677**	**0.9511**	**0.9121**
Fence	BPDF	0.9840	0.9671	0.9458	0.9146	0.8759	0.8153	0.7142	0.5383	0.2133
	MDBUTMF	0.9827	0.9489	0.8565	0.8309	0.8772	0.9025	0.8779	0.7935	0.4382
	DBAIN	0.9933	0.9833	0.9682	0.9445	0.9121	0.8619	0.7923	0.6810	0.5020
	NAFSMF	0.9812	0.9671	0.9528	0.9365	0.9193	0.8949	0.8627	0.8121	0.6814
	DAMF	0.9934	0.9874	0.9781	0.9654	0.9505	0.9314	0.9044	0.8624	0.7654
	AWMF	0.9785	0.9772	0.9735	0.9673	0.9577	0.9415	0.9156	0.8736	0.7781
	ARmF	**0.9932**	**0.9888**	**0.9834**	**0.9758**	**0.9651**	**0.9479**	**0.9213**	**0.8785**	**0.7821**
Flowers	BPDF	0.9841	0.9611	0.9272	0.8805	0.8068	0.7058	0.5686	0.3679	0.1790
	MDBUTMF	0.9795	0.9374	0.8455	0.8121	0.8456	0.8596	0.8255	0.7449	0.4691
	DBAIN	0.9889	0.9742	0.9528	0.9215	0.8738	0.8060	0.7114	0.5790	0.3994
	NAFSMF	0.9744	0.9486	0.9223	0.8951	0.8619	0.8279	0.7850	0.7231	0.5975
	DAMF	0.9920	0.9822	0.9692	0.9540	0.9333	0.9065	0.8698	0.8113	0.6919
	AWMF	0.9789	0.9722	0.9640	0.9538	0.9380	0.9139	0.8792	0.8205	0.7021
	ARmF	**0.9935**	**0.9861**	**0.9766**	**0.9651**	**0.9478**	**0.9225**	**0.8864**	**0.8262**	**0.7057**
Garden Table	BPDF	0.9751	0.9432	0.9013	0.8477	0.7784	0.6757	0.5549	0.3896	0.2267
	MDBUTMF	0.9668	0.9206	0.8333	0.7914	0.8057	0.7969	0.7490	0.6649	0.4036
	DBAIN	0.9791	0.9553	0.9229	0.8823	0.8298	0.7527	0.6593	0.5417	0.4017
	NAFSMF	0.9671	0.9345	0.9000	0.8635	0.8239	0.7761	0.7209	0.6547	0.5321
	DAMF	0.9813	0.9619	0.9395	0.9132	0.8826	0.8429	0.7939	0.7274	0.6150
	AWMF	0.9671	0.9532	0.9357	0.9156	0.8895	0.8529	0.8042	0.7365	0.6241
	ARmF	**0.9852**	**0.9698**	**0.9513**	**0.9297**	**0.9025**	**0.8646**	**0.8142**	**0.7444**	**0.6286**

Bold values indicate the best scores

Table 8 SSIM results of the filters for guitar bridge, guitar fret, guitar head, keyboard 1, keyboard 2, lion, and multimeter images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Guitar Bridge	**BPDF**	0.9788	0.9533	0.9239	0.8883	0.8418	0.7791	0.7003	0.5688	0.2842
	MDBUTMF	0.9760	0.9137	0.7811	0.7352	0.8024	0.8358	0.8072	0.7157	0.3713
	DBAIN	0.9835	0.9645	0.9417	0.9110	0.8730	0.8229	0.7588	0.6751	0.5780
	NAFSMF	0.9736	0.9465	0.9207	0.8919	0.8628	0.8297	0.7893	0.7384	0.6298
	DAMF	0.9845	0.9687	0.9515	0.9311	0.9073	0.8778	0.8410	0.7892	0.7091
	AWMF	0.9712	0.9605	0.9477	0.9319	0.9123	0.8853	0.8487	0.7960	0.7167
	ARmF	**0.9873**	**0.9753**	**0.9623**	**0.9460**	**0.9256**	**0.8975**	**0.8593**	**0.8043**	**0.7216**
Guitar Fret	**BPDF**	0.9874	0.9713	0.9480	0.9155	0.8676	0.7955	0.6863	0.5161	0.2784
	MDBUTMF	0.9855	0.9343	0.8019	0.7698	0.8459	0.8993	0.8861	0.8056	0.4755
	DBAIN	0.9905	0.9805	0.9644	0.9424	0.9111	0.8654	0.7973	0.7073	0.5877
	NAFSMF	0.9774	0.9577	0.9404	0.9203	0.9000	0.8773	0.8460	0.8021	0.6765
	DAMF	0.9898	0.9813	0.9708	0.9581	0.9431	0.9246	0.9002	0.8616	0.7826
	AWMF	0.9817	0.9783	0.9726	0.9651	0.9547	0.9390	0.9159	0.8788	0.8041
	ARmF	**0.9933**	**0.9882**	**0.9815**	**0.9733**	**0.9621**	**0.9456**	**0.9216**	**0.8836**	**0.8079**
Guitar Head	**BPDF**	0.9776	0.9520	0.9202	0.8728	0.8103	0.7282	0.6177	0.4492	0.2414
	MDBUTMF	0.9731	0.9261	0.8215	0.7830	0.8226	0.8338	0.7966	0.7198	0.4877
	DBAIN	0.9848	0.9677	0.9425	0.9094	0.8627	0.7954	0.7086	0.5956	0.4513
	NAFSMF	0.9685	0.9391	0.9084	0.8764	0.8424	0.8029	0.7531	0.6930	0.5740
	DAMF	0.9847	0.9715	0.9546	0.9351	0.9102	0.8787	0.8384	0.7787	0.6698
	AWMF	0.9686	0.9606	0.9495	0.9359	0.9159	0.8876	0.8481	0.7882	0.6797
	ARmF	**0.9873**	**0.9769**	**0.9646**	**0.9493**	**0.9282**	**0.8984**	**0.8574**	**0.7957**	**0.6846**
Keyboard 1	**BPDF**	0.9845	0.9625	0.9325	0.8908	0.8336	0.7580	0.6575	0.5187	0.2150
	MDBUTMF	0.9789	0.9259	0.8036	0.7644	0.8231	0.8467	0.8192	0.7452	0.4959
	DBAIN	0.9873	0.9713	0.9489	0.9173	0.8727	0.8064	0.7228	0.6162	0.4966
	NAFSMF	0.9683	0.9424	0.9154	0.8865	0.8530	0.8132	0.7718	0.7108	0.5895

Table 8 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
	DAMF	0.9899	0.9786	0.9652	0.9481	0.9259	0.8972	0.8583	0.8003	0.6944
	AWMF	0.9735	0.9662	0.9558	0.9426	0.9235	0.8970	0.8596	0.8021	0.6993
	ARmF	**0.9904**	**0.9804**	**0.9687**	**0.9539**	**0.9335**	**0.9055**	**0.8663**	**0.8064**	**0.7005**
Keyboard 2	**BPDF**	0.9782	0.9536	0.9214	0.8792	0.8262	0.7503	0.6368	0.4257	0.1368
	MDBUTMF	0.9773	0.9410	0.8558	0.8271	0.8557	0.8575	0.8236	0.7307	0.4098
	DBAIN	0.9860	0.9683	0.9435	0.9104	0.8665	0.8080	0.7251	0.6164	0.4785
	NAFSMF	0.9713	0.9500	0.9269	0.9011	0.8738	0.8391	0.8000	0.7414	0.6195
	DAMF	0.9865	0.9731	0.9554	0.9349	0.9092	0.8804	0.8434	0.7917	0.6959
	AWMF	0.9695	0.9620	0.9512	0.9374	0.9179	0.8919	0.8551	0.8020	0.7051
	ARmF	**0.9883**	**0.9774**	**0.9644**	**0.9491**	**0.9284**	**0.9010**	**0.8633**	**0.8085**	**0.7099**
Lion	**BPDF**	0.9900	0.9767	0.9559	0.9310	0.8973	0.8506	0.7828	0.6899	0.5169
	MDBUTMF	0.9861	0.9224	0.7593	0.7232	0.8294	0.9030	0.8922	0.8011	0.4326
	DBAIN	0.9927	0.9839	0.9707	0.9530	0.9253	0.8880	0.8341	0.7627	0.6649
	NAFSMF	0.9831	0.9678	0.9513	0.9345	0.9158	0.8944	0.8637	0.8230	0.7029
	DAMF	0.9930	0.9867	0.9786	0.9685	0.9556	0.9390	0.9161	0.8796	0.8079
	AWMF	0.9822	0.9790	0.9741	0.9672	0.9572	0.9430	0.9210	0.8847	0.8163
	ARmF	**0.9934**	**0.9885**	**0.9827**	**0.9748**	**0.9637**	**0.9487**	**0.9258**	**0.8889**	**0.8188**
Multimeter	**BPDF**	0.9760	0.9496	0.9193	0.8833	0.8357	0.7745	0.6923	0.5682	0.3638
	MDBUTMF	0.9788	0.9226	0.7867	0.7477	0.8135	0.8523	0.8257	0.7527	0.4743
	DBAIN	0.9803	0.9604	0.9347	0.9036	0.8642	0.8146	0.7521	0.6711	0.5545
	NAFSMF	0.9769	0.9547	0.9314	0.9059	0.8768	0.8464	0.8076	0.7587	0.6396
	DAMF	0.9835	0.9656	0.9443	0.9204	0.8916	0.8610	0.8247	0.7792	0.7064
	AWMF	0.9722	0.9607	0.9449	0.9248	0.9003	0.8709	0.8344	0.7869	0.7151
	ARmF	**0.9856**	**0.9711**	**0.9542**	**0.9340**	**0.9092**	**0.8793**	**0.8416**	**0.7924**	**0.7182**

Bold values indicate the best scores

Table 9 SSIM results of the filters for pencils 1, pencils 2, pillar, plastic, roof, scarf, and screws images

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Pencils 1	**BPDF**	0.9824	0.9619	0.9346	0.8940	0.8290	0.7271	0.5752	0.3305	0.1055
	MDBUTMF	0.9914	0.9598	0.8734	0.8447	0.8874	0.9022	0.8786	0.7850	0.4233
	DBAIN	0.9941	0.9830	0.9648	0.9384	0.8979	0.8365	0.7502	0.6148	0.4546
	NAFSMF	0.9844	0.9678	0.9497	0.9291	0.9072	0.8806	0.8476	0.7931	0.6717
	DAMF	0.9968	0.9903	0.9804	0.9675	0.9516	0.9332	0.9082	0.8666	0.7794
	AWMF	0.9862	0.9829	0.9788	0.9722	0.9628	0.9473	0.9232	0.8806	0.7927
	ARmF	**0.9968**	**0.9926**	**0.9875**	**0.9801**	**0.9695**	**0.9536**	**0.9287**	**0.8856**	**0.7973**
Pencils 2	**BPDF**	0.9822	0.9625	0.9383	0.9025	0.8520	0.7649	0.6392	0.4102	0.1271
	MDBUTMF	0.9877	0.9502	0.8491	0.8182	0.8733	0.9003	0.8777	0.7952	0.4778
	DBAIN	0.9934	0.9818	0.9640	0.9385	0.9005	0.8432	0.7645	0.6414	0.4705
	NAFSMF	0.9850	0.9708	0.9560	0.9378	0.9163	0.8927	0.8576	0.8030	0.6831
	DAMF	0.9957	0.9887	0.9784	0.9645	0.9471	0.9280	0.9023	0.8585	0.7705
	AWMF	0.9834	0.9802	0.9756	0.9691	0.9580	0.9425	0.9169	0.8721	0.7838
	ARmF	**0.9954**	**0.9908**	**0.9852**	**0.9774**	**0.9656**	**0.9492**	**0.9228**	**0.8775**	**0.7885**
Pillar	**BPDF**	0.9861	0.9706	0.9449	0.9138	0.8642	0.7986	0.7014	0.5543	0.2462
	MDBUTMF	0.9726	0.9260	0.8096	0.7746	0.8382	0.8718	0.8473	0.7630	0.3974
	DBAIN	0.9908	0.9794	0.9628	0.9394	0.9055	0.8565	0.7905	0.7035	0.5833
	NAFSMF	0.9747	0.9504	0.9268	0.9061	0.8807	0.8523	0.8151	0.7686	0.6498
	DAMF	0.9876	0.9787	0.9674	0.9548	0.9381	0.9175	0.8888	0.8441	0.7582
	AWMF	0.9804	0.9753	0.9687	0.9607	0.9484	0.9310	0.9031	0.8596	0.7767
	ARmF	**0.9929**	**0.9868**	**0.9791**	**0.9701**	**0.9564**	**0.9381**	**0.9090**	**0.8637**	**0.7794**
Plastic	**BPDF**	0.9735	0.9443	0.9122	0.8732	0.8278	0.7627	0.6651	0.4914	0.1463
	MDBUTMF	0.9747	0.9340	0.8420	0.8049	0.8329	0.8362	0.8018	0.7130	0.3707
	DBAIN	0.9774	0.9547	0.9262	0.8936	0.8542	0.8041	0.7391	0.6635	0.5652
	NAFSMF	0.9785	0.9569	0.9329	0.9079	0.8794	0.8482	0.8104	0.7611	0.6542

Table 9 continued

	Filters	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
	DAMF	0.9808	0.9605	0.9367	0.9109	0.8811	0.8467	0.8060	0.7587	0.6995
	AWMF	0.9705	0.9571	0.9380	0.9155	0.8884	0.8552	0.8138	0.7660	0.7088
	ARmF	**0.9832**	**0.9659**	**0.9451**	**0.9223**	**0.8950**	**0.8612**	**0.8186**	**0.7688**	**0.7095**
Roof	**BPDF**	0.9884	0.9692	0.9426	0.9004	0.8398	0.7580	0.6625	0.5544	0.4146
	MDBUTMF	0.9749	0.9108	0.7618	0.7177	0.8079	0.8690	0.8455	0.7538	0.4181
	DBAIN	0.9884	0.9768	0.9571	0.9277	0.8844	0.8222	0.7425	0.6504	0.5633
	NAFSMF	0.9600	0.9307	0.9062	0.8813	0.8542	0.8222	0.7810	0.7197	0.6024
	DAMF	0.9896	0.9826	0.9721	0.9583	0.9402	0.9170	0.8877	0.8398	0.7418
	AWMF	0.9801	0.9753	0.9706	0.9624	0.9511	0.9311	0.9019	0.8549	0.7572
	ARmF	**0.9944**	**0.9888**	**0.9822**	**0.9729**	**0.9600**	**0.9391**	**0.9090**	**0.8612**	**0.7620**
Scarf	**BPDF**	0.9816	0.9538	0.9115	0.8519	0.7673	0.6506	0.4974	0.3255	0.1450
	MDBUTMF	0.9780	0.9427	0.8752	0.8441	0.8516	0.8381	0.7903	0.7044	0.4862
	DBAIN	0.9853	0.9677	0.9401	0.9017	0.8431	0.7568	0.6387	0.4909	0.3294
	NAFSMF	0.9683	0.9379	0.9030	0.8688	0.8263	0.7815	0.7233	0.6506	0.5226
	DAMF	0.9896	0.9779	0.9621	0.9433	0.9173	0.8840	0.8365	0.7644	0.6258
	AWMF	0.9725	0.9640	0.9529	0.9396	0.9186	0.8881	0.8422	0.7705	0.6325
	ARmF	**0.9906**	**0.9811**	**0.9685**	**0.9532**	**0.9303**	**0.8981**	**0.8506**	**0.7769**	**0.6357**
Screws	**BPDF**	0.9832	0.9572	0.9187	0.8667	0.7921	0.6877	0.5460	0.3648	0.1357
	MDBUTMF	0.9771	0.9429	0.8763	0.8424	0.8468	0.8315	0.7840	0.6966	0.4416
	DBAIN	0.9873	0.9693	0.9439	0.9060	0.8520	0.7715	0.6644	0.5246	0.3621
	NAFSMF	0.9647	0.9313	0.8947	0.8554	0.8144	0.7662	0.7085	0.6334	0.5080
	DAMF	0.9899	0.9777	0.9628	0.9443	0.9207	0.8894	0.8450	0.7747	0.6312
	AWMF	0.9732	0.9650	0.9553	0.9422	0.9241	0.8961	0.8524	0.7822	0.6383
	ARmF	**0.9917**	**0.9824**	**0.9713**	**0.9565**	**0.9362**	**0.9068**	**0.8613**	**0.7892**	**0.6426**

Bold values indicate the best scores

Table 10 SSIM results of the filters for snails, socks, sweets, tomatoes 1, tomatoes 2, tools 1, and tools 2 images

	Filters	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
Snails	**BPDF**	0.9913	0.9786	0.9626	0.9352	0.8967	0.8427	0.7473	0.5672	0.2653
	MDBUTMF	0.9832	0.9289	0.7986	0.7652	0.8590	0.9176	0.9096	0.8365	0.4737
	DBAIN	0.9940	0.9868	0.9759	0.9598	0.9362	0.8994	0.8473	0.7624	0.6293
	NAFSMF	0.9860	0.9736	0.9628	0.9502	0.9369	0.9202	0.8978	0.8584	0.7358
	DAMF	0.9935	0.9879	0.9815	0.9726	0.9619	0.9486	0.9295	0.8989	0.8341
	AWMF	0.9883	0.9856	0.9814	0.9754	0.9674	0.9555	0.9374	0.9078	0.8466
	ARmF	**0.9957**	**0.9918**	**0.9871**	**0.9805**	**0.9718**	**0.9594**	**0.9408**	**0.9106**	**0.8483**
Socks	**BPDF**	0.9688	0.9308	0.8838	0.8294	0.7623	0.6763	0.5619	0.3923	0.1633
	MDBUTMF	0.9690	0.9189	0.8277	0.7808	0.7854	0.7613	0.7076	0.6226	0.4132
	DBAIN	0.9728	0.9432	0.9053	0.8590	0.8023	0.7278	0.6361	0.5269	0.3847
	NAFSMF	0.9674	0.9331	0.8951	0.8545	0.8107	0.7579	0.6987	0.6264	0.5034
	DAMF	0.9774	0.9526	0.9232	0.8905	0.8528	0.8041	0.7471	0.6729	0.5537
	AWMF	0.9634	0.9444	0.9209	0.8932	0.8601	0.8137	0.7568	0.6810	0.5610
	ARmF	**0.9815**	**0.9612**	**0.9371**	**0.9090**	**0.8751**	**0.8274**	**0.7686**	**0.6897**	**0.5659**
Sweets	**BPDF**	0.9895	0.9755	0.9549	0.9239	0.8783	0.8075	0.6966	0.4842	0.1034
	MDBUTMF	0.9911	0.9525	0.8604	0.8345	0.8931	0.9231	0.9032	0.8182	0.4401
	DBAIN	0.9927	0.9843	0.9707	0.9512	0.9201	0.8729	0.8028	0.6945	0.5457
	NAFSMF	0.9870	0.9745	0.9617	0.9480	0.9326	0.9126	0.8836	0.8406	0.7198
	DAMF	0.9950	0.9891	0.9818	0.9715	0.9599	0.9432	0.9196	0.8817	0.8056
	AWMF	0.9865	0.9830	0.9787	0.9720	0.9627	0.9481	0.9252	0.8879	0.8138
	ARmF	**0.9954**	**0.9910**	**0.9857**	**0.9782**	**0.9682**	**0.9528**	**0.9293**	**0.8911**	**0.8161**
Tomatoes 1	**BPDF**	0.9924	0.9807	0.9631	0.9376	0.9000	0.8338	0.7264	0.5077	0.1617
	MDBUTMF	0.9938	0.9502	0.8264	0.8017	0.8892	0.9411	0.9337	0.8539	0.4486
	DBAIN	0.9945	0.9884	0.9787	0.9642	0.9404	0.9040	0.8474	0.7503	0.5836
	NAFSMF	0.9926	0.9846	0.9786	0.9700	0.9601	0.9480	0.9285	0.8923	0.7698

Table 10 continued

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
	DAMF	0.9964	0.9918	0.9860	0.9793	0.9701	0.9587	0.9419	0.9162	0.8533
	AWMF	0.9923	0.9902	0.9871	0.9825	0.9758	0.9658	0.9499	0.9247	0.8663
	ARmF	**0.9972**	**0.9943**	**0.9907**	**0.9858**	**0.9787**	**0.9685**	**0.9524**	**0.9268**	**0.8679**
Tomatoes 2	**BPDF**	0.9951	0.9870	0.9741	0.9554	0.9251	0.8724	0.7878	0.6170	0.3064
	MDBUTMF	0.9885	0.9311	0.7664	0.7358	0.8516	0.9382	0.9384	0.8737	0.5117
	DBAIN	0.9971	0.9928	0.9856	0.9740	0.9578	0.9232	0.8764	0.7908	0.6475
	NAFSMF	0.9936	0.9880	0.9826	0.9768	0.9692	0.9583	0.9394	0.9076	0.7862
	DAMF	0.9984	0.9957	0.9922	0.9869	0.9802	0.9709	0.9578	0.9353	0.8815
	AWMF	0.9942	0.9931	0.9910	0.9879	0.9832	0.9753	0.9630	0.9418	0.8945
	ARmF	**0.9985**	**0.9966**	**0.9943**	**0.9906**	**0.9855**	**0.9775**	**0.9648**	**0.9435**	**0.8959**
Tools 1	**BPDF**	0.9824	0.9594	0.9296	0.8841	0.8242	0.7361	0.6063	0.3888	0.1408
	MDBUTMF	0.9844	0.9522	0.8855	0.8553	0.8745	0.8729	0.8374	0.7473	0.4453
	DBAIN	0.9890	0.9732	0.9523	0.9206	0.8772	0.8171	0.7267	0.6035	0.4427
	NAFSMF	0.9785	0.9564	0.9338	0.9083	0.8788	0.8468	0.8057	0.7419	0.6249
	DAMF	0.9921	0.9811	0.9678	0.9502	0.9282	0.9025	0.8676	0.8126	0.7087
	AWMF	0.9783	0.9710	0.9625	0.9513	0.9348	0.9119	0.8776	0.8223	0.7185
	ARmF	**0.9926**	**0.9842**	**0.9747**	**0.9623**	**0.9448**	**0.9207**	**0.8851**	**0.8288**	**0.7232**
Tools 2	**BPDF**	0.9844	0.9642	0.9385	0.9035	0.8585	0.7936	0.6936	0.5227	0.2524
	MDBUTMF	0.9812	0.9226	0.7653	0.7311	0.8334	0.8973	0.8871	0.8232	0.5301
	DBAIN	0.9875	0.9754	0.9591	0.9357	0.9047	0.8597	0.7963	0.7041	0.5602
	NAFSMF	0.9835	0.9695	0.9558	0.9403	0.9229	0.9026	0.8776	0.8385	0.7200
	DAMF	0.9884	0.9776	0.9652	0.9501	0.9316	0.9104	0.8848	0.8502	0.7902
	AWMF	0.9792	0.9733	0.9641	0.9517	0.9359	0.9158	0.8906	0.8560	0.7986
	ARmF	**0.9898**	**0.9806**	**0.9703**	**0.9573**	**0.9410**	**0.9207**	**0.8947**	**0.8588**	**0.8002**

Bold values indicate the best scores

Table 11 SSIM results of the filters for wood game image

	Filters	10%	20%	30%	40%	50%	60%	70%	80%	90%
Wood Game	**BPDF**	0.9915	0.9793	0.9653	0.9445	0.9186	0.8725	0.7968	0.6439	0.3410
	MDBUTMF	0.9767	0.9028	0.6915	0.6573	0.8076	0.9188	0.9259	0.8290	0.3816
	DBAIN	0.9953	0.9911	0.9839	0.9728	0.9552	0.9303	0.8912	0.8290	0.7366
	NAFSMF	0.9757	0.9624	0.9545	0.9476	0.9413	0.9328	0.9172	0.8904	0.7741
	DAMF	0.9931	0.9877	0.9824	0.9762	0.9680	0.9585	0.9442	0.9225	0.8739
	AWMF	0.9869	0.9859	0.9834	0.9795	0.9739	0.9661	0.9525	0.9324	0.8915
	ARmF	**0.9943**	**0.9916**	**0.9881**	**0.9835**	**0.9772**	**0.9689**	**0.9548**	**0.9344**	**0.8926**

Bold values indicate the best scores

that the noise-removal performances of the filters are more significant in high noise densities, in which noisy pixels outnumber uncorrupted pixels, then performance-based success would be more important in the presence of high noise densities than of others. For example, let

$$[b_{0j}] = \begin{bmatrix} [0,0.01] & [0,0.05] & [0,0.1] & [0.05,0.35] & [0.2,0.45] & [0.25,0.5] & [0.8,0.85] & [0.85,0.9] & [0.9,0.95] \\ [0.9,0.95] & [0.85,0.9] & [0.8,0.85] & [0.25,0.5] & [0.2,0.45] & [0.05,0.35] & [0,0.1] & [0,0.05] & [0,0.01] \end{bmatrix}$$

Thus, the d-matrix $[b_{ij}]$, modelling the SSIM values provided in Tables 5, 6, 7, 8, 9, 10, and 11, is as follows:

$$[b_{ij}] = \left[\begin{matrix} [0,0.01] & [0,0.05] & [0,0.1] & [0.05,0.35] & [0.2,0.45] \\ [0.9,0.95] & [0.85,0.9] & [0.8,0.85] & [0.25,0.5] & [0.2,0.45] \\ [0.9422,0.9696] & [0.8771,0.9348] & [0.8040,0.8943] & [0.7263,0.8506] & [0.6424,0.7997] \\ [0.0029,0.0304] & [0.0074,0.0652] & [0.0154,0.1057] & [0.0250,0.1494] & [0.0430,0.2003] \\ [0.9382,0.9677] & [0.8538,0.9081] & [0.5711,0.7452] & [0.5393,0.7188] & [0.7090,0.8063] \\ [0.0027,0.0323] & [0.0376,0.0919] & [0.0806,0.2548] & [0.1017,0.2812] & [0.0965,0.1937] \\ [0.9488,0.9735] & [0.8965,0.9460] & [0.8340,0.9127] & [0.7636,0.8747] & [0.6865,0.8309] \\ [0.0019,0.0265] & [0.0044,0.0540] & [0.0085,0.0873] & [0.0143,0.1253] & [0.0248,0.1691] \\ [0.9272,0.9613] & [0.8780,0.9346] & [0.8192,0.9036] & [0.7564,0.8712] & [0.6936,0.8381] \\ [0.0045,0.0387] & [0.0087,0.0654] & [0.0121,0.0964] & [0.0140,0.1288] & [0.0174,0.1619] \\ [0.9569,0.9779] & [0.9120,0.9546] & [0.8616,0.9284] & [0.8087,0.9006] & [0.7515,0.8703] \\ [0.0011,0.0221] & [0.0027,0.0454] & [0.0048,0.0716] & [0.0075,0.0994] & [0.0108,0.1297] \\ [0.9336,0.9645] & [0.8990,0.9471] & [0.8582,0.9262] & [0.8130,0.9028] & [0.7620,0.8761] \\ [0.0046,0.0355] & [0.0048,0.0529] & [0.0057,0.0738] & [0.0073,0.0972] & [0.0099,0.1239] \\ [0.9648,0.9818] & [0.9276,0.9626] & [0.8852,0.9406] & [0.8381,0.9161] & [0.7848,0.8880] \\ [0.0012,0.0182] & [0.0025,0.0374] & [0.0040,0.0594] & [0.0059,0.0839] & [0.0088,0.1120] \end{matrix}\right.$$

$$\left.\begin{matrix} [0.25,0.5] & [0.8,0.85] & [0.85,0.9] & [0.9,0.95] \\ [0.05,0.35] & [0,0.1] & [0,0.05] & [0,0.01] \\ [0.5140,0.7240] & [0.3632,0.6306] & [0.2218,0.5204] & [0.0602,0.3613] \\ [0.0660,0.2760] & [0.1019,0.3694] & [0.1810,0.4796] & [0.3377,0.6387] \\ [0.6329,0.8015] & [0.5613,0.7681] & [0.4893,0.7034] & [0.2977,0.4583] \\ [0.0299,0.1985] & [0.0250,0.2319] & [0.0826,0.2966] & [0.3811,0.5417] \\ [0.5934,0.7781] & [0.4946,0.7170] & [0.3514,0.6284] & [0.2074,0.5295] \\ [0.0373,0.2219] & [0.0606,0.2830] & [0.0947,0.3716] & [0.1484,0.4705] \\ [0.6232,0.8009] & [0.5533,0.7614] & [0.4783,0.7147] & [0.3789,0.6262] \\ [0.0214,0.1991] & [0.0305,0.2386] & [0.0489,0.2853] & [0.1264,0.3738] \\ [0.6823,0.8338] & [0.6080,0.7942] & [0.5217,0.7463] & [0.4017,0.6762] \\ [0.0147,0.1662] & [0.0197,0.2058] & [0.0290,0.2537] & [0.0492,0.3238] \\ [0.6951,0.8408] & [0.6199,0.8008] & [0.5305,0.7515] & [0.4071,0.6814] \\ [0.0134,0.1592] & [0.0182,0.1992] & [0.0274,0.2485] & [0.0443,0.3186] \\ [0.7145,0.8510] & [0.6351,0.8088] & [0.5406,0.7568] & [0.4120,0.6840] \\ [0.0125,0.1490] & [0.0175,0.1912] & [0.0269,0.2432] & [0.0440,0.3160] \end{matrix}\right]$$

Finally, we apply the configured method to $[b_{ij}]$. Moreover, we obtain the results herein by MATLAB R2021a.

Step 2. The column matrix $\begin{bmatrix}\alpha_{i1}\\ \beta_{i1}\end{bmatrix}$ is as follows:

$$\begin{bmatrix}\alpha_{i1}\\ \beta_{i1}\end{bmatrix} = \begin{bmatrix} [0.1837,0.5585] & [0.3244,0.6369] & [0.2678,0.6434] \\ [0.0087,0.1125] & [0.0323,0.1490] & [0.0050,0.0924] \\ [0.3383,0.6921] & & \\ [0.0065,0.0948] & & \\ [0.3673,0.7252] & [0.3733,0.7299] & [0.3813,0.7369] \\ [0.0026,0.0723] & [0.0038,0.0755] & [0.0023,0.0623] \end{bmatrix}^T$$

Step 3. The score matrix is as follows:

$$[s_{i1}] = [[0.0712, 0.5497]\ \ [0.1754, 0.6047]\ \ [0.1753, 0.6384]\ \ [0.2436, 0.6856]\ \ [0.2950, 0.7225]\ \ [0.2979, 0.7261]\ \ [0.3190, 0.7347]]^T$$

Step 4. The decision set is as follows:

$$\left\{{}^{[0.1768,0.7705]}\text{BPDF}, {}^{[0.3060,0.8387]}\text{ MDBUTMF}, {}^{[0.3059,0.8805]}\text{ DBAIN}, {}^{[0.3906,0.9392]}\text{ NAFSMF}, {}^{[0.4544,0.9849]}\text{DAMF}, {}^{[0.4580,0.9894]}\text{ AWMF}, {}^{[0.4842,1]}\text{ ARmF}\right\}$$

Step 5. The ranking order

$$\text{BPDF} \prec \text{MDBUTMF} \prec \text{DBAIN} \prec \text{NAFSMF} \prec \text{DAMF} \prec \text{AWMF} \prec \text{ARmF}$$

is valid. Therefore, the performance ranking of the filters shows that ARmF outperforms the other filters.

6 Comparative analysis

In this section, we compare the configured method with five SDM methods, namely iMBR01, iMRB02(I_9), iCCE10, iCCE11, and iPEM, provided in (Arslan et al. 2021). For this reason, first, Table 12 presents the filters' ranking orders provided in (Arslan et al. 2021) when the methods are applied to *ifpifs*-matrix $[a_{ij}]$ (Arslan et al. 2021) obtained using the results in Tables 1, 2, 3, and 4. Second, we construct *ifpifs*-matrix $[c_{ij}]$ using the membership and non-membership functions in (Arslan et al. 2021) and the filters' noise-removal performance results provided in Tables 5, 6, 7, 8, 9, 10, and 11. We then apply five SDM methods to this *ifpifs*-matrix.

$$[c_{ij}] = \begin{bmatrix} \begin{smallmatrix}0.05\\0.9\end{smallmatrix} & \begin{smallmatrix}0.15\\0.8\end{smallmatrix} & \begin{smallmatrix}0.25\\0.7\end{smallmatrix} & \begin{smallmatrix}0.35\\0.6\end{smallmatrix} & \begin{smallmatrix}0.5\\0.5\end{smallmatrix} & \begin{smallmatrix}0.65\\0.3\end{smallmatrix} & \begin{smallmatrix}0.75\\0.2\end{smallmatrix} & \begin{smallmatrix}0.85\\0.1\end{smallmatrix} & \begin{smallmatrix}0.9\\0.05\end{smallmatrix} \\
\begin{smallmatrix}0.9688\\0.0030\end{smallmatrix} & \begin{smallmatrix}0.9308\\0.0079\end{smallmatrix} & \begin{smallmatrix}0.8838\\0.0169\end{smallmatrix} & \begin{smallmatrix}0.8294\\0.0286\end{smallmatrix} & \begin{smallmatrix}0.7623\\0.0510\end{smallmatrix} & \begin{smallmatrix}0.6506\\0.0836\end{smallmatrix} & \begin{smallmatrix}0.4958\\0.1391\end{smallmatrix} & \begin{smallmatrix}0.3162\\0.2581\end{smallmatrix} & \begin{smallmatrix}0.0861\\0.4831\end{smallmatrix} \\
\begin{smallmatrix}0.9668\\0.0028\end{smallmatrix} & \begin{smallmatrix}0.9028\\0.0397\end{smallmatrix} & \begin{smallmatrix}0.6915\\0.0976\end{smallmatrix} & \begin{smallmatrix}0.6573\\0.1239\end{smallmatrix} & \begin{smallmatrix}0.7854\\0.1069\end{smallmatrix} & \begin{smallmatrix}0.7613\\0.0359\end{smallmatrix} & \begin{smallmatrix}0.7076\\0.0316\end{smallmatrix} & \begin{smallmatrix}0.6226\\0.1051\end{smallmatrix} & \begin{smallmatrix}0.3547\\0.4540\end{smallmatrix} \\
\begin{smallmatrix}0.9728\\0.0019\end{smallmatrix} & \begin{smallmatrix}0.9432\\0.0046\end{smallmatrix} & \begin{smallmatrix}0.9053\\0.0092\end{smallmatrix} & \begin{smallmatrix}0.8590\\0.0161\end{smallmatrix} & \begin{smallmatrix}0.8023\\0.0290\end{smallmatrix} & \begin{smallmatrix}0.7278\\0.0457\end{smallmatrix} & \begin{smallmatrix}0.6361\\0.0780\end{smallmatrix} & \begin{smallmatrix}0.4860\\0.1310\end{smallmatrix} & \begin{smallmatrix}0.3059\\0.2189\end{smallmatrix} \\
\begin{smallmatrix}0.9600\\0.0047\end{smallmatrix} & \begin{smallmatrix}0.9307\\0.0092\end{smallmatrix} & \begin{smallmatrix}0.8947\\0.0132\end{smallmatrix} & \begin{smallmatrix}0.8545\\0.0159\end{smallmatrix} & \begin{smallmatrix}0.8107\\0.0203\end{smallmatrix} & \begin{smallmatrix}0.7579\\0.0260\end{smallmatrix} & \begin{smallmatrix}0.6987\\0.0385\end{smallmatrix} & \begin{smallmatrix}0.6264\\0.0641\end{smallmatrix} & \begin{smallmatrix}0.5034\\0.1679\end{smallmatrix} \\
\begin{smallmatrix}0.9774\\0.0011\end{smallmatrix} & \begin{smallmatrix}0.9526\\0.0028\end{smallmatrix} & \begin{smallmatrix}0.9232\\0.0052\end{smallmatrix} & \begin{smallmatrix}0.8905\\0.0083\end{smallmatrix} & \begin{smallmatrix}0.8528\\0.0123\end{smallmatrix} & \begin{smallmatrix}0.8041\\0.0174\end{smallmatrix} & \begin{smallmatrix}0.7471\\0.0242\end{smallmatrix} & \begin{smallmatrix}0.6729\\0.0375\end{smallmatrix} & \begin{smallmatrix}0.5537\\0.0678\end{smallmatrix} \\
\begin{smallmatrix}0.9634\\0.0047\end{smallmatrix} & \begin{smallmatrix}0.9444\\0.0050\end{smallmatrix} & \begin{smallmatrix}0.9209\\0.0061\end{smallmatrix} & \begin{smallmatrix}0.8932\\0.0081\end{smallmatrix} & \begin{smallmatrix}0.8601\\0.0111\end{smallmatrix} & \begin{smallmatrix}0.8137\\0.0157\end{smallmatrix} & \begin{smallmatrix}0.7568\\0.0223\end{smallmatrix} & \begin{smallmatrix}0.6810\\0.0352\end{smallmatrix} & \begin{smallmatrix}0.5610\\0.0611\end{smallmatrix} \\
\begin{smallmatrix}0.9815\\0.0012\end{smallmatrix} & \begin{smallmatrix}0.9612\\0.0026\end{smallmatrix} & \begin{smallmatrix}0.9371\\0.0042\end{smallmatrix} & \begin{smallmatrix}0.9090\\0.0064\end{smallmatrix} & \begin{smallmatrix}0.8751\\0.0098\end{smallmatrix} & \begin{smallmatrix}0.8274\\0.0145\end{smallmatrix} & \begin{smallmatrix}0.7686\\0.0212\end{smallmatrix} & \begin{smallmatrix}0.6897\\0.0343\end{smallmatrix} & \begin{smallmatrix}0.5659\\0.0604\end{smallmatrix}
\end{bmatrix}$$

In Tables 13 and 14, we present the decision sets and the noise-removal filters' ranking orders when five SDM methods are applied to $[c_{ij}]$, respectively. We reveal in Section 5 that the configured method produces the same ranking orders for the filters' SSIM results obtained with 20 traditional test images and 40 test images at nine noise densities. Thus, the configured method confirms the ranking order provided in (Aydın and Enginoğlu 2021a) and those of iCCE10 and iCCE11 in Tables 12 and 14. On the other hand, although iPEM provides the same ranking order as iCCE10 and iCCE11 for 40 test images, iMBR01, iMRB02(I_9), and iPEM generate different ranking orders for 20 traditional test images. Consequently, we observe that the configured method is more consistent than iMBR01, iMRB02(I_9), and iPEM. Thus, these comments exhibit that the SDM method constructed

Table 12 Ranking orders generated by five SDM methods (Arslan et al. 2021)

Methods	Ranking orders
iMBR01	BPDF $\prec$ DBAIN $\prec$ NAFSMF $\prec$ MDBUTMF $\prec$ DAMF $\prec$ AWMF $\prec$ ARmF
iMRB02(I_9)	BPDF $\prec$ DBAIN $\prec$ NAFSMF $\prec$ MDBUTMF $\prec$ DAMF $\prec$ AWMF $\prec$ ARmF
iCCE10	BPDF $\prec$ MDBUTMF $\prec$ DBAIN $\prec$ NAFSMF $\prec$ DAMF $\prec$ AWMF $\prec$ ARmF
iCCE11	BPDF $\prec$ MDBUTMF $\prec$ DBAIN $\prec$ NAFSMF $\prec$ DAMF $\prec$ AWMF $\prec$ ARmF
iPEM	BPDF $\prec$ DBAIN $\prec$ MDBUTMF $\prec$ NAFSMF $\prec$ DAMF $\prec$ AWMF $\prec$ ARmF

with d-matrices is more advantageous in dealing with problems involving multiple measurement results.

7 Conclusion

In this paper, we defined the concept of d-matrices. Furthermore, we introduced its basic operations and investigated some of their basic properties. We then configured the SDM method (Aydın and Enginoğlu 2021a) to operate it in d-matrices space. Moreover, we applied it to two d-matrices constructed with SSIM results of the known noise-removal filters for 40 test images, provided in the TESTIMAGES database (Asuni and Giachetti 2014), and 20 traditional test images. This application results confirmed the one available in Aydın and Enginoğlu (2021a). Thus, the configured method enabled problems containing a large number of data to be processed on a computer. In addition, we applied five state-of-the-art SDM methods constructed with *ifpifs*-matrices to the same problem and compared the ranking performance of the configured method with those of the five methods.

The results in the present study manifested that the configured method was successfully applied to a decision-making problem containing *ivif* uncertainties. Therefore, further research should be focussed on developing effective SDM methods based on group decision making using AND/OR/ANDNOT/ORNOT-products of d-matrices. Moreover, it is possible to render the SDM methods constructed with *fpfs*-matrices (Enginoğlu and Memiş 2018d, 2020; Enginoğlu et al. 2018a, b, 2019c, d, 2021a) and *ifpifs*-matrices (Enginoğlu and Arslan 2020) operable in d-matrices space. Furthermore, the membership and non-membership functions used to obtain an *ivif*-value from multiple intuitionistic fuzzy values can be defined in a different way and used to construct a d-matrix in the first step of the configured method. Thus, these new methods can be applied to the problem featured in the current study and the results of this process can be compared with those herein. In addition, it is necessary and worthwhile to conduct theoretical and applied studies on varied topics, such as distance and similarity measures, by making use of the d-matrices. Researchers can also conduct studies on the various hybrid versions of soft sets and the other generalisations of fuzzy sets, such as hesitant fuzzy sets (Torra 2010), linear Diophantine fuzzy sets (Riaz and Hashmi 2019), spherical linear Diophantine fuzzy sets (Riaz et al. 2021), and picture fuzzy sets (Cuong 2014; Memiş 2021), and their matrices.

Table 13 Decision sets when five SDM methods are applied to $[c_{ij}]$

Methods	Decision sets
iMBR01	$\left\{{}^{0}_{0.8051}\mathrm{BPDF}, {}^{0.1949}_{0.5820}\mathrm{MDBUTMF}, {}^{0.1492}_{0.8333}\mathrm{DBAIN}, {}^{0.2648}_{0.6196}\mathrm{NAFSMF}, {}^{0.4624}_{0.3374}\mathrm{DAMF}, {}^{0.5538}_{0.4059}\mathrm{AWMF}, {}^{0.6895}_{0}\mathrm{ARmF}\right\}$
iMRB02(I_9)	$\left\{{}^{0.7871}_{0.1834}\mathrm{BPDF}, {}^{0.8991}_{0.0512}\mathrm{MDBUTMF}, {}^{0.8820}_{0.1022}\mathrm{DBAIN}, {}^{0.9459}_{0.0415}\mathrm{NAFSMF}, {}^{0.9807}_{0.0131}\mathrm{DAMF}, {}^{0.9858}_{0.0073}\mathrm{AWMF}, {}^{0.9947}_{0}\mathrm{ARmF}\right\}$
iCCE10	$\left\{{}^{0.2468}_{0.0185}\mathrm{BPDF}, {}^{0.3170}_{0.0312}\mathrm{MDBUTMF}, {}^{0.3063}_{0.0099}\mathrm{DBAIN}, {}^{0.3464}_{0.0079}\mathrm{NAFSMF}, {}^{0.3682}_{0.0039}\mathrm{DAMF}, {}^{0.3714}_{0.0043}\mathrm{AWMF}, {}^{0.3770}_{0.0033}\mathrm{ARmF}\right\}$
iCCE11	$\left\{{}^{0.2468}_{0.0185}\mathrm{BPDF}, {}^{0.3170}_{0.0312}\mathrm{MDBUTMF}, {}^{0.3063}_{0.0099}\mathrm{DBAIN}, {}^{0.3464}_{0.0079}\mathrm{NAFSMF}, {}^{0.3682}_{0.0039}\mathrm{DAMF}, {}^{0.3714}_{0.0043}\mathrm{AWMF}, {}^{0.3770}_{0.0033}\mathrm{ARmF}\right\}$
iPEM	$\left\{{}^{0.7200}_{0.2795}\mathrm{BPDF}, {}^{0.8282}_{0.1714}\mathrm{MDBUTMF}, {}^{0.8370}_{0.1629}\mathrm{DBAIN}, {}^{0.9132}_{0.0868}\mathrm{NAFSMF}, {}^{0.9735}_{0.0265}\mathrm{DAMF}, {}^{0.9800}_{0.0200}\mathrm{AWMF}, {}^{1}_{0}\mathrm{ARmF}\right\}$

Table 14 Noise removal filters' ranking orders when five SDM methods are applied to $[c_{ij}]$

Methods	Ranking orders
iMBR01	BPDF ≺ DBAIN ≺ MDBUTMF ≺ NAFSMF ≺ DAMF ≺ AWMF ≺ ARmF
iMRB02(I_9)	BPDF ≺ DBAIN ≺ MDBUTMF ≺ NAFSMF ≺ DAMF ≺ AWMF ≺ ARmF
iCCE10	BPDF ≺ MDBUTMF ≺ DBAIN ≺ NAFSMF ≺ DAMF ≺ AWMF ≺ ARmF
iCCE11	BPDF ≺ MDBUTMF ≺ DBAIN ≺ NAFSMF ≺ DAMF ≺ AWMF ≺ ARmF
iPEM	BPDF ≺ MDBUTMF ≺ DBAIN ≺ NAFSMF ≺ DAMF ≺ AWMF ≺ ARmF

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

Arslan B, Aydın T, Memiş S, Enginoğlu S (2021) Generalisations of SDM methods in *fpfs*-matrices space to render them operable in *ifpifs*-matrices space and their application to performance ranking of the noise-removal filters. J New Theory (36):88–116. https://doi.org/10.53570/jnt.989335

Asuni N, Giachetti A (2014) TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms. STAG - Smart Tools and Apps for Graphics Conference

Atanassov KT (1986) Intuitionistic fuzzy sets. Fuzzy Sets Syst 20(1):87–96. https://doi.org/10.1016/S0165-0114(86)80034-3

Atanassov KT (2020) Interval-valued intuitionistic fuzzy sets. Studies in Fuzziness and Soft Computing, Springer, New York. https://link.springer.com/book/10.1007/978-3-030-32090-4

Atanassov KT, Gargov G (1989) Interval valued intuitionistic fuzzy sets. Fuzzy Sets Syst 31(3):343–349. https://doi.org/10.1016/0165-0114(89)90205-4

Atmaca S (2017) Relationship between fuzzy soft topological spaces and (X, τ_e) parameter spaces. Cumhuriyet Sci J 38(4):77–85. https://doi.org/10.17776/csj.340541

Aydın T (2020) Interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft matrices and their application to a performance-based value assignment problem. PhD Dissertation, Çanakkale Onsekiz Mart University, Çanakkale, Turkey, In Turkish

Aydın T, Enginoğlu S (2019) A configuration of five of the soft decision-making methods via fuzzy parameterized fuzzy soft matrices and their application to a performance-based value assignment problem. In: Kılıç M, Özkan K, Karaboyacı M, Taşdelen K, Kandemir H, Beram A (eds) International Conferences on Science and Technology; Natural Science and Technology, Prizren, Kosovo, pp 56–67

Aydın T, Enginoğlu S (2020) Configurations of SDM methods proposed between 1999 and 2012: A follow-up study. In: Yıldırım K (ed) 4th International Conference on Mathematics: "An Istanbul Meeting for World Mathematicians", Istanbul, Turkey, pp 192–211

Aydın T, Enginoğlu S (2021) Interval-valued intuitionistic fuzzy parameterized interval-valued intuitionistic fuzzy soft sets and their application in decision-making. J Ambient Intell Hum Comput 12(1):1541–1558. https://doi.org/10.1007/s12652-020-02227-0

Aydın T, Enginoğlu S (2021b) Some results on soft topological notions. Journal of New Results in Science 10(1):65–75. https://dergipark.org.tr/tr/pub/jnrs/issue/62194/910337

Çağman N, Enginoğlu S (2010) Soft matrix theory and its decision making. Comput Math Appl 59(10):3308–3314. https://doi.org/10.1016/j.camwa.2010.03.015

Çağman N, Enginoğlu S (2010) Soft set theory and uni-int decision making. Eur J Oper Res 207(2):848–855. https://doi.org/10.1016/j.ejor.2010.05.004

Çağman N, Enginoğlu S (2012) Fuzzy soft matrix theory and its application in decision making. Iran J Fuzzy Syst 9(1):109–119. http://ijfs.usb.ac.ir/article_229.html

Çağman N, Çıtak F, Enginoğlu S (2010) Fuzzy parameterized fuzzy soft set theory and its applications. Turk J Fuzzy Syst 1(1):21–35

Çağman N, Çıtak F, Enginoğlu S (2011a) FP-soft set theory and its applications. Ann Fuzzy Math Inf 2(2):219–226. http://www.afmi.or.kr/papers/2011/Vol-02_No-02/AFMI-2-2(219-226)-J-110329R1.pdf

Çağman N, Enginoğlu S, Çıtak F (2011b) Fuzzy soft set theory and its applications. Iran J Fuzzy Syst 8(3):137–147. http://ijfs.usb.ac.ir/article_292.html

Çıtak F, Çağman N (2015) Soft int-rings and its algebraic applications. J Intell Fuzzy Syst 28(3):1225–1233. https://doi.org/10.3233/IFS-141406

Cuong BC (2014) Picture fuzzy sets. J Comput Sci Cybern 30(4):409–420. https://doi.org/10.15625/1813-9663/30/4/5032

Deli I, Çağman N (2015) Intuitionistic fuzzy parameterized soft set theory and its decision making. Appl Soft Comput 28:109–113. https://doi.org/10.1016/j.asoc.2014.11.053

Deli I, Karataş S (2016) Interval valued intuitionistic fuzzy parameterized soft set theory and its decision making. J Intell Fuzzy Syst 30(4):2073–2082. https://doi.org/10.3233/IFS-151920

Enginoğlu S, Arslan B (2020) Intuitionistic fuzzy parameterized intuitionistic fuzzy soft matrices and their application in decision-making. Comput Appl Math Article Number: 325 39(4):1–20. https://doi.org/10.1007/s40314-020-01325-1

Enginoğlu S, Çağman N (2020) Fuzzy parameterized fuzzy soft matrices and their application in decision-making. TWMS J Appl Eng Math 10(4):1105–1115. http://jaem.isikun.edu.tr/web/images/articles/vol.10.no.4/25.pdf

Enginoğlu S, Memiş S (2018a) Comment on fuzzy soft sets [The Journal of Fuzzy Mathematics 9(3), 2001, 589-602]. International Journal of Latest Engineering Research and Applications 3(9):1–9, https://www.ijlera.com/papers/v3-i9/1.201809134.pdf

Enginoğlu S, Memiş S (2018b) A configuration of some soft decision-making algorithms via *fpfs*-matrices. Cumhuriyet Sci J 39(4):871–881. https://doi.org/10.17776/csj.409915

Enginoğlu S, Memiş S (2018c) A review on an application of fuzzy soft set in multicriteria decision making problem [P. K. Das, R. Borgohain, International Journal of Computer Applications 38 (2012) 33–37]. In: Akgül M, Yılmaz I, İpek A (eds) International Conference on Mathematical Studies and Applications. Karaman, Turkey, pp 173–178

Enginoğlu S, Memiş S (2018) A review on some soft decision-making methods. In: Akgül M, Yılmaz I, İpek A (eds) International conference on mathematical studies and applications. Karaman, Turkey, pp 437–442

Enginoğlu S, Memiş S (2020) A new approach to the criteria-weighted fuzzy soft max-min decision-making method and its application to a performance-based value assignment problem. J New Results Sci 9(1):19–36. http://dergipark.org.tr/tr/pub/jnrs/issue/53974/709375

Enginoğlu S, Öngel T (2020) Configurations of several soft decision-making methods to operate in fuzzy parameterized fuzzy soft matrices space. Eskişehir Technical University Journal of Science and Technology A-Applied Sciences and Engineering 21(1):58–71. https://doi.org/10.18038/estubtda.562578

Enginoğlu S, Çağman N, Karataş S, Aydın T (2015) On soft topology. El-Cezerî J Sci Eng 2(3):23–38. https://doi.org/10.31202/ecjse.67135

Enginoğlu S, Memiş S, Arslan B (2018a) Comment (2) on soft set theory and uni-int decision-making [European Journal of Operational Research, (2010) 207, 848–855]. J New Theory (25):84–102. https://dergipark.org.tr/download/article-file/594503

Enginoğlu S, Memiş S, Öngel T (2018b) Comment on soft set theory and uni-int decision-making [european journal of operational research, (2010) 207, 848-855]. J New Results Sci 7(3):28–43. https://dergipark.org.tr/en/pub/jnrs/issue/40346/482909

Enginoğlu S, Ay M, Çağman N, Tolun V (2019a) Classification of the monolithic columns produced in Troad and Mysia Region ancient granite quarries in Northwestern Anatolia via soft decision-making. Bilge International Journal of Science and Technology Research 3(Special Issue):21–34. https://doi.org/10.30516/bilgesci.646126

Enginoğlu S, Erkan U, Memiş S (2019) Pixel similarity-based adaptive Riesz mean filter for salt-and-pepper noise removal. Multimed Tools Appl 78:35401–35418. https://doi.org/10.1007/s11042-019-08110-1

Enginoğlu S, Memiş S, Çağman N (2019c) A generalisation of fuzzy soft max-min decision-making method and its application to a performance-based value assignment in image denoising. El-Cezerî J Sci Eng 6(3):466–481. https://doi.org/10.31202/ecjse.551487

Enginoğlu S, Memiş S, Karaaslan F (2019d) A new approach to group decision-making method based on TOPSIS under fuzzy soft environment. J New Results Sci 8(2):42–52. https://dergipark.org.tr/tr/download/article-file/904374

Enginoğlu S, Aydın T, Memiş S, Arslan B (2021a) Operability-oriented configurations of the soft decision-making methods proposed between 2013 and 2016 and their comparisons. J New Theory (34):82–114. https://dergipark.org.tr/en/pub/jnt/issue/61070/896315

Enginoğlu S, Aydın T, Memiş S, Arslan B (2021b) SDM methods' configurations (2017–2019) and their application to a performance-based value assignment problem: A follow up study. Ann Optim Theory Pract 4(1):41–85. https://doi.org/10.22121/AOTP.2021.287404.1069

Erkan U, Gökrem L (2018) A new method based on pixel density in salt and pepper noise removal. Turk J Electr Eng Comput Sci 26(1):162–171. https://doi.org/10.3906/elk-1705-256
Erkan U, Gökrem L, Enginoğlu S (2018) Different applied median filter in salt and pepper noise. Comput Electr Eng 70:789–798. https://doi.org/10.1016/j.compeleceng.2018.01.019
Esakkirajan S, Veerakumar T, Subramanyam AN, PremChand CH (2011) Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter. IEEE Signal Process Lett 18(5):287–290. https://doi.org/10.1109/LSP.2011.2122333
Garg H, Arora R (2020) TOPSIS method based on correlation coefficient for solving decision-making problems with intuitionistic fuzzy soft set information. AIMS Math 5(4):2944–2966. https://doi.org/10.3934/math.2020190
Jiang Y, Tang Y, Chen Q, Liu H, Tang J (2010) Interval-valued intuitionistic fuzzy soft sets and their properties. Comput Math Appl 60(3):906–918. https://doi.org/10.1016/j.camwa.2010.05.036
Karaaslan F (2016) Intuitionistic fuzzy parameterized intuitionistic fuzzy soft sets with applications in decision making. Ann Fuzzy Math Inf 11(4):607–619. http://www.afmi.or.kr/papers/2016/Vol-11_No-04/PDF/AFMI-11-4(607-619)-H-150813-1R1.pdf
Kumar K, Garg H (2018) TOPSIS method based on the connection number of set pair analysis under interval-valued intuitionistic fuzzy set environment. Comput Appl Math 37(2):1319–1329. https://doi.org/10.1007/s40314-016-0402-0
Liu Y, Jiang W (2020) A new distance measure of interval-valued intuitionistic fuzzy sets and its application in decision making. Soft Comput 24(9):6987–7003. https://doi.org/10.1007/s00500-019-04332-5
Maji PK, Biswas R, Roy AR (2001) Fuzzy soft sets. J Fuzzy Math 9(3):589–602
Maji PK, Roy AR, Biswas R (2002) An application of soft sets in a decision making problem. Comput Math Appl 44(8–9):1077–1083. https://doi.org/10.1016/S0898-1221(02)00216-X
Memiş S (2021) A study on picture fuzzy sets. In: Çuvalcıoğlu G (ed) 7th IFS and Contemporary Mathematics Conference. Mersin, Turkey, pp 125–132
Memiş S, Enginoğlu S (2019) An application of fuzzy parameterized fuzzy soft matrices in data classification. In: Kılıç M, Özkan K, Karaboyacı M, Taşdelen K, Kandemir H, Beram A (eds) International Conferences on Science and Technology; Natural Science and Technology, Prizren, Kosovo, pp 68–77
Memiş S, Enginoğlu S, Erkan U (2019) A data classification method in machine learning based on normalised Hamming pseudo-similarity of fuzzy parameterized fuzzy soft matrices. Bilge Int J Sci Technol Res 3(Special Issue):1–8. https://doi.org/10.30516/bilgesci.643821
Memiş S, Arslan B, Aydın T, Enginoğlu S, Camcı Ç (2021a) A classification method based on Hamming pseudo-similarity of intuitionistic fuzzy parameterized intuitionistic fuzzy soft matrices. Journal of New Results in Science 10(2):59–76, https://dergipark.org.tr/en/pub/jnrs/issue/64701/981326
Memiş S, Enginoğlu S, Erkan U (2021) Numerical data classification via distance-based similarity measures of fuzzy parameterized fuzzy soft matrices. IEEE Access 9:88583–88601. https://doi.org/10.1109/ACCESS.2021.3089849
Min WK (2008) Interval-valued intuitionistic fuzzy soft sets. J Korean Inst Intell Syst 18(3):316–322. https://doi.org/10.5391/JKIIS.2008.18.3.316
Mishra AR, Rani P (2018) Interval-valued intuitionistic fuzzy WASPAS method: application in reservoir flood control management policy. Group Decis Negot 27(6):1047–1078. https://doi.org/10.1007/s10726-018-9593-7
Molodtsov D (1999) Soft set theory-first results. Comput Math Appl 37(4–5):19–31. https://doi.org/10.1016/S0898-1221(99)00056-5
Molodtsov D (2004) The theory of soft sets. URSS Publishers, Moscow, Russia ((**in Russian**))
Petchimuthu S, Garg H, Kamacı H, Atagün AO (2020) The mean operators and generalized products of fuzzy soft matrices and their applications in MCGDM. Comput Appl Math Article Number: 68 39(2):1–32. https://doi.org/10.1007/s40314-020-1083-2
Riaz M, Hashmi MR (2017) Fuzzy parameterized fuzzy soft topology with applications. Ann Fuzzy Math Inf 13(5):593–613. https://doi.org/10.30948/afmi.2017.13.5.593
Riaz M, Hashmi MR (2019) Linear Diophantine fuzzy set and its applications towards multi-attribute decision-making problems. J Intell Fuzzy Syst 37(4):5417–5439. https://doi.org/10.3233/JIFS-190550
Riaz M, Hashmi MR, Farooq A (2018) Fuzzy parameterized fuzzy soft metric spaces. J Math Anal 9(2):25–36. http://www.ilirias.com/jma/repository/docs/JMA9-2-3.pdf
Riaz M, Hashmi MR, Pamucar D, Chu Y (2021) Spherical linear Diophantine fuzzy sets with modeling uncertainties in MCDM. Comput Model Eng Sci 126(3):1125–1164. https://doi.org/10.32604/cmes.2021.013699
Senapati T, Shum KP (2019) Atanassov's interval-valued intuitionistic fuzzy set theory applied in KU-subalgebras. Discr Math Algorithms Appl 11(2):16. https://doi.org/10.1142/S179383091950023X

Şenel G (2016) A new approach to hausdorff space theory via the soft sets. Mathematical Problems in Engineering 2016:Article ID 2196743, pages 6, https://doi.org/10.1155/2016/2196743
Şenel G (2018) Analyzing the locus of soft spheres: Illustrative cases and drawings. European Journal of Pure and Applied Mathematics 11(4):946–957. https://doi.org/10.29020/nybg.ejpam.v11i4.3321
Sezgin A (2016) A new approach to semigroup theory I: Soft union semigroups, ideals and bi-ideals. Algebra Letters Article ID 3, 2016:1–46, http://scik.org/index.php/abl/article/view/2989
Sezgin A, Çağman N, Çıtak F (2019) α-inclusions applied to group theory via soft set and logic. Communications Faculty of Sciences University of Ankara Series A1 Mathematics and Statistics 68(1):334–352. https://doi.org/10.31801/cfsuasmas.420457
Srinivasan KS, Ebenezer D (2007) A new fast and efficient decision-based algorithm for removal of high density impulse noises. IEEE Signal Process Lett 14:189–192. https://doi.org/10.1109/LSP.2006.884018
Sulukan E, Çağman N, Aydın T (2019) Fuzzy parameterized intuitionistic fuzzy soft sets and their application to a performance-based value assignment problem. Journal of New Theory (29):79–88, https://dergipark.org.tr/tr/download/article-file/906764
Tang Z, Yang Z, Liu K, Pei Z (2016) A new adaptive weighted mean filter for removing high density impulse noise. In: Eighth International Conference on Digital Image Processing (ICDIP 2016), International Society for Optics and Photonics, vol 10033, pp 1003353/1–5, https://doi.org/10.1117/12.2243838
Thomas J, John SJ (2016) A note on soft topology. Journal of New Results in Science 5(11):24–29, https://dergipark.org.tr/tr/pub/jnrs/issue/27287/287227
Toh KKV, Isa NAM (2010) Noise adaptive fuzzy switching median filter for salt-and-pepper noise reduction. IEEE Signal Process Lett 17(3):281–284. https://doi.org/10.1109/LSP.2009.2038769
Torra V (2010) Hesitant fuzzy sets. Int J Intell Syst 25(6):529–539. https://doi.org/10.1002/int.20418
Ullah A, Karaaslan F, Ahmad I (2018) Soft uni-Abel-Grassmann's groups. European Journal of Pure and Applied Mathematics 11(2):517–536. https://doi.org/10.29020/nybg.ejpam.v11i2.3228
Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: From error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612. https://doi.org/10.1109/TIP.2003.819861
Xu Z, Yager RR (2006) Some geometric aggregation operators based on intuitionistic fuzzy sets. Int J Gen Syst 35(4):417–433. https://doi.org/10.1080/03081070600574353
Xue Y, Deng Y, Garg H (2021) Uncertain database retrieval with measure-based belief function attribute values under intuitionistic fuzzy set. Inf Sci 546:436–447. https://doi.org/10.1016/j.ins.2020.08.096
Zadeh LA (1965) Fuzzy sets. Inf Control 8(3):338–353. https://doi.org/10.1016/S0019-9958(65)90241-X