Informatica

An International Journal of Computing and Informatics

Special Issue:

   **End-user Privacy, Security, and
      Copyright issues**

Guest Editors:

   **Nilanjan Dey
   Surekha Borra
   Suresh Chandra Satapathy**

1977

Editorial Boards

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor from the Editorial Board can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the list of referees. Each paper bears the name of the editor who appointed the referees. Each editor can propose new members for the Editorial Board or referees. Editors and referees inactive for a longer period can be automatically replaced. Changes in the Editorial Board are confirmed by the Executive Editors.

The coordination necessary is made through the Executive Editors who examine the reviews, sort the accepted articles and maintain appropriate international distribution. The Executive Board is appointed by the Society Informatika. Informatica is partially supported by the Slovenian Ministry of Higher Education, Science and Technology.

Each author is guaranteed to receive the reviews of his article. When accepted, publication in Informatica is guaranteed in less than one year after the Executive Editors receive the corrected version of the article.

Editors' Introduction to the Special Issue on "End-user Privacy, Security, and Copyright issues"

**"All artists are protected by copyright...
and we should be the first to respect copyright"
— Billy Cannon**

With the rapid growth in the internet technology, end-user privacy has tremendous impact on the society and a firm's business. Technological solutions to minimize the digital piracy such as unofficial downloading of E-books, product models, songs, movies and commercial software, is the need of the hour for marketers. Investigating effective techniques and measures for reduction of piracy protects human resources and skills, and resolves the nightmare of marketers and investors. Further, it ensures fairness and accountability, and builds economic growth of a country.

This special issue is organized to promote the marketers and investors trust, and to enhance businesses by publishing the state-of-art research and developments in privacy, security and copyright concerns of multimedia content, with emphasis on organizational and end user computing. This special issue includes original research works; insightful research and practice notes, case studies, and surveys on vulnerabilities, requirements, attacks, challenges, reviews, mechanisms, tools, policies, emerging technologies and technological innovations for minimizing the digital piracy. This special issue includes six articles as follows:

In the first article **Ahmad *et al.*** proposed a novel hybrid watermarking method based on three different transforms: Discrete Wavelet Transform (DWT), Discrete Shearlet Transform (DST) and Arnold transform one after the other. To evaluate the proposed method authors used six performance measures namely peak signal to noise ratio (PSNR), mean squared error (MSE), root mean squared error (RMSE), signal to noise ratio (SNR), mean absolute error (MAE) and structural similarity (SSIM), which indicated a stable outcome under any type of attack and performs significantly better than state-of-the-art watermarking approaches. According to the results achieved, it is recommended to consider extending this hybrid method for other multimedia data such as video, text and audio.

Wang *et al.* in the second article proposed an improved gene expression programming algorithm based on niche technology of outbreeding fusion (OFN-GEP). This algorithm uses population initialization strategy of gene equilibrium for improving the quality and diversity of population. Further, the algorithm introduced the outbreeding fusion mechanism into the niche technology, to eliminate the kin individuals, fuse the distantly related individuals, and promote the gene exchange between the excellent individuals from niches. The authors then verified the effectiveness and competitiveness of the algorithm through function finding problems, and by relating it to literatures. The results being effective in convergence speed, quality of solution and in restraining

the premature convergence phenomenon; the OFN-GEP algorithm promises a great application value in solving practical problems related to privacy and security related issues such as data hiding, hash function generation and Boolean function evolution etc.

Secret key generation and establishment plays main role in launching the data sharing sessions, and is consequently used for authentication, confidentiality and integrity of data. In contrast to traditional key establishment protocols, where one user decides the key and communicates it to other user, the key agreement protocols involve all the users in the communication in key establishment process. **Sivaranjani and Surekha** in the third article presented an enhanced ID-based authenticated key agreement protocol using hybrid mixing of bilinear pairing and Malon-Lee approach for secure communication between two users. The results proved that the protocol stands secure and satisfies desired security properties at minimum time. The authors also extended the algorithms for multiple users.

Traffic and road accident safety are a big issue in every country. Data science, being assisting in analyzing different factors behind traffic and road accidents, **Prayag Tiwari *et al.*** in fourth article analyzed different clustering and classification techniques such as Decision Tree, Lazy classifier, and Multilayer perceptron classifier to classify datasets based on casualty class. Further, authors proposed clustering techniques which are k-means and hierarchical to cluster dataset. After analyzing dataset without and with clustering, the authors reported a noticeable improvement in the accuracy level by using clustering techniques on dataset compared to a dataset which was classified without clustering. Better results for reducing the accident ratio and improving the safety are reported after using hierarchical clustering as compared to k mode clustering techniques.

In the fifth article, **Siba and Ajanta** presented an enhanced distributed fault tolerant architecture and the related algorithms for connectivity maintenance in Wireless Sensor Networks (WSN) of various surveillance applications. The algorithms and hence the recovery actions are initiated based on fault diagnosis notifications, data checkpoints and state checkpoints in a distributed manner.

In the sixth article, **Thuong *et al.*** proposed a simple and less expensive hybrid approach based on the multiscale Curvelet analysis and the Zernike moment for detecting image forgeries. The results proved its effectiveness in copy-move detection.

As guest editors, we hope the research work covered under this special issue will be effective and valuable for multitude of readers/researchers. In addition, the technical standard and quality of published content is based on the strength and expertise of the submitted papers. We are grateful to the authors for their imperative research contribution to this issue and their patience

during the revision stages done by experts in the editorial board. We take this opportunity to give our special thanks to Prof. Matjaz Gams, Editor-in-chief of the Informatica: An International Journal Of Computing And Informatics, for all his support, and competence rendered to this special issue.

Nilanjan Dey

Surekha Borra

Suresh Chandra Satapathy

A Hybrid Wavelet-Shearlet Approach to Robust Digital Image Watermarking

Ahmad B. A. Hassanat[†], V. B. Surya Prasath[*], Khalil I. Mseidein[†], Mouhammd Al-awadi[†] and Awni Mansoar Hammouri[†]
[†]Department of Information Technology, Mutah University, Karak, Jordan
[*]Computational Imaging and VisAnalysis (CIVA) Lab, Department of Computer Science, University of Missouri-Columbia, USA
E-mail: prasaths@missouri.edu

Watermarking systems are one of the most important techniques used to protect digital content. The main challenge facing most of these techniques is to hide and recover the message without losing much of information when a specific attack occurs. This paper proposes a novel method with a stable outcome under any type of attack. The proposed method is a hybrid approach of three different transforms, discrete wavelet transform (DWT), discrete shearlet transform (DST) and Arnold transform. We call this new hybrid method SWA (shearlet, wavelet, and Arnold). Initially, DWT applied to the cover image to get four sub-bands, we selected the HL (High-Low) sub band of DWT, since HL sub-band contains vertical features of the host image, where these features help maintain the embedded image with more stability. Next, we apply the DST with HL sub-band, at the same time applying Arnold transform to the message image. Finally, the output that obtained from Arnold transform will be stored within the Shearlet output. To evaluate the proposed method we used six performance evaluation measures, namely, peak signal to noise ratio (PSNR), mean squared error (MSE), root mean squared error (RMSE), signal to noise ratio (SNR), mean absolute error (MAE) and structural similarity (SSIM). We apply seven different types of attacks on test images, as well as apply combined multi-attacks on the same image. Extensive experimental results are undertaken to highlight the advantage of our approach with other transform based watermarking methods from the literature. Quantitative results indicate that the proposed SWA method performs significantly better than other transform based state-of-the-art watermarking approaches.

Povzetek: Opisana je robustna metoda digitalnega vodnega tiska, tj. vnosa kode v sliko.

1 Introduction

With the exponential growth of digital contents in the internet, there is a strong need to protect these contents with more security automatically. This open online environment needs more efficient techniques to save original content creators, and author's rights. Digital watermarking systems can significantly contribute to protect information and files. Generally, people need an easy-to-use model to protect their files, texts, and images. The availability of automatic tools that provide such services for documents are currently restricted and difficult to use. Although, there are many previous works focused on certain solutions to solve this security problem, we certainly need more research to improve the efficiency of these existing methods. This paper provides a new hybrid approach based on some efficient transforms and provides an overview of the relevant issues and definitions related to digital image watermarking. Copyright nowadays is one of the most important research areas; digital watermarking is considered as one of the important automatic signal/image processing technique that enables us to hide our information behind a noisy signal. This signal may be image, audio or video etc.

An important extension of watermarking area is the image watermarking; here we embedded a watermark image within a cover image. The produced watermarked image is a combination of cover image and the watermark. The watermark is the information to be embedded, on the other hand the host or cover image is the signal where the watermark is embedded. In general, Watermarks can be classified depending on several criteria such as: domain that can be applied by the watermark, type the watermark used visibility of watermark to user and the application used to create the watermark. Figure 1 shows these broad classifications.

1.1 The usage of image watermarking

The growth of digital computing technology in recent times was the reason beyond the widespread use of digital media such as video digital, documents and digital images. As a result of the increase in speed of transmission and distribution it is easy to obtain digital content. Despite the abundance of digital products, however this technology lacks protection because of illegal use, and imitations. The protection of intellectual property rights for digital media has been the attention the focus of many researchers in the past. Using a digital watermark technology is a suc-
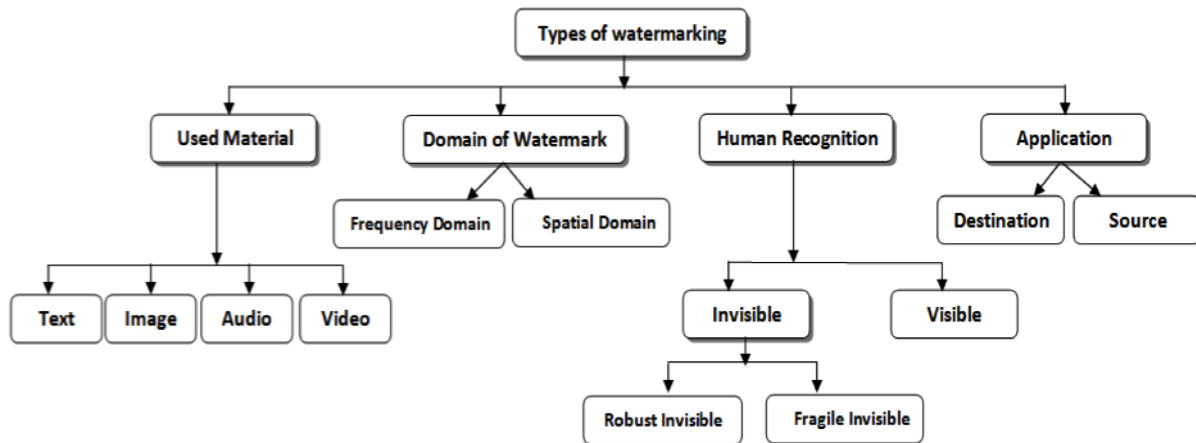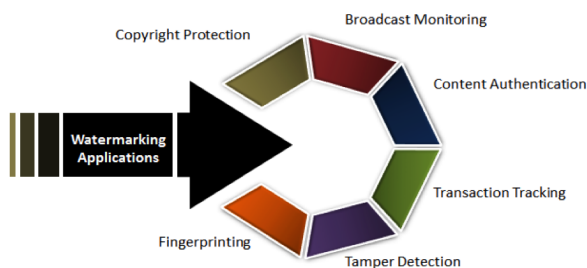
Figure 1: Main types of Watermarking techniques.

Figure 2: Application areas of watermarking techniques.

cessful choice to solve the digital content protection problem. There are, in general, six types of watermarking applications are presented; copyright protection, fingerprinting, broadcast monitoring, content authentication, transaction tracking, and tamper detection, these applications are mentioned in Figure 2.

Recently, the advance of editing capabilities and the wide availability due to internet penetration in the world, digital forgery has become easy, and difficult to prevent in general. Therefore, there is a need to protect end-user privacy, security, and copyright issues for content generators [13]. The application areas include biometrics, medical imagery, telemedicine, etc [4, 38, 39, 40, 41, 42].

Nowadays, digital watermarking is used mainly for the protection of copyrights. Moreover, it was used early to send sensitive confidential information across the communicative signal channels. Thus, applying watermark has been occupying the attention of researchers in the last few decades. As a consequence of tremendous growth of computer networks and data transfer over the web; huge amounts of multimedia data are vulnerable to unauthorized access, for e.g., web images which can easily be duplicated and distributed without eligibility. Image watermarking provides copyright protection for documents and multimedia in order to protect intellectual property and data security.

1.2 Contributions

In this work we will provide an overview five different transform based methods one of which is a new proposed hybrid transform approach. We considered the transform of an image using discrete cosine transform (DCT), discrete wavelet transform (DWT), DWT with DCT, and DWT with Arnold transform in addition to our proposed hybrid method which combines discrete shearlet transform (DST) with these previous transforms. Also, we considered adding attacks like crop image, salt and pepper noise, Gaussian noise, and etc. The schemes we are providing here are resilient to these types of attacks and novel in this scenario. In this paper, we will study in details the common methods that used to protect the original image after adding attacks. This work also presents a comprehensive experimental study of these transform based watermarking algorithms. Another important contribution of this study is a hybrid method which is based on a fusion of different transforms taking advantages of each transforms discriminability. Initially, we will design our new method; then we test it and compare our results with the results of the previous transform and combination of them. To ensure the efficiency of our hybrid, we will use different performance measures to quantitatively benchmark it on various test images and attacks.

In the digital image watermarking area, efficiency of any proposed algorithm can be evaluated based on a host of image and embedded image (message). One of the most common image is the "Lena" which is traditionally used as a host. Also, the copyright image has been used widely as a message. Apart from these images, there are other standard test images such as Cameraman, Baboon, Peppers, Airplane, Boat, Barbara, Elaine, Man, Bird, Couple, House, and Home which are widely used in benchmarking evaluations. With the development of a new watermarking technology, it is necessary to protect the images against mul-

tiple types of attack, such as compression, Gaussian filter, pepper and salt noise, median filter, cropping, resize, and rotation. The Watermark is said to be robust against some attack if we can detect the message after that particular attack. In our proposed approach, to ensure and reduce influence of watermarked image in any attack, shearlet transform is applied at the HL (high-low) sub-band, where this sub-band retains features of original image. shearlets generate a series of matrices that obtained from the HL sub-band. In this way, shearlets applied to specific pixels of the original image with these features represented in 61 different matrices. As a consequence, any applied attack is distributed to all of these matrices, and the probability of changing the value of the pixel that has been embedding by the attack is low. Based on this observation, our proposed hybrid SWA approach's results were stable or semi-static whatever the type of attack and its value. Finally, the Arnold transform was applied to the message image to achieve the best protection against the unauthorized change in pixel values.

We organized the rest of the paper as follows. Section 2 provides a brief overview of literature on watermarking with special emphasis on transform domain watermarking techniques. Section 3 provides a detail explanation of the proposed hybrid shearlet, wavelet, and Arnold (SWA) method. Section 4 provides detailed experimental results and discussions with Section 5 concluding the paper.

2 Literature review

Recently, research activities in image watermarking area has seen a lot of progress, and become more specialized. Some researchers were focusing on improving the properties of the watermark or applications, while others were focusing on improving the efficiency of the techniques used to embed, and extract the watermarked image, taking into consideration attacks. Watermarking techniques can be classified generally in to two broad domains; spatial domain, and transform domain.

2.1 Spatial domain watermarking

Classical techniques do watermarking in the spatial domain, where the message image is included by adjusting the pixel values of the original (host) image . Least Significant bit (LSB) method is considered one of the most important and most common examples of the use of the spatial domain techniques for the watermark. There are two main procedures to any watermarking technique model; embedding procedure and extraction procedure. For technique of LSB in embedding phase, the host image (original or cover) and the message image (watermark) being read, both images must be gray. However, this method suffers from the problem of an impaired ability to hide information [20], therefore it is easy to extract the image hidden within the original image by unauthorized persons, moreover, the quality of watermarking is not good when embed-

ding procedure, and extraction procedure are combined. In other words, the results achieved by spatial domain methods are not good enough, especially when the intensity of the pixel changed directly, which affects the value of this pixel.

Since typical spatial domain methods suffer from much vulnerability, many authors have tried to improve these. For example, [50] introduced two different methods to improve the technique of LSB, in the first method LSB substituted with a pseudo-noise (PN) sequence, and a second method that adds both together. However, this improvement also gives unsatisfactory results after adding any type of noise.

2.2 Transform domain watermarking

These techniques rely on hiding images in the transformed coefficients, thereby giving further information that helps secreting against any attack. As a consequence, majority of recent studies in the field of digital image watermarking use the frequency domain. Use of the frequency domain gives results better than the spatial domain in terms of robustness [28] . There are many transforms can be used for images watermarking based on frequency domain (FD), such as continuous wavelet transform (CWT), discrete cosine transform (DCT), short time Fourier transform (STFT), discrete wavelet transforms (DWT), Fourier transform, and combinations of DCT and DWT. We very briefly review these well-known transforms as they are relevant to the hybrid method proposed here (Section 3).

2.2.1 Discrete cosine transform (DCT)

One of the most important methods that based on the frequency domain is the discrete cosine transform (DCT). Typically in DCT based approaches, the image is represented in the form of a set of sinusoids with changeable magnitudes and frequencies, where the image is split into three different sections of frequencies; low frequency (LF), medium frequency (MF) and high frequency (HF). Data or message will be hidden in the medium frequency region; since it is considered to be the best place, whereas if the message is stored in low frequency regions it will be visible to the naked human eyes. Thus, for the areas of higher frequency, if the message is stored in this region, the resulting image will be distorted because this frequency spreads the biggest place of the block on the bottom right corner. Consequently, this will cause local deformation combined with the edges, thus the places where the areas are of the medium frequency, does not affect the quality of the image. The DCT is utilized in a number of earlier studies, see for e.g. [49] who proposed a new model based DCT technique within a specific scheme for the watermark, in which DCT increased the resistance against attacks, mainly JPEG compression attacks.

2.2.2 Discrete wavelet transform (DWT)

Discrete wavelet transform (DWT) is based on wavelets and is sampled discretely. The goal of using DWT is to convert an image from the spatial domain to the frequency domain in a locality preserving way. In DWT transform, coefficients separate the high and low frequency information in an image on a pixel-to-pixel basis. Original signal is divided into four mini signals - low-low (LL), low-high (LH), high-low (HL), and high-high(HH), these wavelets are generated from the original signal by dilations and translations. It is commonly used in various image processing applications, and in particular in the application of watermarking [48]. DWT can find appropriate area to embed the message efficiently, this means that the message will be invisible to the naked human eyes.

2.2.3 Joint transforms (DCT and DWT)

According to the advantages of the last two methods, namely DCT and DWT, we notice that each one is been characterized by certain positive aspects, however there are limitations that restrict their application efficiency. To improve the performances, several studies combined these two transforms based techniques, see for e.g. [8, 2, 5]. DCT achieves high results in the robust of hiding data, but it produces a distorted image, DWT produces high-quality image, but it achieves bad results with the addition of the attack. One of the proposed solutions to resolve this issue is a hybrid method combined two techniques; this hybrid method usually called joint DWT-DCT, the joint method is common use in signal processing application.

3 Proposed hybrid approach

The proposed approach is based on a hybrid approach combining three different transforms namely: discrete wavelet transform (DWT), discrete shearlet transform (DST) and Arnold transform. We named this a new hybrid model SWA, this abbreviation comes from the name of transforms utilized here; Shearlet, Wavelet, and Arnold transforms respectively. In this section, we will explain the salient points of these three transforms as well as our method of merging them together for the purposes of digital image watermarking.

3.1 Discrete wavelet transform (DWT)

As mentioned in Section 2.2.2, DWT is perhaps one of the most commonly used transform in the field of watermarking, where it is most widely used in image and videos. In the proposed model, we decompose the host image into four normal sub-bands according; LL, LH, HL, HH by level one DWT, the high frequency (LH, HL and HH) sub-bands are suitable for watermark embedding, as embedding watermark in LL sub-band causes the deterioration of image quality [47], the HL sub-band has been adopted for this

model since it contains a mixture of both high and low frequency contents.

3.2 Discrete shearlet transform (DST)

Recently, many of the multi-scale transforms appeared and became widely applied in image processing, such as the curvelets, contourlets and Shearlets. These transforms merge between the multi-scale analysis and directional wavelets to find an optimal directional representation. Generally these are adopted by building a pyramid waveform which consists of different directions as well different scales. The nature of transforms architecture enables us to use directions in multi-scale systems. For this reason, these transforms are widely used in many applications such as sparse image processing applications, operators decomposition, inverse problems, edge detection, and image restoration [19].

In this study, we utilize the discrete shearlet transform which has many advantages, the most important is that it is a simple mathematical construct, where it depends on the affine systems theory, the best solution for sparse multidimensional representation [35]. Further, Shear transform applied to a fixed function, and exhibit the geometric and mathematical properties like directionality, elongated shapes, scales, oscillations [31]. With host image A, the Shearlet transform for the transformed output image B are computed by Laplacian pyramid scheme and directional filtering according to equation below,

$$ShImg \rightarrow SH\{HostImg(a, s, t)\}, \tag{1}$$

where, a is the scale parameter; $a > 0$, s is the shear parameter or sometimes it called the direction; $s \in \mathbb{R}$, t is the translation parameter or sometimes it called the location; $t \in \mathbb{R}$; a, s, and t called Shearlet basis functions [33].

Shearlet transform are calculated by dilating, shearing and translation [33]. It is defined in equations below:

$$SH\{HostImg(a, s, t)\}$$
$$= \int HostImg(y)\, \Psi(x - y)\, dy$$
$$= HostImg \times \Psi(x), \quad (2)$$

where Ψ is a generating function is computed by equation,

$$\psi(x) = |detA(a, s)| - 0.5\, \Psi(A(a, s - 1)(x - t)) \tag{3}$$

Where a and s are geometrical transforms and dilation, and are 2×2 matrices calculated by:

$$\mathbf{a} = \begin{bmatrix} a & 0 \\ 0 & \sqrt{a} \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix}, \tag{4}$$

$$A(a, s) = \begin{bmatrix} a & s\sqrt{a} \\ 0 & \sqrt{a} \end{bmatrix}. \tag{5}$$

DST is an appropriate way for image watermarking, that it achieved high performance for determine the

directional features also the optimal localization [1]. In this paper, we used Finite Discrete Shearlet Transform[1] (FDST) for spread spectrum watermarking [24]. One of the main advantages of the FDST is that, it is only based on the Fast Fourier transform (FFT) for which very fast implementations are available. Further, using band-limited shearlets one can construct a Parseval frame that provides a simple and straightforward inverse shearlet transform. In the notations of the MATLAB software utilized here, the shearlet transform is applied to the image according to the following command:

```
[ST, Psi] = shearletTransformSpect(A,
numOfScales, realCoefficients);
```

This command returns two most important functions are ST and Psi, ST is the Shearlet coefficients as a three-dimensional matrix of size $M \times N \times \eta$ and Psi is the same size and contains the respective shearlet spectra. Each one of these functions stores 61 matrices and each matrix has the same size of the original image. According FDST toolbox described above, the matrix Psi(18) is considered good matrix for applying shearlet transform, where it is defined far from the external boundaries, which reduces the overlap.

3.3 Arnold transform (AT)

Arnold transform can be used to improve the security of the logo image that used in many watermarking applications [16, 55]. Suppose that M is n-dimensional matrix $(n \times n)$ which represents the original image. Arnold transform can be calculated from M as the following formula by equation,

$$\begin{bmatrix} x^* \\ y^* \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \ (mod \ n), \qquad (6)$$

where n is the size of the original image, (x, y) are the original pixel coordinates, (x^*, y^*) represents the coordinates of the pixel after applying Arnold transform. The principle of Arnold transform is based on modifying the location of the original pixels many times. This means the possibility of implementing it on the image periodically [46] which leads to the improvement of security in the watermarking image. Moreover, the mathematical characteristics [16] of the Arnold transform enables it to be used widely in the area of image watermarking. Note that the pixel location changes frequently, until it comes back to its original position after a number of iterations of Arnold transform, so the original image is recovered. The anti-Arnold transform which brings back to the original locations suffers from high time complexity that is needed in the reverse calculations [55].

---

[1]FDST is available as a MATLAB programs presented for applying DST. This software is available for free: http://www.mathematik.uni-kl.de/imagepro/software/ffst/
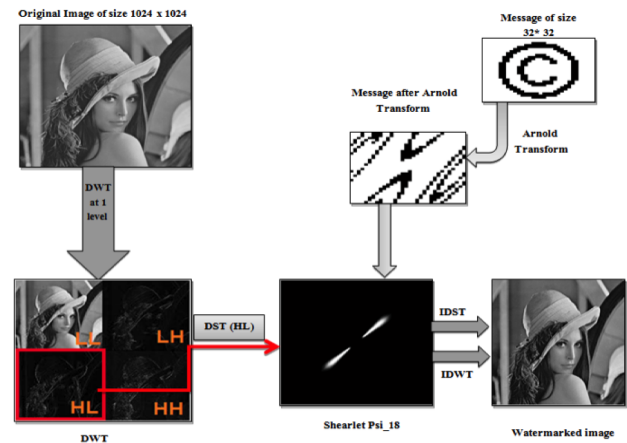
Figure 3: Watermark embedding algorithm of the proposed SWA method.

3.4 The hybrid SWA watermarking

A general digital image watermarking algorithm includes two procedures: (i) the first is the *embedding*, through this procedure the copyright image (message) hidden inside the original image (the host), the resulting image is called watermarked image [23], (ii) the second is the *extraction*, through this procedure the copyright image is extracted from the watermarked image. The proposed SWA watermarking model uses DWT, DST, and Arnold Transform for embedding procedure, in addition to using these for extraction procedure as well. Thus, our watermarking method consists of two major phases, (1) message embedding, and (2) message extraction, which we describe below in detail.

3.4.1 SWA embedding algorithm

Embedding algorithm procedure in the proposed SWA method consists of several steps, as shown in Figure 3. In the beginning, the host image (the original image) is being read, where its size is (1024×1024), also the copyright image is being read, where its size is (32×32).

The DWT applied to host image, where four of the sub-bands produced through applying this transform are; HH, HL, LH, and LL respectively. After that, the DST performed based on vertical frequency (HL) at level one DWT, From applying the DST produces two types of parameter (Ψ, ST) where each type consists of a 61 matrices, with the size of matrices to be 512×512.

The Arnold transform applied with copyright image, then each pixel of the resulting image after applying Arnold transform hide or embedded in the resulting host image obtained after applying DST. The embedding is performed at the matrix number (18) which resulted from applying DST at Ψ parameter. The size of this matrix is 512×512, this matrix divided into 32 blocks (the size of matrix (18) by copyright image (512/32), the pixel which obtained from applying Arnold embedded at the last pixel of each block

Figure 4: Watermark extracting algorithm of the proposed SWA method.

of the matrix (18), pixels 16×16, 16×32, etc. The following formula is calculated for all the values in the last Pixel from the matrix 18,

$$New\_value = SD(LPixl)^2/\alpha, \qquad (7)$$

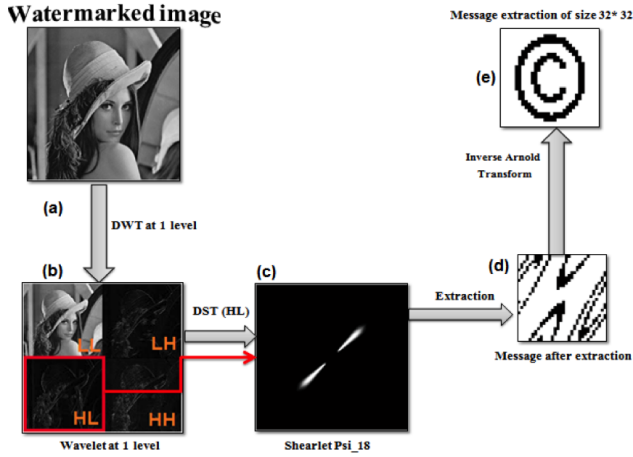where (SD) is the standard deviation, $LPixl$ is last pixel in matrix 18 and value of α is set to 10000. From copyright image after applying Arnold, when the pixel of the (New_value) is 1 then New_value is stored as it is, otherwise the negative of (New_value) stored. Finally, inverse discrete wavelet transform (IDWT) and inverse discrete shearlet transform (IDST) are performed to return the image to normal shape.

3.4.2 Extraction algorithm SWA model

The steps to extract the message computed by the inverse procedure can be described as follows. The proposed extraction algorithm for the SWA watermarking model is depicted in Figure 4. Initially, the watermarked image is being read, where its size is 1024×1024. DWT applied with watermarked image, After that, the DST performed based on horizontal frequency (HL) at level one DWT, the Extraction performed at the matrix number (18) which resulting from Ψ parameter. The size of this matrix is 512×512, this matrix divided into 32 blocks, when the value of the last pixel of each block of the matrix (18) is positive, then return 1 otherwise return 0, Thus, the previous Arnold transformed image was generated, and finally inverse Arnold transform was performed to get the copyright image.

The pseudo codes given in Algorithm 1 and Algorithm 2 summarizes the proposed embedding and extraction algorithms for the proposed SWA watermarking approach.

Algorithm 1 Embedding of SWA model

Input: Host image (1024×1024) and copyright image (32×32)

Output: Watermarked image of size (1024×1024)

1: Apply 2D_DWT (Host image)
2: Output: (LL, LH, HL, HH)
3: Apply DST (HL at step 2)
4: Output: 61 matrices (Ψ) of size 512×512
5: Get matrix Ψ_{18} (number 18 from matrix Ψ), block size 16×16
6: Apply Arnold Transform (Copyright image)
7: Let $k0 = - (std(Host)^2)/\alpha$
8: Let $k1 = (std(Host)^2)/\alpha$
9: Let $\Psi_{18}\_W(x, y) = \Psi_{18}(x, y)$
10: For each pixel from copyright image after Arnold transform (message vector (pixel)).
11: If (message vector (pixel) is 0) then $\Psi_{18}\_W(y + blocksize - 1, x + blocksize - 1) = k0$
12: Else $\Psi_{18}\_W(y + blocksize - 1, x + blocksize - 1) = k1$
13: If (message vector is last pixel) Go To step 15
14: Next pixel (message vector (pixel++)) Go To step 11
15: Let $\Psi_{18}(x, y) = \Psi_{18}\_W(x, y)$
16: Let C= IDST (Ψ, ST)
17: Let Watermarked image= 2D_IDWT (LL, LH, C, HH)
18: Output Watermarked image of size (1024×1024)
19: Calculate Performance evaluation measures for (Watermarked image)

---

4 Experimental results

4.1 Setup and error metrics

Our experimental results were conducted on a Toshiba laptop with Intel (R) Core i5-2450M processor with CPU 2.50 GHz, RAM 6.00 GB in MATLAB environment. Most of the studies in the digital image watermarking literature are based on standard images such as Lena, Baboon, Boat, Cameraman, Peppers and Barbara images etc. These images taken from USC-SIPI miscellaneous database[2], it consist of 44 images, 16 colors and 28 monochromes. The sizes of images are 256×256, 512×512, and 1024×1024.

We gauge the performance of different watermarking techniques quantitatively by using six most common error metrics utilized in image processing literature which are given below.

– **Mean Square Error (MSE)**: MSE [27, 52] between original image and watermarked image is calculated using the formula:

$$MSE = \frac{1}{N} \sum_i \sum_j (f(i,j) - g(i,j))^2,$$

where sum j and k is taken over all the pixels in the image, N is the total number of pixels in the image.

---

[2]http://sipi.usc.edu/database/database.php?volume=misc

Algorithm 2 Extraction of SWA model

Input: Watermarked image (1024×1024)

Output: Message image (32×32)

1: Apply 2D_DWT (Watermarked image)
2: Output: (LL_W, LH_W, HL_W and HH_W).
3: Apply DST (HL_W at step 2).
4: Output: 61 matrices (Ψ_W), 61 matrices (ST_W) size of each matrix is 512×512
5: Get matrix Ψ_{18}_W (number of 18 from matrix Ψ_W), block size 16×16
6: For each last element from block at Ψ_{18}_W matrix
7: If ($\Psi_{18}\_W(y + blocksize - 1, x + blocksize - 1) = negative value$) then (message vector(pixel)=0)
8: Else (message vector(pixel)=1)
9: If (Ψ_{18}_W is last block) Go To step 10.
10: Message_EX= reshape(message vector(pixels))
11: Go To step 6
12: Apply Inverse Arnold Transform (Message_EX) .
13: Output Message_EX image.
14: Calculate Performance evaluation measures for (Message_EX image).
15: End

It is worth mentioning here that a good watermarking system should satisfy minimum MSE.

- **Root Mean Square Error (RMSE)**: RMSE [9] equals the square root of Mean Square Error ($MSE^{0.5}$); it can also be calculated using the formula,

$$RMSE = \sqrt{\frac{1}{N} \sum_i \sum_j (f(i,j) - g(i,j))^2}.$$

It is worth mentioning here that a good watermarking system should satisfy minimum RMSE.

- **Peak Signal to Noise Ratio (PSNR)**: PSNR [56] is used for measuring the quality of the watermarked image and defined using the formula:

$$PSNR =$$
$$10 \log_{10} \frac{(max)^2}{\frac{1}{m \times n} \sum_j \sum_k (f(i,j) - g(i,j))^2},$$

where, $m \times n$ is the image size, (max) is the maximum value of the pixels values in the image f is the host image, g is the watermarked images. It is worthily mentioned here that a higher value of PSNR (dB) is good because it means that the ratio of signal to noise is higher.

- **Signal to Noise Ratio (SNR)**: SNR [59] is mainly used to measure the sensitivity of the image. It measures the power of a signal strength relative to the background noise. It is calculated by the formula:

$$SNR = \frac{P_{signal}}{P_{noise}}$$

Higher values of SNR (dB) shows better performance.

- **Structural Similarity (SSIM)**: Another popular measure for similarity comparison between two images is SSIM [54] with values in the range of $[0, 1]$, where 1 is acquired when two images are identical. Mean structural similarity index is in the range $[0, 1]$ and is known to be a better error metric than traditional signal to noise ratio [54]. It is the mean value of the structural similarity (SSIM) metric[3]. The SSIM is calculated between two windows ω_1 and ω_2 of common size $N \times N$, and is given by,

$$SSIM(\omega_1, \omega_2)$$
$$= \frac{(2\mu_{\omega_1}\mu_{\omega_2} + c_1)(2\sigma_{\omega_1\omega_2} + c_2)}{(\mu_{\omega_1}^2 + \mu_{\omega_2}^2 + c_1)(\sigma_{\omega_1}^2 + \sigma_{\omega_2}^2 + c_2)}$$

where μ_{ω_i} the average of ω_i, $\sigma_{\omega_i}^2$ the variance of ω_i, $\sigma_{\omega_1\omega_2}$ the covariance, and c_1, c_2 stabilization parameters. The MSSIM value near 1 implies the optimal denoising capability of a method and we used the default parameters.

- **Mean Absolute Error (MAE)**: This measure is used to compute the average magnitude of the errors in a set of forecasts, without considering their direction. It uses to compute the accuracy with continuous values [25]. MAE is calculated using the formula:

$$MAE = \frac{1}{N} \sum_i \sum_j |f(i,j) - g(i,j)|$$

Lower values of MAE indicates better performance.

4.2 Attacks

With the development of our proposed SWA watermarking technology, it is necessary to protect the images against several different types of attacks, which they are exposed to, such as compression, Gaussian filter, pepper and salt noise, median filter, cropping, resize, and rotation attacks. A watermarking method is said to be robust against some attack if we can recover the original image after that particular attack. Some of the common types of attack will be briefly explained below.

- **JPEG compression attack**: JPEG compression [60] and [3], is aimed to reduce the size of an image, this resize operation enables the users to upload and download images efficiently. Moreover, compression reduces the complexity time of sending multimedia material.

- **Salt and pepper noise attack**: Another type of attacks is Salt and pepper noise [12] and [30], which affected watermarking images. This kind of noise is not

---

[3]We use the default parameters for SSIM and the MATLAB code is available online at https://ece.uwaterloo.ca/ z70wang/research/ssim/.

only damaging the image through software used for that purpose, but it can also happen during the process of image acquisition, by affecting the used camera, or in the stage of storing in memory. Regarding to 8 bit grayscale image, salt and pepper noise randomly alter the pixel value to either minimum (0) or maximum ($2^8 - 1 = 255$).

– **Cropping attack**: Image cropping is the operation of cutting the outer part of the original image with specific measurements, to improve the image or get rid of the excess parts. This operation can be carried out using different block size ratio [60].

– **Median filter attack**: Mean filter is an image processing technique which changes the center value of a block (for example 3×3 block) of an image with the median value of the pixels, this leads to smooth image and reduces the variation of the density between any pixel and its neighbors. The main challenge of median filter attack [32], is the use of small number of pixels to reconfigure the original image

– **Rotation attack**: The principle of image rotation is the inverse transformation for each pixel of the original image, so the image is calculated using the interpolation [43]. Rotation attack could affect the image to various degrees based on the image rotation angle [21].

– **Resize (scaling) attack**: Image scaling Usually used to resize digital images for printing purposes, which does not change the actual pixels in the original image [14], But to keep the image without affected by scaling, users must take into account not to reduce the original image to less than half, or not to enlarge it to more than double as proven in studies [36]. Generally there two types of scaling attack; down-scaling attack and up-scaling attack [36].

– **Gaussian filter noise**: Gaussian filter is a geometric method which modify the original image by removing high frequency pixels [3], to reduce the amount of pixel density variation with the adjacent pixels, this can be done according to a specific formula depends on variance and mean [51]. The noise which caused by Gaussian filter resulted from adding random noise values to the actual pixel.

4.3 Detailed results

Our main experiments are divided into three parts each consist of multiple sub-experiments. The first part is comparison of our proposed SWA watermarking method with four common transforms based watermarking approaches available in the literature.

1. **DWT** - [57, 15, 37, 48, 58], and [7],

2. **DCT** - [44, 34, 49], and [10].

3. **DWT_DCT_ Joint** [2, 28, 5], and [6],

4. **DWT_ Arnold** - [55, 16] and [29].

All these techniques were implemented in MATLAB. In these experiments, we verify the performance improvements when we applied the proposed algorithm.

The second part is a comparison of quantitative results of our proposed SWA model, and four different approaches with seven attacks on *Lena* image based on PSNR (dB), MSE, RMSE, SNR (dB), SSIM, and MAE error metrics. We also provide the ranking of these methods with respect to each of these error metrics.

The third part is a new way to compare different watermarking methods performances by combining multiple attacks together. These multi-attacks are applied on *Lena* image and we compute the PSNR (dB), and SSIM error metrics as representative benchmarking of four methods from the literature with our proposed SWA watermarking method.

4.3.1 Comparison with other methods on multiple standard test images

Table 1 shows a comparison between our proposed approach with four of state-of-the-art watermarking approaches on multiple USC-SIPI standard test images. In order to test the advantage of the proposed SWA approach against these methods from the literature, we utilized six of the standard test images widely used (as a host image), and the copyright image which was used as embedded image.

As can be seen by comparing the different error metrics reported in Table 1 (with no attacks), the proposed SWA model achieved the best results across different images. The MSE measure with Boat image, the percentage of squares errors between the original image and the image, after embedding is (0.0015), and this low percentage indicates that our proposed approach obtains good result for the embedding step. Nevertheless, DWT_DCT Joint method obtained a decent result, it achieved (6.2047) whereas DWT achieved (104.8672), which shows that this method is not satisfactory. Similarly, outcomes of PSNR, which refers to the ratio of the noise signal of the image, as when the values of this measure are high, the quality of the image after embedding is good, the proposed method got the highest result (76.4063) and the lowest value presented when applying DWT method, the result was (27.9584). The MAE, which expresses the absolute error value between the original image and the embedded image, whenever the value of this measure was low the quality watermarked image is better. The proposed method got least absolute error value (0.0797) for the Lena image, while DWT method achieved the highest absolute error (8.1849). The SNR, which indicates confusion of the signals between the original image and the watermarked image, the perfect result was obtained with the proposed method (0.000) across all test images. The SSIM, which refers to the amount of similarity between the structure of the original image and

Image/Methods	MSE	RMSE	PSNR	MAE	SNR	SSIM
Lena						
DWT	105.0586/5	10.2498/5	27.9505/5	8.1849/5	-0.0258/5	0.5839/5
DCT	25.4757/4	5.0473/4	34.1035/4	4.0007/4	-0.0058/4	0.8388/4
DWT_DCT_Joint	5.9893/3	2.4473/3	40.3910/3	0.9294/2	-0.0014/2	0.9532/3
DWT_ Arnold	4.6186/2	2.0903/2	41.7607/2	1.2615/3	0.0018/3	0.9721/2
Our SWA model	**0.0085/1**	**0.0919/1**	**68.8949/1**	**0.0797/1**	**0.0000/1**	**1.0000/1**
Baboon						
DWT	104.9922/5	10.2466/5	27.9532/5	8.1817/5	-0.0247/4	0.8225/5
DCT	57.1129/4	7.5573/4	30.5975/4	5.0679/4	-0.0060/3	0.9160/4
DWT_DCT_Joint	11.5060/2	3.3920/2	37.5556/2	1.1783/2	-0.0017/2	0.9915/2
DWT_ Arnold	34.6575/3	5.8871/3	32.7668/3	4.1454/3	0.0348/5	0.9650/3
Our SWA model	**0.0199/1**	**0.1410/1**	**65.1835/1**	**0.1152/1**	**0.0000/1**	**1.0000/1**
Barbara						
DWT	103.9729/5	10.1967/5	27.9956/5	8.1256/5	-0.0303/5	0.6931/5
DCT	28.8601/4	5.3722/4	33.5618/4	4.1810/4	-0.0074/3	0.8816/4
DWT_DCT_Joint	10.0737/2	3.1739/2	38.1329/2	1.0764/2	-0.0019/2	0.9694/2
DWT_ Arnold	20.8456/3	4.5657/3	34.9747/3	2.6838/3	0.0269/4	0.9633/3
Our SWA model	**0.0079/1**	**0.0890/1**	**69.1725/1**	**0.0702/1**	**0.0000/1**	**1.0000/1**
Cameraman						
DWT	101.1671/5	10.0582/5	28.1144/5	8.0435/5	-0.0245/5	0.5771/5
DCT	23.7020/4	4.8685/4	34.4170/4	3.8271/4	-0.0051/3	0.8375/4
DWT_DCT_ Joint	5.1074/2	2.2599/2	41.0828/	0.8677/2	-0.0012/2	0.9482/2
DWT_ Arnold	20.5353//3	4.5316/3	35.0398/3	2.2142/3	0.0096/4	0.9299/3
Our SWA model	**0.0161/1**	**0.1268/1**	**66.0995/1**	**0.1015/1**	**0.0000/1**	**0.9999/1**
Peppers						
DWT	102.5565/5	10.1270/5	28.0552/5	8.0597/5	-0.0319/5	0.6091/5
DCT	27.3814/4	5.2327/4	33.7902/4	4.1240/4	-0.0075/4	0.8457/4
DWT_DCT_Joint	9.7729/2	3.1262/2	38.2646/2	1.0404/2	-0.0019/2	0.9603/2
DWT_ Arnold	11.4553/3	3.3846/3	37.5747/3	2.0799/3	0.0036/3	0.9392/3
Our SWA model	**0.0089/1**	**0.0943/1**	**68.6709/1**	**0.0823/1**	**0.0000/1**	**1.0000/1**
Boat						
DWT	104.8672/5	10.2405/5	27.9584/5	8.1716/5	-0.0214/5	0.6388/5
DCT	27.5275/4	5.2467/4	33.7671/4	4.1335/4	-0.0051/4	0.8586/4
DWT_DCT_Joint	6.2047/2	2.4909/2	40.2376/2	0.9238/2	-0.0011/2	0.9478/3
DWT_ Arnold	8.7440/3	2.9570/3	38.7477/3	1.7856/3	0.0042/3	0.9615/2
Our SWA model	**0.0015/1**	**0.0387/1**	**76.4063/1**	**0.0323/1**	**0.0000/1**	**1.0000/1**

Table 1: Comparison results of our proposed SWA model and four different approaches without attack on embedded copyright image. We show different error metric values for each method along with ranks. Best results are given in **boldface**.
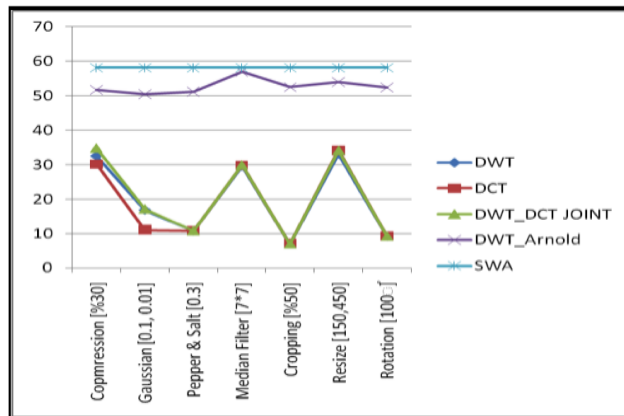
Figure 5: Graph showing the comparison of our proposed SWA model with other four methods based on PSNR (dB) values for the value of each type of attack on Lena image and copyright image.
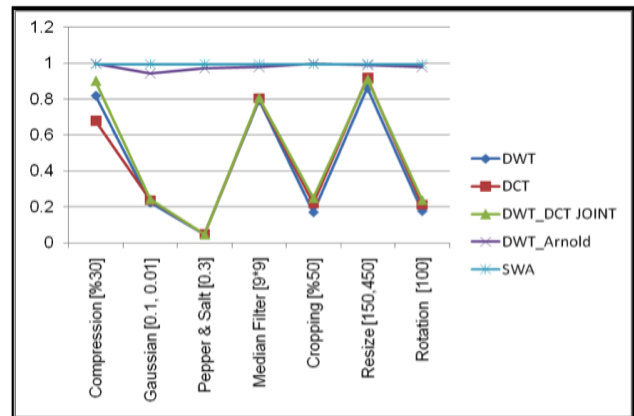

Figure 6: Graph showing the comparison of our proposed SWA model with other four methods based on SSIM values for the value of each type of attack on Lena image and copyright image.

the image after embedding, the proposed method got the highest results (1.000) or close to optimal.

4.3.2 Comparison of different attacks on Lena image with various error metrics

We next compare our proposed SWA model with other approaches with different attack methods under various error metrics. The watermarked image is exposed to several types of attacks with different parameter values as indicated appropriately. The compression attack expresses the ratio maintaining the image quality, for instant when the compression ratio is 10% which means that 90% of the image quality may be lost, with the maintaining 10% of the quality of the image, while this may not be discernible to the naked human eyes. Similarly, when the compression ratio is 90% means the 90% of the image quality has been preserved. For Gaussian noise the parameters indicate the mean and standard deviations indicating the amount of noise added to the image. Pepper and salt is a multiplicative noise with the probabilities given. Mean filtering is applied using the window size. Cropping uses the percentage of crop applied to the image. Resize is given in terms of the final size values. Rotation is performed at the angles given.

Figure 5 and Figure 6 shows the comparison of different attacks with respect to the PSNR (dB), and SSIM error metrics respectively. From the figures it is clear that the proposed SWA model performs the best with DWT_Arnold performing the next best. The results with other DWT, DCT, DWT_DCT_Joint perform rather poorly.

Table 2 shows the PSNR (dB) values, which is used to measure the quality of the image after embedding, comparing all transform techniques with our proposed method. When (10%) compression ratio is applied, the proposed SWA model achieved a high PSNR = 58.0975 dB value, and the worst result is achieved by DCT, with PSNR = 30.7130 dB. Except under median filtering our proposed SWA outperforms the other transform based approaches

with many different types of attacks with various parameter settings.

Table 3 examines the impact of applying different methods on seven types of attack using Lena image with respect to the mean square error (MSE) measure. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.1016) error value. Although, it is clear from the results that DWT_ Arnold algorithm was better than our when we apply compression attack, except 10% compression ratio. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_ Arnold method with median and cropping attacks, but, with a little difference.

Similar observations can be made about RMSE metric on different attacks. Table 4 examines the impact of applying different methods on seven types of attack using Lena image with respect to the root mean square error (RMSE) measure. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.3187) error value. Although, it is clear from the results that DWT_Arnold algorithm was better than our when we apply compression attack, except (10%) compression ratio. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_Arnold method with median and cropping attacks, but, with a little difference.

Next, Table 5 investigates the impact of applying different methods on seven types of attack using Lena image with respect to signal to noise ratio (SNR) error metric. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (-0.0629) error value. Although, the DWT_Arnold algorithm was better than ours when we apply compression attacks in (80%), and (90%) compression ratio, and

Attack/Methods		DWT	DCT	DWT_DCT_Joint	DWT_Arnold	Our SWA
Compression	[10]	30.7969/5	30.7130/4	30.9133/3	51.7164/2	**58.0975/1**
	[30]	32.4773/4	30.1909/5	34.7766/3	51.6497/2	**58.0975/1**
	[50]	30.4796/5	32.7898/4	35.5173/3	51.6591/2	**58.0975/1**
	[70]	28.0910/5	32.7397/4	35.3425/3	51.6591/2	**58.0975/1**
	[90]	27.4059/5	33.4065/4	38.0909/3	51.6591/2	**58.0975/1**
Gaussian Noise	[0.03, 0.003]	22.6204/5	23.7255/4	24.0203/3	51.0994/2	**58.0975/1**
	[0.09, 0.009]	17.4078/5	17.6835/4	17.7568/3	50.6410/2	**58.0975/1**
	[0.1, 0.01]	16.7995/4	11.0168/5	17.1067/3	50.3720/2	**58.0975/1**
	[0.3, 0.03]	8.1811/5	8.1858/4	10.1418/3	50.5888/2	**58.0975/1**
	[0.5, 0.05]	7.2834/5	7.4875/4	7.4862/3	50.7320/2	**58.0975/1**
Pepper and Salt	[0.01]	23.5386/5	24.9051/4	25.3305/3	51.7357/2	**58.0975/1**
	[0.05]	18.0867/5	18.4106/4	18.5091/3	51.5284/2	**58.0975/1**
	[0.09]	15.7175/5	15.9571/4	15.9726/3	51.2608/2	**58.0975/1**
	[0.3]	10.7093/5	10.7678/4	10.7559/3	51.1582/2	**58.0975/1**
	[0.5]	8.5105/5	8.5513/3	8.5415/4	51.1078/2	**58.0975/1**
Median Filter	[1×1]	27.9228/5	34.1035/4	40.3910/3	**58.7254/1**	58.0975/2
	[3×3]	33.3072/5	34.9873/4	36.1247/3	**58.6774/1**	58.0975/2
	[5×5]	31.3862/5	31.9093/4	32.1208/3	**58.4451/1**	58.0975/2
	[7×7]	29.5287/5	29.8056/4	29.8890/3	56.9006/2	**58.0975/1**
	[9×9]	28.2176/5	28.4383/4	28.4967/3	51.8829/2	**58.0975/1**
Cropping	[10]	5.7366/4	5.7368/3	5.7368/3	51.8929/2	**58.0975/1**
	[30]	6.11805/5	6.1198/4	6.1205/3	52.4585/2	**58.0975/1**
	[50]	6.9294/5	6.9355/4	6.9379/3	52.4245/2	**58.0975/1**
	[70]	8.4311/5	8.4487/4	8.4543/3	52.1612/2	**58.0975/1**
	[90]	13.0356/5	13.1217/4	13.1446/3	52.2802/2	**58.0975/1**
Resize	[100,300]	31.0705/5	31.2293/2	31.1997/3	52.3018/2	**58.0975/1**
	[150,450]	33.0241/5	34.1921/3	34.1008/4	53.8430/2	**58.0975/1**
	[200,600]	33.6675/5	36.1633/4	36.2744/3	55.7881//2	**58.0975/1**
	[250,750]	33.9085/5	36.9390/4	37.9065/3	57.1619/2	**58.0975/1**
	[300,900]	33.7695/5	36.8726/4	39.1607/3	57.2298/2	**58.0975/1**
Rotation	[25]	8.2973/5	8.3022/4	8.3050/3	52.9275/2	**58.0975//1**
	[70]	8.5009/5	8.5064/4	8.5105/3	51.7745/2	**58.0975/1**
	[100]	9.1657/5	9.1750/4	9.1810/3	52.3239/2	**58.0975/1**
	[200]	8.2427/5	8.2487/4	8.2517/3	52.8147/2	**58.0975/1**
	[300]	8.0150/5	8.0195/4	8.0217/3	52.4019/2	**58.0975/1**

Table 2: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on PSNR (dB) metric with ranks. Best results are given in **boldface**.

Attack/Methods		DWT	DCT	DWT_DCT_Joint	DWT_Arnold	Our SWA
Compression	[10]	5.2654/5	4.6638/3	5.2194/4	0.1367/2	**0.1016/1**
	[30]	4.5640/4	6.3203/5	3.2932/3	**0.0928/1**	0.1016/2
	[50]	5.8867/5	4.5083/4	3.0358/3	**0.0918/1**	0.1016/2
	[70]	7.9038/5	4.6638/4	3.1663/3	**0.0879/1**	0.1016/2
	[90]	8.7213/5	4.3617/4	2.2306/3	**0.0879/1**	0.1016/2
Gaussian Noise	[0.03, 0.003]	15.1372/5	13.3999/4	12.9221/3	0.4395/2	**0.1016/1**
	[0.09, 0.009]	28.2383/5	27.4299/4	27.2864/3	0.6592/2	**0.1016/1**
	[0.1, 0.01]	30.4388/5	29.7588/4	29.5494/3	0.6592/2	**0.1016/1**
	[0.3, 0.03]	70.9105/3	71.0183/5	71.0116/4	0.7363/2	**0.1016/1**
	[0.5, 0.05]	99.7833/3	99.8240/4	99.9451/5	0.7188/2	**0.1016/1**
Pepper and Salt	[0.01]	9.32670/5	5.21240/4	2.2142/3	0.1299/2	**0.1016/1**
	[0.05]	14.1145/5	10.1257/4	7.3025/3	0.3057/2	**0.1016/1**
	[0.09]	18.8807/5	14.9787/4	12.2862/3	0.3584/2	**0.1016/1**
	[0.3]	44.0520/5	41.2690/4	38.8275/3	0.4590/2	**0.1016/1**
	[0.5]	67.9147/5	65.6387/4	64.3678/3	0.4834/2	**0.1016/1**
Median Filter	[1×1]	88.1478/5	4.0007/4	0.9288/3	**0.0879/1**	0.1016/2
	[3×3]	3.96670/5	3.0514/4	2.2597/3	**0.0889/1**	0.1016/2
	[5×5]	4.18210/5	3.6768/4	3.3332/3	**0.0938/1**	0.1016/2
	[7×7]	4.86260/5	4.4538/4	4.2133/3	0.1338/2	**0.1016/1**
	[9×9]	5.56320/5	5.1595/4	4.9448/3	0.4248/2	**0.1016/1**
Cropping	[10]	123.8123/5	123.7751/4	123.7279/3	0.2656/2	**0.1016/1**
	[30]	114.3665/5	114.0056/4	113.6263/3	**0.0977/1**	0.1016/2
	[50]	96.12130/5	95.12180/4	94.06350/3	**0.0684/1**	0.1016/2
	[70]	69.48680/5	67.5075/4	65.7319/3	**0.0820/1**	0.1016/2
	[90]	29.8192/5	26.4739/4	23.8796/3	**0.0498/1**	0.1016/2
Resize	[100,300]	4.1504/5	3.9113/3	3.9644/4	0.3857/2	**0.1016/1**
	[150,450]	3.8638/5	2.9439/3	3.0264/4	0.2705/2	**0.1016/1**
	[200,600]	3.9695/5	2.5879/4	2.5079/3	0.1729/2	**0.1016/1**
	[250,750]	3.9839/5	2.5813/4	2.1687/3	0.1260/2	**0.1016/1**
	[300,900]	4.1283/5	2.7542/4	1.9236/3	0.1240/2	**0.1016/1**
Rotation	[25]	81.9561/5	81.8842/4	81.8376/3	0.3340/2	**0.1016/1**
	[70]	80.5659/5	80.4934/4	80.4498/3	0.4355/2	**0.1016/1**
	[100]	74.5938/5	74.5079/4	74.4594/3	0.3838/2	**0.1016/1**
	[200]	84.3401/5	84.2893/4	84.2609/3	0.3428/2	**0.1016/1**
	[300]	85.6902/5	85.6161/4	85.5812/3	0.3770/2	**0.1016/1**

Table 3: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on MSE metric with ranks. Best results are given in **boldface**.

Attack/Methods		DWT	DCT	DWT_DCT_Joint	DWT_Arnold	Our SWA
Compression	[10]	7.3665/5	5.9055/3	7.2874/4	0.3698/2	**0.3187/1**
	[30]	3.1988/3	7.9195/5	4.6710/4	**0.3046/1**	0.3187/2
	[50]	7.7434/4	5.8715/5	4.2892/3	**0.3030/1**	0.3187/2
	[70]	10.0878/5	5.9055/4	4.3764/3	**0.2965/1**	0.3187/2
	[90]	10.9318/5	5.4691/4	3.1893/3	**0.2965/1**	0.3187/2
Gaussian Noise	[0.03, 0.003]	18.9023/5	16.7106/4	16.1062/3	0.6629/2	**0.3187/1**
	[0.09, 0.009]	34.4540/5	33.3613/4	33.1316/3	0.8119/2	**0.3187/1**
	[0.1, 0.01]	36.9761/5	35.9851/4	35.7148/3	0.8149/2	**0.3187/1**
	[0.3, 0.03]	79.7524/5	79.6753/4	79.6042/3	0.8581/2	**0.3187/1**
	[0.5, 0.05]	108.0238/3	108.044/4	108.0808/5	0.8478/2	**0.3187/1**
Pepper and Salt	[0.01]	16.8436/5	14.2325/4	13.8030/3	0.3604/2	**0.3187/1**
	[0.05]	31.8880/5	30.5307/4	30.4602/3	0.5529/2	**0.3187/1**
	[0.09]	41.6718/5	40.5566/4	40.6133/4	0.5978/2	**0.3187/1**
	[0.3]	74.6177/5	74.4625/4	74.0317/3	0.6775/2	**0.3187/1**
	[0.5]	95.9893/5	95.6146/4	95.8191/4	0.6953/2	**0.3187/1**
Median Filter	[1×1]	10.2124/5	5.0473/4	2.4467/3	**0.2965/1**	0.3187/2
	[3×3]	5.51580/5	4.5591/4	3.9998/3	**0.2981/1**	0.3187/2
	[5×5]	6.90620/5	6.4979/4	6.3416/3	**0.3062/1**	0.3187/2
	[7×7]	8.55710/5	8.2786/4	8.1997/3	0.3658/2	**0.3187/1**
	[9×9]	9.93440/5	9.6901/4	9.6252/3	0.6518/2	**0.3187/1**
Cropping	[10]	132.2546/5	132.2520/4	132.2506/3	0.5154/2	**0.3187/1**
	[30]	126.5733/5	126.4560/3	126.5357/4	**0.3125/1**	0.3187/2
	[50]	115.2859/5	115.2027/4	115.1709/3	**0.2615/1**	0.3187/2
	[70]	96.98050/5	96.78460/4	96.7225/3	**0.2864/1**	0.3187/2
	[90]	57.07360/5	56.51360/4	56.3652/3	**0.2232/1**	0.3187/2
Resize	[100,300]	7.1552/5	7.0271/3	7.0514/4	0.6211/2	**0.3187/1**
	[150,450]	5.7099/5	4.9961/3	5.0494/4	0.5201/2	**0.3187/1**
	[200,600]	5.3152/5	3.9818/4	3.9311/3	0.4158/2	**0.3187/1**
	[250,750]	5.1489/5	3.6416/4	3.2583/3	0.3549/2	**0.3187/1**
	[300,900]	5.2512/5	3.6695/4	2.8195/3	0.3522/2	**0.3187/1**
Rotation	[25]	98.4825/5	98.4311/4	98.3983/3	0.5779/2	**0.3187/1**
	[70]	96.2016/5	96.1435/4	96.0986/3	0.6600/2	**0.3187/1**
	[100]	89.1230/5	89.0206/4	88.9595/3	0.6195/2	**0.3187/1**
	[200]	99.0954/5	99.0390/4	99.0047/3	0.5855/2	**0.3187/1**
	[300]	101.7332/5	101.6871/4	101.6610/3	0.6140/2	**0.3187/1**

Table 4: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on RMSE metric with ranks. Best results are given in **boldface**.

Attack/Methods		DWT	DCT	DWT_DCT_Joint	DWT_Arnold	Our SWA
Compression	[10]	0.0068/2	0.0071/4	**0.0062/1**	-0.2669/5	-0.0629/3
	[30]	0.0047/2	0.0096//3	**0.00037/1**	-0.0224/4	-0.0629/5
	[50]	0.0114/3	0.0035/2	**-0.00011/1**	-0.0179/4	-0.0629/4
	[70]	0.0236/4	0.0052/3	0.00180/2	**0.0000/1**	-0.0629/5
	[90]	0.0929/5	0.0059/3	0.00150/2	**0.0000/1**	-0.0629/4
Gaussian Noise	[0.03, 0.003]	0.5236/4	0.5069/3	0.5027/2	-2.1947/5	**-0.0629/1**
	[0.09, 0.009]	1.4204/4	1.4076/3	1.4058/2	-4.6177/5	**-0.0629/1**
	[0.1, 0.01]	1.5562/4	1.5474/3	1.5424/2	-4.8161/5	**-0.0629/1**
	[0.3, 0.03]	3.6103/2	3.6132/3	3.6138/4	-5.9191/5	**-0.0629/1**
	[0.5, 0.05]	4.6983/2	4.7012/3	4.7047/4	-5.6487/5	**-0.0629/1**
Pepper and Salt	[0.01]	0.0607/3	0.0403/2	**0.0383/1**	-0.1870/5	-0.0629/4
	[0.05]	0.2006/4	0.1825/3	0.1803/2	-1.3078/5	**-0.0629/1**
	[0.09]	0.3399/4	0.3255/3	0.3053/2	-1.6464/5	**-0.0629/1**
	[0.3]	0.9922/4	0.9804/2	0.9850/3	-2.5172/5	**-0.0629/1**
	[0.5]	1.4573/4	1.4037/2	1.5374/4	-2.7211/5	**-0.0629/1**
Median Filter	[1×1]	0.0254/4	0.0058/3	0.0014/2	**0.0000/1**	-0.0629/5
	[3×3]	-0.0154/4	-0.0147/3	-0.0137/2	**-0.0045/1**	-0.0629/5
	[5×5]	-0.0377/4	-0.0333/3	-0.0317/2	**-0.0269/1**	-0.0629/5
	[7×7]	-0.0553/4	-0.0485/3	-0.0455/2	-0.2527/5	**-0.0629/1**
	[9×9]	-0.6910/4	-0.0606/3	-0.0561/2	-2.0421/5	**-0.0629/1**
Cropping	[10]	-19.0662/3	-19.0817/5	-19.07895/4	-0.9658/2	**-0.0629/1**
	[30]	-10.1601/3	-10.1779/4	-10.1840/5	**-0.0179/1**	-0.0629/2
	[50]	-5.9778/3	-5.9973/4	-6.00370/5	0.1232/2	**-0.0629/1**
	[70]	-3.2358/3	-3.2558/4	-3.2615/5	**0.0267/1**	-0.0629/2
	[90]	-0.8318/3	-0.8516/4	-0.8563/5	0.2219/2	**-0.0629/1**
Resize	[100,300]	**-0.0202/1**	-0.0204/2	-0.0205/3	-1.8058/5	-0.0629/4
	[150,450]	**-0.0096/1**	-0.0115/2	-0.0116	-1.0502/5	-0.0629/4
	[200,600]	**-0.0040/1**	-0.0071/2	-0.0071/2	-0.4660/4	-0.0629/3
	[250,750]	**-0.0013/1**	-0.0044/2	-0.0052/3	-0.2056/5	-0.0629/4
	[300,900]	**0.0008/1**	-0.0027/2	-0.0040/3	-0.2056/5	-0.0629/4
Rotation	[25]	-2.4793/3	-2.4844/4	-2.4871/5	-1.4738/2	**-0.0629/1**
	[70]	-2.1654/2	-2.1707/3	-2.1742/4	-2.2096/5	**-0.0629/1**
	[100]	-1.2720/2	-1.2780/3	-1.2817/4	-1.7788/5	**-0.0629/1**
	[200]	-2.1649/3	-2.1708/4	-2.1736//5	-1.5179/2	**-0.0629/1**
	[300]	-2.7311/3	-2.7365/4	-2.7391/5	-1.7187/2	**-0.0629/1**

Table 5: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on SNR (dB) metric with ranks. Best results are given in **boldface**.
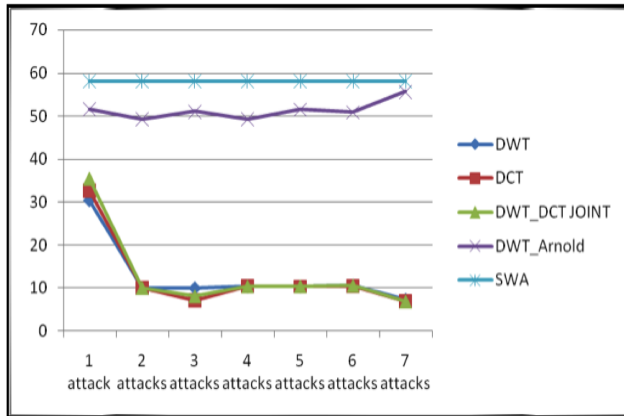
Figure 7: Graph showing the comparison of our proposed SWA model with other four methods based on PSNR (dB) values for multi-attacks on Lena image and copyright image.
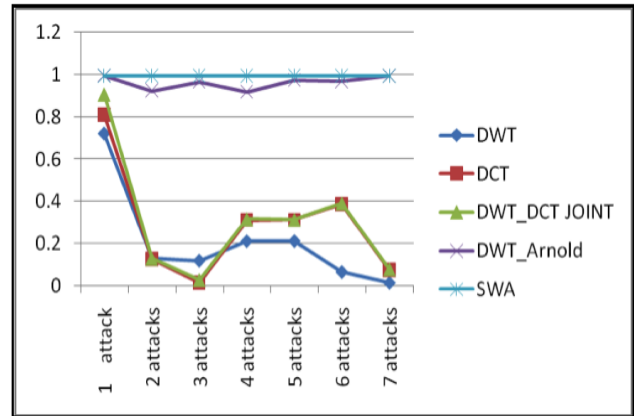

Figure 8: Graph showing the comparison of our proposed SWA model with other four methods based on SSIM values for multi-attacks on Lena image and copyright image.

DWT_DCT_Joint in (10%-50%) compression ratios. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_Arnold method with median and cropping attacks, but, with a little difference. Under Resize attack the DWT performed better under all sizes.

Perhaps the best error metric is SSIM which measures the performance in terms of preserving structural similarity between original and watermarking images. Table 6 investigates the impact of applying different methods on seven types of attack using Lena image with SSIM error metric. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.9950) similarity value. Although, it is clear from the results that DWT_Arnold algorithm was similar to our results when we apply compression attackss. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_Arnold method with median and cropping attacks, but, with a little difference.

Finally, Table 7 investigates the impact of applying different methods on seven types of attack using Lena image with respect to the mean absolute error (MAE) metric. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.1016) error value. Although, the DWT_Arnold algorithm was similar to our results when we apply compression attack, except (10%) compression ratio as well in Median Filter, and Cropping attacks. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation.

4.3.3 Comparison of multi-attacks on Lena image with PSNR and SSIM error metrics

When an attack occurs in an image that may lead to the loss of information or quality of the image (extracted message) that will be extracted. The performance gauged by error metrics which are used to evaluate the efficiency of the algorithms used in the embedding and extraction processes are classified into two types: subjective techniques, which are based on a view of humans, and objective techniques. We selected the SSIM, and PSNR (dB) metrics as the subjective and objective representative error measures for the evaluation next. We expose the image to different number of the attacks sequentially. This is a new comparative method in the field of digital image watermarking with multi-attacks.

Table 8 presents these multi-attacks and their corresponding results for different watermarking methods. We performed different combinations of attacks and we started with the compression attack (with 50% ratio) applied on image first. As can be seen, the proposed SWA method achieved significantly superior results compared to the rest of methods with PSNR = 58.0975 dB, and SSIM = 0.9950. The DWT achieved the worst results among others with PSNR = 30.4512 dB, and SSIM = 0.7207. When the image was further exposed to noise and filtering attacks with random values, along with cropping, resize, rotation attacks, the proposed SWA method achieved the best results consistently with PSNR = 58.0975 dB, and SSIM = 0.9950. Note that the DCT, DWT_DCT_Joint methods obtained the worst results with PSNR = 6.9578, SSIM = 0.0773 and PSNR = 0.0792, respectively. These results indicate the robustness of our proposed SWA model against multi-attacks. Figure 7 and Figure 8 shows the comparison of sequential multi-attacks with respect to the PSNR (dB), and SSIM error metrics respectively. From the figures it is clear that the proposed SWA model performs the best with DWT_Arnold performing the next best. The results with other DWT, DCT, DWT_DCT_Joint perform rather poorly.

Attack/Methods		DWT	DCT	DWT_DCT_Joint	DWT_Arnold	Our SWA
Compression	[10]	0.8271/5	0.8272/4	0.8297/3	**0.9951/1**	0.9950/2
	[30]	0.8204/5	0.6798/4	0.9031/3	**0.9950/1**	**0.9950/1**
	[50]	0.7204/5	0.8108/4	0.9062/3	**0.9950/1**	**0.9950/1**
	[70]	0.5985/5	0.7946/4	0.8820/3	**0.9950/1**	**0.9950/1**
	[90]	0.6004/5	0.8125/4	0.9259/3	**0.9950/1**	**0.9950/1**
Gaussian Noise	[0.03, 0.003]	0.3726/5	0.4320/4	0.4508/3	0.9751/2	**0.9950/1**
	[0.09, 0.009]	0.2376/5	0.2539/4	0.2599/3	0.9457/2	**0.9950/1**
	[0.1, 0.01]	0.2250/5	0.2390/4	0.2446/3	0.9428/2	**0.9950/1**
	[0.3, 0.03]	0.1322/5	0.1343/4	0.1368/3	0.9336/2	**0.9950/1**
	[0.5, 0.05]	0.1403/5	0.1427/4	0.1451/3	0.9338/2	**0.9950/1**
Pepper and Salt	[0.01]	0.4789/5	0.6486/4	0.7248/3	0.9949/2	**0.9950/1**
	[0.05]	0.2564/5	0.2937/4	0.3081/3	0.9877/2	**0.9950/1**
	[0.09]	0.1633/5	0.1750/4	0.1820/3	0.9826/2	**0.9950/1**
	[0.3]	0.0479/5	0.0487/4	0.0490/3	0.9704/2	**0.9950/1**
	[0.5]	0.0234/5	0.0238/4	0.0232/3	0.9704/2	**0.9950/1**
Median Filter	[1×1]	0.5839/5	0.8388/4	0.9532/3	**0.9950/1**	**0.9950/1**
	[3×3]	0.8477/5	0.9034/4	0.9264/3	**0.9950//1**	**0.9950/1**
	[5×5]	0.8525/5	0.8720/4	0.8810/3	**0.9951/1**	0.9950/2
	[7×7]	0.8214/5	0.8338/4	0.8392/3	**0.9955/1**	0.9950/2
	[9×9]	0.7950/5	0.8041/4	0.8087/3	0.9793/2	**0.9950 /1**
Cropping	[10]	0.0075/5	0.0085/4	0.0091/3	0.9901/2	**0.9950/1**
	[30]	0.0593/5	0.0765/4	0.0871/3	**0.9958/1**	0.9950/2
	[50]	0.1716/5	0.2206/4	0.2514/3	**0.9965/1**	0.9950/2
	[70]	0.3163/5	0.4249/4	0.4829/3	**0.9958/1**	0.9950/2
	[90]	0.4950/5	0.6946/4	0.7881/3	**0.9962/1**	0.9950/2
Resize	[100,300]	0.8609/5	0.8750/4	0.8722/3	0.9816/2	**0.9950/1**
	[150,450]	0.8621/5	0.9182/4	0.9122/3	0.9896/2	**0.9950/1**
	[200,600]	0.8450/5	0.9316/4	0.9318/3	0.9932/2	**0.9950/1**
	[250,750]	0.8368/5	0.9285/4	0.9444/3	0.9954/2	**0.9950/1**
	[300,900]	0.8248/5	0.9166/4	0.9528/3	0.9948/2	**0.9950/1**
Rotation	[25]	0.1498/5	0.1723/4	0.1875/3	0.9834/2	**0.9950/1**
	[70]	0.1534/5	0.1751/4	0.1952/3	0.9703/2	**0.9950/1**
	[100]	0.1784/5	0.2130/4	0.2398/3	0.9779/2	**0.9950/1**
	[200]	0.1459/5	0.1692/4	0.1851/3	0.9823/2	**0.9950/1**
	[300]	0.1337/5	0.1534/4	0.1651/3	0.9780/2	**0.9950/1**

Table 6: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on SSIM metric with ranks. Best results are given in **boldface**.

Attack/Methods		DWT	DCT	DWT_DCT_Joint	DWT_Arnold	Our SWA
Compression	[10]	5.2654/5	4.6638/3	5.2194/4	0.1367/2	**0.1016/1**
	[30]	4.5640/4	6.3203/5	3.2932/3	**0.0928/1**	0.1016/2
	[50]	5.8867/5	4.5083/4	3.0358/3	**0.0918/1**	0.1016/2
	[70]	7.9038/5	4.6638/4	3.1663/3	**0.0879/1**	0.1016/2
	[90]	8.7213/5	4.3617/4	2.2306/3	**0.0879/1**	0.1016/2
Gaussian Noise	[0.03, 0.003]	15.1372/5	13.3999/4	12.9221/3	0.4395/2	**0.1016/1**
	[0.09, 0.009]	28.2383/5	27.4299/4	27.2864/3	0.6592/2	**0.1016/1**
	[0.1, 0.01]	30.4388/5	29.7588/4	29.5494/3	0.6592/2	**0.1016/1**
	[0.3, 0.03]	70.9105/4	71.0183/5	71.0116/3	0.7363/2	**0.1016/1**
	[0.5, 0.05]	99.7833/4	99.8240/5	99.9451/3	0.7188/2	**0.1016/1**
Pepper and Salt	[0.01]	9.32670/5	5.21240/4	2.2142/3	0.1299/2	**0.1016/1**
	[0.05]	14.1145/5	10.1257/4	7.3025/3	0.3057/2	**0.1016/1**
	[0.09]	18.8807/5	14.9787/4	12.2862/3	0.3584/2	**0.1016/1**
	[0.3]	44.0520/5	41.2690/4	38.8275/3	0.4590/2	**0.1016/1**
	[0.5]	67.9147/5	65.6387/4	64.3678/3	0.4834/2	**0.1016/1**
Median Filter	[1×1]	88.1478/5	4.0007/4	0.9288/3	**0.0879/1**	0.1016/2
	[3×3]	3.96670/5	3.0514/4	2.2597/3	**0.0889/1**	0.1016/2
	[5×5]	4.18210/5	3.6768/4	3.3332/3	**0.0938/1**	0.1016/2
	[7×7]	4.86260/5	4.4538/4	4.2133/3	0.1338/2	**0.1016/1**
	[9×9]	5.56320/5	5.1595/4	4.9448/3	0.4248/2	**0.1016/1**
Cropping	[10]	123.8123/5	123.7751/4	123.7279/3	0.2656/2	**0.1016/1**
	[30]	114.3665/5	114.0056/4	113.6263/3	**0.0977/1**	0.1016/2
	[50]	96.12130/5	95.12180/4	94.06350/3	**0.0684/1**	0.1016/2
	[70]	69.48680/5	67.5075/4	65.7319/3	**0.0820/1**	0.1016/2
	[90]	29.8192/5	26.4739/4	23.8796/3	**0.0498/1**	0.1016/2
Resize	[100,300]	4.1504/5	3.9113/4	3.9644/3	0.3857/2	**0.1016/1**
	[150,450]	3.8638/5	2.9439/4	3.0264/3	0.2705/2	**0.1016/1**
	[200,600]	3.9695/5	2.5879/4	2.5079/3	0.1729/2	**0.1016/1**
	[250,750]	3.9839/5	2.5813/4	2.1687/3	0.1260/2	**0.1016/1**
	[300,900]	4.1283/5	2.7542/4	1.9236/3	0.1240/2	**0.1016/1**
Rotation	[25]	81.9561/5	81.8842/4	81.8376/3	0.3340/2	**0.1016/1**
	[70]	80.5659/5	80.4934/4	80.4498/3	0.4355/2	**0.1016/1**
	[100]	74.5938/5	74.5079/4	74.4594/3	0.3838/2	**0.1016/1**
	[200]	84.3401/5	84.2893/4	84.2609/3	0.3428/2	**0.1016/1**
	[300]	85.6902/5	85.6161/4	85.5812/3	0.3770/2	**0.1016/1**

Table 7: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on MAE metric with ranks. Best results are given in **boldface**.

Attack	DWT		DCT		DWT_DCT_Joint		DWT_Arnold		Our SWA	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Compr.50	30.4512	0.7207	32.7898	0.8108	35.5095	0.9063	51.6591	0.9950	**58.0975**	**0.9950**
Gaussian Noise 0.3, 0.03	10.1372	0.1291	10.1321	0.1281	10.1422	0.1298	49.3246	0.9242	**58.0975**	**0.9950**
Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3	10.0453	0.1198	7.2584	0.0164	8.1910	0.0286	51.0662	0.9662	**58.0975**	**0.9950**
Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter 1×1	10.4129	0.2116	10.5051	0.3132	10.5011	0.3169	49.3081	0.9184	**58.0975**	**0.9950**
Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter 1×1 Cropping 30	10.4051	0.2123	10.4917	0.3140	10.5022	0.3140	51.5746	0.9743	**58.0975**	**0.9950**
Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter 1×1 Cropping 30 Resize 200,600	10.6769	0.0664	10.6127	0.3886	10.6494	0.3916	50.8800	0.9676	**58.0975**	**0.9950**
Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter 1×1 Cropping 30 Resize 200,600 Rotation 50	7.3280	0.0153	6.9578	0.0773	6.9597	0.0792	55.6671	0.9929	**58.0975**	**0.9950**

Table 8: Comparison results of our proposed SWA model and four different approaches with multi-attacks on Lena image based on PSNR (dB), SSIM error metrics. Compression (50%) attack was applied first and the remaining attacks are applied sequentially. Best results are given in **boldface**.

Method	Embed.	Extr.	Total
DWT	7.6934	5.1184	12.8118
DCT	1.9716	1.3098	3.2814
DWT_DCT_Joint	5.8207	3.1113	8.932
DWT_Arnold	1.1819	1.4375	2.6194
Our SWA	6.081	7.1644	13.2454

Table 9: Watermarking (embedding/extraction) time consumed (in seconds) by different approaches using Lena image and copyright message.

Ref.	Approach	PSNR
[53]	Arnold	63.25
[45]	Arnold + DCT	43.82
[29]	DWT + Arnold	62.79
[22]	DWT + DCT	57.67
[35]	DWT+Shearlet	67.54
[18]	Entropy + Hadamard	42.74
[26]	Log-average luminance	62.49
[17]	DWT + DCT + SVD	57.09
[11]	DFT + 2D histogram	49.45
Our	DWT + DST + Arnold	**68.89**

Table 10: Comparison of PSNR (dB) values of the watermarked Lena image using some recent methods with our proposed SWA method.

4.4 Timing and other watermarking methods comparison

In Table 9 we show the total time consumed (in seconds) for each methods compared here in terms of the embedding and extraction of the message from a 1024×2014 image. It can be noted that the proposed SWA consumed the highest amount of time particularly in the extraction phase, this is due to the use of three different types of transforms. The DW_Arnold transform based method takes overall the least amount of time.

Finally, in Table 10 we show the PSNR (dB) comparison results with some more recent studies available in the literature. Note that these methods also utilize the standard test Lena image, and the values indicate that our proposed SWA outperforms them with highest PSNR = 68.89 dB, followed by [35] which has PSNR = 67.54 dB, with pure Arnold transform based approaches fairing better than other transforms.

5 Conclusions and future works

Digital watermarking is an active area of research within security and there are many automatic systems that have been presented to secure the ownership information of the digital image based on watermarking. These systems utilized available techniques from image processing and data mining areas and apply them to digital images. In this pa-

per, we studied a new hybrid model called SWA which is based on shearlet, wavelet, and Arnold transforms for efficient, and robust digital image watermarking. This model combines three transforms - discrete wavelet (DWT), discrete shearlet transform (DST), and Arnold transform - and their respective advantages. We utilized standard error image metrics to evaluate the proposed SWA method with seven types of attacks consists of different values of parameters; as well as multi-attacks which was used as a new way for comparing the effects of the attack on the image. Our results show that the proposed method is not affected by multi-attacks since applying shearlet at HL sub-band reduced the influence of watermarked image in any attack. Shearlet transform generates a series of matrices that obtained from the HL sub-band, where this sub band contains various features of the original image. In this way, the proposed SWA algorithm preserves a great deal of information. As a consequence, the attack is distributed to all of the shearlet derived matrices and the probability of changing the value of the pixel that has been embedding by the attack is very low. Based on this analysis, the results were stable or semi-static whatever the type of attack and its values.

Comparison results proved the robustness and strength of the proposed SWA method again other transform based state of the art methods. These results show that the proposed SWA model can be useful in securing image component in watermarking area. According to the results achieved by the proposed hybrid model, we consider that it is encouraging to apply this hybrid with several multimedia systems such as Video, text and audio, we expect that this system will achieve satisfactory results. Since the current research trends towards the multicore computing systems, our model may increase protection of videos transmission ownership.

Although the performance of the proposed method was the best comparing to state-of-the-art methods, it consumes more computational time than other approach, reducing this is one of the important future works. In this work, we applied shearlet transform at HL sub-band from level one in the wavelet transform, as another future work, we propose to go ahead towards applying shearlet transform with wavelet transform at the second and third level as well. Further, we can perform with other sub-bands such as LL, LH and HH to see if the robustness can be increased when certain types of attacks are applied. In our proposed method, we applied shearlet transform on the results of the wavelet transform, shearlet transform can also be applied with several other enhanced transforms like joint DWT with DCT etc.

References

[1] B. Ahmederahgi, F. Kurugollu, P. Milligan, A. Bouridane (2013) Spread spectrum image watermarking based on the discrete shearlet transform, *4th Euro-*

pean Workshop Visual Information Processing (EU-VIP), pp. 178–183.

[2] A. Al-Haj (2007) Combined DWT-DCT digital image watermarking, *Journal of computer science*, Vol. 3, pp. 740–746.

[3] Z. N. Y. Al-Qudsy (2011) An efficient digital image watermarking system based on contourlet transform and discrete wavelet transform *PhD Dissertation*, Middle East University, Turkey.

[4] Y. B. Amar, I. Trabelsi, N. Dey, M. S. Bouhlel (2016). Euclidean distance distortion based robust and blind mesh watermarking. International Journal of Interactive Multimedia and Artificial Inteligence, Vol. 4.

[5] S. K. Amirgholipour, A. R. Naghsh-Nilchi (2009) Robust digital image watermarking based on joint DWT-DCT, *International Journal of Digital Content Technology and its Applications*, pp. 42–54.

[6] R. Anju, Vandana (2013) Modified algorithm for digital image watermarking using combined DCT and DWT, *International Journal of Information and Computation Technology*, Vol. 3, No. 7, pp. 691–700.

[7] D. Baby, J. Thomas, G. Augustine, E. George, N. R. Michael (2015) A novel DWT based image securing method using steganography, *Procedia Computer Science*, No. 46, pp. 612–618.

[8] N. Y. Baithoon (2014) Zeros Removal with DCT Image Compression Technique, *Journal of Kufa for Mathematics and Computer* Vol. 1, No. 3.

[9] I. Belhadj, Z. Kbaier (2006) A novel content preserving watermarking scheme for multispectral images, *2nd International Conference on Information and Communication Technologies*, pp. 322–327.

[10] T. Bhaskar, D. Vasumathi (2013) DCT based watermark embedding into mid frequency of DCT coefficients using luminance component, *Elektronika ir Elektrotechnika*, Vol. 19, No. 4.

[11] F. G. M. Cedillo-Hernandez, M. Nakano-Miyatake, H. Manuel Pérez-Meana (2014) Robust hybrid color image watermarking method based on DFT domain and 2D histogram modification, *Signal, Image and Video Processing* Vol. 8, No. 1, pp. 49–63.

[12] C. H. V. Reddy, P. Siddaiah (2015) Medical image watermarking schemes against salt and pepper noise attack, *International Journal of Bio-Science and Bio-Technology* Vol. 7, No. 6, pp. 55–64.

[13] S. Chakraborty, S. Chatterjee, N. Dey, A. S. Ashour, A. E. Hassanien (2017). Comparative approach between singular value decomposition and randomized singular value decomposition-based watermarking. In Intelligent Techniques in Signal Processing for Multimedia Security (pp. 133-149). Springer.

[14] W. O. O. Chaw-Seng (2007) Digital image watermarking methods for copyright protection and authentication, *PhD Dissertation*, Queensland University of Technology, Australia.

[15] P.-Y. Chen, H.-J. Lin A (2006) DWT based approach for image steganography, *International Journal of Applied Science and Engineering*, Vol. 4, No. 3 pp. 275–290.

[16] M. F. M. El Bireki, M. F. L. Abdullah, M. Ali Abdrhman (2016) Digital image watermarking based on joint (DCT-DWT) and Arnold Transform, *International Journal of Security and Its Applications*, Vol. 10, No. 5, pp. 107–118.

[17] S. Fazli, M. Moeini (2016) A robust image watermarking method based on DWT, DCT, and SVD using a new technique for correction of main geometric attacks, *Optik-International Journal for Light and Electron Optics*, Vol. 2, No. 127, pp. 964–972.

[18] V. F. Rajkumar, G. R. S. Manekandan, V. Santhi (2011) Entropy based robust watermarking scheme using Hadamard transformation technique, *International Journal of Computer Applications* Vol. 12, No. 9.

[19] X. Gibert, V. M. Pate;, D. Labate, R. Chellappa (2014) Discrete shearlet transform on GPU with applications in anomaly detection and denoising, *EURASIP J. Adv. Signal Process*, pp. 1–14.

[20] B. L. Gunjal, R. R. Manthalkar (2010) An overview of transform domain robust digital image watermarking algorithms, *Journal of Emerging Trends in Computing and Information Sciences* Vol. 2, No. 1, pp. 37–42.

[21] X. Guo-juan, W. Rang-ding (2009) A blind video watermarking algorithm resisting to rotation attack, *International Conference on Computer and Communications Security (ICCCS)*. Hong Kong, pp. 111–114.

[22] I. I. Hamid, E. M. Jamel (2016) Image watermarking using integer wavelet transform and discrete cosine transform, *Iraqi Journal of Science* Vol. 57, No. 2B, pp. 1308–1315.

[23] A. E. Hassanien, M. Tolba, and A. T. Azar (2014) Advanced machine learning technologies and applications. *Second International Conference*, Egypt, AML,Springer.

[24] S. Häuser, and G. Steidl (2012) Fast finite shearlet transform, *arXiv preprint*.

[25] K. K. Hiran, R. Doshi (2013) Robust & secure digital image watermarking technique using concatenation process, *International Journal of ICT and Management*

[26] J. A. Hussein (2010) Spatial domain watermarking scheme for colored images based on log-average luminance, *Journal of Computing* Vol. 2, no. 1.

[27] X. Kang, J. Huang, Y. Q Shi, Y. Lin (2003) A DWT-DFT composite watermarking scheme robust to both affine transform and JPEG compression, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 8, pp. 776–786.

[28] S. A. Kasmani, A. NaghshNilchi (2008) A new robust digital image watermarking technique based on joint DWT-DCT Transformation, *IEEE International Conference on Convergence and Hybrid Information Technology (ICCIT)*, pp. 539–544.

[29] R. Keshavarzian, A. Aghagolzadeh (2016) ROI based robust and secure image watermarking using DWT and Arnold map, *AEU-International Journal of Electronics and Communications*, pp. 278–288.

[30] S. K. A. Khalid, M. Mat Deris, and K. M. Mohamad (2013) A robust digital image watermarking against salt and pepper using sudoku, *The Second International Conference on Informatics Engineering and Information Science (ICIEIS)*.

[31] D. Labate, W.-Q. Lim, G. Kutyniok, G. Weiss (2005) Sparse multidimensional representation using shearlets. *Optics & Photonics*, pp. 59140U-59140U.

[32] J. C. Lee Analysis of attacks on common watermarking techniques.

[33] W.-Q. Lim (2010) The discrete shearlet transform: A new directional transform and compactly supported shearlet frames, *IEEE Transactions on Image Processing*, pp. 1166–1180.

[34] Lin, Shinfeng D., and C.-F. Chen (2000) A robust DCT-based watermarking for copyright protection, *IEEE Transactions on Consumer Electronics*, Vol. 3, No. 46, pp. 415–421.

[35] M. Mardanpour, Mohammad Ali Zare, Chahooki (2016) Robust hybrid image watermarking based on discrete wavelet and shearlet transforms, *arXiv preprint*.

[36] Y. Naderahmadian, S. Beheshti (2015) Robustness of wavelet domain watermarking against scaling attack, *IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE)*, Canada, pp. 1218–1222.

[37] N. Dey, R. A. Bardhan, D. Sayantan (2011) A novel approach of color image hiding using RGB color planes and DWT, *International Journal of Computer Applications*.

[38] N. Dey, M. Pal, A. Das (2012). A session based blind watermarking technique within the NROI of retinal fundus images for authentication using DWT, spread spectrum and Harris corner detection. arXiv preprint 1209.0053.

[39] N. Dey, B. Nandi, P. Das, A. Das, S. S. Chaudhuri (2013). Retention of electrocardiogram features insignificantly devalorized as an effect of watermarking for a multi-modal biometric authentication system. Advances in biometrics for secure human authentication and recognition, 175.

[40] N. Dey, S. Samanta, X. S. Yang, A. Das, S. S. Chaudhuri (2013). Optimisation of scaling factors in electrocardiogram signal watermarking using cuckoo search. International Journal of Bio-Inspired Computation, Vol. 5, No. 5, pp. 315–326.

[41] N. Dey, G. Dey, S. Chakraborty, S. S. Chaudhuri (2014). Feature analysis of blind watermarked electromyogram signal in wireless telemonitoring. In Concepts and Trends in Healthcare Information Systems (pp. 205-229). Springer.

[42] N. Dey, M. Dey, S. K. Mahata, A. Das, S. S. Chaudhuri (2015). Tamper detection of electrocardiographic signal using watermarked biohash code in wireless cardiology. International Journal of Signal and Imaging Systems Engineering, Vol. 8, No. 1-2, pp.46–58.

[43] F. O. Owalla, E. Mwangi (2012) A robust image watermarking scheme invariant to rotation, scaling and translation attacks, *16th IEEE Mediterranean Electrotechnical Conference*, Hammamet, pp. 379-382.

[44] A. Piva, B. Mauro, B. Franco, C. Vito (1997) DCT-based watermark recovering without resorting to the uncorrupted original image, *IEEE International Conference on Image Processing*, pp. 520–523.

[45] C. Pradhan, V. Saxena, A. K. Bisoi (2012) Imperceptible watermarking technique using Arnold's transform and cross chaos map in DCT Domain, *International Journal of Computer Applications*, Vol. 55, No. 15.

[46] N. Saikrishna, M. G. Resmipriya (2016) An invisible logo watermarking using Arnold transform, *Procedia Computer Science*, pp. 808–815.

[47] C.N. Sujatha, P. Satyanarayana (2015) Hybrid color image watermarking using multi frequency band, *International Journal of Scientific and Engineering Research*, Vol. 6, No. 1, pp. 948–951.

[48] S. Swami (2013) Digital image watermarking using 3 level discrete wavelet transform, *Conference on Advances in Communication and Control Systems*.

[49] T. Tewari, V. Saxena (2010) An improved and robust DCT based digital image watermarking scheme, *International Journal of Computer Applications*, Vol. 1, No. 3, pp. 28–32.

[50] Van Schyndel, Ron G., Andrew Z. Tirkel, and Charles F. Osborne (1994) A digital watermark, *IEEE International Conference on Image Processing*, Vol. 86–90.

[51] J. Veerappan, G. Pitchammal (2012) Interpolation based image watermarking resisting to geometrical attacks, *IEEE International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME)*, pp. 252–256.

[52] H.-J. Wang, P.-C. Su, and C-C. J. Kuo (1998) Wavelet-based digital image watermarking, *Optics Express*, Vol. 3, No. 12, pp. 491–496.

[53] H.-Q. Wang, J.-C. Hao, F.-M. Cui (2010) Colour image watermarking algorithm based on the Arnold transform, *IEEE International Conference on Communications and Mobile Computing (CMC)*, pp. 66–69.

[54] Z. Wang, E. P. Simoncelli, A. C. Bovik (2004) Multiscale structural similarity for image quality assessment, *Thirty-Seventh Asilomar Conference Signals, Systems and Computers*, pp. 1398–1402.

[55] L. Wu, J. Zhang, W. Deng, D. He (2009) Arnold transformation algorithm and anti-Arnold transformation algorithm, *IEEE International Conference on Information Science and Engineering*, pp. 1164–1167.

[56] Y. Wu, G. Xin, M. S. Kankanhalli, Z. Huang (2001) Robust invisible watermarking of volume data using the 3D DCT, *Computer Graphics International*, pp. 359–362.

[57] X. G. Xia, B. Charles, and G. Arce (1998) Wavelet transform based watermark for digital images, *Optics Express*, Vol. 3, No. 12, pp. 497–511.

[58] S. Zagade, S. Bhosale (2014) Secret data hiding in images by using DWT Techniques, *International Journal of Engineering and Advanced Technology* Vol. 3, no. 5.

[59] X. Y. Zhao, H. Wang (2006) A novel synchronization invariant audio watermarking scheme based on DWT and DCT, *IEEE Transactions on Signal Processing*, Vol. 54, No. 12, pp. 4835–4840.

[60] T.-R. Zong, Y. Xiang, S. Elbadry, S. Nahavandi (2016) Modified moment-based image watermarking method robust to cropping attack, *International Journal of Automation and Computing*, Vo. 13, No. 3, pp. 259–267.

An Improved Gene Expression Programming Based on Niche Technology of Outbreeding Fusion

Chao-xue Wang, Jing-jing Zhang, Shu-ling Wu, Fan Zhang
School of Information and Control Engineering, Xi'an University of Architecture and Technology, Xi'an, China
E-mail: 13991996237@163.com
http://www.xauat.edu.cn/en

Jolanda G. Tromp
Department of Computer Science, State University of New York Oswego, USA
E-mail: jolanda.tromp@oswego.edu

An improved Gene Expression Programming (GEP) based on niche technology of outbreeding fusion (OFN-GEP) is proposed to overcome the insufficiency of traditional GEP in this paper. The main improvements of OFN-GEP are as follows: (1) using the population initialization strategy of gene equilibrium to ensure that all genes are evenly distributed in the coding space as far as possible; (2) introducing the outbreeding fusion mechanism into the niche technology, to eliminate the kin individuals, fuse the distantly related individuals, and promote the gene exchange between the excellent individuals from niches. To validate the superiority of the OFN-GEP, several improved GEP proposed in the related literatures and OFN-GEP are compared about function finding problems. The experimental results show that OFN-GEP can effectively restrain the premature convergence phenomenon, and promises competitive performance not only in the convergence speed but also in the quality of solution.

Povzetek: V prispevku je predstavljena izboljšava genetskih algoritmov na osnovi niš in genetskega zapisa.

1 Introduction

Gene expression programming (GEP) was invented by Candida Ferreira in 2001[1, 2], which is a new achievement of evolutionary algorithm. It inherits the advantages of Genetic Algorithm (GA) and Genetic Programming (GP), and has the simplicity of coding and operation of GA and the strong space search ability of GP in solving complex problems [3]. GEP has more simplify on data set reduction than other intelligent computing technologies such as rough set, clustering, and abstraction. [4-6]. Currently, GEP becomes a powerful tool of function finding and has been widely used in the field of mechanical engineering, materials science and so on [7, 8], but the problem of low converging speed and readily being premature still exists like other evolutionary algorithms.

So far, the domestic and foreign scholars proposed different improvements about the traditional GEP in the field of function finding. Yi-shen Lin introduced the improved K-means clustering analysis into GEP, by adjusting the min clustering distance to control the number of niches, to improve the global searching ability [9]. Tai-yong Li designed adaptive crossover and mutation operators, and put forward the measure method of population diversity with weighted, to maintain the population diversity in the process of evolution [10]. Yong-qiang Zhang introduced the superior population

producing strategy and various population strategy to improve the convergence speed of the algorithm and the diversity of population [11]. Shi-bin Xuan proposed the control of mixed diversity degree (MDC-GEP) to ensure the different degree in the process of evolution and avoid trapping into local optimal [12]. Hai-fang Mo adopted the clonal selection algorithm with GEP code for function modelling (CSA-GEP), to maintain the diversity of population and increase the convergence rate [13]. Yan-qiong Tu proposed an improved algorithm based on crowding niche, and the algorithm contributed to push out the premature individuals by penalty function and made the better individuals have greater probability to evolve [14].

To further improve the performance of the GEP, this paper proposes an improved gene expression programming based on niche technology of outbreeding fusion (OFN-GEP). The main ideas are as follows: (1) using the population initialization strategy of gene equilibrium to ensure that all genes are evenly distributed in the coding space as far as possible; (2) introducing the outbreeding fusion mechanism into the niche technology, to eliminate the kin individuals, fuse the distantly related individuals, and promote the gene exchange between the excellent individuals from different sub-populations. The experiments compared with other GEP algorithms

proposed in the related literatures about function finding problems are executed，and the results show that OFN-GEP can overcome the premature convergence phenomenon effectively during the evolutionary process, and has high solution quality, fast convergence rate.

2 Standard gene expression programming

Standard gene expression programming (ST-GEP), which was firstly put forward by Candida Ferreira in 2001[1, 2], could be defined as a nine-meta group: $GEP = \{C, E, P_0, M, \varphi, \Gamma, \Phi, \Pi, T\}$, where

C is the coding means; E is the fitness function; P_0 is the initial population; M is the size of population; φ is the selection operator; Γ is the crossover operator; Φ is the point mutation operator; Π is the string mutation operator; T is the termination condition. In GEP, individual is also called chromosome, which is formed by gene and linked by the link operator. The gene is a linear symbol string which is composed of head and tail. The head involves the functions from function set and the variables from the terminator set, but the tail merely contains the variables from the terminator set. Like GA and GP, GEP follows the Darwinian principle of the survival of the fittest and uses populations of candidate solutions to a given problem to evolve new ones, and the basic steps of ST-GEP are as follows [1, 2]:

(1) Inputting relevant parameters, creating the initial population;
(2) Computing the fitness of each individual;
(3) If the termination condition is not met, go on the next step, otherwise, terminate the algorithm;
(4) Retaining the best individual;
(5) Selecting operation;
(6) Point mutating operation;
(7) String mutating operation (IS transposition, RIS transposition, Gene transposition);
(8) Crossover operation (1-point recombination, 2-point recombination, Gene recombination);
(9) Go to (2).

3 Gene expression programming based on niche technology of outbreeding fusion

The flowchart of the OFN-GEP is schematically represented in Fig 1. Its main steps are as follows:

Step 1: Adopt the population initialization strategy of gene equilibrium to generate the population P, and set up the maximum *MAX* and the minimum *MIN* about the number of individuals in the niche. Then divide the initial population into several equal niches;

Step 2: Perform the genetic operators within each niche, which including point mutation, string mutation (IS, RIS, Gene transposition) and recombination (1-point, 2-point, gene recombination). Then use the pre-selection operator to protect the best individual in every niche;
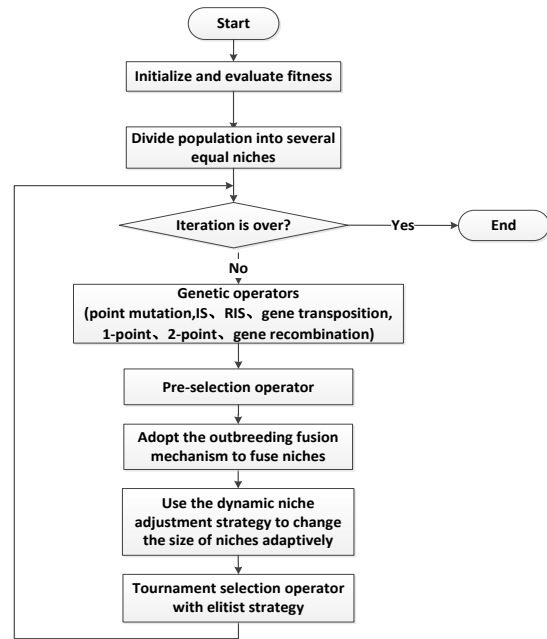

Figure 1: The flowchart of OFN-GEP.

Step 3: Use the outbreeding fusion mechanism to eliminate the kin individuals and fuse the distant relatives between two niches, and introduce some random individuals at the same time;

Step 4: Adopt the dynamic adjustment strategy to change the size of niches according to the maximum *MAX* and the minimum *MIN*;

Step 5: Perform the tournament selection operator with elitist strategy;

Step 6: Go to Step 2 until the iteration is over.

3.1 Population initialization

This algorithm adopts the population initialization strategy of gene equilibrium to increase the initial population diversity. The idea of this strategy is to let all genes are distributed uniformly in the coding space, so that the initial population diversity is rich. This strategy can reduce the time of search process and achieve the global optimal solution at a rapid speed [15].

3.2 Fitness function

In statistics, the method to assess the relevance degree between two groups of data usually uses the correlation coefficient. References [16], the fitness function is devised as: *fitness*=R^2=1-*SSE/SST*, where

$$SSE = \sum_{j=1}^{m} (y_j - \hat{y}_j)^2 \qquad (1)$$

$$SST = \sum_{j=1}^{m} (y_j - \overline{y})^2 \qquad (2)$$

where, y_j is the observation data; $\hat{y}_j$ is the forecast data which is computed with formula and observation data; $\overline{y}$ is the mean of y ; *SSE* is the residual sum of squares;

SST is the total sum of squares of deviations; *m* is the size of data.

3.3 Pre-selection

The pre-selection operator is: the offspring individual can instead of his father and access to the next generation only when the fitness of the new individual is bigger than his father. Due to the similarity of the offspring individual and his father, an individual can be replaced by his structure similar individual, which can maintain the diversity of population and protect the best individual in population.

3.4 A niche technology of outbreeding fusion

The niche technology of outbreeding fusion includes two aspects: one is to use the outbreeding fusion mechanism to eliminate the kin individuals, fuse the distant relatives between two niches and promote the gene exchange between the best individuals, which improving the diversity of population and the quality of solutions; the other is to use the dynamic niche adjustment strategy to change the size of niches adaptively [17], which maintaining the genetic diversity of niches.

Aiming at the judgment of the distant individuals in outbreeding fusion, this paper adopts the calculation methods of recessive hamming distance (individual fitness, the essence differences between individuals), and dominant hamming distance (the appearance differences between individuals), and the judge rules between kin and distant relatives as well in literature [18], to judge the kinship between individuals.

The niche technology of outbreeding fusion operators is as follows:

(1) Select two niches randomly, and merge all individuals of the two niches (which are supposed as N1 and N2, and before fusion, their sizes are S_1 and S_2 respectively) into niche N1; go to (2);

(2) Adopt the outbreeding fusion strategy (Algorithm 1) to eliminate the kin individuals, and then obtain the size S_1' of the modified N1; go to (3);

(3) If S_1' is bigger than the maximum *MAX*, corresponding *MAX* individuals will be selected out by tournament selection; then adjust S_1' and go to (5); else go to (4);

(4) If S_1' is smaller than minimum *MIN*, the new individuals will be introduced randomly until the smallest size *MIN* is satisfied; then adjust S_1' and go to (5);

(5) Construct N2 by the individuals generated randomly, and the size S_2' of N2 satisfies the equation $S_2' = S_1 + S_2 - S_1'$.

Algorithm 1: Outbreeding judgment

- Sort the fitness of the fusion individuals in ascend (or descend);

- Compute the dominant and recessive hamming distance between two adjacent individuals;
- If the dominant hamming distance between two adjacent individuals is less than the setting threshold M_1, and the recessive hamming distance is less than the setting threshold M_2, then the two individuals are kin relatives; otherwise, they are the distant relatives;
- Eliminate the lower fitness individual between the kin relatives, and retain the other one.

4 Experiments and results

In this section, two experiments are designed to justify the effectiveness and competitiveness of OFN-GEP for function finding problems, the general parameters setting of experiments are shown as Table 1. The source codes are developed by MATLAB 2009a, and run on a PC with i7-2600 3.4 GHz CPU, 4.0 GB memory and Windows 7 professional sp1.

4.1 Test for the effectiveness of OFN-GEP

To evaluate the improved effect of OFN-GEP, this paper adopts the F function, which was used in literature [1] as shown in equation (3), and the 10 groups of training data are produced by F. OFN-GEP is compared with the DS-GEP in literature [19]. The test results are shown in Table 2, the evolution curve is shown in Fig 1, Fig 2

Table 1: The parameter settings of experiments.

Option	Test A	Test B
Times of runs	50	50
Max evolution generation	200	200
Size of population	100	100
Function set	+,-,*,/	+, -, *, /, ln, exp, S, Q, sin, cos, tan, cot
Terminator set	a	
Link operator	+	+
The length of head	6	6
Number of gene	5	5
Point mutation rate	0.4	0.4
IS and RIS rate	0.3	0.3
Crossover rate	0.3	0.3
Recombination rate	0.3	0.3
Length of IS element	{1,2,3,4,5}	{1,2,3,4,5}
Length of RIS element	{1,2,3,4,5}	{1,2,3,4,5}
Size of tournament	3	3
The number of niches	5	5
The minimum threshold of the size of niche	10	10
The maximum threshold of the size of niche	60	60
The threshold of the recessive hamming distance	0.1	0.1
The threshold of the dominant hamming distance	0.5	0.5

Note: S is Square, Q is Sqrt, and Exp is e^x.

separately, where both the optimizing rate and the average generations of convergence of OFN-GEP are obviously superior to DS-GEP.

$$F : 5a_n^4 + 4a_n^3 + 3a_n^2 + 2a_n + 1 \tag{3}$$

Table 2: The results of experiment A.

Option	DS-GEP	OFN-GEP
Times of runs	50	50
Times of hit	41	48
Optimizing ratio	82%	96%
Average generations of convergence	87	41

As is shown in Table2, the average generations of convergence that achieves the optimal solution in OFN-GEP algorithm is less than the one in DS-GEP, so the OFN-GEP algorithm can improve convergence speed efficiently. From the evolution curves in Fig 2 and Fig 3 the volatility of average fitness in OFN-GEP is greater than the one in DS-GEP, and this says the differences between individuals are greater and the population diversity is better in OFN-GEP.
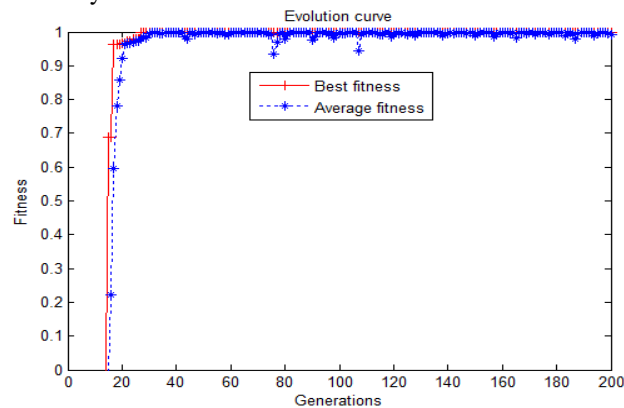
Figure 2: The evolution curve of DS-GEP.
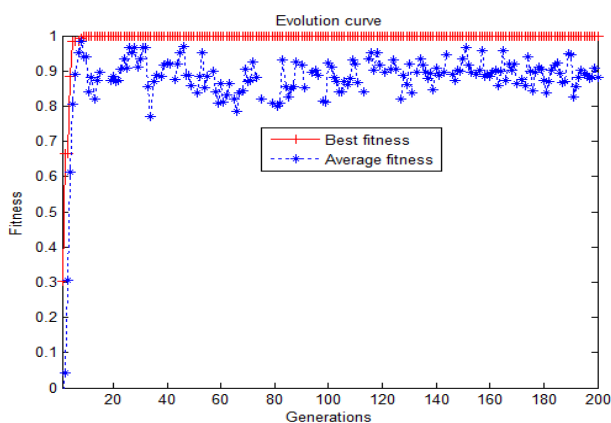
Figure 3: The evolution curve of OFN-GEP.

4.2   Test for the competitiveness of OFN-GEP

To test the competitiveness of OFN-GEP, MDC-GEP [12] and S-GEP [20] are chosen to compare with OFN-GEP. Test functions are partly the same with the ones in [12] and [20]. They are shown as (4) - (14). The test results are shown in Table 3.

$$F1 : 8 + 2e^{1-x^2} \cos(2\pi x), x \in [0,5] \tag{4}$$

$$F2 : \cos^2(x^2), x \in [0,3] \tag{5}$$

$$F3 : \frac{1}{2 + \sin(x)}, x \in [0,5\pi] \tag{6}$$

$$F4 : 5 + \log(\cos^2(2\pi x) + x^2), x \in [0,5\pi] \tag{7}$$

$$F5 : \frac{3x^3 + 2x^2 + 1}{5x^3 + 2}, x \in [0,5] \tag{8}$$

$$F6 : \cos(10^x), x \in [0,2] \tag{9}$$

$$F7 : \frac{3x^2 + 5x + 1}{5y^2 + 3}, x, y \in [0,5] \tag{10}$$

$$F8 : \frac{\sin(x^2 + 2)}{\cos(y^3 + 2.5) + 3}, x, y \in [-5,5] \tag{11}$$

$$F9 : xye^{-x^2 - y^2}, x, y \in [-2,2] \tag{12}$$

$$F10 : \frac{\sin x_1 + \cos x_2}{\sqrt{e^{x_3}}} + \tan(x_4 - x_5) \tag{13}$$

$$x_i \in [0, 2\pi], i = 1, 2, ...5$$

$$Fv : 4.251a^2 + \ln(a^2) + 7.243e^a, a \in [-1,1] \tag{14}$$

Table 3: Test results of experiment B.

Function	Algorithm	Max fitness	Min fitness	Average fitness
F1	MDC-GEP	0.9675	0.8231	0.9263
	OFN-GEP	0.96825	0.8782	0.9334
F2	MDC-GEP	0.9991	0.7865	0.9371
	OFN-GEP	1	0.8112	0.9446
F3	MDC-GEP	0.9916	0.8645	0.9843
	OFN-GEP	0.9959	0.8901	0.9505
F4	MDC-GEP	0.9812	0.8743	0.9476
	OFN-GEP	0.9898	0.9430	0.9696
F5	MDC-GEP	0.9587	0.6856	0.8735
	OFN-GEP	0.9413	0.7641	0.8408
F6	MDC-GEP	0.9954	0.8237	0.9465
	OFN-GEP	1	0.9603	0.9913
F7	MDC-GEP	0.9462	0.8133	0.8956
	OFN-GEP	0.9792	0.8387	0.9317
F8	MDC-GEP	0.9473	0.7012	0.8653
	OFN-GEP	0.9525	0.7464	0.8719
F9	MDC-GEP	0.9673	0.6782	0.8750
	OFN-GEP	0.9415	0.7099	0.8777
F10	MDC-GEP	0.9771	0.8954	0.9520
	OFN-GEP	0.9909	0.9109	0.9605
Fv	S-GEP	0.9991		
	OFN-GEP	0.9988	0.9796	0.9926

From Table 3, for most functions, the max fitness, min fitness and average fitness increase obviously in OFN-GEP compared with the MDC-GEP, S-GEP. This shows the effectiveness and competitiveness of OFN-GEP. For relatively simple function F2 and F6, the fitness of OFN-GEP can achieve 1, but for F5, F9, Fv, their fitness is less than (very close to) the results in MDC-GEP. The reasons are that GEP algorithm is a random algorithm, the algorithm parameters have great influence on the results of experiment, and the parameters of every function to obtain the best fitness are different. So, this situation exists which few functions can't obtain a better fitness value under the same parameters.

5 Conclusion

This paper puts forward an improved gene expression programming based on niche technology of outbreeding fusion (OFN-GEP), and verifies the effectiveness and competitiveness of the proposed algorithm about the function finding problems. The improvements in the paper are that: (1) using the population initialization strategy of gene equilibrium to ensure that all genes are evenly distributed in the coding space as far as possible; (2) introducing the outbreeding fusion mechanism into the niche technology, to eliminate the kin individuals, fuse the distantly related individuals, and promote the gene exchange be-tween the excellent individuals from different sub-populations. To validate the effectiveness and competitiveness of OFN-GEP, several improved GEP proposed in the related literatures and OFN-GEP are compared as regards function finding problems. The experimental results show that OFN-GEP can effectively restrain the premature convergence phenomenon, and promises competitive performance not only in the convergence speed but also in the quality of solution.

6 Acknowledgment

7 References

[1] Ferreira C (2001), Gene Expression Programming: A New Adaptive Algorithm for Solving Problems, *Complex System*, Complex Systems Publications, vol.13 no 2, pp 87–129.

[2] Ferreira C (2003), Function Finding and the Creation of Numerical Constants in Gene Expression Programming, In: Benítez J.M., Cordón O., Hoffmann F., Roy R. (Eds), *Advances in Soft Computing*, Springer-verlag, pp 257–266.

[3] Gerald Schaefer (2016), Gene Expression Analysis based on Ant Colony Optimisation Classification, *International Journal of Rough Sets and Data Analysis*, IGI Global, vol. 3, no.3, pp.51-59.

[4] Debi Acharjya, A. Anitha (2017), A Comparative Study of Statistical and Rough Computing Models in Predictive Data Analysis, *International Journal of Ambient Computing and Intelligence*, IGI Global, vol.8, no.2, pp.32-51.

[5] Hans W. Guesgen, Stephen Marsland (2016), Using Contextual Information for Recognising Human Behaviour, *International Journal of Ambient Computing and Intelligence*, IGI Global, vol. 7, no.2, pp.27-44

[6] Ch. Swetha Swapna, V. Vijaya Kumar, J.V.R Murthy (2016), Improving Efficiency of K-Means Algorithm for Large Datasets, *International Journal of Rough Sets and Data Analysis*, IGI Global, vol.3, no.2, pp.1-9.

[7] A.H. Gandomi, A.H. Alavi (2013), Intelligent Modeling and Prediction of Elastic Modulus of Concrete Strength via Gene Expression Programming, *Lecture Notes in Computer Science*, Springer-verlag, vol. 7928I, pp 564–571.

[8] Y. Yang, X.Y. Li (2016), Modeling and impact factors analyzing of energy consumption in CNC face milling using GRASP gene expression programming, *International Journal of Advanced Manufacturing Technology*, Springer-verlag, Vol.87, no.5, pp.1247–1263.

[9] Y.S. Lin, H. Peng, J. Wei (2008), Function Finding in Niching Gene Expression Programming, *Journal of Chinese Computer Systems*, Chinese Computer Society, vol.29, pp.2111–2114.

[10] T.Y.LI, C.J. Tang (2010), Adaptive Population Diversity Tuning Algorithm for Gene Expression Programming, *Journal of University of Electronic Science and Technology of China*, UESTC Press, vol. 39, no.2, pp. 279–283.

[11] Y.Q. Zhang, J. Xiao (2010), A New Strategy for Gene Expression Programming and Its Applications in Function Mining, *Universal Journal of Computer Science and Engineering Technology*, Springer-verlag, vol.1, no.2, pp.122–126.

[12] S.B. Xuan, Y.G. Liu (2012), GEP Evolution Algorithm Based on Control of Mixed Diversity Degree, *Pattern Recognition and Artificial Intelligence*, Chinese Association Automation, vol. 25. no.2, pp.187–194.

[13] H.F. MO (2013), Clonal Selection Algorithm with GEP Code for Function Modeling, *Pattern Recognition & Artificial Intelligence*, Chinese Association Automation, vol.26, no9, pp.878–884, 2013.

[14] Y.Q. Tu, X. Wang (2013), Application of improved gene expression programming for evolutionary modeling, *Journal of Jiangxi University of Science and Technology*, JUST Press, vol. 34, no.5, pp.77–81.

[15] L. Yao, H. Li (2012), An Improved GEP-GA Algorithm and Its Application, Communications in Computer & Information Science, Springer, vol.316, pp 368-380.

[16] J. Zuo (2004), *Research on Core Technology of Gene Expression Programming*, Sichuan: Sichuan University.

[17] Y. L. Chen, F. Y. Li, J. Q. Fan (2015), Mining association rules in big data with NGEP, *Cluster Computing*, Kluwer Academic Publishers, vol.18, no2, pp.577-585.

[18] Y. Jiang, C. J. Tang (2007), Outbreeding Strategy with Dynamic Fitness in Gene Expression Programming, *Journal of Sichuan University (Engineering Science Edition)*, Sichuan University Press, vol.39. no2, pp.121-126.

[19] C. X. Wang, K. Zhang, H. Dong (2014), Double System Gene Expression Programming and its application in function finding, *the Proceedings of International Conference on Mechatronics, Control and Electronic Engineering(MCE2014)*, Atlantis Press, pp 357-361.

[20] Y. Z. Peng, C. A. Yuan, X. Qin, J. T. Huang (2014), An improved Gene Expression Programming approach for symbolic regression problems, *Neurocomputing*, Elsevier, vol.137, pp 293-301.

Identity-based Signcryption Groupkey Agreement Protocol Using Bilinear Pairing

Sivaranjani Reddi,
Anil Neerukonda Institute of Technology and Science, Bhimunipatnam, India
E-mail: sivaranjani.cse@anits.edu.in

Surekha Borra
K.S. Institute of Technology, Bangalore, India
E-mail: borrasurekha@gmail.com

This paper proposes a key agreement protocol with the usage of pairing and Malon-Lee approach in key agreement phase, where users will contribute their key contribution share to other users to compute the common key from all the users key contributions and to use it in encryption and decryption phases. Initially the key agreement is proposed for two users, later it is extended to three users, and finally a generalized key agreement method, which employs the alternate of the signature method and authentication with proven security mechanism, is presented. Finally, the proposed protocol is compared with the against existing protocols with efficiency and security perspective.

Povzetek: Razvit je nov varnostni protokol za uporabo več ključev.

1   Introduction

Key Establishment is the procedure in which more than one user launches the session key, and is consequently used in accomplishing the cryptographic services like confidentiality or integrity. In general, key establishment protocols follow the key transfer approach, where one user decides the key and communicate it to other user. In contrast, for key agreement protocols all the users in the communication are involved in key establishment process. Further, these key agreement protocols provides the implicit authentication if the user assures that no other user or intruder involved in the communication knows the confidential key value. Hence, a protocol which possesses the implied key authentication to all the users involved in the group communication is called authenticated group key agreement protocol. Key Confirm is one property of the group key agreement protocol where one user involved in group communication assures that the other user in the group is under the control of the confidential key. When a protocol possesses both implicit authentication and key confirmation, that protocol is called as explicit key authentication. More details about key agreement protocols are discussed in [1, 21, 22, 23, 24].

This paper emphasis is on an authentic key agreement technique. Diffie-Hellman [2] proposed first key agreement. However, it is insecure against middle attack. Afterwards, many key agreement methodologies were published by various authors, but some users prerequisite a Public Key Infrastructure (PKI), needs more calculation and preserving efforts. Shamir[4] had initiated the concept called cryptosystem using user identity in which users public key can be calculated using the users unique attributes (e.g. Email, mobile no.

etc), his private key is estimated by the trustworthy user referred as Private Key Generator (PKG). After that public key crypto system is formulated using user identity, which had simplified the process of key administration thus become a substitute to certificate centred PKI. Later, Joux[3] had proposed, Bilinear pairing based group key agreement protocol.

Boneh[5],formally published an ID based encryption scheme using bilinear pairings. Many protocols were proposed [13, 11, 10, 8, 15], analyzed and some of them were broken [14,9,17,12,16]. Few pairing based applications use a pairing-friendly elliptic curve of prime numbers. There are different coordinate systems that can be used to represent points on elliptic curves such as Jacobian, Affine and Homogeneous. Inversion to multiplication ratio threshold can be used to decide the efficiency of coordinate system. In this work timing results of pairing is being reported for both affine and projective coordinates using BN-curve. All fast algorithms to compute pairings on elliptic curves are based on *so as* Miller's algorithm [26]. In this paper, focus is on ID based authenticated key agreement using pairings with the two users. It is based on the signature scheme suggested Malone-Lee [6]. Furthermore, it is elaborated and evaluated against some of the existing ones in terms of efficiency and security. Pairing based mathematical properties were discussed in section 2, Marko Hölbl protocol the existing protocol was discussed in section 3, the proposed protocol was explained in section 4 and the next talks about performance of proposed technique against the existing protocols and finally it was concluded.

2 Preliminaries

This section presents a notation of bilinear pairing operations which are to be used next.

Bilinear maps[5] [6]: Let $(G_1,+)$, $(G_2,+)$ and $(G_T, \cdot)$ are the two additive and one multiplicative group of prime order $q > 2k$ for a security parameter $k \in N$, then there exists a bilinear map $\hat{e} : G_1 \times G_2 \to G_T$ that has the following properties:

1. Bilinearity: $\hat{e}(aP, bQ) = \hat{e}(P,Q)^{ab}$, where $P,Q \in G_1$, and $a, b \in Z$, can be reformulated as:
   $e(P + Q,R) = e(P,R) e(Q,R)$ and $e(P,Q + R) = e(P,Q)e(P,R)$ for $P,Q,R \in G_1$
2. Non-degeneracy: $\hat{e}(P, Q) = 1$, if $Q \in G_2$ iff $P = 1 \in G_1$.
3. Computability: $\hat{e}(P, Q)$ is efficiently computable if $P \in G_1$ and $Q \in G_2$.

When $G_1=G_2$ and $P=Q$ then that group is termed as symmetric bilinear map.

2.1 Signcryption

Signcryption is a type of crypto mechanism and offers security services. It performs encryption and data signing in a single operation, and satisfies the requirements of smaller bandwidth and less computational cost by doing the operations sequentially. In symmetric encryption schemes it is computationally impossible to extract the plaintext from the signcrypted message without receiver's private key. As in symmetric digital signature, creation of signcrypted text without using the private key of the sender is computationally infeasible. Some of the existing signcryption mechanisms are as follows:

A. Malone -Lee ID-based encryption scheme[6]

The detailed description of the Malonee Lee identity based encryption is as follows:

Step 1: (Setup): A PKG considers hash functions $H_1:\{0,1\}^* \to G_1$, $H_2:\{0,1\}^* \to Z_q^*$, $H_3: G_2 \to \{0,1\}^l$ and a generator P. The PKG can choose a random integer as master private key s and calculates $P_{pub}=sP$. Finally publishes the parameters $<P, \hat{e}, P_{pub}, H_1, H_2, H_3>$, by keeping PKG's secret keys as secret.

Step 2: (Extract): For given user identification $ID \in \{0,1\}^*$, the PKG calculates the public key $QID= H_1(ID)$ and secret key $SID= s*QID$.

Step 3: (sign): For the given secret key SID and message $M \in \{0,1\}^*$, the sender selects random number $r \in Z_q^*$, and U=rP, then computes $r=H_2(U||M)$, W= $r*P_{pub}$, V=r*SID+W, y=e(W,QID), x=$H_3(y)$ and C=x$\oplus$ M, finalizes the signature as (C,U,V) and then send it to receiver side.

Step 4:(unsigncrypt): Upon receiving the signature (C,U,V), receiver computes public key of the sender using his identity $QID= H_1((ID)$, parse the signature

(C,U,V) then computes y=e(SID,U), x=$H_3(y)$, M=x$\oplus$ C, r=$H_2(U||M)$, and then accepts M if e(V,P)= e(U,P_{pub})*e($QID, P_{pub})^r$

Advantages: Eliminates distribution of the public key. Authentication of the public key is implicitly guaranteed as long as individual user kept his private key secure.

Disadvantage: Establishment of the secure channel is required between the user and the PKG.

B. Boneh IBE cryptosystem[5]

Boneh has proposed an identity based encryption technique to encrypt the message using pairing. It mainly contains four algorithms described as follows:

Step 1: (Setup): A PKG considers two hash functions, H_1 and H_3. The PKG can choose random s $\in Z_q$ master private key, and calculates P_{pub}=sP. Finally, publishes the parameters $<P, \hat{e}, P_{pub}, H_1, H_3>$, by keeping PKG's secret keys as secret.

Step 2: (Extract): For the given user identity $(ID) \in \{0,1\}^*$ the PKG calculates publickey $QID= H_1((ID)$ and secret key $SID= s*QID$.

Step 3: (encrypt): An user can choose r, then calculates ciphertext(C) for M, be C= (rP, M$\oplus H_3(g_{ID}{}^r$)) where g_{ID}= e (QID, P_{pub})

Step 4: (decrypt): from the received C= (U, V) receiver computes V $\oplus H_3$(e (SID, U)) in order to extract M.

Advantage: This mechanism is secure against forgery under the chosen plaintext attack under Strong Diffie Hellman(SDH) assumption without using oracle model.

Disadvantages: All the hash functions are random hash functions. Further, as the public keys are directly computed, it leads to avoidance of certificate maintenance.

C. Hesse identity based signature[25]

A signature is computed and enclosed to M before sending onto other side. Upon receiving M along with the signature; the receiver tries to verify the signature before accepting the M. The detailed Hesse mechanism is as follows:

Step 1: (Setup): A PKG considers hash function $H_1, H:\{0,1\}^* X G_2 \to Z_q^*$. The PKG can choose s master private key and calculates P_{pub}=sP. Finally publish the parameters $<P, \hat{e}, P_{pub}, H_1, H>$, by keeping PKG's secret key s as secret.

Step 2: (Extract): For given user with identity (ID), the PKG calculates the public key $QID= H_1(ID)$ and the secret key $SID= s*QID$.

Step 3: (Sign): for the given secret keySID and M $\in \{0,1\}^*$, the sender selects $P_1 \in G_1$ and k $\in Z_q^*$, and then computes r= $e(P_1,P)^k$, v=H(M,r) and u=v*SID + $k*P_1$., finalizes the signature is (u,v).

Step 4: (Verify): for a given public key QID , the received M and the signature is (u,v). The receiver computes r= $e(u,P)\mathrm{e}(QID, -P_{pub})^k$ and accept if v=H (M, r).

Advantages: It is secure against adaptive chosen message attack in the random oracle model.

Disadvantages: As PKG is generating the private keys of user, there may be a scope to decrypt or sign any message without any authorization. Hence it may not be fit to attain non repudiation

2.2   Security analysis

The protocol mechanism presented in this paper is equipped with the following listed attributes:
(i) Known key Security: For each session, the participant randomly selects h_i and r_i, results separate independent group encryption key and decryption keys for other sessions. A leakage of group decryption keys in one session will not help in derivation of other session group decryption keys.
(ii) Unknown key share: In proposed protocol, each participant U_i generates a signature ρ_i using x_i. Therefore, group participants can verify the ρ_i if it is from authorized person or not. Hence, no non group participant can be impersonated.
(iii) Key compromise impersonate: Due to generation of unforgeable signature by the participant $U_{i,}$ the challenger cannot create the valid signature on behalf of U_i. Even if participant U_j's private key is compromised by the adversary, he cannot mimic other participant U_i with U_j's private key. Hence, key is not impersonated in the proposed protocol.

3   Marko Hölbl protocol [7]

This is an ID-based signature technique using the Hess algorithm. It is a two party ID-based authenticated key agreement protocol requiring PKG. Mainly divided into system setup, private key estimation and key agreement phase.
Phase 1 (setup): In this phase PKG decides the parameters called system parameters, which helps in the derivation of common group key agreement by all the users in the communication. A PKG formulates G_1 , G_2 and $\hat{e}$ and computes the cryptographic function H, P, a random integers as PKG's private key and P_{pub} as PKGs publickey. All elements are of order q. Finally he publishes all the parameters $<G_1, G_2, P, \hat{e}, P_{pub}, H>$, by keeping PKG's secret keys as secret.
Where mapping function $\hat{e}$: $G_1 \times G_1 \rightarrow G_2$
Primitive Generator P: P $\in G_1$
Random integer s: s $\in Z_q^*$
Public Key P_{pub} =sP
Hash function H: $Z_q^* \rightarrow G_1$

Phase 2 (Private key extraction): In this phase PKG derives the public key Q_i and private key S_i of individual user by using their identity ID_i and then broad casts the public key and firmly send the privatekey to the

respective user through secured channel, where Q_i = H(ID_i) and S_i = s*Q_i.

Phase 3 (Key agreement): Since signature verification will authenticate the data in deciding which user issued this, a message generated from this phase will be used later to derive the session key. After choosing the receiver (B), sender (A) decides the message and then signed the message. Later on both message and the signature are sent to the receiver. The receivers compute the signature from the received message and then compare against the received signature, before deriving the key sent by sender. Procedure 1 shows the operations summary in key agreement phase.

Global Parameters $<G_1, G_2, P, \hat{e}, P_{pub}, H>$
User A Key Generation   a $\in Z_q^*$ $T_A = aP, U_A = \hat{e}(S_A,P)^a$, $V_A = H(T_A, r_A), W_A = \mathrm{H}(V_A S_A + a S_A)$
User B Key Generation   b $\in Z_q^*$ $T_B = bP, U_B = \hat{e}(S_B,P)^b$, $V_B = H(T_B, r_B)$, $W_B = \mathrm{H}(V_B S_B + b S_B)$
Calculation of secret key by User A $U_B' = \hat{e}(W_B,P)\hat{e}(Q_B, -P_{pub})^{V_B}$ $V_B = \mathrm{H}(T_B, U_B')$ $K_{AB} = aT_B = abP$
Calculation of secret key by User B $U_A' = \hat{e}(W_A,P)\hat{e}(Q_A, -P_{pub})^{V_A}$ $V_A = \mathrm{H}(T_A, U_A')$ $K_{AB} = bT_A = abP$

Procedure 1: Marko Hölbl protocol.

Marko Hölbl protocol mechanism results in the following computational requirements:
- In order to exchange message, each user has to compute two scalar multiplications, exponentiation, hash function and summation.
- In session key computation, 2 pairings and 2 hashing operation, scalar multiplication and exponentiation are required.

4   Proposed protocols

Group key agreement is the mechanism where two or more users are involved in the derivation of the group key used to encrypt/decrypt the data. The major phases in the proposed algorithm are: setup, extract, signcrypt and unsigncrypt phases as shown in Fig.1. This section describes the key agreement protocol between two users, three users and n numbers of users.

4.1 Proposed protocol for two users

This protocol is designed based on the Malone-Lee [6] ID-based crypto system scheme. It is protected against chosen random oracle model under BDH. The advantage of this algorithm is to perform the message encryption and decryption in only one step to attain security services more efficiently, instead of first signing and then encryption. This scheme is the combination of Boneh IBE cryptosystem with the variant of Hesses Identity based signature.

Step 1: (Setup): This phase usually finalizes the number of users willing to join the group communication. Once the number of users is decided, then PKG will finalize the common parameters to be used in the derivation of other phase parameters. A PKG considers three hash functions H_1, H_2, H_3 and P. PKG can choose a random integer s, master private key and calculates $P_{pub}=sP$. Finally publishes the parameters $<P, ê, P_{pub}, H_1, H_2, H_3>$, by keeping PKG's secret key s as secret.

Step 2: Extract: PKG employs user's identity information in the derivation of secret and public keys. The input for this phase is user identity and produces QID and D. PKG uses user A Identity $(ID_A) \in \{0,1\}^*$ and calculates public key $QID_A= H_1(ID_A)$ and secret key $SID_A= s* QID_A$. Once generated SID_A is securely sent to user A. This process repeats for user B, in calculating QID_B and SID_B using the identity(ID_B).

Step 3: Signcrypt: Both users A and B can execute this phase in parallel, where individual user uses their SID, along with other users public key QID and their key contribution k in the derivation of ciphertext and the signature generation.
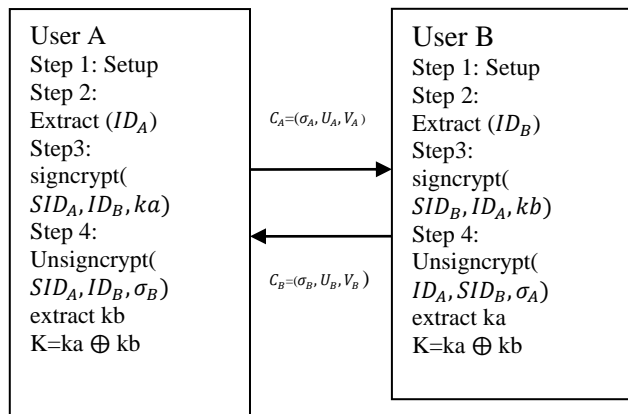

Figure 1: Group key agreement protocol.

The steps for the signcrypt at user A side is as follows:
a. User A selects ka $\in \{0,1\}^l$, computes $QID_B= H_1(ID_B)$. ------(1)
b. User A chooses a random number $X_A \leftarrow Z_q^*$ and set $U_A = X_A P$ -----(2)
c. Calculates $R_A= H_1(U_A ||ka)$, $W_A= X_A.P_{pub}$, $V_A= R_A.SID_A + W_A$, $Y_A =e(W_A, QID_B)$, $T_A=H_3(Y_A)$. ---(3)

d. *Finally computes $\sigma_A= T_A \oplus$ ka and then sends $C_A=(\sigma_A, U_A, V_A)$ to B. ----(4)*

Here A chooses the key ka and communicates to B by adding a signature for the verification. Parallely B also chooses his contribution in key agreement kb, User B follows the above steps, uses his private key SID_B and A's public key QID_A and then sends $C_B=(\sigma_B, U_B, V_B)$ to A.
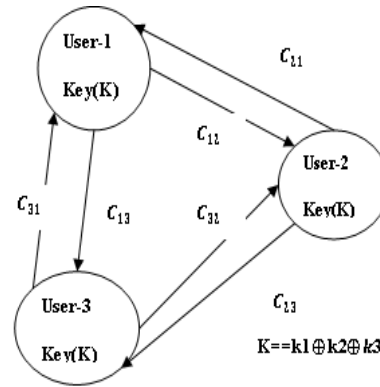

Figure 2: Key agreement among three users.

Step 4: Unsigncrypt: Key contribution of A can be extracted from C_A after comparing the signature validation condition. B uses the following steps in the derivation of ka from received C'_A.
a) Computes the A's public key $QID_A=H_1(ID_A)$ ---(5)
b) parse $C'_A=(\sigma'_A, U'_A, V'_A)$, compute $Y'_A =e(SID_B, U'_A)$, $T'_A=H_3(Y'_A)$, $ka'= T'_A \oplus \sigma'_A$ and $R'_A= H_1(U'_A ||ka)$. ---(6)
c) Accept ka' when $e(V'_A, P) = e(QID_A, P_{pub})^{R'_A}.e(U'_A, P_{pub})$ ------(7)

Limitations of the work:
- Proposed technique withstands outsider attacks (i.e. adversary is not permitted to exhibit the sender's private key with which the cipher text was created).
- Another limitation is due to the procedure used by the receiver in non repudiation. The receiver needs to prove to the third party that sender is the authorized person of a given plaintext.

4.2 Group key agreement with three users

The proposed algorithm is extended to three users and their arrangement is shown in Figure 2, where, the setup and extraction phase is same as described in section 3. During the signcrypt phase, user-1 uses other users public key with whom he wants to share the key and then computes the respective value C1, j where j $\in\{3,2\}$. From the diagram, user-1 calculates C_{12} and C_{13} and send to user-2 and user-3 respectively. Similarly user-2 calculates their contributions C_{21} and C_{23} and then send to user 1 and 3. After signcrypt phase each user will receive the encrypted contributions from other users in the group. All the keys will be decrypted and then extract the individual key user contributions after validating the signature. Once all user signatures were satisfied, individual user adds his contribution and apply the XOR

operation on all the users in group in order to derive the session group key.

4.3 Generalized group key agreement

Step 1: (Setup): This phase usually finalizes the number of users willing to join in the group communication. Once the users joining task gets completed, then PKG will finalize the common parameters to be used in the derivation of other phase parameters. A PKG considers hash functions H_1, H_2, H_3 and P. PKG can choose a random integer s, master private key and calculates P_{pub}=s*P. Finally publish the parameters $<P, \hat{e}, P_{pub}, H_1, H_2, H_3>$, by keeping PKG's secret key s as secret.

Step 2: Extract: PKG uses individual user's identity information in the derivation of secret and public keys. The input for this phase is user identity and produces QID and SID which represents public and private keys respectively. PKG uses user i ($1 \leq i \leq n$) identity (ID_i)and computes $QID_i = H_1(ID_i)$ and secretkey $SID_i = $ s*QID_i, then sends SID_i securely to i.

    For i=1 to n
    Calculate $QID_i = H_1(ID_i)$ ---(8)
    Calculate $SID_i = $s* QID_i ---(9)

Step 3: Signcrypt: Each user derives the parameters individually to other participant and communicates. User-1 in the group will first decide ka and then calculates other variables:X_1, U_1, R_1, W_1, $Y_{1,i}, V_1$ and $T_{1,i}$. Similarly user-i uses the signcrypt algorithm to securely share his key contribution ki.

a.   A selects ki $\in \{0,1\}^l$, computes $QID_j = H_1(ID_j)$ ($1 \leq$ j $\leq$n, j$\neq$ i) ---(10)
b. Afterwards he chooses a random number $X_i \leftarrow Z_q^*$ and set $U_i = X_iP$ ---(11)
c. Calculates $R_i = H_1(U_i ||ki)$, $W_i = X_i.P_{pub}$, $V_i = R_i.SID_i + W_i$ ---(12)
d. For each user j (j$\neq$ i) , user i computes $Y_{i,j} = e(W_i, QID_j)$, $T_{i,j}=H_3(Y_i)$.---(13)
e. Finally computes $\sigma_{i,j} = T_{i,j} \oplus$ ka and then sends $C_{i,j}=(\sigma_{i,j}, U_A, V_A)$ user –j ($1 \leq$ j $\leq$n, j$\neq$ i).---(14)

    Step 4: Unsigncrypt: User-j uses the following steps in the derivation of ki from received $C'_{i,j}$. key contribution of ith user can be extracted from $C_{i,j}$ after comparing the signature validation condition
a.. Computes the i's public key $QID_i=H_1(ID_i)$ ---(15)
b.Parse $C'_{i,j}=(\sigma'_{i,j}, U'_i, V'_i)$, compute $Y'_i =$e(SID_j, U'_i), $T'_i=H_3(Y'_i)$, ki'= $T'_i \oplus \sigma'_i$ and $R'_i = H_1((U'_i ||ki)$. -(16)
c.Accept ki' when e$(V'_i, P)=e(QID_i, P_{pub})^{R'_i}.e(U'_i, P_{pub})$.

--- (17)

5 Performance analyses

Proposed protocol is compared with Wang [16], Yuan-Li [18], Chow–Choo without escrow[19], Choie-jeong-Lee[20] and Marko Hölbl et.al [7]. Tables 1&2 illustrate

comparison of the suggested protocol against the existing protocols. The efficiency is estimated by considering the communication cost and the execution cost. Communication cost includes number of rounds and the length of message transmitted through the network during protocol execution. Overall number of rounds in protocol
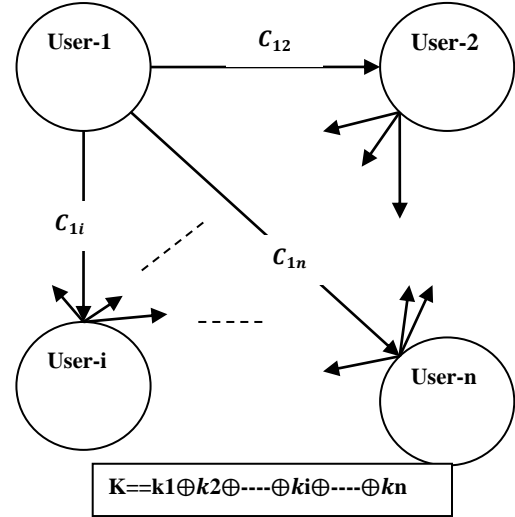

Figure 3: Generalized key agreement Protocol.

is the primary concern in practical environments where the group users are more in number. Yuan-Li has one round operation in key agreement phase, used one multiplication and exponentiation, one addition. Protocol is secured against the key impersonation, backward and forward secrecy. Wang's method almost uses the same number of operations as yuan's method, but computation time is more.   Chow–Choo without escrow key agreement protocol mainly contains two rounds: one is extract phase and the other is key agreement phase. During the extract phase, one hash function and pairing function, remaining operations were used during the key agreement phase.

Protocol Name	Computation Cost						Commu-nication Cost
	pairing	Mul	Exp	Add	Hash	XOR	
[16]	1	3	0	3	3	0	1
[18]	1	3	0	2	1	0	1
[19]	1	4	0	2	1	0	2
[20]	2	4	0	0	2	0	1
[7]	3	3	2	1	3	0	3
Proposed	1	5	0	1	4	1	3

P2P: total point to point communication per user: Pairing: total number of mapping or pairing operations per user: Add: Total number of addition operations per user: Exp : total exponentiations performed per user.: Mul: total scalar multiplications computed : XoR: total XOR operations computed; Hash: total hash functions evaluated per user : Rounds: Number of Rounds

Table 1: Efficiency Comparison with other protocols.

Protocol name	KKS	FoS	UKS	BS	KC	KI
[16]	√	√	√	√	√	√
[18]	√	√	√	√	√	√
[19]	√	√	√	√	√	√
[20]	√	√	√	√	√	√
[7]	√	√	√	√	√	√
Proposed	√	√	√	√	√	√

KI:Key Impersonation BS:Backward Secrecy UKS:Unknown
Key Share   FoS:Forward Secrecy; KC:Key control

Table 2: Security Analysis with existing protocols.

Marko Hölbl et.al method uses three multiplications, three pairings, two exponentiations, one addition, and three hashing operations in three rounds for finalizing group key using pairing based key agreement. roposed algorithm has three rounds setup, private key extraction and common key agreement in the group. The computation time for proposed protocol is less compared to [7] and [20] protocols because of less number of pairing operations. The proposed protocol requires more time in scalar multiplication and XOR operation. The protocol does not require any exponential operations. Inspite of more number of hash functions, the proposed protocol requires less computation time because of involvement of less expensive operations.

6   Conclusion

An enhanced ID-based authenticated key agreement protocol is proposed and discussed, which employs signatures to authenticate participated user and verifies correctness of transferred messages between two users. The effectiveness and security of proposed technique showed all desired security properties and was compared against existing protocols in terms of efficiency and security. The protocol further confirms all the security properties with minimum time efficiency. In future, the protocol can be extended to hierarchical and cluster based network environment for establishing a secured communication. Also it can be applied in IoT based machine to machine communication, and machine to device communication.

7   References

[1] A. Menezes, P.C. Van Oorschot, S. Vanstone,(1997) Handbook of Applied Cryptography, CRC Press,.

[2] W. Diffie, M. Hellman,(1976), New directions in cryptography, IEEE Trans. Inform. Theory 22 (6),pp. 644–654.

[3] A. Joux(2000), A one round protocol for tripartite Diffie–Hellman, in: 4th International Symposium on Algorithmic Number Theory, in: Lecture Notes inComput. Sci., vol. 1838, Springer, New York, pp. 385–394.

[4] A. Shamir(1985), Identity-based cryptosystems and signature schemes, in: Advances in Cryptology – CRYPTO'84, Springer, New York, pp. 47–53.

[5] D. Boneh, M. Franklin (2003), Identity-based encryption from the Weil pairing, SIAM J. Comput. Vol,32 issue-3,pp: 586–615.

[6] J.Malonee-Lee(2002), "Identitity based signcryption, Available at *http://eprint.iacr.org/2002/098*

[7] Marko Hölbl et.al(2012)," An improved two-party identity-based authenticated key agreement protocol using pairings,journal of computer and system sciences ,vol:78,pp.142-150.

[8] L. Chen, C. Kudla(2003), Identity based authenticated key agreement protocols from pairings, in: Computer Security Foundations Workshop, IEEE, USA,pp. 219–233.

[9] K.K.R. Choo, McCullagh–Barreto(2005)," two-party ID-based authenticated key agreement protocols",Internat. J. Netw. Secur.vol-1,issue-3,pp.154–160.

[10] N. McCullagh, P.S.L.M. Barreto(2004), A new two-party identity-based authenticated key agreement, Cryptology ePrint Archive Report .

[11] K. Shim(2003), Efficient ID-based authenticated key agreement protocol based on Weil pairing, Electronics Lett. 39 (8) ,pp.653–654.

[12] K. Shim(2005), Cryptanalysis of two ID-based authenticated key agreement protocols from pairings, Cryptology ePrint Archive Report 2005/357.

[13] N.P. Smart(2002), Identity-based authenticated key agreement protocol based on Weil pairing, Electronics Lett. 38 (13) ,pp.630–632.

[14] H.M. Sun, B.T. Hsieh(2003), Security analysis of Shim's authenticated key agreement protocols from pairings, Cryptology ePrint Archive Report 2003/113.

[15] Y. Wang(2005), Efficient identity-based and authenticated key agreement protocol, Cryptology ePrint Archive Report2005/108.

[16] S.B. Wang, Z.F. Cao, H.Y. Bao(2005), Security of an efficient ID-based authenticated key agreement protocol from pairings, in: Parallel and Distributed Processingand Applications – ISPA2005, in: Lecture Notes in Comput. Sci., vol. 3759, Springer, New York, pp. 342–349.

[17] G. Xie(2004), Cryptanalysis of Noel McCullagh and Paulo S.L.M. Barreto's two-party identity-based key agreement, Cryptology ePrint Archive Report 2004/308.

[18] Q. Yuan(2005), S. Li, A new efficient ID-based authenticated key agreement protocol, Cryptology ePrint Archive Report 2005/309.

[19] Z. Cheng, L. Chen(2007), " On security proof of McCullaghBarretos key agreement protocol and its variants" , Internat. J. Secur. Networks 2 ,pp.251–259.

[20] Y.J. Choie, E. Jeong, E. Lee (2005), Efficient identity-based authenticated key agreement protocol from pairings, Appl. Math. Comput. 162 (1) ,pp.179–188.

[21] Chakraborty, S., Chatterjee, S., Dey, N., Ashour, A. S., & Hassanien, A. E. (2017). Comparative

Approach Between Singular Value Decomposition and Randomized Singular Value Decomposition-based Watermarking. In Intelligent Techniques in Signal Processing for Multimedia Security (pp. 133-149). Springer International Publishing..

[22] Dey, N., Samanta, S., Yang, X. S., Das, A., & Chaudhuri, S. S. (2013). Optimisation of scaling factors in electrocardiogram signal watermarking using cuckoo search. International Journal of Bio-Inspired Computation, 5(5), 315-326. NilanjanDey et al.(2015), "Tamper Detection of Electrocardiographic Signal using Watermarked Bio-hash Code in Wireless ", International Journal of Signal and Imaging Systems Engineering , Volume 8, Issue 1-2 .

[23] Dey, N., Pal, M., & Das, A. (2012). A Session Based Blind Watermarking Technique within the NROI of Retinal Fundus Images for Authentication Using DWT, Spread Spectrum and Harris Corner Detection. arXiv preprint arXiv:1209.0053..

[24] Hess, F. (2002, August). Efficient identity based signature schemes based on pairings. In International Workshop on Selected Areas in Cryptography (pp. 310-324). Springer Berlin Heidelberg..

[25] Beuchat, J. L., González-Díaz, J. E., Mitsunari, S., Okamoto, E., Rodríguez-Henríquez, F., & Teruya, T. (2010, December). High-speed software implementation of the optimal ate pairing over Barreto–Naehrig curves. In International Conference on Pairing-Based Cryptography (pp. 21-39). Springer Berlin Heidelberg.

Performance Evaluation of Lazy, Decision Tree Classifier and Multilayer Perceptron on Traffic Accident Analysis

Prayag Tiwari
National University of Science and Technology MISiS
Department of Computer Science and Engineering, Moscow, Russia
E-mail: prayagforms@gmail.com

Huy Dao
Microsoft Corp, Redmond, Washington, USA
E-mail: huydao@microsoft.com

Gia Nhu Nguyen
Duy Tan University, Danang, VietNam
E-mail: nguyengianhu@duytan.edu.vn, tel: (+84) 901 964444

Traffic and road accident are a big issue in every country. Road accident influence on many things such as property damage, different injury level as well as a large amount of death. Data science has such capability to assist us to analyze different factors behind traffic and road accident such as weather, road, time etc. In this paper, we proposed different clustering and classification techniques to analyze data. We implemented different classification techniques such as Decision Tree, Lazy classifier, and Multilayer perceptron classifier to classify dataset based on casualty class as well as clustering techniques which are k-means and Hierarchical clustering techniques to cluster dataset. Firstly we analyzed dataset by using these classifiers and we achieved accuracy at some level and later, we applied clustering techniques and then applied classification techniques on that clustered data. Our accuracy level increased at some level by using clustering techniques on dataset compared to a dataset which was classified with-out clustering.

Povzetek: Predstavljena je analiza prometnih nesreč z odločitvenimi drevesi in večnivojskimi perceptroni.

1 Introduction

Traffic and road accident are one of the important problems across the world. Diminishing accident ratio is most effective way to improve traffic safety. There are many types of research has been done in many countries in traffic accident analysis by using a different type of data mining techniques. Many researchers proposed their work in order to reduce the accident ratio by identifying risk factors which particularly impact in the accident [1-5]. There are also different techniques used to analyze traffic accident but it's stated that data mining technique is more advance technique and shown better results as compared to statistical analysis. However, both methods provide an appreciable outcome which is helpful to reduce accident ratio [6-13, 28, 29, 37-44].

From the experimental point of view, mostly studies tried to find out the risk factors which affect the severity levels. Among most of the studies explained that drinking alcoholic beverage and driving influenced more in an accident [14]. It identified that drinking an alcoholic beverage and driving seriously increase the accident ratio. There are various studies which have focused on restraint devices like a helmet, seat belts influence the severity level of accident and if these devices would have been used to accident ratio had

decreased at a certain level [15]. In addition, few studies have focused on identifying the group of drivers who are mostly involved in an accident. Elderly drivers whose age are more than 60 years, they are identified mostly in road accident [16]. Many studies provided a different level of risk factors which influenced more in severity level of accident.

Lee C [17] stated that statistical approaches were good option to analyze the relation between in various risk factors and accident. Although, Chen and Jovanis [18] identified that there is some problem like large contingency table during analyzing big dimensional dataset by using statistical techniques. As well as statistical approach also have their own violation and assumption which can bring some error results [30-33]. Because of this limitation in statistical approach, Data techniques came into existence to analyze data of road accident. Data mining often called as knowledge or data discovery. This is set of techniques to achieve hidden valuable knowledge from huge amount of dataset. There are many observed implementations of data mining techniques in transportation system like pavement analysis, roughness analysis of road and road accident analysis.

Data mining methods have been the most widely used techniques in a field like agriculture, medical, transportation, business, industries, engineering and many other scientific fields [21-23]. There are many diverse data mining methodologies like clustering, association rules, and classification techniques have been extensively used for analyzing a dataset of road accident [19-20]. Geurts K [24] analyzed dataset by using association rule mining to know the, unlike circumstances that happen at a very high rate in road accident areas on Belgium road. Despair [25] analyzed a dataset of a road accident in Belgium by using different clustering techniques and stated that clustered based data might fetch information at a higher level as compared without clustered data. Kwon analyzed dataset by using Decision Tree and NB classifiers to factors which are affecting more in a road accident. Kashani [27] analyzed dataset by using classification and regression algorithm to analyze accident ratio in Iran and achieved that there are factors such as wrong overtaking, not using seat belts, and badly speeding affected the severity level of accident. Tiwari [34, 36] used K-modes, Hierarchical clustering and Self-Organizing Map (SOM) to cluster dataset of Leed City, UK and run classification techniques on that clustered dataset, accuracy in-creased up to some level around 70% as compared to the classified dataset without clustering. Hassan [35] used multi-layer perceptron (MLP) fuzzy adaptive resonance theory (ART) used a dataset of Central Florida Area and his result shown that inter-section in rural areas is more dangerous in a situation of injury severity of driver than intersection in urban areas.

2 Methodology

This research work focuses on casualty class based classification of a road accident. The paper describes the k-means and Hierarchical clustering techniques for cluster analysis. Moreover, Decision Tree, Lazy classifier and Multilayer perceptron used in this paper to classify the accident data.

2.1 Clustering techniques

2.1.1 Hierarchical clustering

Hierarchical clustering is also known as HCS (Hierarchical cluster analysis). It is un-supervised clustering techniques which attempt to make clusters hierarchy. It is divided into two categories which are Divisive and Agglomerative clustering.

Divisive Clustering: In this clustering technique, we allocate all of the inspection to one cluster and later, partition that single cluster into two similar clusters. Finally, we continue repeatedly on every cluster till there would be one cluster for every inspection.

Agglomerative method: It is bottom up approach. We allocate every inspection to their own cluster. Later, evaluate the distance between every cluster and then amalgamate the most two similar clusters. Repeat steps second and third until there could be one cluster left. The algorithm is given below

X set A of objects {a1, a2,.........an}
Distance function is d1 and d2
For j=1 to n
dj={aj}
end for
D= {d1, d2,.....dn}
Y=n+1
 while D.size>1 do
- *(dmin1, dmin2)=minimum distance (dj, dk) for all dj, dk in all D*
- *Delete dmin1 and dmin2 from D*
- *Add (dmin1, dmin2) to D*
- *Y=Y+1*
  end while

It is essential to find out proximity matrix consisting distance between every point utilizing distance function before clustering implementation. There is three methods is used to find out the distance between clusters.

Single Linkage: Distance between two different clusters is defined as the minimum distance between two points in every cluster. For example a and b is the two clusters and distance is given by this formula:

$$L\ (a,\ b) = min\ (D(Y_{ai},\ Y_{bj}))   \qquad (1)$$

Complete Linkage: Distance between two different clusters is defined as the longest distance between two points in every cluster. For example a and b is the two clusters and distance is given by this formula:

$$L\ (a,\ b) = max\ (D(Y_{ai},\ Y_{bj}))   \qquad (2)$$

Average Linkage: Distance between two different clusters is defined as the average distance between two points in every cluster. For example a and b is the two clusters and distance is given by this formula:

$$L\ (a,\ b) = \frac{1}{NaNb}\sum_{i=1}^{Na}\sum_{i=1}^{Nb} D(Yai, Ybi)   \quad (3)$$

2.1.2   K-modes clustering

Clustering is a data mining technique which uses unsupervised learning, whose major aim is to categorize the data features into a distinct type of clusters in such a way that features a group would more alike than the features in different clusters. K-means technique is an extensively used clustering methodologies for large numerical dataset analysis. In this, the dataset is grouped into k-clusters. In this, there are diverse clustering techniques available but the assortment of appropriate clustering algorithm rely on the nature and type of data. Our major objective of this work is to differentiate the accident places on their rate occurrence. Let's assume that X and Y is a matrix of m by n matrix of categorical data. The straightforward closeness coordinating measure amongst X and Y is the quantity of coordinating quality estimations of the two values. The more noteworthy the quantity of matches is more the comparability of two items. K-modes algorithm can be explained as:

$$d\ (X_i, Y_i) = \sum_{i=1}^{m} \delta(X_i, Y_i)   \qquad (4)$$

$$Where\ \delta(X_i, Y_i) = \begin{cases} 1, & if\ X_i = Y_i \\ 0, & if\ X_i \neq Y_i \end{cases}   \qquad (5)$$

2.2 Classification techniques

2.2.1 Lazy classifier

Lazy classifier saves the training instances and do no genuine work until classification time. Lazy classifier is a learning strategy in which speculation past the preparation information is postponed until a question is made to the framework where the framework tries, to sum up the training data before getting queries. The main ad-vantage of utilizing a lazy classification strategy is that the objective scope will be exacted locally, for example, in the k-nearest neighbor. Since the target capacity is approximated locally for each question to the framework, lazy classifier frameworks can simultaneously take care of various issues and arrangement effectively with changes in the issue field. The burdens with lazy classifier incorporate the extensive space necessity to store the total preparing dataset. For the most part boisterous pre-paring information expands the case bolster pointlessly, in light of the fact that no idea is made amid the preparation stage and another detriment is that lazy classification strategies are generally slower to assess, however, this is joined with a quicker preparing stage.

2.2.2 K-Star

The K star can be characterized as a strategy for cluster examination which fundamentally goes for the partition of n perception into k-clusters, where every perception has a location with the group to the closest mean. We can depict K star as an occurrence based learner which utilizes entropy as a separation measure. The advantages are that it gives a predictable way to deal with the treatment of genuinely esteemed attributes, typical attributes, and missing attributes. K star is a basic, instance-based classifier, like K-Nearest Neighbor (K-NN). New data instance, x, are doled out to the class that happens most every now and again among the k closest information focuses, yj, where j = 1, 2… k. Entropic separation is then used to recover the most comparable occasions from the informational index. By the method for entropic remove as a metric has a number of advantages including treatment of genuinely esteemed qualities and missing qualities. The K star function can be ascertained as:

$$K^*(yi, x) = -ln\ P^*(yi, x) \qquad (6)$$

Where P* is the likelihood of all transformational means from instance x to y. It can be valuable to comprehend this as the likelihood that x will touch base at y by means of an arbitrary stroll in IC highlight space. It will be performed streamlining over the percent mixing proportion parameter which is closely resembling K-NN 'sphere of influence', before appraisal with other Machine Learning strategies.

2.2.3 IBK (K - nearest neighbor)

It's a k-closest neighbor classifier technique that utilizes a similar separation metric. The quantity of closest neighbors may be illustrated unequivocally in the object editor or determined consequently utilizing blow one

cross-approval center to a maximum point of confinement provided by the predetermined esteem. IBK is the knearest-neighbor classifier. A sort of divorce pursuit calculations might be used to quicken the errand of identifying the closest neighbors. A direct inquiry is a default yet promote decision blend ball trees, KD-trees, thus called "cover trees". The dissolution work used is a parameter of the inquiry strategy. The rest of the thing is alike one the basis of IBL—which is called Euclidean separation; different alternatives blend Chebyshev, Manhattan, and Minkowski separations. Forecasts higher than one neighbor may be weighted by their distance from the test occurrence and two unique equations are implemented for altering over the distance into a weight. The quantity of preparing occasions kept by the classifier can be limited by setting the window estimate choice. As new preparing occasions are included, the most seasoned ones are segregated to keep up the quantity of preparing cases at this size.

2.2.4 Decision tree

Random decision forests or random forest are a package learning techniques for regression, classification and other tasks, that perform by building a legion of decision trees at training time and resulting in the class which would be the mode of the mean prediction (regression) or classes (classification) of the separate trees. Random decision forests good for decision trees' routine of overfitting to their training set. In different calculations, the classification is executed recursively till each and every leaf is clean or pure, that is the order of the data ought to be as impeccable as would be prudent. The goal is dynamically speculation of a choice tree until it picks up the balance of adaptability and exactness. This technique utilized the 'Entropy' that is the computation of disorder data. Here Entropy $\vec{X}$ is measured by:

$$Entropy\ (\vec{X}) = -\sum_{i=1}^{n} \frac{|xi|}{|x|} log(\frac{|xi|}{|x|}) \qquad (7)$$

$$Entropy\ (i|\vec{X}) = \frac{|xi|}{|x|} log(\frac{|xi|}{|x|}) \qquad (8)$$

Hence so total gain = Entropy $(\vec{X})$ - Entropy $(i|\vec{X})$ (9)

Here the goal is to increase the total gain by dividing total entropy because of diverging arguments $\vec{X}$ by value i.

2.2.5 Multilayer perceptron

An MLP might be observed as a logistic regression classifier in which input data is firstly altered utilizing a non-linear transformation. In this, alteration deal the input dataset into space, and the place where this turn into linearly separable. This layer as an intermediate layer is known as a hidden layer. One hidden layer is enough to create MLPs.
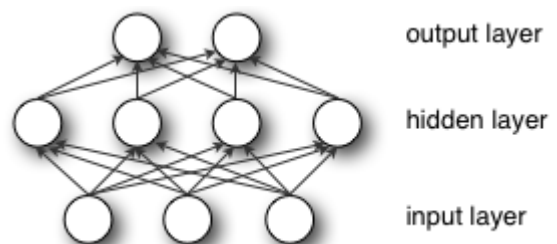
Formally, a single hidden layer Multilayer Perceptron (MLP) is a function of $f: Y^I \rightarrow Y^O$, where I would be the input size vector x and O is the size of output vector $f(x)$, such that, in matrix notation

$$F(x) = g(\theta^{(2)} + W^{(2)}(s(\theta^{(1)} + W^{(1)}x))) \qquad (10)$$

3 Description of dataset

The traffic accident data is obtained from the online data source for Leeds UK [8]. This data set comprises 13062 accident which happened since last 5 years from 2011 to 2015. After carefully analyzed this data, there are 11 attributes discovered for this study. The dataset consist attributes which are Number of vehicles, time, road surface, weather conditions, lightening conditions, casualty class, sex of casualty, age, type of vehicle, day and month and these attributes have different features like casualty class has driver, pedestrian, passenger as well as same with other attributes with having different features which was given in data set. These data are shown briefly in table 2.

Table 2.

S.NO.	Attribute	Code	Value	Total	Casualty Class		
					Driver	Passenger	Pedestrian
1.	No. of vehicles	1	1 vehicle	3334	763	817	753
		2	2 vehicle	7991	5676	2215	99
		3+	>3 vehicle	5214	1218	510	10
2.	Time	T1	[0-4]	630	269	250	110
		T2	[4-8]	903	698	133	71
		T3	[6-12]	2720	1701	644	374
		T4	[12-16]	3342	1812	1027	502
		T5	[16-20]	3976	2387	990	598
		T6	[20-24]	1496	790	498	207
3.	Road Surface	OTR	Other	106	62	30	13
		DR	Dry	9828	5687	2695	1445
		WT	Wet	3063	1858	803	401
		SNW	Snow	157	101	39	16
		FLD	Flood	17	11	5	0
4.	Lightening Condition	DLGT	Day Light	9020	5422	2348	1249
		NLGT	No Light	1446	858	389	198
		SLGT	Street Light	2598	1377	805	415
5.	Weather Condition	CLR	Clear	11584	6770	3140	1666
		FG	Fog	37	26	7	3
		SNY	Snowy	63	41	15	6
		RNY	Rainy	1276	751	350	174
6.	Casualty Class	DR	Driver				
		PSG	Passenger				
		PDT	Pedestrian				
7.	Sex of Casualty	M	Male	7758	5223	1460	1074
		F	Female	5305	2434	2082	788
8.	Age	Minor	<18 years	1976	454	855	667
		Youth	18-30 years	4267	2646	1158	462
		Adult	30-60 years	4254	3152	742	359
		Senior	>60 years	2567	1405	787	374
9.	Type of Vehicle	BS	Bus	842	52	687	102
		CR	Car	9208	4959	2692	1556
		GDV	GoodsVehicle	449	245	86	117
		BCL	Bicycle	1512	1476	11	24
		PTV	PTWW	977	876	48	52
		OTR	Other	79	49	18	11
10.	Day	WKD	Weekday	9884	5980	2499	1404
		WND	Weekend	3179	1677	1043	458
11.	Month	Q1	Jan-March	3017	1731	803	482
		Q2	April-June	3220	1887	907	425
		Q3	July-September	3376	2021	948	406
		Q4	Oct-December	3452	2018	884	549

4 Accurecy measurement

The accuracy is defined by different classifiers of provided dataset and that is achieved a percentage of dataset tuples which is classified precisely by help of different classifiers. The confusion matrix is also called as error matrix which is just layout table that enables to visualize the behavior of an algorithm. Here confusing matrix provides also an important role to achieve the efficiency of different classifiers. There are two class labels given and each cell consist prediction by a classifier which comes into that cell.

Table 1.

Confusion Matrix		
	Correct Labels	
	Negative	**Positive**
Negative	TN (True negative)	FN (False negative)
Positive	FP (False positive)	TP (True positive)

Now, there are many factors like Accuracy, sensitivity, specificity, error rate, precision, f-measures, recall and so on.

$TPR\ (Accuracy\ or\ True\ Positive\ Rate) = \frac{TN+TP}{All}$　　(11)

$FPR\ (False\ Positive\ Rate) = \frac{FP}{TN+FP}$　　　　(12)

$Error\ rate = see\ Accuracy$　　　　　　　(13)

$Precision = \frac{TP}{FP+TP}$　　　　　　　　(14)

$Sensitivity\ or\ Recall = \frac{TP}{FN+TP}$　　　(15)

$F\text{-}measures = 2\ (Precision*Recall)\ /\ (Precision + Recall)$
　　　　　　　　　　　　　　　(16)

And there are also other factors which can find out to classify the dataset correctly.

5 Results and discussion

Table 2 describe all the attributes available in the road accident dataset. There are 11 attributes mentioned and their code, values, total and other factors included. We divided total accident value on the basis of casualty class which is Driver, Passenger, and Pedestrian by the help of SQL.

5.1 Directed classification analysis

We utilized different approaches to classifying this bunch of dataset on the basis of casualty class. We used classifier which are Decision Tree, Lazy classifier, and Multilayer perceptron. We attained some result to few level as shown in table 3.

Table 3.

Classifiers	Accuracy
Lazy classifier(K-Star)	67.7324%
Lazy classifier (IBK)	68.5634%
Decision Tree	70.7566%
Multilayer perceptron	69.3031%

We achieved some results to this given level by using these three approaches and then later we utilized different clustering techniques which are Hierarchical clustering and K-modes.
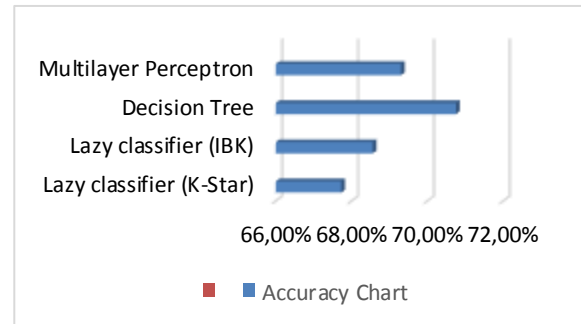

Figure 1: Direct classified Accuracy.

5.2 Analysis by using clustering techniques

In this analysis, we utilized two clustering techniques which are Hierarchical and K-modes techniques, Later we divided the dataset into 9 clusters. We achieved better results by using Hierarchical as compared to K-modes techniques.

5.3 Lazy Classifier Output

K-Star: In this, our classified result increased from 67.7324 % to 82.352%. It's sharp improvement in the result after clustering.

Table 4.

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.956	0.320	0.809	0.956	0.876	0.679	0.928	0.947	Driver
0.529	*0.029*	*0.873*	*0.529*	*0.659*	*0.600*	*0.917*	*0.824*	*Passenger*
0.839	0.027	0.837	0.839	0.838	0.811	0.981	0.906	Pedestrian

IBK: In this, our classified result increased from 68.5634% to 84.4729%. It's sharp improvement in the result after clustering.

Table 5.

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.945	0.254	0.840	0.945	0.890	0.717	0.950	0.964	Driver
0.644	0.048	0.833	0.644	0.726	0.651	0.940	0.867	Passenger
0.816	0.018	0.884	0.816	0.849	0.826	0.990	0.946	Pedestrian

5.4 Decision Tree Output

In this study, we used Decision Tree classifier which improved the accuracy better than earlier which we achieved without clustering. We achieved accuracy 84.4575 % which is almost more than 15% earlier without clustering.

Table 6.

TP Rate	FP Rate	Precis ion	Recall	F-Me asure	MCC	ROC Area	PRC Area	Class
0.922	0.220	0.856	0.922	0.888	0.717	0.946	0.961	Driver
0.665	0.057	0.814	0.665	0.732	0.652	0.936	0.861	Passenger
0.868	0.027	0.841	0.868	0.855	0.830	0.988	0.939	Pedestrian

5.5   Multilayer Perceptron Output

In this study, our accuracy increased from 69.3031% to 78.8301% after using clustering technique.

Table 7.

TP Rate	FP Rate	Precis ion	Recall	F-Me asure	MCC	ROC Area	PRC Area	Class
0.929	0.338	0.796	0.929	0.857	0.627	0.892	0.916	Driver
0.452	0.036	0.824	0.452	0.584	0.520	0.855	0.720	Passenger
0.849	0.053	0.726	0.849	0.783	0.746	0.955	0.818	Pedestrian

We achieved error rate, precision, TPR (True positive rate), FPR (False positive rate), Precision, recall for every classification techniques as shown in given tables and also achieved different confusion matrix for different classification techniques. We can see the performance of different classifier techniques by the help of confusion matrix.

Here in the next table, we have shown the overall accuracy of analysis with clustering with the help of table 8, as we can compare this table from the previous table that our accuracy increased in each classification techniques after doing clustering.

Table 8.

Classifiers	Accuracy
Lazy classifier (K-Star)	82.352%
Lazy classifier (IBK)	84.4729%
Decision Tree	84.4575%
Multilayer perceptron	78.8301%

We have shown accuracy level of table 8 in given figure 2 with the help of chart and we can see from the chart that it's improved after doing clustering in accuracy chart also.
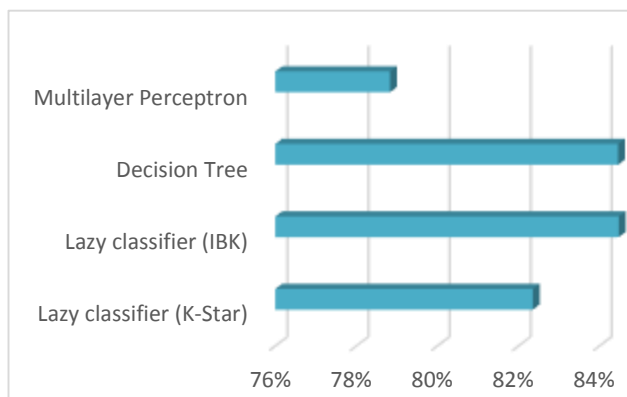
Figure 3: Accuracy after clustering.

As we can see from table 3 and 8 that our accuracy level increased after clustering. We have shown comparison chart in fig. 3 without clustering and with clustering.

6   Conclusion and future suggestions
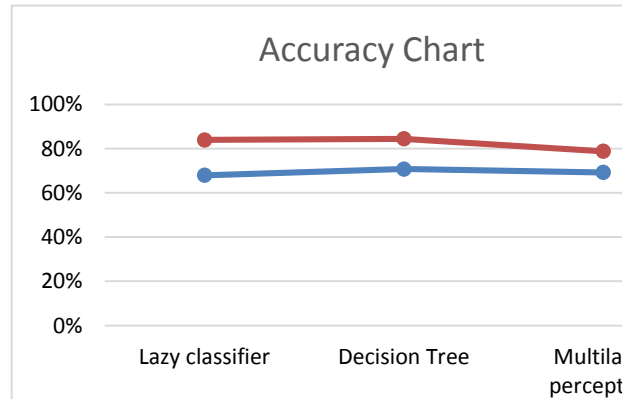
Figure 2: Compared accuracy chart with clustering and without clustering.

In this study, we analyzed dataset without clustering and with clustering. Generally, we use clustering techniques on accident dataset that could find a homogenous pattern of same accident data which could be used to increase the accuracy of classifiers. We achieved a better result when we used hierarchical clustering as compared to k mode clustering techniques. We used different classifier such as Decision Tree, Lazy classifier, and Multilayer perceptron to classify our dataset and they have shown optimized performance after clustering as well as we used different classifiers technique also but they did not show better accuracy as these classifiers shown. If accuracy would be higher our classified result will be better. We achieved better accuracy on the basis of casualty class (Driver, Passenger, and Pedestrian) and we can see from tables that which factors are affecting more in an accident on the basis of casualty class. The future work will include a comprehensive evaluation of all clusters with an aim to determine the various other factors and locations behind traffic accident.

7   References

[1]   Depaire B, Wets G, Vanhoof K. Traffic accident segmentation by means of latent class clustering. Accid Anal Prev.2008;40(4):1257–66.

[2]   Miaou SP. The Relationship between truck accidents and geometric design of road sections-poisson versus negative binomial regressions. Accid Anal Prev. 1994;26(4):471–82.

[3]   Miaou SP, Lum H. Modeling vehicle accidents and highway geometric design relationships. Accid Anal Prev. 1993;25(6):689–709.

[4]   Ma J, Kockelman K. Crash frequency and severity modeling using clustered data from Washington state. In: IEEE intelligent transportation systems conference. Toronto; 2006.

[5]   Savolainen P, Mannering F, Lord D, Quddus M. The statistical analysis of highway crash-injury

severities: a review and assessment of methodological alternatives. Accid Anal Prev. 2011;43(5):1666–76

[6] Abellan J, Lopez G, Ona J. Analyis of traffic accident severity using decision rules via decision trees. Expert Syst Appl. 2013;40(15):6047–54.

[7] Kumar S, Toshniwal D. A data mining approach to characterize road accident locations. J Mod Transp. 2016;24(1):62–72.

[8] Chang LY, Chen WC. Data mining of tree based models to analyze freeway accident frequency. J Saf Res. 2005;36(4):365–75.

[9] Kashani T, Mohaymany AS, Rajbari A. A data mining approach to identify key factors of traffic injury severity. Promet- Traffic Transp. 2011;23(1):11–7.

[10] Kumar S, Toshniwal D. Analyzing road accident data using association rule mining, International conference on computing, communication and security. Mauritius: ICCCS-2015; 2015. doi:10.1109/CCCS.2015.7374211

[11] Oña JD, López G, Mujalli R, Calvo FJ. Analysis of traffic accidents on rural highways using latent class clustering and Bayesian networks. Accid Anal Prev. 2013;51(2013):1–10.

[12] Kumar S, Toshniwal D. A data mining framework to analyze road accident data. J Big Data. 2015;2(1):1–26.

[13] Karlaftis M, Tarko A. Heterogeneity considerations in accident modeling. Accid Anal Prev. 1998;30(4):425–33

[14] Zajac, S., Ivan, J., 2003. Factors influencing injury severity of motor vehicle crossing pedestrian crashes in rural Connecticut. Accident Anal. Prev. 35 (3), 369–379.

[15] Bedard, M., Guyatt, G., Stones, M., Hirdes, J., 2002. The independent contribution of driver, crash, and vehicle characteristics to driver fatalities. Accident Anal. Prev. 34 (6), 717–727.

[16] Zhang, J., Lindsay, J., Clarke, K., Robbins, G., Mao, Y., 2000. Factors affecting the severity of motor vehicle traffic crashes involving elderly drivers in Ontario. Accident Anal. Prev. 32 (1), 117–125.

[17] Lee C, Saccomanno F, Hellinga B (2002) Analysis of crash precursors on instrumented freeways. Transp Res Rec. doi:10. 3141/1784-01.

[18] Chen W, Jovanis P (2000) Method for identifying factors contributing to driver-injury severity in traffic crashes. Transp Res Rec. doi:10.3141/1717-01

[19] Tan PN, Steinbach M, Kumar V (2006) Introduction to data mining. Pearson Addison-Wesley, Boston.

[20] Barai S (2003) Data mining application in transportation engineering. Transport 18:216–223. doi:10.1080/16483840.2003. 10414100.

[21] Shaw, M. J., Subramaniam, C., Tan, G. W., & Welge, M. E. (2001). Knowledge management and data mining for marketing. Decision Support Systems, 31(1), 127–137.

[22] Rygielski, C., Wang, J.-C., & Yen, D. C. (2002). Data mining techniques for customer relationship management. Technology in Society, 24(4), 483–502.

[23] Valafar, H., & Valafar, F. (2002). Data mining and knowledge discovery in proton nuclear magnetic resonance (1H-NMR) spectra using frequency to information transformation. Knowledge-Based Systems, 15(4), 251– 259.

[24] Geurts K, Wets G, Brijs T, Vanhoof K (2003) Profiling of high frequency accident locations by use of association rules. Transp Res Rec. doi:10.3141/1840-14.

[25] Depaire B, Wets G, Vanhoof K (2008) Traffic accident segmentation by means of latent class clustering. Accid Anal Prev 40:1257–1266. doi:10.1016/j.aap.2008.01.007.

[26] Kwon OH, Rhee W, Yoon Y (2015) Application of classification algorithms for analysis of road safety risk factor dependencies. Accid Anal Prev 75:1–15. doi:10.1016/j.aap.2014.11.005

[27] Kashani T, Mohaymany AS, Rajbari A (2011) A data mining approach to identify key factors of traffic injury severity. Promet- Traffic Transp 23:11–17. doi:10.7307/ptt.v23i1.144

[28] Tiwari Prayag, Brojo Kishore Mishra, Sachin Kumar and Vivek Kumar. "Implementation of n-gram Methodology for Rotten Tomatoes Review Dataset Sentiment Analysis," International Journal of Knowledge Discovery in Bioinformatics (IJKDB) 7 (2017): 1, accessed (March 02, 2017), doi:10.4018/IJKDB.2017010103.

[29] Prayag Tiwari. Article: Comparative Analysis of Big Data. International Journal of Computer Applications 140(7):24-29, April 2016. Published by Foundation of Computer Science (FCS), NY, USA

[30] P. Tiwari, "Improvement of ETL through integration of query cache and scripting method," 2016 International Conference on Data Science and Engineering (ICDSE), Cochin, India, 2016, pp. 1-5.doi:10.1109/ICDSE.2016.7823935

[31] P. Tiwari, "Advanced ETL (AETL) by integration of PERL and scripting method," 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India,2016,pp.1-5.doi:10.1109/INVENTIVE.2016.7830102

[32] P. Tiwari, S. Kumar, A.C. Mishra, V. Kumar, B. Terfa. Improved Performance of Data Warehouse. "International Conference on Inventive Communication and Computational Technologies (ICICCT 2017)"

[33] Tiwari P, Mishra AC, Jha AK (2016) Case Study as a Method for Scope Definition. Arabian J Bus Manag Review S1:002

[34] Tiwari P., Kumar S., K. Denis. Road user specific analysis of traffic accidents using Data mining techniques, Communications in Computer and Information Science (Springer)

[35] Hassan Abdelwahab and Mohamed Abdel-Aty Transportation Research Record: Journal of the Transportation Research Board 2001 1746:, 6-13

[36] K Sachin, Shemwal V.B., Tiwari P, K Denis. A Conjoint Analysis of Road Accident Data using K-modes Clustering and Bayesian Networks, Annals of Computer Science and Information System

[37] Virmani J, Dey N, Kumar V (2015) PCA-PNN and PCA-SVM based CAD systems for breast density classification. Applications of intelligent optimization in biology and medicine: current trends and open problems"

[38] Karaa WBA, Ashour AS, Sassi DB, Roy P, Kausar N, Dey N (2016) MEDLINE text mining: an enhancement genetic algorithm based approach for document clustering. Applications of intelligent optimization in biology and medicine. Springer International Publishing, Switzerland, pp 267–287

[39] Dey N, Ashour AS, Beagum S, Pistola DS, Gospodinov M, Gospodinova EP, Tavares JMRS (2015) Parameter optimization for local polynomial approximation based intersection confidence interval filter using genetic algorithm: an application for brain MRI image de-noising. J Imaging 1:60–84

[40] Dey, Nilanjan, et al. "Firefly algorithm for optimization of scaling factors during embedding of manifold medical information: an application in ophthalmology imaging." Journal of Medical Imaging and Health Informatics 4.3 (2014): 384-394.

[41] Naik, Anima, et al. "Social group optimization for global optimization of multimodal functions and data clustering problems." Neural Computing and Applications (2016): 1-17.

[42] Van Hoof, J., et al. "Ambient assisted living and care in The Netherlands: the voice of the user." Pervasive and Ubiquitous Technology Innovations for Ambient Intelligence Environments 205 (2012).

[43] Mokhtar, Sonia Ben, et al. "Interoperable semantic and syntactic service discovery for ambient computing environments." Innovative Applications of Ambient Intelligence: Advances in Smart Systems: Advances in Smart Systems 213 (2012).

[44] Zappi, Piero, et al. "Collecting datasets from ambient intelligence environments." Innovative Applications of Ambient Intelligence: Advances in Smart Systems. IGI Global, 2012. 113-127.

Distributed Fault Tolerant Architecture for Wireless Sensor Network

Siba Mitra and Ajanta Das
Department of Computer Science and Engineering, Birla Institute of Technology
Mesra, Kolkata Campus, India
E-mail: sibamitra@bitmesra.ac.in, ajantadas@bitmesra.ac.in

Smart applications use wireless sensor network for surveillance of any physical property of that area, to realize the vision of ambient intelligence. Since wireless sensor network is resource constrained and for unattended deployment scenario faults are quite trivial. Reliability and dependability of the network depends on its fault detection, diagnosis and recovery techniques. Detecting faults in wireless sensor network is challenging and recovery of faulty nodes is very crucial task. In this research article, a distributed fault tolerant architecture is proposed. This paper also proposes fault recovery algorithms. Recovery actions are initiated based on fault diagnosis notification. The novelty of this paper is to perform recovery actions using data checkpoints and state checkpoints of the node, in a distributed manner. Data checkpoint helps to recover the old data and the state checkpoint tells the previous trust degree of the node. Moreover, the result section explains, that after replacement of a faulty node, the topology and connectivity between rests of the nodes are maintained in WSN.

Povzetek: Opisana je arhitektura brezžičnega senzorskega omrežja.

1   Introduction

The use of wireless sensor network (WSN) nowadays has seen a huge growth in the field of ambience intelligence. WSN is resource constrained in nature but can be integrated with any system by using Dynamic Adaptive System Infrastructure (DAiSI) proposed by Klus and Niebuhr (2009) in [11]. Component reconfiguration and dynamic integration can be done with the help of this. Another interesting application that uses WSN to detect and track presence of human and human motion in an environment is presented in the research of Graham et al. (2011) in [7]. It is also shown in the work that appropriate device placement scheme can improve network performance.

The unit of WSN is a tiny sensor node, which communicates to other sensor nodes through radio transmission. Small sensor nodes constituting of sensing unit, tiny memory, a microcontroller, a transceiver and an omni-directional antenna are deployed in the target area. Sensor nodes send relevant data to the nearest base station (BS), which is used for some meaningful decision making. Fault in WSN is trivial because of its resource constraints and unattended deployment scenario. Therefore to make the WSN reliable and dependable, fault tolerance must be implemented in it.

Various types of node faults are classified in the Figure 1. Faults in WSN can be permanent, transient or intermittent in nature. Fault management is the process to monitor the nodes, detect and diagnose fault and perform necessary recovery tasks to make WSN fault tolerant. Permanent failures generally have no option for recovery but for transient and intermittent faults the recovery actions should prevail. Proper recovery schedules should be there for occurred faults to make it fault tolerant and

help application to make correct decision, even in presence of faults. Some of the critical factors in recovery process are the available residual energy of the sensor node, the network traffic scenario, connectivity issues and current topological structure of the WSN.

A distributed adaptive fault detection scheme for WSN is proposed in [28], where each node detect any unnatural event by fetching neighbor sensor nodes' reading with queries. A three-bit control packet exchange is done during the fault detection phase in order to reduce communication overhead. Here moving average filter was employed for implementing fault tolerance in WSN. The article claims to have reached high detection accuracy and low false alarm rate.

WSN may comprise of static or mobile sensor nodes. Borawake-Satao and Prasad (2017) [4] presents a study of effects of sensor node mobility on various performance parameters of WSN. A proposal of mobile sink with mobile agent mobility model for WSN is also presented. In [12] Kumar and Nagarajan (2013) proposed Incorporated Network Topological control and Key management (INTK) for relay nodes of WSN, for privacy and security measures in the network. The proposed scheme includes hierarchical routing architecture in WSN for better performance and security. Another novel research proposal by Mukherjee et al. (2016) [22] presents a model for disaster aware mobile Unmanned Aerial Vehicle (UAV) for flying Ad-hoc network. The nodes can perform collaborative job by relaying useful message in a post-disaster situation of any ecosystem.
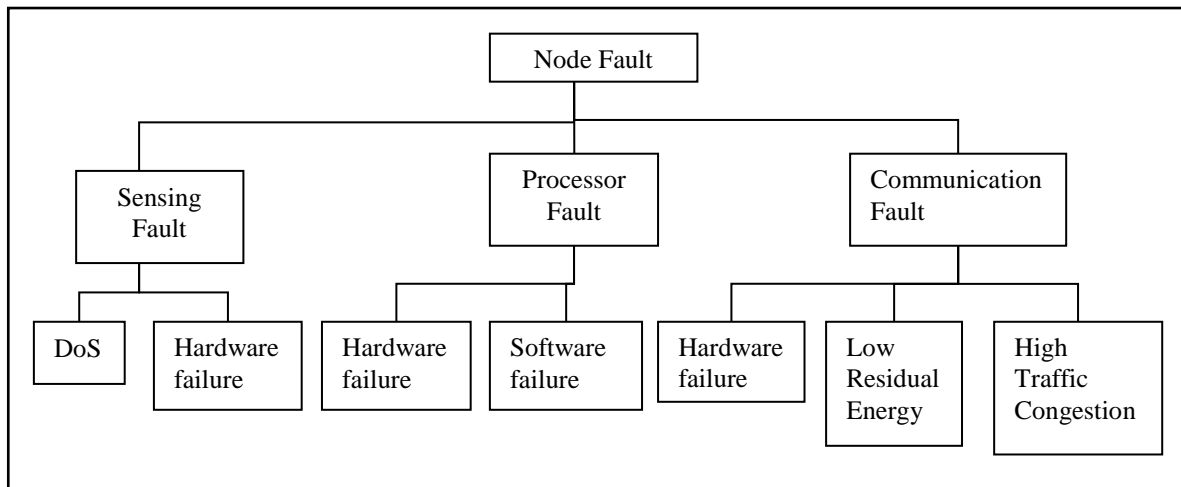
Figure 1: Node Fault Classification.

The analytical comparison presented in [3] by Bathla & Jindal 2016, where two distributed self-healing recovery techniques, Recovery by In-ward Motion (RIM) and Least Disruptive Topology Repair (LeDiR) are analyzed and compared with respect to their efficiency in various applications. Both the approaches are distributed in nature.

The RIM method aims to replace a failed node by a healthy node, by moving the latter towards the former's location. Here all nodes must have a 1-hop neighbor list and should be aware of their neighbor's locality and proximity. Now the goal of LeDiR is to restore the connectivity among the sensor nodes. But it also takes care that after the recovery action the shortest path length among the nodes is not extended compared to the pre-failure topology.

A fault recovery algorithm for WSN is proposed in [13] by Lakamana et al. 2015, which enhances the routing efficiency in the WSN. Battery depletion is a major issue and that is taken care of over here by reducing the number of node replacements and by reusing the historic routing paths. According to the authors' claim the network longevity is increased over here.

In WSN, now sensor node(s) when gets disconnected from the network due to some reason may generate partitions or isolations in the network, which is not good for reliability and dependability of the network. Moreover, it is crucial to maintain the connectivity throughout its longevity. So the objective of this research is to design a distributed fault tolerant architecture for WSN, which includes fault detection, diagnosis and recovery. However the architecture proposed here is an improved version of a fault tolerant framework already proposed in our previous research work available in [18] by Mitra & De Sarkar (2014). Moreover this article also proposes a novel fault recovery model, which is integrated with the proposed architecture. This research also proposes some algorithms for connectivity maintenance, and recovery tasks to be performed. The

novelty of the proposed recovery technique is to initiate the recovery actions after proper diagnosis of the detected fault. Recovery tasks are done once after it gets notification from the diagnosis layer about the fault-type. The recovery model has two phases; the first one being set action and start recovery.

The remainder part of the article is sub-divided into sections namely, related work done in the current field, followed by proposed Distributed Fault Tolerant Architecture and supporting fault recovery architecture and algorithms; and then the results and discussions section is presented. Finally, the conclusion and the references to the article are presented.

2   Related work

This section mainly presents some of the valuable researches carried out by many scholars in the field of WSN. Many existing fault management techniques are available, which are used for fault tolerance in WSN. A review work of the same is presented in our previous research work, Mitra, De Sarkar and Roy (2012) in ref. [20] and a few of them are mentioned here also. Moreover in this article a study on some of the existing recovery schemes is presented. Data communication is important factor in WSN hence routing decision is significant. Leskovec, et al. (2005) [14] proposed a novel link quality estimation model for sensor network, which uses link quality map to estimate a link in sensor network. This work also optimizes power consumption of radio transmission signal, while scheduling the communication task and taking routing decision.

An analytical study and comparison of various recovery techniques are presented in our recent research work, in [19] (Mitra, Das & Mazumdar 2016). Some of those recovery schemes are also discussed briefly here. Among them CRAFT (Checkpoint/Recovery-based scheme for Fault Tolerance) [26] for WSN, proposed by Saleh, Eltoweissy and Agbaria (2007) is studied; another scheme proposed by Ma, Lin, Lv & Wang (2009) [16] called ABSR, which recovers some compromised sensor

nodes in a heterogeneous sensor network. Various types of sensor nodes each playing specific role are used here. Reghunath, Kumar & Babu (2014) proposed Fault Node Recovery (FNR) algorithm, which is a combination of Genetic algorithm with Grade Diffusion algorithm. A rank based replacement strategy for the sensor nodes is presented in [25].

In [6] Chen, Kher & Somani (2006) proposed DLFS (distributed localized fault sensors) detection algorithm, for locating and identifying faulty nodes in WSN. Each node can be either in good health or can be faulty depending upon the node behavior. The technique here uses probabilistic approach. The implemention of the algorithm claim that execution complexity of the same is much low and detection accuracy is high. Haboush, Mohanty, Pattanayak and Al-Tarazi (2014) [8] have proposed a faulty node replacement algorithm for hybrid WSN. Mobile sensor nodes are considered over here. Any node having low residual energy may seek a replacement; after replacement maintenance of the topology etc. are taken care of. Redundancy is used to avoid faulty results and also adaptive threshold policy is employed for rectification of the faults and optimizing the network lifetime. The research in [2] Akbari et al. (2010) presents a survey of faults in WSN due to energy crunch and the role of cellular architecture and clustering for network sustain purpose. The cluster-based fault detection and recovery techniques was observed to be quite efficient, robust and fast for WSN sustain and longevity. Another cluster maintenance technique is designed by them for nodes having energy crunch as mentioned in [1] (Akbari, Dana, Khademzadeh & Beikmahdavi 2011). First of all, nodes with highest residual energy are selected as primary cluster head, and nodes second in residual energy becomes the secondary cluster head. So the technique is energy aware in nature and consequentially selects the cluster head as per the nodes' residual energy.

An FNR algorithm is proposed by Brahme, Gadadare, Kulkarni, Surana & Marathe (2014) in [5], for fault recovery in WSN to enhance network lifetime. Researchers employed genetic algorithm and grade diffusion algorithm for designing the scheme. Moreover researchers, Mishal, Narke, Shinde, Zaware & Salve (2015) in [17] have worked upon FNR and improved it performing lesser number of node replacement for fault recovery, and basically old routing paths are reused; however better result is claimed over here. A proposal on a distributed fault detection algorithm for detecting coverage holes in the WSN is presented in Kang et al. 2013 [9]. The research do not maintain any node coordinates. The critical information of a node can be collected from the neighbors and that can be used for detection and recovery purpose for WSN. On demand checkpoint based recovery technique for WSN is proposed in [23] by Nithilan & Renold (2015). In this scheme checkpoint coordination and non-blocking checkpoint is used for consistency and some backup nodes maintains and checks the health of a node by monitoring the checkpoints. A localized tree based method for fault detection is proposed by Wan, Wu & Xu

(2008) [27]. The recovery scheme uses elected new parent technique for avoiding isolation of children node of the tree. This technique enhances the network lifetime.

The main objective of this research work is to design a distributed fault tolerant architecture for WSN, with intrinsic parts for fault detection, diagnosis and recovery. In this research we mainly concentrate to propose a distributed fault recovery model for WSN with a set of algorithms, which are employed for performing node, data and network recovery. For fault detection, existing detection algorithm proposed in our previous work in [18] is used. Thereafter the proposed recovery technique is employed to maintain the fault tolerance. The major job is to increase the reliability and dependability of the WSN for correct decision making. The novelty of this research work is to perform recovery actions, using data checkpoints and state checkpoints of the node, in a distributed manner. Also topology maintenance is being performed by each node during the recovery process.

3 Proposed distributed fault tolerant architecture for WSN

This section details on a proposal of a fault tolerant framework for WSN. Event detection is important for implementing fault tolerance in WSN, where the event can be presence of hole in the network. A distributed, lightweight, hole detection algorithm proposed by Nguyen et al. (2016) in [24] monitors and reports about any hole in the network. The present proposal is an improvisation of an already proposed framework mentioned in Mitra et al (2014). The architecture as mentioned in Figure 2 can be embedded in each sensor node of WSN and the node can independently perform fault management in a distributed way.

In a centralized system fault management scheme there is a central manager, who monitors and controls the network. So each node has to report the central manager with relevant data for fault tolerance in the WSN. Therefore too much of communication will result and in the WSN huge overhead will be incurred in terms of energy and bandwidth, which may affect network performance. In centralized approach the traffic flow is towards a single central manager creating overheads and resulting in bottlenecking. However this is not desirable in WSN since it is resource constrained and infrastructure less. This critical bottleneck problem can be avoided in the distributed architecture of fault management scheme. In distributed fault management, the network is partitioned and self fault management is implemented. Moreover in comparison with the centralized system the communication cost is less in distributed system. Therefore this research work mainly aims for distributed architecture. The proposed architecture has three main phases viz. *Fault Detection, Fault Diagnosis* and *Fault Recovery* respectively.

3.1 Fault detection

The detection phase has three significant tasks namely *Node and Link Monitoring, Fault Isolation* and *Fault*

Prediction. Fault detection algorithm is already proposed in our previous work presented in Mitra et al. (2014). A brief discussion is presented hereafter. All the tasks are computed in an energy aware mode. In the monitoring stage the sensor node listener carefully monitors and examines the health of the sensor node and detects if any unnatural event occurs; and then it scans the attributes of the event; it also evaluates some useful parameter-value required for detecting faults in the node. First of all a neighbor table for each node is created and after that node performs self-checking. Sensor nodes evaluates own tendency by comparing the average of neighbors' reading with the own read value. Again the nodes do a similar comparison with its own previous read value and current value. The tendency of the node quantizes whether it is trustworthy or not. If a node is not trustworthy then the trust degree (TD) value is zero and if it is trustworthy then the TD value is one. So the TD value isolates rather detects the fault in the WSN.

In the *Prediction* module the residual energy analysis of the node is carried out and any fault-to-be are forecasted. The forecast is done on the basis of some comparative study of the fault evaluation parameters namely residual energy of the node. If the residual energy of the node goes below a threshold then the built-in fault predictor invokes two actions; firstly it broadcast the information of its low energy state. Secondly some query packets are broadcasted asking for a node with high residual energy for offloading its own responsibility. Finally the node is sent to sleep mode.
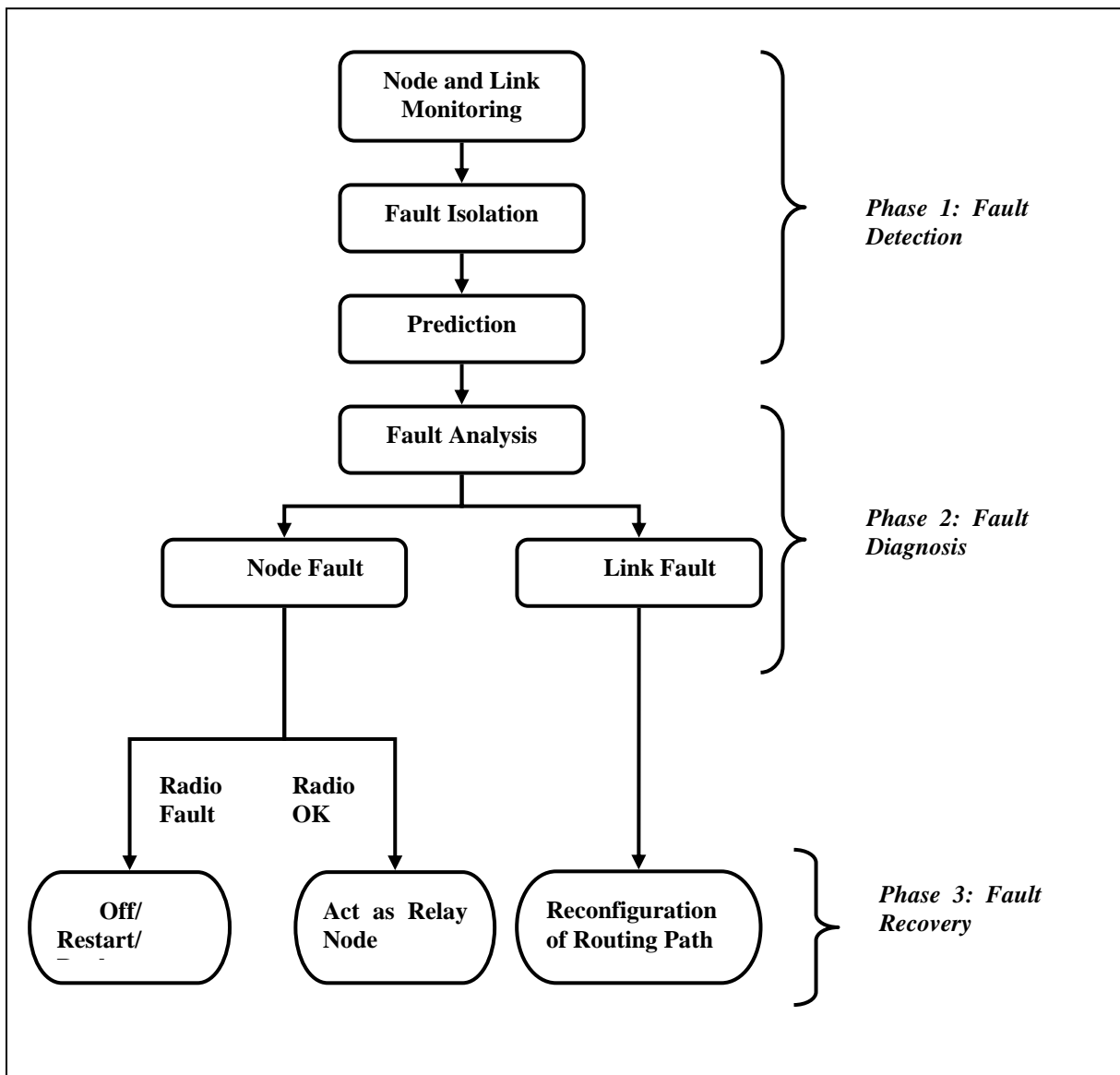
Figure 2: Distributed Fault Tolerant Architecture for WSN.

3.2   Fault diagnosis

The second phase of the fault tolerant architecture is *fault diagnosis* and it is done after the analysis of the occurred event. Fault analysis is a reactive process, and the fault category in WSN can be either a node fault or communication fault. For diagnosing the node fault, the assigned TD value is taken into consideration as available in [18]. Evaluation of TD value of a node is computed on the basis of self analysis and neighbor analysis for a fixed number of iterates. And depending on the iteration count the decision of node fault is finalized. Now for communication fault diagnosis, two critical parameters received signal strength (RSS) and link utilization of the sensor nodes are taken into account. The average RSS of all the neighbors of the sensor node are computed for communication fault analysis. Moreover the sensor node also computes the average link utilization parameter to check self performance. Once self-fault diagnosis is completed then a notification is forwarded to the next phase i.e. *Fault Recovery.*

4   Proposed fault recovery scheme

This section presents the fault recovery scheme for WSN. This scheme can be integrated in each sensor node such that distributed fault recovery is possible. The recovery process is invoked when a sensor node is suffering of some fault. The next sub-sections present the network model and the fault recovery model.

4.1   Network model

The WSN model in this research work can be represented as a graph structure, $G(S, E)$, where S is a set of sensor nodes $S = \{S_1, S_2, ... S_n\}$, which is deployed in random or planned way in the target area. The area can be represented as a two-dimensional plane whose origin is (x_0, y_0). Now $E = \{E_1, E_2, ... E_n\}$ is a set of communication links in between a pair of nodes S_i and S_j, which transmits within its communication range but has a sensing range lesser than communication range, in [15] given by $R_S < R_C$ where R_S and R_C, are sensing range and communication range respectively. The necessary condition for a node S_i to transmit signal to a node S_j is the Euclidean distance between the two nodes should conform Equation 1. Each node maintains a list of neighbors, which may dynamically change with time as per availability of the node in the communication process. It is very general to perform low power transmission in WSN, where node's transmission power is directly proportional to the distance. To send data with good quality signal strength a node may have to adjust its transmission power. For the current problem scope if $P_{i,j}$ is the power of transmission for communication of S_i and S_j. It is quite obvious that Equation 2 will satisfy if and only if Equation 3 is true. Moreover the maximum value of transmission power is also limited. The assumption is that any node S_i will perform low power

transmission for nodes within R_C and may sometimes, as required, perform high power communication with S_j if and only if Equation 4 is true.

$$\|S_i S_j\| \le R_C \qquad \textit{Equation (1)}$$

$$P_{i,j} \le P_{k,r} \qquad \textit{Equation (2)}$$

$$\|S_i S_j\| \le \|S_k S_r\| \qquad \textit{Equation (3)}$$

$$R_S < \|S_i S_j\| < R_C \qquad \textit{Equation (4)}$$

4.2   Connectivity issue

Now not all the nodes can directly transmit data to the sink or BS; so any node unable to do the same will employ some intermediary forwarding parent nodes to send the data to the BS. At the run time one or more sensor nodes may not work properly due to faults and then the recovery actions of the nodes may be initiated to recover the node, data or network. Any recovering node may need to stop its scheduled tasks for self-recovery. In that case other affected neighbor nodes may have to update their own neighbor list and exclude the recovering node from any current activity. This scenario is explained in Figure 3.

It is well understandable from the figure that the normal nodes need to maintain the connectivity even in absence of the faulty nodes marked as black nodes. Hence the normal nodes have to find alternate suitable nodes within its communication range for forwarding data. But if it is unable to find one then it should perform a high power transmission to the nodes, which are in its communication range, rather than getting isolated. In the figure the dotted arrows demarcate the unstable or sometimes unavailable links. To transmit in high power the node should increase its transmission power by a multiplicative factor, somewhat proportionate to the increment in the distance given by $|D_{i,j} - D_{i,k}|$, where $D_{i,j}$ and $D_{i,k}$ are explained in Equation 5 and 6. In the equations i-th node transmits to k-th node in the place of recovering j-th node.

$$D_{i,j} = \|S_i S_j\| \qquad \textit{Equation (5)}$$

$$D_{i,k} = \|S_i S_k\| \qquad \textit{Equation (6)}$$

4.3   Fault recovery model

The proposed fault recovery model is depicted in Figure 4, where there is a *Fault Recovery Process*, which has two main phases viz. *Set Action* and *Start Recovery*. The *Fault Recovery Process* gets notification from the fault diagnosis layer along with the information on the fault type; depending upon that the *Set Action* decides on what kind of recovery activity has to be invoked. Again *Start Recovery* actually begins the specific recovery task. Faults can be due to hardware or software failure, bad link quality or power depletion of the node.
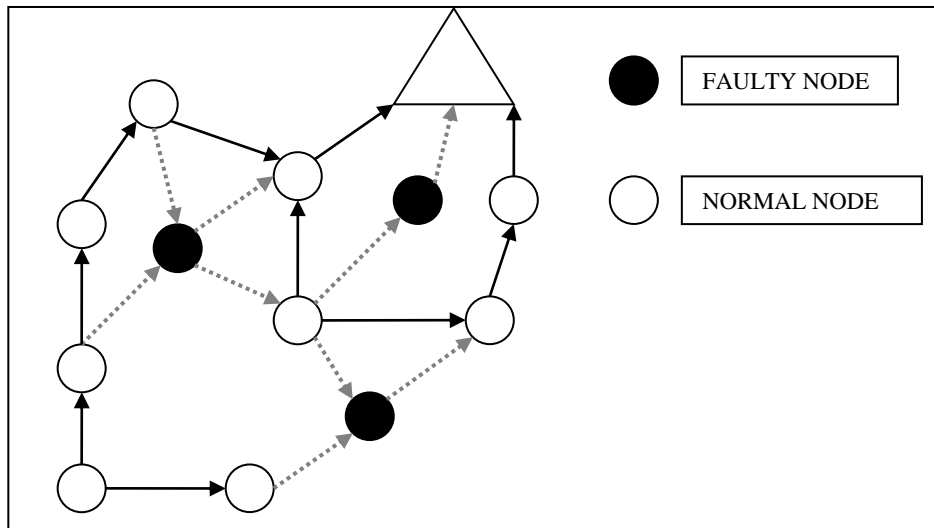
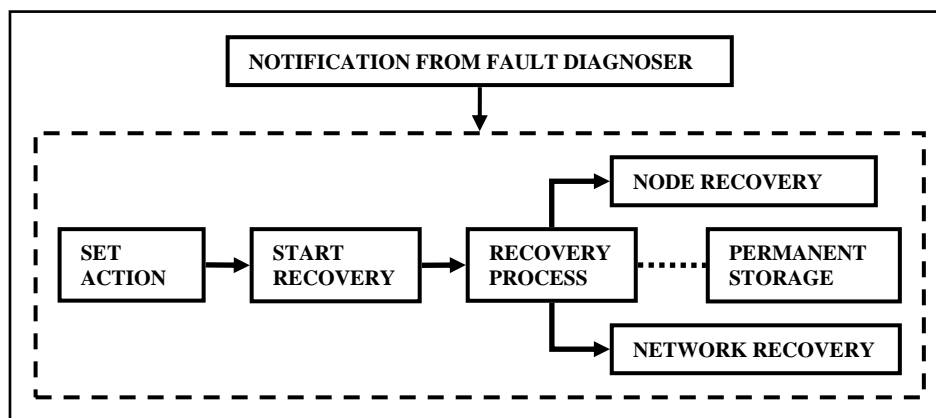Figure 3: Connectivity Issue in WSN.

Figure 4: Fault Recovery Model for WSN.

The Recovery Process performs and maintains node recovery and network recovery and it also communicates with the permanent storage for any kind of query checking. The permanent storage contains node status and data checkpoint before occurrence of the fault. The Recovery Process fetches the necessary information to perform the recovery tasks smoothly. It also performs various types of communication before the node gets reinitialized. Node recovery means data recovery from the node and also checking the node state and performing activities to preserve the node functionality. Network recovery deals with reconfiguring the network by performing the path quality estimation already proposed by Mitra, Roy & Das (2015) in [21]. The recovery jobs are vividly explained later in Algorithm 3.

5   Proposed algorithms for fault recovery

The proposed fault recovery algorithm consists of various parts, where each node carry out some self-checking task and some of them are already mentioned in [18]. Since the total process is being carried out in a distributed atmosphere so the nodes perform self-

evaluation and self-recovery. All the symbols and notations used in the algorithm are mentioned in Table 1. The proposed fault recovery scheme uses some sort of check pointing for performing the data recovery task. Each node maintains a data checkpoint and a state checkpoint by using two variables TD (Trust Degree) and DCkpt (Data Checkpoint) respectively, in the permanent storage of the node i.e. even if the node is restarted the data remains intact for future reference as mentioned in Algorithm 1 and presented in Figure 5. TD is already proposed, explained and used in our prior research mentioned in [18]. This previous work also presented a novel fault detection scheme, which is used over here to detect and diagnose faults. Now HF means sensing unit or hardware failure, where the node is unable to sense any ambient signal, or it may be transceiver fault that occurs when transmitter or receiver is not in working mode and microcontroller fault means when a node cannot perform its computations at par.

Each node has a delivered packet counter as DPC, which keeps the count of the delivered packets. Whenever a node delivers 200 packets it stores TD, as state checkpoint, in the permanent storage and stores the current read value of the node in DCkpt, as data

checkpoint, in permanent memory. After completion of these steps the DPC is reinitialized to zero so that it can again count the next set of 100 and 200 delivered packets respectively. The checkpoint creation process continues for each node until it goes to recovery state. When a node gets a notification from the fault diagnosis layer, that a fault has occurred, it fetches the fault type and performs some internal necessary actions that is mentioned in Algorithm 2 and presented in Figure 6. As mentioned the fault type can be either hardware fault (HF) or software fault (SF). Depending upon fault-type the recovery process is initiated as mentioned in Algorithm 3 and presented in Figure 7.

Notation	Meaning
S_i	i-th node
S_r	Node at Recovery mode
S_r.NBR	Neighbor list of S_r
S_i.CURR-VAL	Current reading of S_i
DCkpt	Data Checkpoint
DPC	Delivered packet count
TD	Trust degree
CR	Communication range
$P_{j,k}$	Power to transmit data from S_j to S_k
PRR	Packet reception ratio
PDR	Packet delivery ratio

Table 1: Symbols and Notations.

In all these cases the node needs a third party or human intervention to get the problem fixed. Software failure refers to logical or runtime faults in the software, which is again needs third party intervention. If the packet reception ratio (PRR) and packet delivery ratio (PDR) are much low then there must be some disturbances in data transmission and receiving; hence a communication failure may occur in near future. So the nodes goes for a self-recovery process. Finally the node has to be shut down if the residual energy is much less than the threshold value, which may be specified as per application requirement. The node recovery module for any faulty node starts with low-power transmission of probe packets to the neighbors, which again go for topology maintenance as mentioned in Figure 8 as Algorithm 4. The recovery activity takes place by reinitializing the sensor nodes so that it releases all its resources and take a fresh start. The last state checkpoint and data checkpoint is recovered from the permanent memory. Data checkpoint helps to recover the old data and the state checkpoint tells the previous trust degree of the node.

```
Create Checkpoint ( )
{
    For each node Sᵢ do this
    {
        Initialize DPC=0;
        For each packet delivery
        {
            DPC++;
            If (DPC = 200)
            {
                Store TD in permanent storage;
                Store Sᵢ.CURR-VAL in DCkpt;
                Set DPC=0;
            }
        }
    }
}
```

Figure 5: Algorithm 1.

```
For each node Sᵣ with detected fault
{
    Get Notification (fault-type)
    If (fault-type=HF OR fault-type=SF)
    {
        Third party assistance needed
        Initiate Node Recovery ( )
    }
    If (PRR very low OR PDR very low)
        Initiate Node Recovery ( )
    If (Residual energy << Threshold)
        Shut down Sensor Node
}
```

Figure 6: Algorithm 2.

```
Initiate Node Recovery ( )
{
    Send probe packets to all of Sᵣ.NBR
    For each node Sⱼ ∈ Sᵣ.NBR
    {
        Maintain Topology ( )
    }
    Start recovery action (Sᵣ)
    {
        Reinitialize the sensor node;
        Fetch the last data from Sᵣ.DCkpt;
        Get Sᵣ.TD;
        Perform LQE;
    }
}
```

Figure 7: Algorithm 3.

Finally the node performs link quality estimation given by LQE, which is again proposed in our work mentioned in [21]. In the node recovery algorithm any node which is in recovery mode sends some probe packets to its neighbor, stating its unavailability for some instance of time. The neighbors in turn, update their own neighbor tables and get prepared for running the topology maintenance schedule.

```
Maintain Topology ( )
{
    Get parent list of S_r
    For each parent S_k of S_r
    {
        If ‖S_jS_k ≤ CR‖
        {
            Update neighbor list
            Update routing table
            Transmit through S_k
        }
        Else
        {
            Estimate transmission power P_{j,k}
            If  P_{j,k} < P_{j,k-1}
                Store current P_{j,k}
            Else
                Keep the previous P_{j,k-1}
        }
    }
    Select S_k with minimum P_{j,k}
    Update neighbor list and routing table
    Transmit through S_k
}
```

Figure 8: Algorithm 4.

6 Results and discussions

In this section the results are displayed and corresponding discussion is presented. For simulation purpose and preparing the results MATLAB version 7.11.0.584 (R2010b) and Microsoft Office Excel 2003 was used. The sensor node specifications considered and the simulation environment is mentioned in the Tables 2 and Table 3 respectively. Table 4 next shows the computed energy consumption for each task done by each node. Initially the nodes are deployed randomly and then they are initialized and they start to do their normal task. In an area of 100×100 m^2 thirty sensor nodes were randomly deployed considering a uniform communication range of 25 meters.

6.1 Fault detection and diagnosis

The nodes were deployed randomly and then gradually nodes become faulty in the WSN. The faults are detected

and consequentially the recovery is carried out by the nodes.

Just after the nodes are deployed, the scenario is presented in the first quadrant of the Figure 9, which is followed by the edge development of the nodes, depending upon the transmission radius of the sensor nodes and presented in second quadrant of Figure 9. Now for the detection of faults the fault detection algorithm proposed by Mitra and De Sarkar (2014) is used and the nodes demarcated by red colors in the third and fourth quadrant of Figure 9. It was observed that five out thirty nodes were detected to be faulty. Moreover in the Figure 10 especially the faulty nodes with the affected links are represented.

Parameter	Value
Frequency Range	2.4 – 2.48 GHz
Data Rate	250 Kbps
Current Draw	16 mA @ Receive mode
	17 mA @ Transmit mode
	8 mA @ Active mode
	8 µA @ Sleep mode

Table 2: Sensor Node Specifications.

Parameter	Value
No. of Nodes Deployed	30
Area Covered	100×100 m^2
Communication Range	25 meter
Node Density (ρ)	0.003nodes/ m^2

Table 3: Simulation Environment.

Task Performed	Energy Consumed (in mJ)
Data Sensing	0.0018
Data Processing	0.0513
Data Transmission	0.1864152
Data Receiving	0.0627456
Self Evaluation	0.12

Table 4: Energy Consumption for various tasks performed by Sensor Nodes. [18]

6.2 Fault recovery

After the faults are detected then the recovery activities are started. Faulty nodes go for recovery and here they are named as recovering node (RN) and the affected nodes (AN) are their neighbors. Now as in Figure 10 the red nodes are RN and red links are affected links, which will get defunct later on. A list of susceptible parents for each set of ANs is mentioned in Table 5. However the ANs have to select a suitable node to maintain the connectivity even in absence of the corresponding RN.
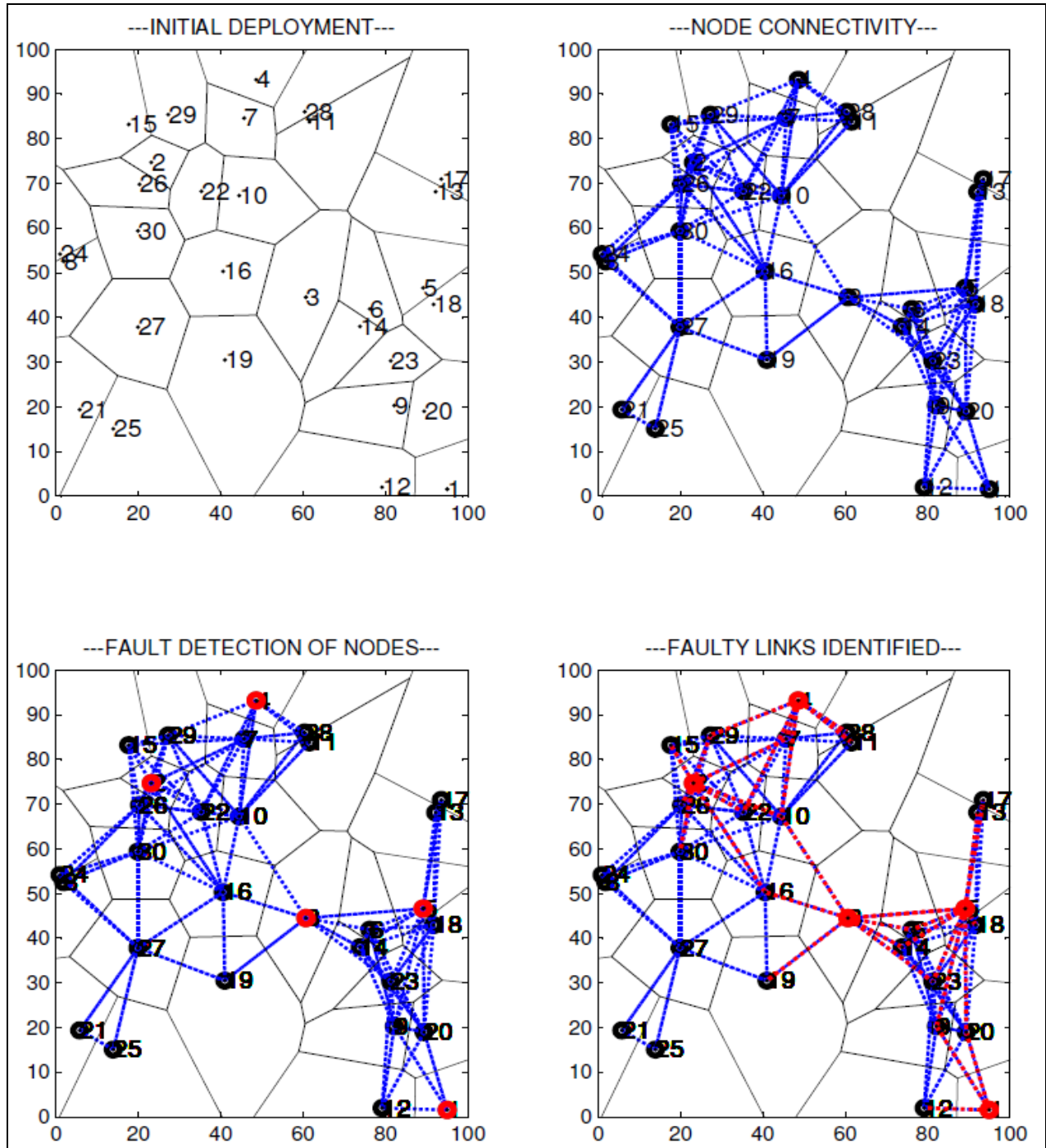
Figure 9: Node Deployments and Fault Scenario.

Case 1 (Recovery for Node 1): Here the node ID 1 is RN after getting faulty goes to recovery mode; now this node sends probe packets to its neighboring nodes, which are actually in its communication range. The IDs of the ANs are 9, 12 and 20. Now these nodes will update their neighbor list and each of them will try to find out a node, which will act as their new parent. There are multiple susceptible parents for nodes 9, 12 and 20 and they select node IDs 18, 23 and 6 respectively, as their new parent. Node ID 9 have 5 susceptible parents and out of which it selects node ID 18 since the transmission power factor is minimum among all other possible parents (as presented in Table 5). Similarly node ID 12 and 20 has 5 and 6

susceptible parents respectively but node IDs 23 and 6 are selected as actual parents because of low transmission factor. So in all the 3 situation the tasks are carried out to minimize the power consumption. The necessary results to support new parent selection, from the each node's susceptible parent list, is elaborately presented in Table 5.

Case 2 (Recovery for Node 4): In the second case node ID 4 is RN and its ANs are 7, 11, 28 and 29. Just similarly like case 1 here all ANs find a suitable parent for transmitting data. In this case there are multiple susceptible parents out of which nodes 7, 11, 28 and 29 (mentioned in Table 5) but each node selects some

specific node as their new parent. Moreover they update their neighbor list also.

Node ID 7 and 11 have 6 susceptible parents and out of which node ID 7 selects node ID 10 as its immediate parent and node 11 selects 22 as its new parent. This selection is done on the basis of the minimum power factor for these nodes. Lower power factor means lower power consumption for transmission. Similarly node ID 28 and 29 selects node ID 2 and 30 as their new parent respectively. So in all the 4 situations specific selections are made to keep the power consumption of the node low, in comparison to others. The necessary results to support new parent selection, from the each node's susceptible parent list, is elaborately presented in Table 5.

7   Discussion

Moreover in Table 6 all the ANs are mentioned along with their transmission power and distance from the current parent. After simulation it is inferred that the ANs select those nodes as their new parent, in absence of the RN, through which they can forward data towards BS.

Node 28 has to raise its multiplication factor as high as 2.82, in order to avoid isolation. As in the cases of nodes 11, 28 and 29 the distance with the new parent is greater than their distance with node 4 so they have to raise their transmission power.

Here the activities for two of the nodes with ID 1 and 4 are shown the same activities are carried out for other RNs. All RNs send the probe packets to their neighbors, and the ANs in turn perform the topology maintenance tasks and then the RNs are reinitialized and the values from system variable DCkpt and TD are fetched, since they contain the last data checkpoint and state checkpoint of the node. After that the link quality estimation is done to check the vicinity traffic situation and finally the node comes back to the network.
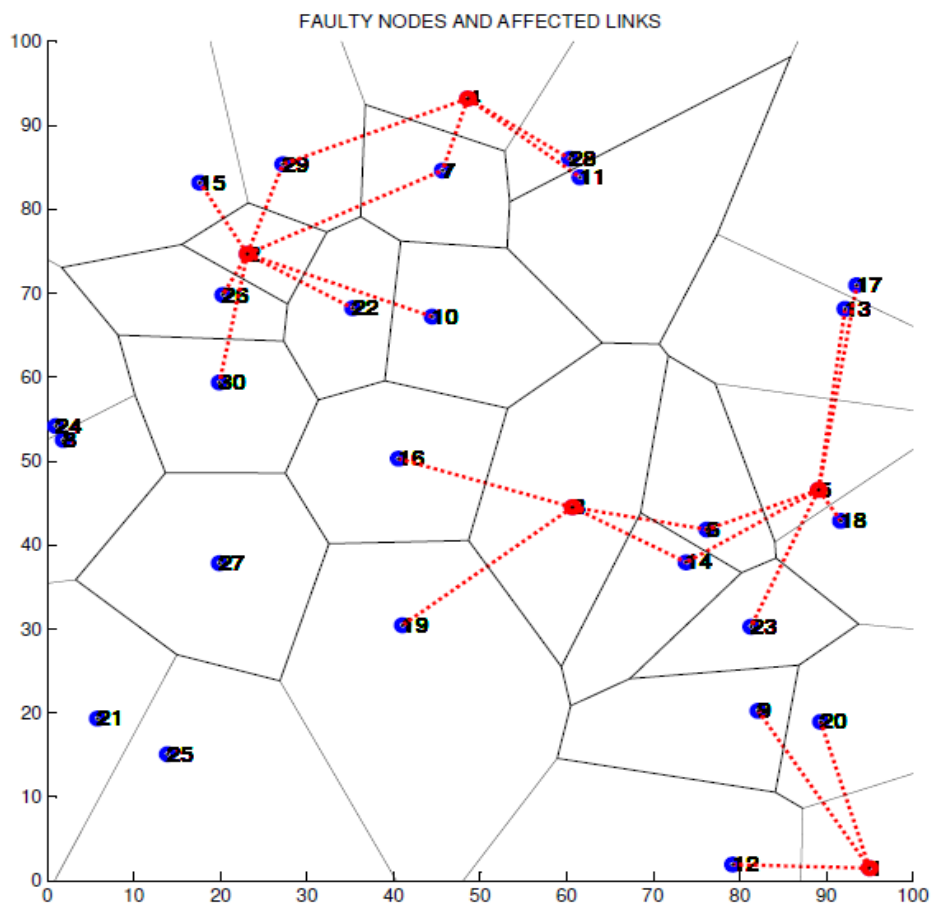
Figure 10: Affected Links due to Node Fault.

AN ID	Susceptible Parent IDs	Power Factor For Each Parent
9	3, 5, 13, **18**, 19	1.42, 1.20, 2.15, **1.08**, 1.86
12	5, 6, 14, 18, **23**	2.89, 2.53, 2.30, 2.30, **1.79**
20	3, 5, **6**, 13, 17,19	2.10, 1.51, **1.44**, 2.69, 2.85, 2.71
7	2, **10**, 15, 16, 26, 30	2.72, **1.93**, 3.10, 3.83, 3.25, 3.99
11	2, 3, 13, 17, **22**, 29	2.47, 2.46, 2.15, 2.16, **1.91**, 2.15
28	**2**, 3, 15, 16, 26	**2.82**, 3.01, 3.11, 2.96, 3.14
29	11, 16, 28, **30**	1.51, 1.65, 1.46, **1.19**

Table 5: Susceptible Parent List for Affected Nodes.

RN ID	AN ID	New Parent Node ID	Power Factor	Distance of AN with new Parent (in meters)
1	9	18	1.08	24.56
	12	23	1.79	28.39
	20	6	1.44	26.41
4	7	10	1.93	17.45
	11	22	1.91	30.54
	28	2	2.82	38.95
	29	30	1.19	27.02

Table 6: New Parent Selections.

8   Conclusion

WSN is used widely nowadays for various field surveillance and distributed fault tolerance in necessary in the same for reliability and dependability of WSN. Novel fault recovery architecture is designed and proposed in this paper; the recovery architecture is destined to be integrated with a fault tolerant framework for wireless sensor network. This paper also presented proposed algorithms for fault recovery and connectivity maintenance in WSN. This algorithm explains details of recovery tasks are carried out. A brief discussion is presented to identify the detection of faults and then different cases for recovery are done.

This proposed recovery technique takes care of recovery actions related to the faults due to hardware or software failure. It also improves link quality or connectivity among the nodes during recovery phases. However, the noise-related measurement or error due to presence of noise is not scope of this paper. This research will enhance the recovery scheme with self-organization and noise-related measurement based recovery in future. As a future work this research will also present a result-interpretation based comparative study of recovery schemes.

9   References

[1]   Akbari A., Dana A., Khademzadeh A. & Beikmahdavi N. (2011) "Fault Detection and Recovery in Wireless Sensor Network using Clustering" in International in Journal of Wireless & Mobile Networks (IJWMN) Vol. 3, Issue 1, 130-138

[2]   Akbari A., Beikmahdavi N., Khosrozadeh A., Panah O., Yadollahi M. & Jalali S. V. (2010) "A Survey Cluster-Based and Cellular Approach to Fault Detection and Recovery in Wireless Sensor Networks" in World Applied Sciences Journal Vol. 8 Issue 1 76-85

[3]   Bathla G. & Jindal S. (2016) "A Review of RIM and LeDiR recovery mechanism for node recovery in Wireless Sensor Actor Network" in International Journal of Engineering Development and Research Vol. 4, Issue 2, 2145-2147

[4]   Borawake-Satao R. & Prasad R. S. (2017) "Mobile Sink with Mobile Agents: Effective Mobility Scheme for Wireless Sensor Network" published in International Journal of Rough Sets and Data Analysis Vol. 4 Issue 2 24-35

[5]   Brahme C., Gadadare S., Kulkarni R., Surana P. & Marathe M.V. (2014) "Fault Node Recovery Algorithm for a Wireless Sensor Network" in International Journal of Emerging Engineering Research and Technology Vol. 2, Issue 9, 70-76 ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

[6]   Chen J., Kher S. & Somani A. (2006) "Distributed Fault Detection of Wireless Sensor Networks" in Proc. of Workshop on Dependability issues in Wireless Ad hoc Networks and Sensor Networks (DIWANS), pp. 65-72 published by ACM, Los Angeles, CA, USA

[7]   Graham B., Tachtatzis C., Franco F. D., Bykowski M., Tracey D. C., Timmons N. F. & Morrison J. (2011) "Analysis of the Effect of Human Presence on a Wireless Sensor Network" published in

International Journal of Ambient Computing and Intelligence (IJACI), Vol. 3 Issue 1, 1-13

[8] Haboush A., Mohanty M. N., Pattanayak B. K. & Al-Tarazi M. (2014) "A Framework for Wireless Sensor Network Fault Rectification" published in International Journal of Multimedia and Ubiquitous Engineering Vol. 9 Issue 1 133-142

[9] Kang Z., Yu H. & Xiong Q. (2013) Detection and Recovery of Coverage Holes in Wireless Sensor Networks in Journal Of Networks, Vol. 8, Issue 4, 822-828

[10] Karl H. & Willig A (2005) "Protocols and Architectures for Wireless Sensor Networks" West Sussex, England, John Wiley & Sons Ltd.

[11] Klus H. & Niebuhr D. (2009) "Integrating Sensor Nodes into a Middleware for Ambient Intelligence" published in International Journal of Ambient Computing and Intelligence (IJACI), IGI Global Vol. 1, Issue 4, 1-11

[12] Kumar S. & Nagarajan (2013) N. "Integrated Network Topological Control and Key Management for Securing Wireless Sensor Networks" published in International Journal of Ambient Computing and Intelligence (IJACI), Vol. 5 Issue 4, 12-24

[13] Lakamana V. S. S. K. & Rani S. J. (2015) "Fault Node Prediction Model in Wireless Sensor Networks Using Improved Generic Algorithm" in International Journal of Computer Science and Information Technologies, Vol. 6 Issue 4, 3501-3503

[14] Leskovec J., Sarkar P. & Carlos Guestrin (2005) "Modelling Link Qualities in a Sensor Network" published in Informatica Vol. 29 445–451

[15] Liu X. (2006) "Coverage with Connectivity in Wireless Sensor Networks" in Proc. Of Basenet 2006, in conjunction with BroadNets, San Jose, CA

[16] Ma C., Lin X., Lv H. & Wang H. (2009) "ABSR: An Agent based Self-Recovery Model for Wireless Sensor Network" in Proc. Of Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, pp. 400-404, Chengdu, China

[17] Mishal M.D., Narke V.A., Shinde S.P., Zaware G.B. & Salve S. (2015) "Fault Node Recovery For A Wireless Sensor Network" in Multidisciplinary Journal of Research in Engineering and Technology, Vol. 2, Issue 2, 476-479

[18] Mitra S. & Sarkar A. D. (2014) "Energy Aware Fault Tolerent Framework in Wireless Sensor Network" in Proc. Of AIMoC 2014 pp. 139-145 published by IEEE, Kolkata, India

[19] Mitra S., Das A. & Mazumdar S. (2016) "Comparative Study of Fault Recovery Techniques in Wireless Sensor Network" in Proc. Of WIECON-ECE 2016 pp. 130-133, published by IEEE, AISSMS, Pune, India

[20] Mitra S., Sarkar A. D. & Roy S. (2012) "A Review of Fault Management System in Wireless Sensor Network" in Proc. of International Information Technology Conference, CUBE pp. 144-148 published by ACM, Pune India

[21] Mitra S., Roy S. & Das A. (2015) "Parent Selection Based on Link Quality Estimation in WSN" in Advances in Intelligent Systems and Computing (AISC) Vol. 379, Proc. of IC3T, pp. 629-637, published by Springer, Hyderbad, India

[22] Mukherjee A., Dey N., Kausar N., Ashour A. S., Taiar R. & Hassanien A. E. (2016) "A Disaster Management Specific Mobility Model for Flying Ad-hoc Network" published in International Journal of Rough Sets and Data Analysis Vol. 3 Issue 3 72-103

[23] Nithilan N. & Renold A. P. (2015) "On-Demand Checkpoint And Recovery Scheme For Wireless Sensor Networks" in ICTACT Journal On Microelectronics, Vol. 1, Issue 1, 35-40, ISSN online (2395-1680)

[24] Nguyen K. V., Nguyen P. L., Phan H. & Nguyen T. D. (2016) "A Distributed Algorithm for Monitoring an Expanding Hole in Wireless Sensor Networks" published in Informatica Vol. 40 181–195

[25] Reghunath E. V., Kumar P. & Babu A. (2014) "Enhancing the Life Time of a Wireless Sensor Network by Ranking and Recovering the Fault Nodes" published in International Journal of Engineering Trends and Technology (IJETT) Vol. 15 Issue 8, 410-413

[26] Saleh I., Eltoweissy M., Agbaria A. & El-Sayed H. (2007) "A Fault Tolerance Management Framework for Wireless Sensor Networks" published in Journal of Communications, Vol. 2, Issue 4, 38-48

[27] Wan J., Wu J. & Xu X. (2008) "A Novel Fault Detection and Recovery Mechanism for Zigbee Sensor Networks" in Proc. Of Second International Conference on Future Generation Communication and Networking, pp. 270-274, Hainan Island, China published by IEEE

[28] Yim S. J., & Choi Y.H. (2010) "An Adaptive Fault-Tolerant Event Detection Scheme for Wireless Sensor Networks" published in Sensors Journal, Vol. 10 Issue 3 2332-2347

Combined Zernike Moment and Multiscale Analysis for Tamper Detection in Digital Images

Thuong Le-Tien, Tu Huynh-Kha, Long Pham-Cong-Hoan, An Tran-Hong
Dept of Electronics and Electrical Eng., Bach Khoa University of Ho Chi Minh City, Vietnam
E-mail: thuongle@hcmut.edu.vn, hktu@hcmiu.edu.vn, phamcong.hoanlong@gmail.com, an.tranhong94@gmail.com

Nilanjan Dey
Techno India College of Technology, Rajarhat, Kolkata, India
E-mail: neelanjandey@gmail.com

Marie Luong
University Paris 13, France
E-mail: marie.luong@univ-paris13.fr

The paper proposes a new approach as a combination of the multiscale analysis and the Zernike moment based for detecting tampered image with the formation of copy – move forgeries. Although the traditional Zernike moment based technique has proved its ability to detect image forgeries in block based method, it causes large computational cost. In order to overcome the weakness of the Zernike moment, a combination of multiscale and Zernike moments is applied. In this paper, the wavelets and curvelets are candidates for multiscale role in the proposed method. The Wavelet transform has successful in balancing the running time and precision while the precision of the algorithm applied the Curvelets does not meet the expectation. The comparison and evaluation of the performance between the Curvelets analysis and the Wavelets analysis combining with the Zernike moments in a block based forgery detection technique not only prove the effective of the combination of feature extraction and multiscale but also confirm Wavelets to be the best multiscale candidate in copy-move detection.

Povzetek: Razvita je metoda za pospešitev ugotavljanja prekopiranih ponarejenih slik.

1 Introduction

In the world we are living in, most information is created, captured, transmitted and stored in digital form. It is really the flourished time of digital information. However, the digital information can easily be interfered and forged without being recognized by naked eyes, and thus create a significant challenge when analyzing digital evidences such as images. There are many kinds of counterfeiting images which are classified in two groups: active and passive. The active techniques are known as watermarking or digital signature in which the information of original image is given while there is no any prior information in the passive ones. In recent years, the passive, also called the blind, methods are more interested and challenge the researchers and scientists in the field of image processing and image forensics. And among the manipulations creating faked images in the passive/blind kind, copy-move is the most popular. A copy-move tampered image is created by copying and pasting content within the same image. Nowadays, various software for images processing are getting more sophisticated and easily accessed by anybody. That is why the studies on digital image forgeries are getting

more attention lately. Actually, image forgery detection not only belongs to image processing but also is a field of image security. Standing out from other studies, the researches about Zernike moments had shown its advance in detecting forged region in copy-move digital image forgery. The Zernike moment has proved to be superior compare to other types of moment; however the Zernike moment based algorithm requires a large computational cost [. In addition to the moments, studies about the multiscale analyses have long been applied to digital signal processing due to their advantages. A proposed method using a combination of the Zernike moment and the Wavelet transform has been studied and evaluated [1] in which the Wavelets analysis has been used in order to reduce the computational cost of the Zernike moment based technique. The combination of the Zernike moment and the Wavelets has shown its feasibility in detecting the copy-move forgeries and the reduction in running time of the algorithm. An example of a typical copy-move forgery image is shown in Figure 1.
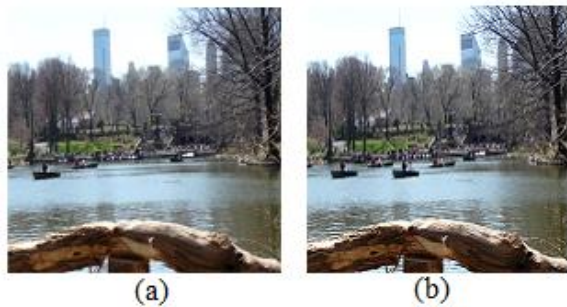
Figure 1: Example image of a typical copy-move forgery; (a). the original image; (b). the tampered image [2].

In this study, the Curvelets is also used to combine with the Zernike moment based technique instead of the Wavelets because of the ability to represent the curves of the objects. A block based technique using the Zernike moment to extract feature for the image forgery detection is used. However, instead of the original testing images, the images which pre-processed by the multiscale analysis is going to be used to calculate the Zernike moments. The goal of this paper is to evaluate the performance of the Wavelets and Curvelets in different perspectives and to examine which multiscale representation is more suitable with the Zernike moment based for detecting copy-move tampered images. The experiments are simulated on MATLAB R2013a with.

2   Related works

In which images' feature extraction techniques are divided into two groups: extracting features directly with and without transformation. And from which, we also chose Zernike moments based techniques for further research. Suggested by Seung – Jin Ryu et al. in [3], Zernike moments technique has advance in feature representation capability, rotation invariance, fast computation, multi-level representation for describing shapes of patterns and low noise sensitivity. The disadvantage of this technique is that it's weak against scaling and other tampering type based on affine transform and it has high computational cost.

Generally, copy – move forgeries detection using block based techniques requires 7 steps [4]; the steps go from dividing the input image into overlapping blocks then calculate features of blocks and final steps are comparing blocks for forgery detection. From [1], a combination between Zernike moments and Wavelet transform was applied to reduce the running time while keeping precision of the original algorithm. Another approach from [5] uses neighborhood sorting, in which G. Li et al. used the Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD). As discussed in the introduction, Wavelet transform based techniques has weakness in processing objects in detailed image with lines and curves, and to work-around that disadvantage, the Curvelet transform was propose [6, 7] by Candès and Donoho in 2000 as a Multi-resolution geometric analysis. The base theory of the Curvelet transform is The Ridgelets transform, proposed by the

same authors. The Curvelet transform was born to analyze local line or curve singularities, which were a difficulty for original Ridgelets. The Curvelet transform allows an almost optimal sparse representation of objects with singularities along smooth curves. Recently, Ying et al. [8] has extended the Curvelet transform to three dimensions. Comparing with Wavelets, the Curvelets in image processing is still new and also a new orientation [6–15].

In this paper, we will carry out and comparison and evaluation of two combinations that are: Zernike moments based technique and Wavelet transform, the other is Zernike moments based technique and Curvelet transform; in order to examine which multiscale representation is more suitable with the Zernike moment based for detecting copy-move tampered images.

3   Methodology

The main purpose of paper is developing an algorithm to balance the running time and exactness using a combination of multiscale and feature extraction, from which the performances of multiscale methods are evaluated to propose the most suitable multiscale method in image forgery detection. The workflow of the copy-move image forgery detection algorithm is shown in Figure 2. In the pre-processing step, the tampered image is going through some morphological techniques after converted into the grayscale. The foreground component is extracted and divided into overlapping blocks of the same size before going to the next step. The Curvelet transform or the Wavelet transform will be applied to analyze the image before extracting features by Zernike moments. The detection results are concluded using the Euclidean distances of each pair of blocks if they are lower than a threshold. Taking account to the similarity between neighbour blocks, the actual distances of a pair of blocks must higher than a threshold before the algorithm can conclude them as the copy-move regions.

3.1   Pre-processing

The tampered images will be converted into gray scale before enhancing by the morphological techniques and extracting the foreground components. Moreover, by converting to gray scale, the computational cost of the Zernike moment will be reduced since we only need to calculate the Zernike moment in one dimension instead of three color dimensions in the color images.

Morphological processing for gray scale images requires more sophisticated mathematical development. For binary image, the pixel has only two values are bit 1 and bit 0, black and white; in gray scale image, pixels have their scale from 0 to 255. Therefore, the operation requires more works. Structuring elements in grayscale morphology come in two categories: non-flat and flat [16]. To reduce the redundant information in background and concentrate in searching the information of the objects belonging to foreground only, a foreground extraction is applied. Among the morphological functions that used for extracting the foreground
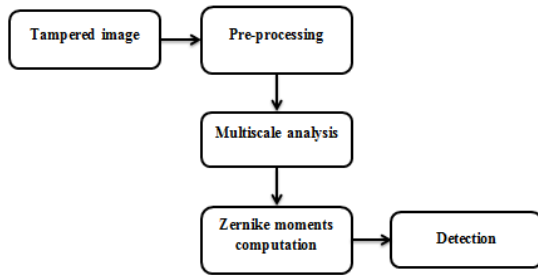
Figure 2: Block diagram of the algorithm.

components such as dilation, erosion, image filling, etc, a combination of dilation and erosion can build all other operations [11].

The dilation of f by a flat structuring element (SE) b at any location (x, y) is defined as the maximum value of the image in the window outlined by $\hat{b}$ when the origin of is $\hat{b}$ at (x, y). That is:

$$[f \oplus b](x, y) = \max_{(s,t)\in b} \{f(x-s, y-t)\} \qquad (1)$$

When we used that

$$\hat{b} = b(-x, -y) \qquad (2)$$

Where, similarly to the correlation, x and y are incremented through all values required so that the origin of b visits every pixel in f. That is, to find the dilation of f by b, the structuring element is reflected about the origin. The dilation is the maximum value of f from all values of f in the region of f coincident with b.

For a non-flat structuring element b_N, the dilation of f is defined as:

$$[f \oplus b_N](x, y) = \max_{(s,t)\in b_N} \{f(x-s, y-t) + b_N(s,t)\}$$
$$(3)$$

Since we add values to f, the dilation with a non-flat SE is not bounded by the values of f, which may present a problem while interpreting results. Grayscale SEs are rarely used.

The erosion of f by a flat structuring element b at any location (x, y) is defined as the minimum value of the image in the window outlined by $\hat{b}$ when the origin of $\hat{b}$ is at (x, y). Therefore, the erosion at (x, y) of an image f by a structuring element b is given by:

$$[f \ominus b](x, y) = \max_{(s,t)\in b} \{f(x+s, y+t)\} \qquad (4)$$

The explanation is similar to one for dilation except for using minimum instead of maximum and that we place the origin of the structuring element at every pixel location in the image.

Similarly, the erosion of image f by non-flat structuring elements b_N is defined as the following equation:

$$[f \ominus b_N](x, y) = \max_{(s,t)\in b_N} \{f(x+s, y+t) - b_N(s,t)\}$$
$$(5)$$

Erosion and dilation, separately, are not very useful in gray-scale image processing. These operations become powerful when used in combination to develop high-level algorithms.

In the image, the components differ greatly from the background image can be segmented. Operators that can calculate the gradient of the image can be used to detect the changes in contrast. Therefore, the edges of the input can be extracted easily and are then processed by dilation and filling holes to create the mask, which is call binary gradient mask, for extracting the foreground component (see Figure 3).
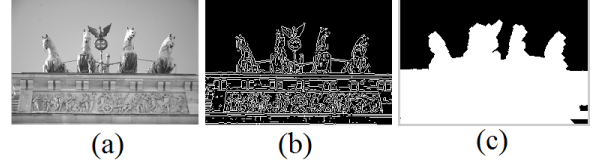

Figure 3: Example of creating the mask from an input image; (a). Input image; (b). Binary gradient mask; (c). Mask that used to extract foreground component [2].

3.2 Wavelets analysis

According to [11, 17-31], the Wavelet transform is easier to compress, transmit and analyze in image processing compared to the Fourier transform. The Wavelets analysis can be used in order to divide the information of the image into approximation and detail sub-signal. The general trend of pixel values is shown in the approximation sub-band while the three detail sub-bands are so small that they can be neglected. Therefore, it is feasible to use just the approximation sub-band to detect the image forgery. Figure 4 shows the steps to get the approximate image from an original image.

According to [11], the two-dimensional images which requires a two-dimensional scaling function, $\varphi(x, y)$, and three two-dimensional wavelets, $\psi^H(x, y)$, $\psi^V(x, y)$ and $\psi^D(x, y)$. They are the product of one-dimensional scaling function φ and corresponding wavelet ψ,

$$\varphi(x, y) = \varphi(x)\varphi(y) \qquad (6)$$

$$\psi^H(x, y) = \psi(x)\varphi(y) \qquad (7)$$

$$\psi^V(x, y) = \varphi(x)\psi(y) \qquad (8)$$

$$\psi^D(x, y) = \psi(x)\psi(y) \qquad (9)$$

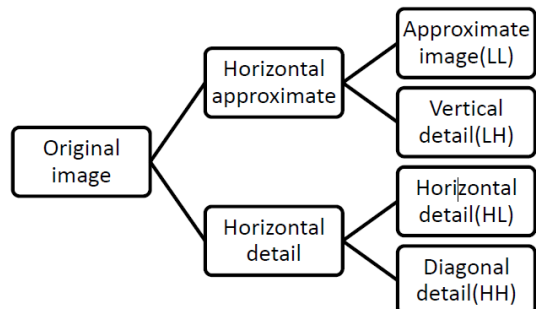In which ψ^H measures variations along horizontal

Figure 4: Wavelets transform block diagram.

edges, ψ^v measures variations along vertical edges and ψ^D measures variations along diagonals. After finding two-dimensional scaling and wavelet functions, based on one-dimensional discrete wavelet transform, we define the scaled and translated basis functions:

$$\varphi_{j,m,n}(x,y) = 2^{j/2}\varphi(2^j x - m, 2^j y - n) \quad (10)$$

$$\psi^i_{j,m,n}(x,y) = 2^{j/2}\varphi(2^j x - m, 2^j y - n) \quad (11)$$

$$W_\varphi(j_0,m,n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\varphi_{j0,m,n}(x,y) \quad (12)$$

$$W^i_\psi(j,m,n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\psi^i_{j,m,n}(x,y) \quad (13)$$

i = {H, V, D}, where j_0 is an arbitrary starting scale and the $W_\varphi(j_0,m,n)$ coefficients define an approximation of f(x, y)at scale j_0. The $W^i_\psi(J, m, n)$ coefficients add horizontal, vertical, and diagonal details for scales $j \geq j_0$. Normally we let $j_0 = 0$ and select $N = M = 2^J$ so that we can get $j = 0, 1, 2.., J - 1$ and $m, n = 0, 1, 2,..., 2^J - 1$. Given the W_φ and W^i_ψ of 2 equations (12) (13), $f(x, y)$ is obtained by inverse discrete wavelet transform [11].

$$f(x,y) = \frac{1}{\sqrt{MN}}\sum_m\sum_n W_\varphi(j_0,m,n)\varphi_{j0,m,n}(x,y) +$$
$$\frac{1}{\sqrt{MN}}\sum_{i=H,V,D}\sum_{j=j0}^{\infty}\sum_m\sum_n W^i_\psi(j_0,m,n)\psi_{ij,m,n}(x,y) \quad (14)$$

The wavelet transform used in the algorithm is selected from the Wavelet families, such as "db1", "haar", "db2",…with similar results. Optionally, the wavelets "haar" is used to demonstrate wavelets transform level 1 in Figure 5.

From the Figure 5, the reduction in size of the approximate image which analyzed by the Wavelet transform can be clearly seen. The approximate image is only a quarter of the original image.

3.3 Curvelets analysis

In this research, the first generation of the Curvelet transform is used to analyze the image. From [6, 7, 32], the idea of the first generation Curvelet transform is to decompose the image into a set of wavelet bands and analyze each band by a local Ridgelet transform as shown on Figure 6. Different sub-bands of a filter bank
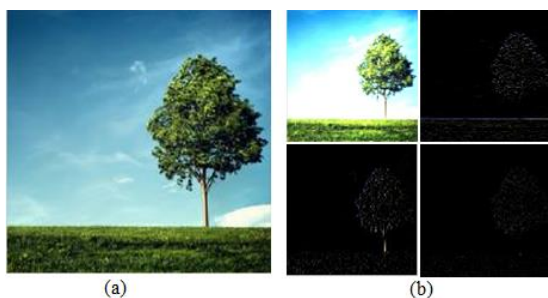

Figure 5: Image analyzed using Wavelet transform level 1; (a). Original image, (b). The approximate and detail images after applying the Wavelet transform [2].

output are represented by different levels of Ridgelet pyramid. Moreover, a relationship between the width and length of the crucial frame elements is contained in this sub-band decomposition. The first generation discrete Curvelet transform of a continuous function $f(x)$ is making use of a dyadic sequence of scales and a bank of filters with characteristic that the band bass filter Δ_j is concentrated near the frequencies of $[2^{2j}, 2^{2j+2}]$, as follow:

$$\Delta_j(f) = \Psi_{2j} * f \quad (15)$$

$$\hat{\Psi}_{2j}(v) = \hat{\Psi}(2^{-2j}v) \quad (16)$$

Following the research in [33,34], the decomposition of the first generation discrete Curvelet transform is a sequence of the following steps:

Sub-band decomposition: Decomposing the object f into sub-bands.

$$f \rightarrow (P_0 f, \Delta_1 f, \Delta_2 f,...) \quad (17)$$

Each layer contains details of different frequencies. P_0 is low pass filter, Δ_1, Δ_2,… are high-pass (band-pass) filters. The sub-band decomposition can be approximated using the well-known wavelet transform; f is decomposed into S_0, D_1, D_2, D_3, etc. $P_0 f$ is partially constructed from S_0 and D_1, and may include also D_2 and D_3. $\Delta_s f$ is constructed from D_{2s} and D_{2s+1}.

Smooth partitioning: The sub-bands are smoothly windowed into squares of a suitable scale. The scale is optional, but in the proposed method, the scale is 2 to reduce the size of image by a half corresponding to a wavelets decomposition level 1. A grid of dyadic squares is defined as.

$$Q_{(s,k_1,k_2)} = \left[\frac{k_1}{2^s}, \frac{k_1+1}{2^s}\right] \times \left[\frac{k_2}{2^s}, \frac{k_2+1}{2^s}\right] \in Q_s \quad (18)$$

Where, Q_s is all the dyadic squares of the grid. Let w be a smooth windowing function with "main" support of size $2^{-s}x2^{-s}$. For each square, w_Q is a displacement of w localized near Q. Multiplying $\Delta_s f$ with w_Q ($\forall Q \in Q_s$) produces a smooth dissection of the function into "squares".
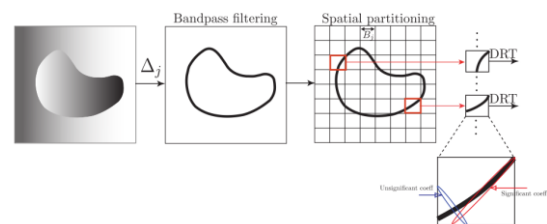

Figure 6: Local Ridgelet transform on bandpass filtered image. The red ellipse is significant coefficient while the blue one is insignificant coefficient [6].

$$h_Q = w_Q \cdot \Delta_s f \quad (19)$$

Renormalization: is centering each dyadic square to unit square. For each Q, the operator T_Q is defined as:

$$(T_Q f)(x_1, x_2) = 2^s f(2^s x_1 - k_1, 2^s x_2 - k_2) \quad (20)$$

Each square is renormalized:

$$g_Q = T_Q^{-1} h_Q \quad (21)$$

Ridgelet analysis: DRT is used to analyze each square.

$$\alpha_{(Q,\lambda)} = \langle g_Q, \rho_\lambda \rangle \qquad (22)$$

In which the Ridgelet element has a formula in the frequency domain as:

$$\hat{\rho}_\lambda(\xi) = \frac{1}{2} |\xi|^{-\frac{1}{2}} \left(\hat{\psi}_{j,k}(|\xi|) \cdot \omega_{i,l}(\theta) + \hat{\psi}_{j,k}(-|\xi|) \cdot \omega_{i,l}(\theta + \pi) \right) \qquad (23)$$

Where, $\omega_{i,l}$ are periodic wavelets for $[-\pi, \pi)$. i is the angular scale and $l \in [0, 2^{i-1}-l]$ is the angular location. $\psi_{j,k}$ are Meyer wavelets for $\mathcal{R}$. j is the Ridgelet scale and k is the Ridgelet location.

Below is an example result for the Curvelet transform with scale of 2 and the image's edges are shown (see Figure 7).
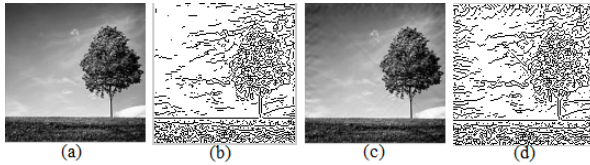
Figure 7: Image analyzed using the Curvelet transform. (a), (b) Original image and its edges; (c), (d) Image processed with scale of 2 and its edges [2].

3.4 Zernike moment's properties

According to the relative research paper about Zernike moments in [3, 35], we can summarize the Zernike moments through some mathematical background. In this section, we describe the Zernike moments as following functions.

The 2D Zernike moment of order n with repetition of m for a continuous image function f(x,y) that vanishes outside the unit circle is:

$$Z_{nm} = \frac{n+1}{\pi} \iint_{x^2 y^2 \leq 1} f(x,y) V_{nm}^* \qquad (24)$$

Where n is a non-negative integer and m is an integer such that (n-|m|) is non-negative and even. The 2D Zernike moment, $V_{nm}(\rho, \theta)$ is defined in polar coordinate (ρ, θ) inside the unit circle as:

$$V_{nm}(\rho, \theta) = R_{nm}(\rho) \exp(jm\theta) \qquad (25)$$

Where $R_{nm}(\rho, \theta)$ is the n-th order of Zernike radial polynomial given by:

$$R_{nm}(\rho) = \sum_{k=0}^{(n-|m|)/2} (-1)^k \frac{(n-k)!}{k! \left[\dfrac{n-2k+[m]}{2}\right]! \left[\dfrac{n-2k-[m]}{2}\right]!} \rho^{n-2k}$$

$$(26)$$

Zernike moments have rotational invariance, and can be made scale and translational invariant, making them suitable for many applications. Zernike moments are accurate descriptors even with relatively few data points. Reconstruction of Zernike moments can be used to determine the amount of moments necessary to make an accurate descriptor. Though the complexity of the computation process compared to geometric and Legendre moments, Zernike moments have proved to be better in terms of their feature representation capability, rotation invariance, multi-level representation for describing the shapes of patterns and low noise sensitivity [3].

3.5 Forgery Detection Conclusion

After calculating Zernike moments, to remove the similar features due to neighbor blocks, the Euclidean distance of each pair of Zernike will be calculated and compare to a threshold D_1 [3]. This threshold is often set equal to the size of a block. In addition to Euclidean distance, the actual distance of each pair of blocks is also calculated in order to avoid the miss-detection between neighbor blocks.

$$\sqrt{\sum (Z_p - Z_{p+1})} < D_1 \qquad (27)$$

$$\sqrt{(i-k)^2 + (j-l)^2} > D_2 \qquad (28)$$

Where $Z_p = V_{ij}$ and $Z_{p+1} = V_{kl}$. Using these equations, the testing blocks are determined if they are tampered region or not.

4 Proposed system

The paper develops an algorithm to detect the forged regions in term of copy-move in images by using the combination of the multiscale analysis and the Zernike moment based technique. Using this algorithm, the performance of the Wavelet transform and the Curvelet transform will be compared and evaluated at pixel level.

The test image is firstly converted to grayscale before extracting foreground using a binary gradient mask which is created by a combination of dilation and erosion. A wavelet transform decomposition level 1 or a fast discrete curvelet transform (FDCT) is applied to the image which is extracted foreground. In case of 1 level DWT, only the approximation subband is considered. The approximation or an image with FDCT is then divided into many overlapping blocks. The blocks' features are collected by calculating their Zernike moments. The Euclidean and actual distance are also calculated to make sure that they are similar, but not neighbors. Vectors satisfying the constraints in two above distances will be considered the suspicious vectors. Blocks corresponding to these vectors are candidates of copied regions. The flowchart is shown in Figure 8.

From a personal collection and database in [2], we choose 12 images with different characteristics and sizes to conduct our experiments. Figure 9 shows the images used in the experiments. The images have different sizes and features. The smallest images have size of 128x128 while the largest have sizes of 1440x1440. Some of them have very little detail on background while the others have detailed background. The purpose of this is that we can conduct experiments on different types of images for diversity.

Error measurement

At pixel level, the important measures are the number of correctly detected forged pixels, T_P, the number of pixels that have been erroneously detected as forged, F_P, and the falsely missed forged pixels F_N. From these
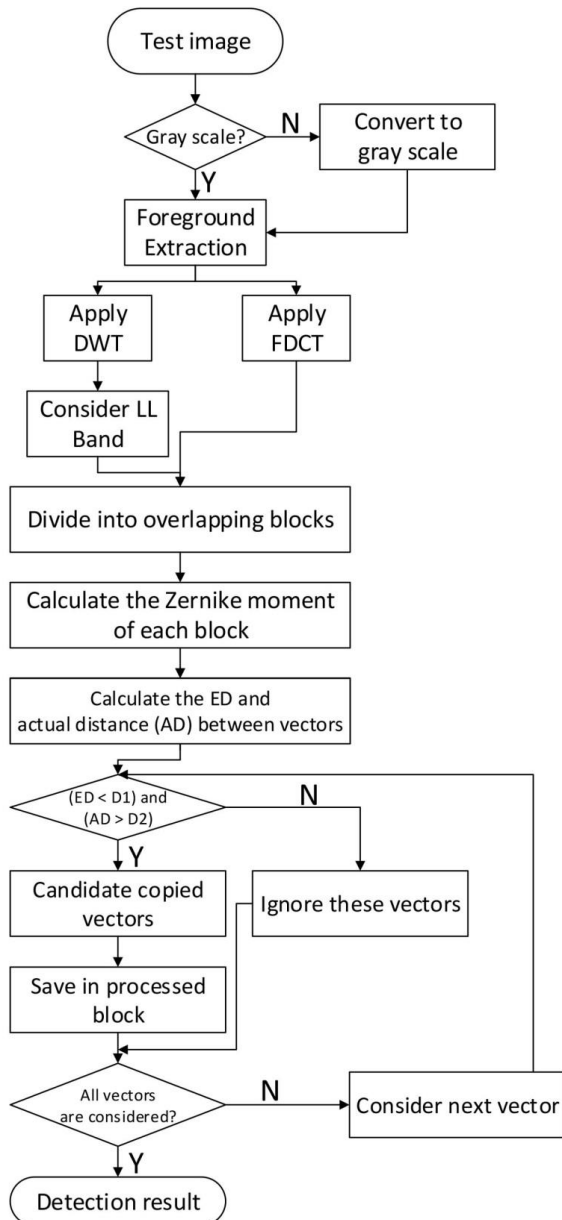
Figure 8: Flowchart of the proposed method.

parameters, we can compute the measure *precision p* and *recall r*. They are defined as:

$$p = T_p / (T_p + F_p) \qquad (29)$$

$$r = T_p / (T_p + F_N) \qquad (30)$$

Precision shows the probability that a detected forgery is truly a forgery, while the recall or true positive rate expresses the probability that a forged pixel is detected. The trade-off between the precision and recall exists; hence, to consider both precision and recall together, F is the combination of precision and recall in a single value.

$$F = 2pr / (p + r) \qquad (31)$$

The three measures are used to evaluate the performance of the copy-move image forgery detection method.
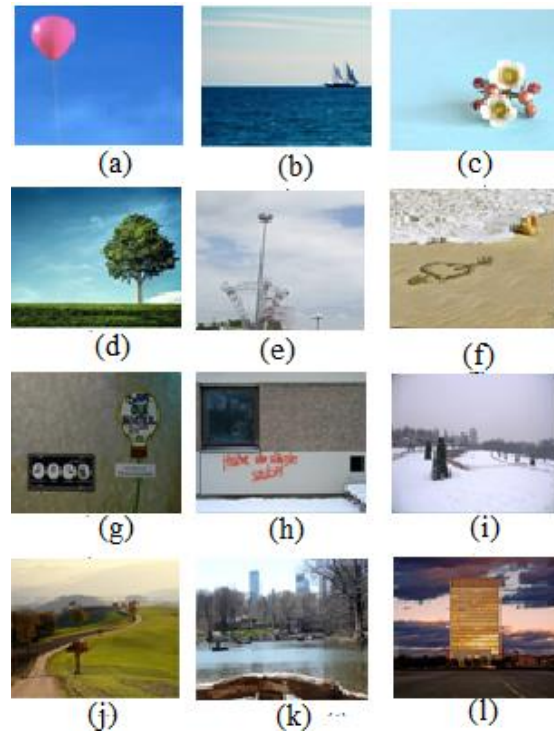
Figure 9: Images used in the experiments [2].

5 Simulations and evaluations

In the experiments, MATLAB program (version 2013a) is used with Window 7 Ultimate 64-bit, CPU Intel Core i5 @ 1.8GHz and 4GB RAM in order to run the simulations.

Using the images in Figure 9, we conducted different experiments by changing the order of Zernike moments, block size, the Curvelet transform's order, etc. Every test image will go through the proposed method and consequently Precision, Recall and F − measure will be evaluated.

Based on the fact that the investigating images are usually color images, there are two options for proposed method. First is to calculate Zernike moments from each color channel and subsequently link the moment values. The other option is to convert the RGB image into a gray-scale image. We choose the latter method because the same copy-move forgery is applied to each color channel similarly. Hence, by only calculating the Zernike moments on one dimension rather than for three dimensions of color image, the computation time of Zernike moments can be minimized.

A. Zernike moment's order

According to [35], a relatively small set of Zernike moments can characterize the global shape of a pattern effectively. The low order moments represent the global shape of a pattern and the higher orders represent the detail. However, the higher the order the more computational cost it takes which result in the increasing in running time. In Figure 10 below show the average running time of proposed method for images with

different Zernike moments' orders in which the higher order is, the more computational time takes and dramatically increases for Curvelets.
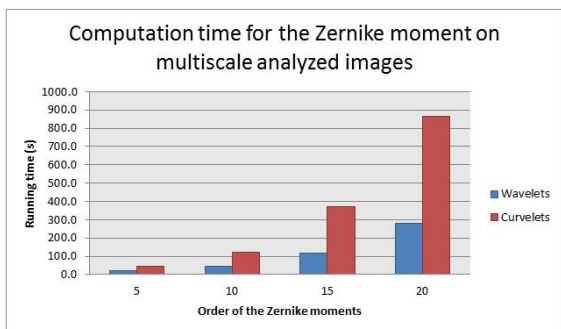

Figure 10: Computation time for the Zernike moment on multiscale analyzed images.
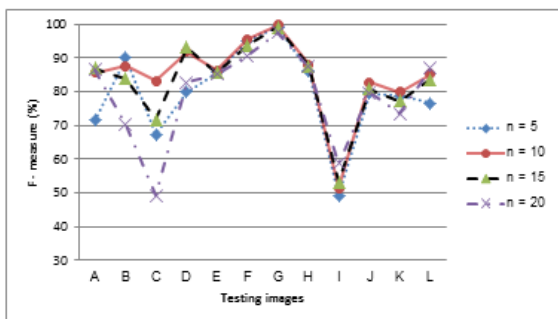

Figure 11: Detection results for forgery images of different Zernike moment's orders with Wavelets analysis.
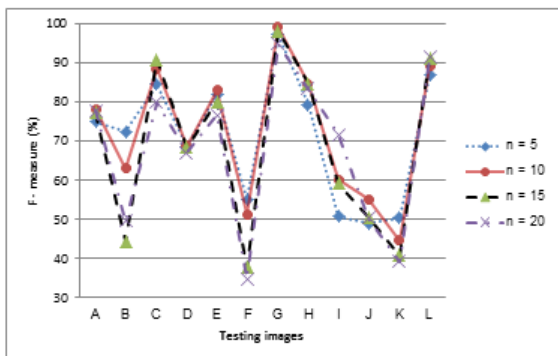

Figure 12: Detection results for forgery images of different Zernike moment's orders with Curvelets analysis.

In this research, the complex Zernike moments based on complex Zernike polynomials as the moment basis set are used in the processing. As we can see from Figure 10, the higher the order, the longer it takes to process. For the Wavelets transform method which only uses the approximation of images to compute the Zernike moments, the running time is much shorter comparing with the Curvelet transform method.

From the Figure 11 and Figure 12, the effect of different Zernike moment's orders on parameter F which shows the relationship between precision and recall was displayed. In this experiment, all images in the dataset will be tested with different Zernike moment's orders and different multiscale analysis. Even though theoretically the Zernike moment with higher order provides better precision compare to the lower ones which means that most part of detected region is correct, the recall which presents the percentage of forgery region detected will relatively reduce as a trade-off for the precision.

Therefore, according to Figure 10 and Figure 11, the F − measure of higher Zernike moment's order detection test is not always higher than the lower orders although it needs more computation time as in Figure 9 shown. Comparing the results of the order of 5, 10, 15 and 20, we can see that the F − measure of 5th order and 10th order is relatively higher than others. For both the Wavelet transform test and the Curvelet transform test, 10th order of Zernike moment is decided to be used as it yields higher F − measure compare to other orders which means the trade-off between the precision and recall of it is better than other order. Hence, for the next experiments, we will use Zernike moment of order 10 to analyze the performance of the Curvelet transform and the Wavelet transform in forgery detection in other different perspectives.

B. Block size

In this experiment, we tested the 13 images in dataset with different dividing block sizes from 8 pixels to 32 pixels in order to analyze the effect of the block size to the detectability and to choose a suitable block size for further experiments. After analyzed by a multiscale method, the testing image will be divided into square blocks with same size before calculating the Zernike moments.

The different in block size effect greatly on the detectability. Figure 12 and Figure 13 show the detection result of forgery images with block sizes of 8, 16, 24 and 32. As we can see from the figures, most of the detection results when dividing the images into block sizes of 24 and 32 pixels fail to detect the forgery regions. This shows that the algorithm is effective if the copied region is comprised by many smaller blocks. This also means if the block size is bigger than the copy-move region, the detectability will drop dramatically since the forgery part in one block is not large enough to provide adequate information for detection. The popular size of divided blocks is 8x8 or 16x16.

From the results shown in Figure 12 and Figure 13, for both the Wavelets analysis and the Curvelets analysis, the image with highest F − measure is G with 99.9% and 98.8% respectively with block size of 16x16. With block size of 24x24, for the Wavelets analysis, the image A, B, F, J and K have visually lower F–measure compare to the highest values, in which the F − values reached 0% in image B and K where the algorithm returns "real image" results. With block size of 32x32, except for the image G, H and I which are large original image with big copy-move region, the other image have 0% for precision and recall. For the Curvelet transform combining with Zernike moment algorithm, the processed images have a bigger size compare to the images analyzed by the

Wavelet transform. Therefore, with the block size of 24x24, only image B have visually lower F – measure (at 0%) compare to other block size and with block size of 32x32, only images B, E, F, J and K have extreme low F – parameters (10% for image F and 0% for the rest).
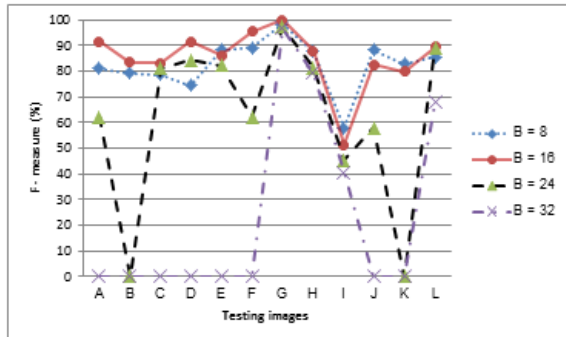

Figure 13: Detection results for forgery images of different block sizes with Wavelets analysis.

For Wavelets analysis, the images' size is reduced to a quarter of the original forgery images which make the copy-move region is shrink to a quarter of original copy-move region. Therefore, compare to the Curvelet analysis detection results, the detectability of bigger block size of Wavelet analysis is lower.

The block size of 16 pixels is chosen since we found this to be a good trade-off between detected image details and the feature robustness. This block size will be fixed across the different analysis, when possible, to allow for a fairer comparison of the feature performance. Note that the majority of the previous research also proposed a block size of 16 pixels.
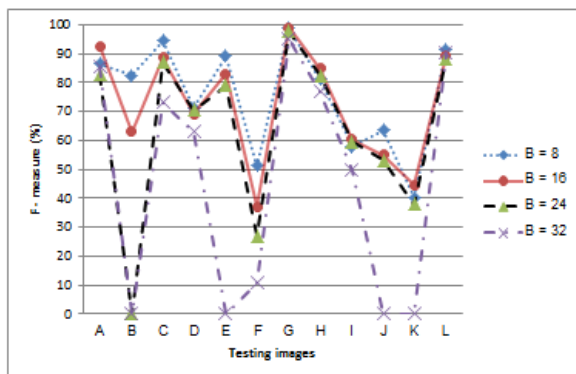

Figure 14: Detection results for forgery images of different block sizes with Curvelets analysis.

Scale of the Curvelet transform
For our experiments, the first-generation Curvelet transform was used to enhance the images. The applications of the first-generation Curvelets are image denoising, image contrast enhancement, etc. The default number of scales including the coarsest wavelet level is equal to $\log_2(N)-3$, in which N is the size of the NxN testing image. In this experiment we will observe the detection results by changing the number of scale from smallest to equal or less than the default value.

Different images will have different default values of scale if they have different sizes. A larger image can be analyzed by larger scale of the Curvelet transform. Therefore, for consistency in this experiment, all 12 images in the dataset will be tested with different number of scale according to the smallest size of the images in dataset which is 128 x 128. With N = 128, we have the number of scale varies from 2, 3 or 4. Hence, the 12 images will be applied Curvelets analysis with one of these scales before going through further steps in the
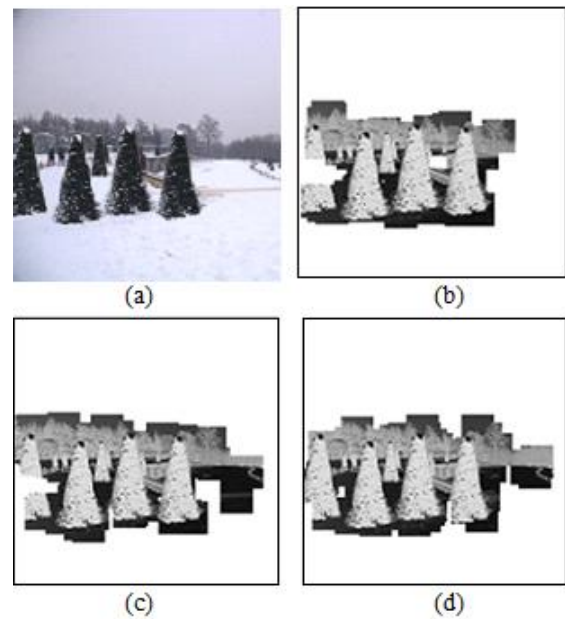

Figure 15: Negative detection results for a forgery image analyzed by the Curvelet transform with different number of scale; (a). Forged image, (b).b n =2,(c) n =3,(d) n =4 [2].

algorithm such as calculating the Zernike moments or computing Euclidean distances.

From the detection results of image J in Figure 15, we can see that with higher number of scale (n = 4) for Curvelet analysis, the percentage of detected forgery region is higher compare to the lower number of scale detection results.

The results of the experiments are shown in Figure 16. Although the experiments are conducted with different number of scales, the F – measure of the detection result is almost the same for most of the testing images except the image A, B, I, J and K. The results of the Curvelets analysis with scale of 2 are slightly higher compare to other scales as we can see from the images A, I, J and K. Although the scale of 4 of the Curvelet analysis algorithm has a higher detection result in image B, the scale of 2 has slightly higher average F – measure compare to other scales. The Curvelet transform with scale of 2 is chosen from the experiments we conducted in this section. Moreover, from [30], the authors also proved that the small scale is more robust. For the next section, we will perform a throughout comparison against the Wavelet transform with the results from the previous experiments.
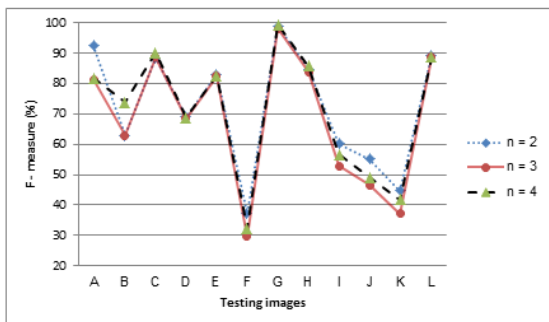
Figure 16: Detection results for forgery images analyzed by the Curvelet transform with different number of scales.

C. Performance comparison between the Curvelet transform and the Wavelet transform combining with the Zernike moment based technique

In Figure 17, the F − measure of both Curvelets analysis and Wavelets analysis is shown. Except for image (C) and (I) where the F − measure of the Curvelet transform method is higher than the Wavelet transform method, the performance of the Curvelet transform is lower for most the testing images. Especially for image (F), the Curvelet transform method's F − measure is lower than 40% while the Wavelet transform method has approximately 95%. In order to analyze the different between two multiscale analyses, we will investigate some of the testing images that have great dissimilarity. The precision and recall of each testing images will also be analyze for further discussions.

For image B, D, F, J and K, they are not the simple images compared to others in the dataset, but they have detailed background which may be the cause for the drop in performance of the Curvelet transform method. Detail statistics about the precision and recall for the images can be analyzed from the Table 1 and 2.

For practical use, the most important aspect is the ability to distinguish tampered and original images. However, the power of an algorithm to correctly annotate the tampered region is also significant, especially when a human expert is visually inspecting a possible forgery. Thus, when evaluating the copy-move algorithm with different multiscale analyses, we analyze their
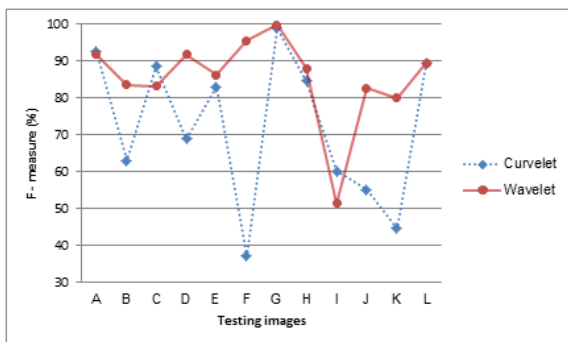

Figure 17: Performance comparison between the Wavelets analysis and the Curvelets analysis combining with the Zernike moment based technique.

Image	Measure (%)			Image	Measure (%)		
	P	R	F		P	R	F
(a)	87.6	96.4	91.8	(g)	99.8	100.0	99.9
(b)	100.0	78.0	87.6	(h)	78.6	100.0	88.0
(c)	74.4	94.3	83.2	(i)	34.7	99.6	51.5
(d)	90.8	92.8	91.8	(j)	93.8	74.1	82.8
(e)	84.2	88.5	86.3	(k)	78.1	81.9	80.0
(f)	91.5	100.0	95.6	(l)	81.1	100.0	89.6
				Average	82.9	92.1	85.7

Table 1: Detection rates for 12 images analyzed by the Wavelet transform combining with Zernike moment

Image	Measure (%)			Image	Measure (%)		
	P	R	F		P	R	F
(a)	89.4	96.0	92.6	(g)	98.1	100.0	99.0
(b)	7.03	57.1	63.0	(h)	73.5	100.0	84.7
(c)	79.8	99.4	88.5	(i)	44.3	94.0	60.2
(d)	76.7	62.7	69.0	(j)	42.5	78.3	55.1
(e)	78.5	88.1	83.0	(k)	30.7	82.2	44.7
(f)	26.5	62.2	37.2	(l)	80.8	100.0	89.4
				Average	65.9	85.0	72.2

Table 2: Detection rates for 12 images analyzed by the Curvelet transform combining with Zernike moment.

performance at pixel level, where we evaluate how accurately tampered regions can be identified through three parameters: Precision, Recall and F − measure.

The experiments conducted are all "plain copy-move" which means we evaluate the performance of each method under ideal conditions. We used the 12 original images and spliced 12 images without any additional modification. We chose per-method optimal thresholds for classifying these 24 images. Although the sizes of the images and the manipulation regions vary on this test set, both tested analyses solved this copy-move problem with a recall rate of above 85% (see Table 1 and 2). However, only the Wavelet transform method has a precision of more than 80%. This means that for the Curvelet transform algorithm, even under these ideal conditions, generate false alarms.

With Curvelet analysis, the algorithm has a difficult time to distinguish the background and detect them falsely as the forgery regions. Through the experiments, it proves that the proposed method which uses the forgery images analyzed by the Curvelet transform is not suitable for the block based method combining with Zernike moment to detect the copy-move forgery images.

Disscusion

Through these experiments, different perspectives of block based technique using a combination of the Zernike moment and multiscale analysis for digital image forgeries are studied. The Zernike moment's orders, block sizes, the number of scale for multiscale analysis and different kinds of multiscale analyses are considered.

For higher Zernike moment's order, the computational time will increase proportionally along with the increasing in orders. Although the complexity in computation increases, the detection results are not worth it. In another word, the F − measure parameter is not going up with higher orders but for higher orders; some

testing images proved that its detectability performance is worse than a lower order.

According the experiments' results and other block based techniques, the block size of 16x16 is more favorable than the others. From the detection results with different block sizes, we can see that the trade-off between precision and recall of 16x16 size is the most suitable for the block based method for detecting image forgeries. Although the block size of 8x8 can also provide a comparable F – measure, for such a small blocks, the information it holds may not enough to be distinctive to other blocks. Therefore, the block size of 16x16 is used for different experiments in this research.

From the test by changing the number of scale of the Curvelet transform, the detection results between different numbers of scale is quite similar to each other. Nevertheless, the scale of two some provides better detection results compare to other number of scale. As shown above, the trade-off between the precision and recall for the scale of two is slightly superior compare to other number of scales.

Although the Curvelets analysis has many superior characteristics compare to the Wavelets analysis, in this block based technique combining with Zernike moment for digital image forgery detection, the detectability of the Curvelets analysis method shown is worse than the Wavelets analysis method's for most of the cases. The performance of the Curvelets analysis has yet reached the expectation of a multiscale analysis which allowed an almost optimal sparse representation of objects with singularities along smooth curves. The detection results have shown that in this digital image forgery detection method, the characteristics of the Curvelet transform have not utilized. Moreover, the precision of the detection results is also lower than the Wavelets analysis which leads to the lower F – measure.

In this research, the proposed idea of using the Curvelets analysis combining with Zernike moments by my study is not suitable. This method not only failed to utilize the characteristics of the Curvelet transform, but also reduced the efficiency of detecting a tampered image. Comparing to the Wavelets analysis, the Curvelets analysis is not feasible for combining with Zernike moments in the block based techniques.

6   Simulations and evaluations

In previous method [1], the combination between the Zernike moments and the Wavelet transform for detecting digital image forgeries has successfully achieve the desired goal. The computational cost reduces significantly while the precision is still acceptable. However, the Wavelets transform has disadvantage when analyzing edges or curves details. Continuing that research paper, another multi-resolution, the Curvelet transform, is used to combine with the Zernike moment in the block based technique for detecting image forgeries in order to solve that problems and increase the precision of the proposed method in the research paper although the computational cost will increase as the compensation for the additional information of the images. The efficiency of the algorithm will be evaluated through three parameters: precision, recall and F – which is combination of precision and recall in one parameter. Other parameter such as Zernike moment's order and the block size are chosen after conducting the experiments to evaluate the effect of them on the performance. From real experiments, we get a conclusion:

The Wavelets transform gives a desired result with low computational time and high precision, its advantages is simplicity, easy to implement and require low computational resources. Most images preprocess by the Wavelets transform have expectation results, however, for some images where the details on edge and curve that neighbor to each other is blended in after the Wavelets preprocessing, therefore having a low precision.

The Curvelet transform provides lower results, contrary to the expectation. Although the edge and curve details are clearer, images preprocessed by the Curvelet transform have lower performance and much higher Zernike moments calculation time than the Wavelets transform combination. From the simulation results, the Curvelet transform is not suitable for the proposed method which used the combination of Zernike moment based and the Curvelet analysis in the block based technique for detecting image forgeries.

The combination of the multiscale analysis and Zernike moment based technique is still not tested with different transformation attack such as rotation and scaling. Although the Zernike moment is known for the rotation invariant, in this block based technique, we have yet conducted experiments for rotation attacks.

For future research, in the problem of identifying the copied areas of a digital image, we may explore the rotation invariant characteristic of the Zernike moments which can robust against rotation attacks. Moreover, by combining with SIFT or SURF, the running time of Zernike moments can be enhanced.

For the Curvelet transform, a new method is needed to utilize its features. There are some feasible algorithms such as keypoint-based algorithms or block based algorithms which do not use Zernike moments but intensity-based or frequency based.

7   Acknowledgement

8   References

[1] Thuong Le – Tien, Marie Luong, Tu Huynh – Kha, Long Pham – Cong – Hoan, An Tran – Hong, "Block Based Technique for Detecting Copy-Move Digital Image Forgeries: Wavelet Transform and Zernike Moments", Proceedings of The Second International Conference on Electrical and Electronic Engineering, Telecommunication Engineering, and Mechatronics, Philippines 2016, ISBN 978-1-941968-30-7.

[2]   Vincent Christlein, Christian Riess, Johannes Jordan, Corinna Riess and Elli Angelopoulou, "An Evaluation of Popular Copy-Move Forgery Detection Approaches", IEEE Transactions on Information Forensics and Security, 2012.

[3]   Seung-Jin Ryu, Min-Jeong Lee, and Heung-Kyu Lee, "Detection of copy-rotate-move forgery using Zernike moments", Lecture Note in Department of Computer Science, Korea Advanced Institute of Science and Technology Volume 6387, pp 51-65, 2010, Daejeon, Republic of Koea.

[4]   Tu Huynh–Kha, Thuong Le–Tien, Khoa Van Huynh, Sy Chi Nguyen, "A survey on image forgery detection techniques," The Proceeding of 11-th IEEE-RIVF International Conference on Computing and Communication Technologies," Can Tho, Vietnam, Jan 25-28 2015.

[5]   G. Li, Q. Wu, D. Tu, and S. Sun, "A Sorted Neighborhood Approach for Detecting Duplicated Regions in Image Forgeries based on DWT and SVD," in Proceedings of IEEE International Conference on Multimedia and Expo, Beijing China, July 2-5, 2007, pp. 1750-1753.

[6]   E.J. Cand`es and D.L. Donoho. Curvelets and curvilinear integrals. J. Approx. Theory., 113:59–90, 2000.

[7]   J.-L. Starck, E. Cand`es, and D.L. Donoho. The curvelet transform for image denoising. IEEE Transactions on Image Processing, 11(6):131–141, 2002.

[8]   Fridrich, D. Soukal, and J. Lukás, "Detection of copy move forgery in digital images," in Proc. Digital Forensic Research Workshop, Aug. 2003.

[9]   P.Subathro, A.Baskar, D. Senthil Kumar, "Detecting digital image forgeries using re-sampling by automatic Region of Interest (ROI)," ICTACT Journal on image and video processing, Vol. 2, issue. 4, May 2012.

[10] E. S. Gopi, N. Lakshmanan, T. Gokul, S. Kumara Ganesh, and P. R. Shah, "Digital Image Forgery Detection using Artificial Neural Network and Auto Regressive Coefficients," Electrical and Computer Engineering, 2006, pp.194-197.

[11] Rafael C. Gonzalez, "Digital Image Processing", Prentice-Hall, Inc, 2002, ISBN 0-201-18075-8.

[12] D.L. Donoho and M.R. Duncan. Digital Curvelet Transform: Strategy, Implementation and Experiments; Technical Report, Stanford University 1999.

[13] E.J. Candès and D.L. Donoho. Curvelets – A Surprisingly Effective Non-adaptive Representation for Objects with Edges; Curve and Surface Fitting: Saint Malo 1999.

[14] Ma, Jianwei, and Gerlind Plonka. "The curvelet transform." IEEE Signal Processing Magazine 27.2 (2010): 118-133.

[15] Jiulong Zhang and Yinghui Wang, "A comparative study of wavelet and Curvelet transform for face recognition", 2010 3rd International Congress on Image and Signal Processing (Vol 4), IEEE, Oct. 2010

[16] Thuong Le - Tien, "Chapter 10: Morphological image processing", Lecture notes, for image/video processing and application, HCMUT, November, 2015.

[17] Dey, Nilanjan, Anamitra Bardhan Roy, and Sayantan Dey. "A novel approach of color image hiding using RGB color planes and DWT." arXiv preprint arXiv:1208.0803 (2012).

[18] Bhattacharya, Tanmay, Nilanjan Dey, and S. R. Chaudhuri. "A novel session based dual steganographic technique using DWT and spread spectrum." arXiv preprint arXiv:1209.0054 (2012).

[19] Dey, Nilanjan, et al. "DWT-DCT-SVD based intravascular ultrasound video watermarking." Information and Communication Technologies (WICT), 2012 World Congress on. IEEE, 2012.

[20] Dey, Nilanjan, et al. "DWT-DCT-SVD based blind watermarking technique of gray image in electrooculogram signal.", IEEE 12th International Conference on Intelligent Systems Design and Applications (ISDA), 2012.

[21] Dey, Nilanjan, Moumita Pal, and Achintya Das. "A Session Based Blind Watermarking Technique within the NROI of Retinal Fundus Images for Authentication Using DWT, Spread Spectrum and Harris Corner Detection." arXiv preprint arXiv:1209.0053 (2012).

[22] Bhattacharya, Tanmay, Nilanjan Dey, and S. R. Chaudhuri. "A novel session based dual steganographic technique using DWT and spread spectrum." arXiv preprint arXiv:1209.0054 (2012).

[23] Dey, Nilanjan, et al. "Lifting wavelet transformation based blind watermarking technique of photoplethysmographic signals in wireless telecardiology." IEEE World Congress on Information and Communication Technologies (WICT), , 2012.

[24] Dey, Nilanjan, et al. "Stationary wavelet transformation based self-recovery of blind-watermark from electrocardiogram signal in wireless telecardiology.", International Conference on Security in Computer Networks and Distributed Systems. Springer Berlin Heidelberg, 2012.

[25] Dey, Nilanjan, et al. "Wavelet based watermarked normal and abnormal heart sound identification using spectrogram analysis.", IEEE International Conference on Computational Intelligence & Computing Research (ICCIC), 2012.

[26] Mukhopadhyay, Sayantan, et al. "Wavelet based QRS complex detection of ECG signal." arXiv preprint arXiv:1209.1563 (2012).

[27] Hemalatha, S., and S. Margret Anouncia. "A Computational Model for Texture Analysis in Images with Fractional Differential Filter for Texture Detection." International Journal of Ambient Computing and Intelligence (IJACI), Vol.7, Issue 2, 2016.

[28] Sharma, Komal, and Jitendra Virmani. "A Decision Support System for Classification of Normal and Medical Renal Disease Using Ultrasound Images: A Decision Support System for Medical Renal

Diseases.", International Journal of Ambient Computing and Intelligence (IJACI) 8.2 (2017): 52-69.

[29] Boulmaiz, Amira, et al. "Design and Implementation of a Robust Acoustic Recognition System for Waterbird Species using TMS320C6713 DSK.", International Journal of Ambient Computing and Intelligence (IJACI) 8.1 (2017): 98-118.

[30] Murray, Niall, et al. "Future Multimedia System: SIP or the Advanced Multimedia System.", International Journal of Ambient Computing and Intelligence (IJACI), 3.1(2011).

[31] Fouad, Khaled Mohammed, Basma Mohammed Hassan, and Mahmoud F. Hassan. "User Authentication based on Dynamic Keystroke Recognition." International Journal of Ambient Computing and Intelligence (IJACI) 7.2 (2016): 1-32.

[32] D.L. Donoho and M.R. Duncan. Digital curvelet transform: strategy, implementation and experiments. In H.H. Szu, M. Vetterli, W. Campbell, and J.R. Buss, editors, Proc. Aerosense 2000, Wavelet Applications VII, volume 4056, pages 12–29. SPIE, 2000.

[33] E.J. Candès and D.L. Donoho. Curvelets – A Surprisingly Effective Non-adaptive Representation for Objects with Edges; Curve and Surface Fitting: Saint Malo 1999.

[34] Ma, Jianwei, and Gerlind Plonka. "The curvelet transform." IEEE Signal Processing Magazine 27.2 (2010): 118-133.

[35] Sundus Y. Hasan, "Study of Zernike moments using analytical Zernike polynomials", Pelagia Research Library, Iraq, 2012.

Aggregation Methods in Group Decision Making: A Decade Survey

Wan Rosanisah Wan Mohd and Lazim Abdullah
School of Informatics and Applied Mathematics
Universiti Malaysia Terengganu, 21030, Kuala Terengganu, Terengganu, Malaysia
E-mail: wanrosanisahwm@gmail.com, lazim_m@umt.edu.my

A number of aggregation method has been proposed to solve various selected problems. In this work, we make a survey of the existing aggregation method which were used in various fields from 2006 until 2016. The information of the aggregation method retrieved from some academic databases and keywords that appeared through international journals from 2006 to 2016 are gathered and analyzed. It is observed that eighteen over ninety five of journal articles or nineteen percent applied the Choquet integral to the selection process. This survey shows this method most prominent compared to the other aggregation method and it is a good indication to the researchers to expand the appropriate aggregation method in MCDM. Besides that, this paper will give the useful information for other researches since the information given in this survey provides the latest evidence about the aggregation operator

Povzetek: Predstavljena je analiza metod za združevanje odločitev skupine.

1   Introduction

Decision making is one of the most widely used management processes in dealing with real world problems which is typically characterized by complex and difficult task. Multiple criteria decision making (MCDM) has been one of the fastest growing knowledge areas in decision sciences and has been used extensively in many disciplines. For example, Roy [1] developed a multi criteria decision analysis for renewable energy sources where it deals with the process of making decisions in the presence of multiple objective. Many researchers used other diverse methods of MCDM to solve the decision problem [2]. Basically, MCDM is a branch of operation research models and a well-known field of decision making. Both quantitative as well as qualitative criteria and attributes can be solved using the methods and it can analyze conflict in criteria and decision makers as well [3]. MCDM problems usually divided into two types: continuous and discrete. MCDM problems have two categories: multi-objective decision making (MODM) and multi-attribute decision making (MADM). By MODM methods, the decision variable values can be determined in a continuous or integer domain as it has a large number of alternative choices. Thus, MADM methods are generally discrete, with a limited number of specified alternatives. Each matrix has four main parts, namely: (a) alternatives, (b) attributes, (c) weight or relative importance of each attribute and (d) measures of performance of alternatives with respect to the attributes [4].

MCDM deals with the problem of helping the decision maker to choose the best alternative according to several criteria. In order to meet this objective, an MCDM method has basically four steps that support making the most efficient and rational decisions. According to Pohekar and Ramachandran [3], the first basic step is structuring the decision process, alternative selection and criteria formulation. The second step is displaying tradeoff among criteria and determine criteria weights. Applying value judgment concerning acceptable tradeoffs and evaluation is the third step in most MCDM. Methods. The fourth step is calculating the final aggregation prior to make a decision. Some literature also suggests that MCDM can be divided into two phase process. The first phase is called as the rating phase where aggregation of the values of criteria for each alternative is made. Ranking or ordering between the alternatives with respect to the global consensus degree of satisfaction is another phase in MCDM. It can be seen that aggregation is one of the fundamental phases in MCDM. Many MCDM methods such as ELECTRE I, II, PROMETHEE use criteria weights in their aggregation process. Weights of criteria play an important role for measuring overall preferences of alternatives. It is necessary to aggregate the available information in order to make decisions. In other words, a central problem in multi-criteria problems are an aggregation of the satisfactions to the individual criteria to obtain a measure of satisfaction to the overall collection of criteria [5]. The overall evaluation value is then used to help select alternatives. It can be seen that aggregation is the fundamental prerequisite of the decision making, in which descriptions on how to aggregate individual experts' preference information on alternatives is succinctly made [6].

Aggregation is easily defined as the process of combining several numerical scores with respect to

each criterion by using an aggregation operator in order to produce a global score [7]. Detyniecki [8] defines an aggregation as 'mathematical object that has the function of reducing a set of numbers into a unique representative value'. Xu [9] defines aggregation is an essential process of gathering relevant information from multiple source. According to the [10], aggregation is made to summarize information in decision making. Omar and Fayek [11] defines aggregation in the multi-criteria decision making environments as a process of combining the values of a set of attributes into one representative value for the entire set of attributes. Aggregation is important in the decision making problem because it is used to derive a collective decision made by the decision makers by representing in the individual opinions. In addition, aggregation of individual judgments or preference is used to transform experts' judgment knowledge and expertise in relative weight.

The interest in the importance of aggregation is enhanced by judgments or preferences made by a group of decision makers. In group decision problem where number of decision makers are multiple, it is assumed that there exists a finite number of alternatives, as well as a finite set of experts. Each expert has their own opinions and may have a variety of ideas about the performance of each alternative and cannot estimate his/her preferences with crisp numerical value. Hence, a more realistic approach to be used to represent the situation of the human expert, instead of using the crisp numerical values. Thus, each variable involved in the problem may be represented in linguistic terms. Under those circumstances, aggregation methods are the key to tackle the mechanism to realize the comprehensive features of group decision making [12]. Several related research have been conducted to deal with multiplicity features in group decision making. Many researchers have studied aggregation operation using different aggregation methods. The way aggregation functions are used depends on the nature of the profile that is given to each user and the description of items [13]. In the real world of decision making problems, decision makers like to pursue more than one aggregation methods to measure the aggregation information. In these kinds of problems, many aggregation methods have been developed in this area for the recent years for judging the alternatives. For example, Ogryczak [14] proposed reference point method and implemented to the fair optimization method in analyzing the efficient frontier. The method was proposed based on the augmented max-min aggregation.

Aggregation operations on fuzzy sets are operations by which several fuzzy sets are combined together in some way to produce a single representative either fuzzy or crisp set. Therefore, aggregation operation needs aggregation operator to deal with the situation where the aggregation operator is commonly tools that can be used to combine the individual preference information into overall

preference information and deriving collective preference values for each alternative. That is to say, the information aggregation is to combine individual experts' preference coming from different sources into a unique representative value by using an appropriate aggregation technique [15]. Aggregation operations are used to rank the alternative decisions an expert or decision support system, which are established and applied in fuzzy logic systems. The information aggregation has received much attention from practitioners and researchers due to its practical and significance in academic [16][17][18][19][20][21].

The first overview of the aggregation operators in 2003 by Xu and Da [22]. The study is reviewed of the existing main aggregation operators and proposed some new aggregation operators that is induced ordered weighted geometric averaging (IOWGA) operator, generalized induced ordered weighted averaging (GIOWA) operator and hybrid weighted averaging (HWA) operator. In 2008, a review of aggregation functions which focus on some special classes of averaging, conjunctive and disjunctive is reviewed by Mesiar et al. [23]. Furthermore, Martinez and Acosta in 2015 [24] have made a review an aggregation operators taking into account of mathematical properties and behavioral measures such as disjunctive degree (orness), dispersion, balance operator, divergence instead of general mathematical properties whose verification might be desirable in certain cases: boundary condition, continuity, increasing, monotonicity etc. Since then, it is important to make a review of the aggregation operator which provide the latest method which will be used to solve the aggregation in MCDM. Obviously, there is no review paper on aggregation operator from year to year.

Our aim in this survey article is to provide an accessible overview of some key methods of aggregation in MCDM. We focus on development of type of aggregation methods that have attracted many researchers in this area without neglecting some technical details of the aggregation methods.

Throughout this survey, the terms aggregation function, aggregation operator, surveys on aggregation in MCDM, an overview of aggregation operation will refer to find the journal articles as well. The collected journals which is regarding the aggregation is retrieving from the various fields such as engineering, medical, operation research, image processing, selection problems, project management selection and etc.

This paper is structured as follows. Section 2 begins by laying out the various methods of aggregation since 2006 until now. Section 3 presents an analysis out of the survey. Section 4 suggests some works that can be extended as future research direction. The last chapter concludes.

2 Review of aggregation method

This chapter reviews aggregation methods that have been developed to aggregate information in order to choose a desirable solution, decision makers have to aggregate their preference information by means of some proper approaches. This review is made by analyzing the method used based on the journals and conference proceedings that are collected from selected popular academic databases such as SpringerLink, Scopus, ScienceDirect, IEEE Xplore Digital Library, ACM Digital Library, and Wiley Online Library from the year 2006 until the year 2016.

The class of aggregation is huge, making the problem of choosing the right method for a given application is a difficult one [25]. In this paper, we review the methods of aggregation and its various applications. Firstly, we segregate the aggregation methods by the basic ones which is the most often used aggregation operators. For example, the average (arithmetic mean), geometric mean and harmonic mean. Then, we proceed to the next aggregation operator by presenting a generalization of the classical one such as Bonferroni Mean (BM), power aggregation operator, fuzzy integral, hybrid aggregation operator, prioritized average operator and the linguistic aggregation operator [26].

2.1 Basic operators

2.1.1 Arithmetic mean operator

In real life decision situation, the aggregation problems in the MCDM are solved using the scoring techniques such as the weighted aggregation operator based on multi attribute theory. The classical weighted aggregation is usually known by the weighted average (WA) or simple additive weighting method. A very common aggregation operator is the ordered weighted averaging (OWA) operator which provides a parameterized family aggregation operator between the minimum, the maximum, the arithmetic average, and the median criteria whose originally introduced by Yager [27]. There are two features have been used to characterize the OWA operators. The first is the orness character and the second is the dispersion. The OWA has been widely used because of its ability to model linguistically expressed aggregation. Since the OWA operator is coming out, the approach has been employed by the most authors in a wide range of applications such as engineering, neural networks, data mining, decision making, and image processing as well as expert systems [28]. However, OWA operator was assumed that the available information includes crisp number or singletons. In the real decision making situation, it is found this may not be the crisp number. Sometimes, the available information is vague or imprecise and it is not possible to analyze it with the crisp numbers. Then, it is necessary to use another approach that is able to represent the uncertainty such as the use of fuzzy numbers (FNs). With the use of

FNs, it is possible to analyze the real situation into the fuzzy value. Here is the definition of OWA.

Definition 1: An OWA operator of dimension n is a mapping OWA: $R^n \rightarrow R$ that has associated weighting vector w of dimension n with $w_j \in [0,1]$ and $\sum_{j=1}^{n} w_j = 1$, such that [27]:

$$\text{OWA}\left(a_1, a_2, ..., a_n\right) = \sum_{j=1}^{n} w_j b_j$$

where b_j is the j th largest of the a_i.

There are some of the authors that being used OWA as aggregation operators such that Xu [29] developed the intuitionistic fuzzy ordered weighted averaging (IFOWA) operator, and the intuitionistic fuzzy hybrid averaging (IFHA) operator. Xu and Chen [30] investigated the interval-valued intuitionistic fuzzy multi-criteria group decision making based on arithmetic aggregation operators such as the interval-valued intuitionistic fuzzy weighted arithmetic aggregation (IIFWA) operator, the interval-valued intuitionistic fuzzy ordered weighted aggregation (IIFOWA) operator and the interval-valued intuitionistic fuzzy hybrid aggregation (IIFHA) operator.

Li and Li [31][32] developed the generalized OWA operators using IFSs to solve MADM in which weights and ratings of alternatives on attributes are expressed in IFS. Chang et al. and Merigo et al. [33][34]proposed the fuzzy generalized ordered weighted averaging (FGOWA) operator as it is an extension of the GOWA operator for the uncertain situation where the information given is in the form of fuzzy numbers. Zhao et al. [35] developed some new generalized aggregation operators such as generalized intuitionistic fuzzy weighted averaging (GIFWA) operator, generalized intuitionistic fuzzy ordered weighted averaging (GIFOWA) operator, generalized intuitionistic fuzzy hybrid averaging (GIFHA) operator, generalized interval-valued intuitionistic fuzzy weighted averaging (GIIFWA) operator, generalized interval-valued intuitionistic fuzzy hybrid average (GIIFHA) operator where the proposed method is the extension of the GOWA operators taking into account of the characterization both of intuitionistic fuzzy sets by a membership function and non membership function and interval-valued intuitionistic fuzzy sets whose fundamental characteristic is the values of its membership function is represented by the interval numbers rather than exact numbers.

Shen et al. [36] presented a new arithmetic aggregation operator which is induced intuitionistic trapezoidal fuzzy ordered weighting aggregation operator and applied to group decision making. Furthermore, in 2011, Casanovas et al. [37] introduced the uncertain induced probabilistic ordered weighted

averaging weighted averaging (UIPOWAWA) operator where it provides a parameterized family of aggregation operators between minimum and maximum in a unified framework between probability, the weighted average and the induced ordered weighted averaging (IOWA) operator. Merigo [38] developed a new aggregation model that unifies the weighted average (WA) and the induced ordered weighted average (IOWA) operator that is called induced ordered weighted averaging-weighted average (IOWAWA) operator by considering the degree of importance that each concept has in the aggregation.

Xu and Wang [39] proposed a new aggregation operator which calling induced generalized intuitionistic fuzzy ordered weighted averaging (I-GIFOWA) operator by considering the characteristics of both the generalized IFOWA and the induced IFOWA operator. In order to deal with the intuitionistic fuzzy preference information in group decision making, the induced generalized intuitionistic fuzzy ordered weighted averaging (I-GIFOWA) operator based on GIFOWA and the I-IFOWA operator. Yu [40] introduced generalized intuitionistic trapezoidal fuzzy weighted averaging operator to aggregate the intuitionistic trapezoidal fuzzy information. Xia et al. [41] proposed several new hesitant fuzzy aggregation operators by extending quasi-arithmetic means to hesitant fuzzy sets under group decision making. Zhou et al. [42] introduced a new operator for aggregating the interval-valued intuitionistic fuzzy values which called the continuous interval-valued intuitionistic fuzzy ordered weighted averaging (C-IVIFOWA) operator. Both intuitionistic fuzzy ordered weighted averaging (IFOWA) operator and the continuous ordered weighted averaging (C-OWA) operator has combined to control the parameter and employed to diminish the fuzziness and improve the preciseness of the decision making.

2.1.2   Geometric mean operator

The geometric mean operator is the traditional aggregation operator that proposed to aggregate information given on a ratio scale measurement in MCDM models. The main characteristics are its guaranties the reciprocity property of the multiplicative preference relations used to provide ratio preferences [43].

Definition 2: An OWG operator of dimension n is a mapping OWG: $R^{+n} \rightarrow R^+$ that has associated weighting vector $w = \left(w^1, w^2, ..., w^n\right)^T$ with $w_j > 0$ and $\sum_{j=1}^{n} w_j = 1$, such that [43]:

$$\text{OWG}_w\left(a_1, a_2, ..., a_n\right) = \prod_{j=1}^{n} b_j^{w_j}$$

where b_j is the j th largest of the $a_j (j = 1,2,...,n)$.

Some authors used geometric mean as aggregation operator. For example, Wu et al. [44] defined same families of geometric aggregation operators to aggregate trapezoidal IFNs (TrIFNs). Xu and Yager [45], Xu and Chen [46], Wei [47], Das et al. [48] developed some geometric aggregation operators based on IFS, such as intuitionistic fuzzy weighted geometric (IFWG) operator, intuitionistic fuzzy ordered weighted geometric (IFOWG) operator and intuitionistic fuzzy hybrid geometric (IFHG) operator. Tan [49] developed a generalized intuitionistic fuzzy geometric aggregation operator for multiple criteria decision making by considering the interdependent or interactive characteristics and preferences.

Wei [50], Xu [51] and Xu and Chen [52] proposed approach based on interval-valued intuitionistic fuzzy weighted geometric (IIFWG) operator, the interval-valued intuitionistic fuzzy ordered weighted geometric (IIFOWG) operator and the interval-valued intuitionistic fuzzy hybrid geometric (IIFHG) operator in different point of view. Verma and Sharma [53] proposed geometric Heronian mean (GHM) under hesitant fuzzy environment by developing some new GHM such that hesitant fuzzy generalized geometric Herinian mean (HFGGHM) operator and weighted hesitant fuzzy generalized geometric Herinian mean (WHFGGHM) operator.

2.1.3   Harmonic mean operator

Harmonic mean is the reciprocal of the arithmetic mean of reciprocal which is a conservative average to be used to provide for aggregation lying between the max and min operators and is widely used as a tool to aggregate central tendency data which is usually expressed in exact numerical values [54].

Definition 3: An ordered weighted harmonic mean operator of dimension n is a mapping OWHM: $R^n \rightarrow R$ that has associated weighting vector $w = \left(w_1, w_2, ..., w_n\right)^T$ with $w_j > 0$ and $\sum_{j=1}^{n} w_j = 1$, such that [54]:

$$\text{OWHM}_w\left(a_1, a_2, ..., a_n\right) = \frac{1}{\sum_{j=1}^{n} \frac{w_j}{a_{\sigma(j)}}}$$

where $(\sigma(1), \sigma(2), ... \sigma(n))$ is a permutation of $(1,2,...,n)$, such that $a_{\sigma(j-1)} \geq a_{\sigma(j)}$ for all $j = 2,...,n$.

Some researchers proposed harmonic mean as a method to solve aggregation in the decision making problem. For example, Xu [55] developed some fuzzy harmonic mean operators, such as fuzzy weighted harmonic mean (FWHM) operator, fuzzy ordered weighted harmonic mean (FOWHM) operator, fuzzy hybrid harmonic mean (FHHM) operator. The aim of this paper is to extend the induced ordered weighted harmonic mean (IOWHM) operator to fuzzy environment and propose a new operator called the

fuzzy induced ordered weighted harmonic mean (FIOWHM) operator.

Wei and Yi [56] proposed an aggregation operator including trapezoidal fuzzy ordered weighted harmonic averaging (ITFOWHA) operator and applied to the decision making.

Wei [57] proposed a new aggregation operator called fuzzy induced ordered weighted harmonic mean (FIOWHM) for fuzzy multi criteria group decision making. Zhou et al. [58] proposed the generalized hesitant fuzzy harmonic mean operators including the generalized hesitant fuzzy weighted harmonic mean operator (GHFWHM), the generalized hesitant fuzzy ordered weighted harmonic mean operator (GHFOWHM), the generalized hesitant fuzzy hybrid harmonic mean operator (GHFHHM) using the technique of obtaining values in the interval to the group decision making under hesitant fuzzy environment.

Liu et al. [59] proposed a generalized interval-valued trapezoidal fuzzy numbers (GIVFTN) is an extended of ordered weighted harmonic averaging operators to solve the problems in multiple attribute group decision making.

2.2 Bonferroni mean (BM)

The Bonferroni mean (BM) originally introduced by Bonferroni [60]. The classical Bonferroni mean is an extension of the arithmetic mean and its generalized by some researchers based on the idea of the geometric mean [61]. The BM is differ from the other classic means such as the arithmetic, the geometric and the harmonic because this mean reflect the interdependent of the individual criterion meanwhile on the classic means the individual criterion is independent [62]. The BM was originally introduced by Bonferroni [60], which was defined as follows:

Definition 4: Let $p, q \geq 0$, and $a_i (i = 1, 2, ..., n)$ be a collection of nonnegative numbers. If

$$B^{p,q}(a_1, a_2, ..., a_n) = \left(\frac{1}{n(n-1)} \sum_{\substack{i, j = 1 \\ i \neq j}}^{n} a_i^p a_j^q \right)^{\frac{1}{p+q}}$$

Then $B^{p,q}$ is called the Bonferroni mean (BM).

The Bonferroni mean (BM) operator is suitable for aggregating crisp data and can capture the expressed interrelationships among criteria, which plays a crucial role in multi-criteria decision making problems [63]. Since the BM introduced, this aggregation operator has received much attention from researchers and practitioners. Among them are, Yager [64] generalized the BM for enhancing its model capability and further Xu and Yager [65] developed intuitionistic fuzzy BM (IFBM) and applied the weighted IFBM to multi criteria decision making. Beliakov et al. [66] gave a

systematic investigation of a family of composing aggregation functions which generalize the Bonferroni Mean (BM).

Zhu et al. [67] explored the geometric Bonferroni mean (GBM) by considering both BM and the geometric mean (GM) under hesitant fuzzy environment. Xia et al. [68] developed the Bonferroni geometric mean, which is a generalization of the Bonferroni mean and geometric mean and can reflect the interrelationships among the aggregated arguments. Wei et al. [69] developed two aggregation operators called the uncertain linguistic Bonferroni mean (ULBM) operator and the uncertain linguistic geometric Bonferroni mean (ULGBM) operator for aggregating the uncertain linguistic information in the multiple attribute decision making (MADM) problems. Park and Park [70] extend the works Sun and Sun [61] by considering the interactions of any three aggregated arguments instead of any two to develop generalized fuzzy weighted Bonferroni harmonic mean (GFWBHM) operator and generalized fuzzy ordered weighted Bonferroni harmonic mean (GFOWBHM) operator. Verma [71] proposed a new generalized Bonferroni mean operator called generalized fuzzy number intuitionistic fuzzy weighted Bonferroni mean (GFNIFWBM) operator which is able to aggregate the fuzzy number intuitionistic fuzzy correlated information.

2.3 Power aggregation operators

Yager [17] was first introduced a power average (PA) operator which uses a non-linear weighted average aggregation tool and a power ordered weighted average (POWA) operator to provide aggregation tools which allow exact arguments to support each other in the aggregation process. The weighting vectors of the PA operator and the POWA operator depend on the input arguments and allow arguments being aggregated to support and reinforce each other. In contrast with most aggregation operators, the PA and POWA operators incorporate information regarding the relationship between the values being combined. Recently, these operators have received much attention in the literature.

Definition 5: The power average (PA) operator is mapping PA: $R^n \rightarrow R$ defined by the following formula [17]:

$$PA(a_i | i = 1, 2, ... n) = \frac{\sum_{i=1}^{n} (1 + T(a_i)) a_i}{\sum_{i=1}^{n} (1 + T(a_i))}$$

where $T(a_i) = \sum_{\substack{j=1 \\ j \neq i}}^{n} Sup(a_i, a_j)$

and $Sup(a_i, a_j)$ is the support for a_i from a_j. The support satisfies the following three properties:

(1) $Sup(a_i, a_j) \in [0,1]$;

(2) $Sup(a_i, a_j) = Sup(a_j, a_i)$;

(3) $Sup(a_i, a_j) \geq Sup(a_s, a_t)$ if $|a_i - a_j| < |a_s - a_t|$

Motivated by the success of the PA and POWA, Xu and Yager [72] proposed a power geometric average (PG) operator and a power ordered weighted average (POWGA) operator. Besides that, power aggregation operators have been further extended to accommodate multi attribute group decision making (MAGDM) under different uncertain environments. For instance, Xu and Cai [73] developed the uncertain power ordered weighted average (UPOWA) operator on the basis of the PA operator and the UOWA operator, Xu [74] introduced the uncertain ordered weighted geometric average (UOWGA) operator based on the PG operator and the UOWA operator.

Xu [75] under intuitionistic fuzzy and interval-valued intuitionistic fuzzy decision making environments, the linguistic power aggregation operators by Zhou et al. [76], generalized argument-dependent power operators by Zhou and Chen [15] to accommodate intuitionistic fuzzy preferences and power aggregation operators under interval-valued dual hesitant fuzzy linguistic environment and the power aggregation operators by Wan [77] under trapezoidal intuitionistic fuzzy decision making environments.

Zhang [78] developed a wide range of hesitant fuzzy power aggregation operators for hesitant fuzzy information such as the hesitant fuzzy power average (HFPA) operators, the hesitant power geometric (HFPG) operators, the generalized hesitant fuzzy power average (GHFPA) operators, the generalized hesitant fuzzy power geometric (GHFPG) operators, the weighted generalized hesitant fuzzy power average (WGHFPA) operators, the generalized hesitant fuzzy power geometric (WGHFPG) operators, the hesitant fuzzy power ordered weighted average (HFPOWA) operators, the hesitant fuzzy power ordered weighted geometric (HFPOWG) operators, the generalized hesitant fuzzy power ordered weighted average (GHPOWA) operators and the generalized hesitant fuzzy power ordered weighted geometric (GHPOWG) operators.

However, the arguments of these power aggregation operators are exact numbers. In practice, we often confront situations in which the input arguments cannot be expressed in the form of exact numerical values instead, they have to take in the form of interval numbers Qi et al. [79], intuitionistic fuzzy numbers (IFNs) [80][81][82], interval-valued intuitionistic fuzzy numbers (IVIFNs) [83], linguistic variables [84][85][86], uncertain linguistic variables [67][87], or 2-tuples [88], hesitant fuzzy sets (HFS) [81]. Gou et al. [81] developed a family of hesitant fuzzy power aggregation operators, for instance the

hesitant fuzzy power weighted average (HFPWA), hesitant fuzzy power weighted geometric (HFPWG) generalized hesitant fuzzy power weighted average (GHFPWA), generalized hesitant fuzzy power weighted geometric (GHFPWG) operators for multi-criteria group decision making problems.

Wang et al. [88] proposed a dual hesitant fuzzy power aggregation operators based on Archimedean t-conorm and t-norm for dual hesitant fuzzy information. Das and Guha [89] proposed some new aggregation operators such as trapezoidal intuitionistic fuzzy weighted power harmonic mean (TrIFWPHM) operator, trapezoidal intuitionistic fuzzy ordered weighted power harmonic mean (TrIFOWPHM) operator, trapezoidal intuitionistic fuzzy induced ordered weighted power harmonic mean (TrIFIOWPHM) operator and trapezoidal intuitionistic fuzzy hybrid power harmonic mean (TrIFhPHM) operator to aggregate the decision information.

2.4 Fuzzy integral

Another types of aggregation operators is fuzzy integrals (FI). There are many types of FI, and most of the well-known fuzzy integral are Choquet and Sugeno integral.

2.4.1 Choquet integral

One of the popular aggregation operator fuzzy integrals is the Choquet integral which is introduced by Choquet [90]. Choquet integral is defined as a subadditive or superadditive to integrate functions with respect to the fuzzy measures [91].

Definition 6. Let f be a real-valued function on X, the Choquet integral of f with respect to a fuzzy measure g on X is defined as [90]:

$$(C)\int f dg = \sum_{i=-1}^{n} \left[f(x_{(i)}) - f(x_{(i-1)}) \right] g(A_{(i)})$$

(1)

or equally by

$$(C)\int f dg = \sum_{i=-1}^{n} \left[g(A_{(i)}) - g(A_{(i+1)}) \right] f(x_{(i)})$$

(2)

where the parentheses used for indices represent a permutation on X such that $f(x_{(1)}) \leq \cdots \leq f(x_{(n)})$, $f(x_{(0)}) = 0$, $A_{(i)} = \{x_{(i)}, \ldots, x_{(n)}\}$, and $A_{(n+1)} = \phi$.

The Choquet integral is a very useful way of measuring the expected utility of an uncertain event

[92]. It is a tool to model the interdependence or correlation among different elements where a new aggregation operators can be defined. Choquet integral has been proposed by many authors as an adequate aggregation operator that extends the weighted arithmetic mean or OWA operator by taking into consideration the interactions among the criteria.

Yager [93] extended the idea of order induced aggregation to the Choquet aggregation and introduced the Choquet ordered averaging (I-COA) operator. Mayer and Roubens [94] aggregated the fuzzy numbers through the Choquet integral. In the other field, Hlinena et al. [95] used Choquet integral with respect to Lukasiewicz filters to present a partial solution to look for an appropriate utility function in a given setting. Ming-Lang et al. [96] proposed analytic network process (ANP) technique to get the relationships of feedback of criteria and Choquet integral is used to eliminate the interactivity of the expert subjective judgment problem and apply in the case study of selection of optimal supplier in supply chain management strategy (SCMS).

Buyukozkan and Ruan [97] proposed a two-addative Choquet integral to the software development experts and managers to enable them to position their projects in terms of associated risks. Murofushi and Sugeno [91] used the Choquet integral to propose the interval-valued intuitionistic fuzzy correlated averaging operator and interval-valued intuitionistic fuzzy correlated geometric operator to aggregate interval-valued intuitionistic fuzzy information and applied them to a practical decision making problem. Angilella et al. [98] proposed a non-additive robust ordinal regression on a set of alternatives by evaluating the utility in terms of Choquet integral which represent the interaction among thecriteria modelled by the fuzzy measure. Huang et al. [99] applied a generalized Choquet integral with a signed fuzzy measure based on the complexity to evaluate the overall satisfaction of the patients.

Demirel et al. [100] proposed generalization Choquet integral by taking consideration of information fusion between criteria and linguistic terms and fuzzy ANP as a fuzzy measure which can handle the dependent criteria and hierarchical problem structure and applied to the multi-criteria warehouse location.

Tan and Chen [101] proposed intuitionistic fuzzy Choquet integral based on t-norms and t-conorms meanwhile Tan [102] extended the TOPSIS method by combining the interval-valued intuitionistic fuzzy geometric aggregation operator with Choquet integral-based Hamming distance to deal with multi-criteria interval-valued intuitionistic fuzzy group decision making problems.

Bustince et al. [103] proposed a new MCDM method for interval-valued fuzzy preference relation which was based on the definition of interval-valued Choquet integrals. Yang and Chen [104] introduced some new aggregation operator including the 2-tuple correlated averaging operator, the 2-tuple correlated

geometric operator and the generalized 2-tuple correlated averaging operator based on the Choquet integral. Belles-sampera [10] developed the extensions of the degree of balance, the divergence, the variance indicator and Renyi entropies to characterize the Choquet integral.

Islam et al. [105] proposed Choquet integral using goal programming to multi-criteria based learning which combines both experts' knowledge and data.

In addition, Choquet integral is applied in the hesitant fuzzy environment. Some authors that used the Choquet integral in the hesitant fuzzy environment are Yu et al. [106], Xia et al. [40]. For example, Yu et al. [106] proposed Choquet integral aggregation operator for hesitant fuzzy elements (HFEs) and applied it to the MCDM problems, Xia et al. [40] applied the Choquet integral to get the weights of criteria for group decision making, Peng et al. [107] proposed Choquet integral methods which is an approach to multi-criteria group decision making (MCGDM) problem to rank the alternatives where the criteria are interdependent or interactive. Wang et al. [108] developed some Choquet integral aggregation operators with interval 2-tuple linguistic information and applied them to MCGDM problems.

2.4.2   Sugeno integral

Sugeno integral is one of the fuzzy integral which introduced by M. Sugeno in the year of 1974 [109]. Sugeno integral is proposed to compute an average value of some function with respect to a fuzzy measure. In particular, the Sugeno integral uses only weighted maximum and minimum functions [16]. The definition of Sugeno integral [109] as follows:

Definition 7: The (discrete) Sugeno Integral of a function $f : X \to [0,1]$ with respect to μ is defined as

$$\int f(x)d\mu = \max_{1 \le i \le n}\left(\min\left(f(x_i), \mu(A_i)\right)\right)$$

where $\{f(x_1), f(x_2), f(x_3), ..., f(x_n)\}$ are the ranges and they are defined as $f(x_1) \le f(x_2) \le f(x_3) \le .... \le f(x_n)$.

In recent years, some authors and practitioners that used Sugeno integral are Mendoza and Melin [110], Liu et al. [111], Tabakov and Podhorska [112], Dubois et al. [113].

Mendoza and Melin [110] extended the Sugeno integral with the interval type-2 fuzzy logic. The generalization composed the modifying the original equations of the Sugeno measures and Sugeno integral. This method is used to combine the simulation vectors into only one vector and lastly the system will be decided the best choice of recognition in the same manner than made with only one monolithic neural network, but the problem of complexity resolved.

Liu et al. [111] extended the componentwise decomposition theorem of lattice-valued Sugeno integral by introducing the concept of interval fuzzy-valued, intuitionistic fuzzy-valued and interval intuitionistic fuzzy-valued Sugeno integral. As a result,

the intuitionistic fuzzy-valued Sugeno integrals and the interval fuzzy-valued Sugeno integrals are mathematically equivalent. It shows that the interval intuitionistic fuzzy-valued Sugeno integral can be decomposed into the interval fuzzy-valued and intuitionistic fuzzy-valued Sugeno integrals or the original Sugeno integrals.

Tabakov and Podhorska [112] proposed fuzzy Sugeno integral as an aggregation operator of an ensemble of fuzzy decision trees in order to classify the corresponding HER-2/neu classes. They used three different fuzzy decision trees which are built over different image characteristics, colour values and structural factors and texture information. The fuzzy Sugeno integral has been used as an aggregation operator to design fuzzy trees and the final medical decision support information generated.

Dubois et al. [113] proposed two new variants of weighted minimum and maximum where the criteria weights play an important role of tolerance. The Sugeno integral is proposed to the residuated counterparts, which means, the weight support on subsets of criteria. Then, the dual aggregation operations called disintegrals are evaluated in terms of its defects rather than in terms of its positive features proposed. The maximal disintegral is when no defects at all are present and maximal integral when all the merits are sufficiently present.

2.5 Hybrid aggregation operators

The hybrid aggregation operator has been proposed by several authors. It is important to propose more than one aggregation operator so that a wide range of fuzzy aggregation operators can be used in a wide range of application in decision making problems. For instance, Jianqiang and Zhong [114] developed the intuitionistic trapezoidal weighted average arithmetic average operator and the intuitionistic trapezoidal weighted geometric average operator.

Zhang and Liu [115] proposed the weighted arithmetic averaging operator and the weighted geometric average operator to aggregate triangular fuzzy intuitionistic fuzzy information and applied it to the decision making problem.

Then, Merigo and Casanovas [116] proposed fuzzy generalized hybrid aggregation operators where further generalize the fuzzy geometric hybrid averaging (FGHA) and the fuzzy induced geometric hybrid average (FIGHA) by using quasi-arithmetic means and the new result are Quasi-FHA and the Quasi-FIHA operator.

Xia and Xu [117] first proposed fuzzy weighted averaging (HFWA), hesitant fuzzy weighted geometric (HFWG) operators, generalized hesitant fuzzy weighted averaging (GHFWA), generalized hesitant fuzzy weighted geometric (GHFWG) operators in solving decision making problems.

Yu et al. [118] proposed the interval-valued intuitionistic fuzzy prioritized weighted average (IVIFPWA) operator and interval-valued intuitionistic

fuzzy prioritized weighted geometric (IVIFPWG) operator to aggregate the IVIFNs.

Verma and Sharma [119] developed some prioritized weighted aggregation operators for aggregating trapezoid fuzzy linguistic information motivated by the idea of prioritized weighted average introduced by Yager [122] such that the trapezoid linguistic prioritized weighted average (TFLPWA) operator, the trapezoid linguistic prioritized weighted geometric (TFLWG), and the trapezoid linguistic prioritized weighted harmonic (TFLWH) operator.

Liao and Xu [120] introduced some new aggregation operators including the generalized hesitant fuzzy hybrid weighted averaging operator, the generalized hesitant fuzzy hybrid weighted geometric operator, and the generalized quasi hesitant fuzzy hybrid weighted geometric operator and their induced forms.

In 2016, Verma [121] proposed a new aggregation operator that based on the generalization of mean called generalized trapezoid fuzzy linguistic prioritized weighted average (GTFLPWA) operator for fusing the trapezoid fuzzy linguistic information. The prominent characteristics of the proposed operator does not only take into account the prioritization among the attributes and decision makers but also has a flexible parameter.

2.6 Prioritized operator

Prioritized Average (PA) operator is one of the aggregation operators which has a great interest among scholars. In practical situations, decision-makers usually consider different criteria priorities. To deal with this issue, Yager [122] developed prioritized average (PA) operators by modeling the criteria priority on the weights associated with the criteria, which depend on the satisfaction of higher priority criteria. The PA operator has many advantages over other operators. For example, the PA operator does not need to provide weight vectors and, when using this operator, it is only necessary to know the priority among the criteria.

Wei [123] extended the prioritized aggregation operator to hesitant fuzzy sets and developed some prioritized hesitant fuzzy operators in multicriteria decision making. As Yager [122] only discussed the criteria values and weights in the real number domain, thus Wang et al. [124] developed some prioritized aggregation operators for aggregating interval-valued hesitant fuzzy linguistic information.

Recently, some researchers have focused on fuzzy prioritized aggregation operator into intuitionistic fuzzy sets (IFSs) such as Yu et al. [117], Chen [125] proposed some interval-valued intuitionistic fuzzy aggregation operators such as the interval-valued intuitionistic fuzzy prioritized weighted average (IVIFPWA) operator and the interval-valued intuitionistic fuzzy prioritized weighted geometric (IVIFPWG) operator, Verma and Sharma [126] proposed two new aggregation operators such as intuitionistic fuzzy Einstein prioritized weighted

average (IFEPWA) operator and the intuitionistic fuzzy Einstein prioritized weighted geometric (IFEPWG) operator for aggregating intuitionistic fuzzy information meanwhile Liang et al. [127], Dong et al. [128] developed some new aggregation operator called generalized intuitionistic trapezoidal fuzzy prioritized weighted average operator and generalized intuitionistic trapezoidal fuzzy prioritized weighted geometric operator and apply to the multi-criteria group decision making. Verma and Sharma [129] also proposed two new prioritized aggregation operators for aggregate triangular fuzzy information called quasi fuzzy prioritized weighted average (QFPWA) operator and the quasi fuzzy prioritized weighted ordered weighted average (QFPWOWA) operator.

2.7   Linguistic aggregation operator

Often, human decision making is too complex or too weakly defined to be represented by the numerical analysis. It is always considered the available information is vague or imprecise and impossible to analyze it with numerical values. However, this may not represent the real situation found in the decision making problem. Therefore, the possible way to solve such situation, it is necessary to use a qualitative approach which is the linguistic variable to aggregate the fused information. The linguistic aggregation operators are offered when the situations of the information cannot be assessed with numerical values, but it is possible to use linguistic assessment [130]. There are some authors used the linguistic variables to aggregate the information in MCDM. For instance, Wang and Hao [131] presented a 2-tuple fuzzy linguistic evaluation model for selecting appropriate agile manufacturing system in relation to MC production. Herrera et al. [132] proposed a fuzzy linguistic methodology to deal with unbalanced linguistic term sets. Chang et al. [33] proposed a linguistic MCDM aggregation model to tackle to solve two problems which are the aggregation operators are usually independent of aggregation situation and there must be a feasible operator for dealing with the actual evaluation scores.

Xu and Chen [52] extended the well-known harmonic mean to represent the information in the linguistic situation and developed some linguistic harmonic mean aggregation operators such as the linguistic weighted harmonic mean (LWHM) operator, the linguistic ordered weighted harmonic mean (LOWHM) operator, and the linguistic hybrid harmonic mean (LHHM) operator for aggregating linguistic information.

Wei [133] proposed a method for multiple attribute group decision making based on the ET-WG and ET-OWG operators with 2-tuple linguistic information. Shen et al. [36] developed the belief structure-linguistic ordered weighted averaging (BS-LOWA), the BS linguistic hybrid averaging (BS-LHA) and a wide range of particular cases. Wei [130] extended the TOPSIS method for 2-tuple linguistic multiple attribute

group decision making with incomplete weight information. Then, Merigo et al. [130] developed linguistic weighted generalized mean (LWGM) and the linguistic generalized OWA (LGOWA) operator and applied to the decision making problems.

Besides that, there are some authors that used 2-tuple linguistic variables such as Wei [134] proposed the GRA-based linear programming methodology for multiple attribute group decision making with 2-tuple linguistic assessment information. Wei [135] utilized the gray relational analysis method for 2-tuple linguistic multiple attribute group decision making with incomplete weight information. Xu and Wang [136] developed some 2-tuple linguistic power aggregation operators. Wei [137] proposed some new aggregation operators which is the 2-tuple linguistic weighted harmonic averaging (TWHA), 2-tuple linguistic ordered weighted harmonic averaging (TOWHA) and 2-tuple linguistic combined weighted harmonic averaging (TCWHA) operators for multiple attribute group decision making. Zadeh [83] developed some new linguistic aggregation operators such as 2-tuple linguistic harmonic (2TLH) operator, 2-tuple linguistic weighted harmonic (2TLWH) operator, 2-tuple linguistic ordered weighted harmonic (2TLOWH) operator and 2-tuple linguistic hybrid harmonic (2TLHH) operator to utilize to aggregate preference information considering linguistic variables in the decision making problem. Then, Li et al. [138] developed a new multiple attribute decision making approach for dealing with 2-tuple linguistic variable based on induced aggregation operators and distance measure by presenting 2-tuple linguistic induced generalized ordered weighted averaging distance (2LIGOWAD) operator which extension of the induced generalized ordered weighted distance (IGWOD) with 2-tuple linguistic variables. The 2LIGOWAD basically uses the IOWA operator represented in the form of 2-tuple linguistic variables.

Furthermore, Liu and Jin [139] introduced operational laws, expected value definitions, score functions and accuracy functions of intuitionistic uncertain linguistic variables and proposed two approaches with intuitionistic uncertain linguistic information to the weighted geometric average (IULWGA) operator and ordered weighted geometric (IULOWG) operator for multi attribute group decision making.

3   Observation

Throughout this study, hundred three journal articles have been reviewed with different aggregation methods within 2006 until 2016 searching via IEEE explore, Science Direct, Springer Link and Wiley online Library. In this paper, the methods and applications of aggregation are discussed in various fields. Based on the Table 1, the most popular aggregation operator is Choquet integral which is 17.48%, then it follows by linguistic aggregation operator (15.53%), arithmetic average operator

(14.74%), power aggregation operator and geometric mean operator (9.71%) respectively, prioritized aggregation operator (8.74), Benferroni Mean and Hybrid aggregation operator (7.77%), harmonic mean (4.85%) and Sugeno integral (3.88%). The details of the percentage are shown in the Figure 1.

Table 1: Numbers and percentage of the aggregation methods.

Aggregation Methods	No. of Authors	Percentage (%)
Arithmetic Mean	15	14.56
Geometric Mean	10	9.71
Harmonic Mean	5	4.85
Bonferroni Mean (BM)	8	7.77
Power	10	9.71
Choquet Integral	18	17.48
Sugeno Integral	4	3.88
Hybrid	8	7.77
Prioritized	9	8.74
Linguistic	16	15.53
Total	103	100

Based on the Table 1 and Figure 1, the Choquet integral have attracted more attention because it is usually known in the literature as a flexible aggregation operator and it is a generalization of the weighted average (WA) or simple additive weighting method, the ordered weighted average, and the max-min operator [130].

In addition, the Choquet integral is the appropriate tool to solve the interactions among the criteria in decision making problem as the traditional multi-criteria decision making (MCDM) methods are based on the additive concept along with the independence assumption where, in fact, each individual criterion is not completely independent [95].

Furthermore, the aggregation operator based on the linguistic variable has received considerable attention too. The linguistic information is used when the information available is vague or imprecise, but unable to analyze it using numerical values [131].
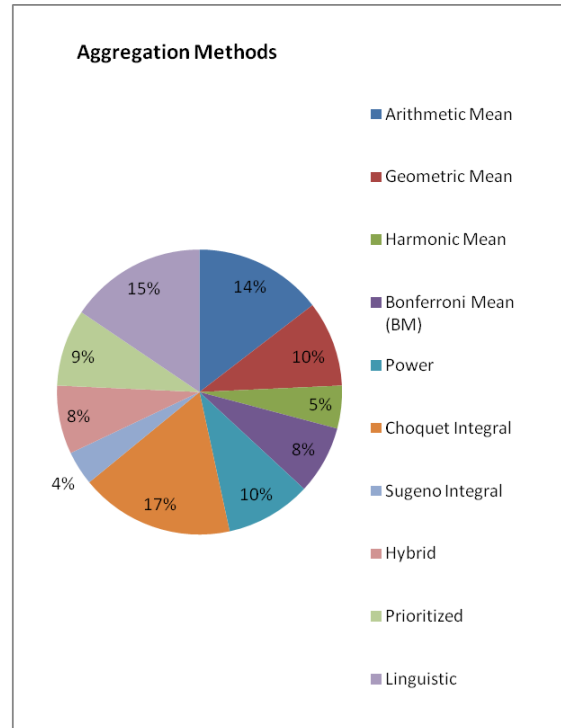
Figure 1: Percentage of aggregation methods.

Other than that, one of the conventional aggregation operators which is arithmetic mean also received attention from many scholars. They have been widely used because the first aggregation operator which introduced by Yager in 1988 is OWA which provide parameterized the arithmetic mean. Then, it is easy to compute. Since then, many researchers have developed aggregation operator that based on the arithmetic since because it is practical in the decision making problem.

Besides that, there are many aggregation operators used in the decision making problem depending on various kinds of factors investigated. For example, most of these operators, however, can only be used in situations where the input arguments are the exact values, and few of them can be used to aggregate the linguistic preference information.

4 Future work

From the observations, there are many types of aggregation operator that have been used by the researchers. Recently, the most appealing and great attention of researchers is Choquet integral because this method can represent the interaction between criteria, ranging from negative interaction to positive interaction. Even the classical of aggregation operator, power operator and linguistic variables have attracted numbers of researchers to apply to this field. It is suggested that the Choquet integral in order to build a more robust method and can be improved or extended by taking into account a weighted combination of both experts' knowledge and data. This suggestion is based on the only expert opinion can be overly subjective and may not result in desired performance.

The Choquet integral derives from the large numbers of coefficients $\left(2^n - 2\right)$ associated with a fuzzy measure, however, this flexibility can drive to be a serious drawback, especially when assigning real values to the importance of all possible combinations.

Further research could be further in selecting the fuzzy measure to the Choquet integral. It is because different fuzzy measure will be impacted to the Choquet integral.

Since the linguistic aggregation operator shows more than fifty percent of overall, it is possible to further to the next phase. In the future, it may extend this approach to other situations that can be assessed with other linguistic approaches and introducing the new aspects in the formulation by integrating them with other types of aggregation operators.

The arithmetic aggregation operator also shows the highest percentage among other aggregation operator. It is expected to expand in the future by using a generalized aggregation operator, distance measures and unified aggregation operators. Moreover, it can be represented in the uncertain environment using fuzzy numbers and linguistic variables.

5   Conclusion

The main purpose of this study is to find the appropriate aggregation operator that is able to present aggregation by taking account the importance of the data that being fused.

In this paper, we have analyzed the method used based on the journals and conference proceedings that are collected from selected popular academic databases.

From the collected journal, we have separated them according to the method that being used by the authors. Each of the aggregation method has been presented in the percentage. See Table 1 and Figure 1 in section 3.

From the observation, most of the criteria of the classical and linguistic aggregation operator in decision-making methods mentioned above are assumed to be independent of one another, but in reality, the criteria of the problems are often interdependent or interactive. For real decision making problems, it does not need the assumption that criteria or preferences are independent of one another and was used to show as a powerful tool for modeling interaction phenomena in decision making [100]. Usually, there is interaction among preference of decision makers. This phenomenon is called correlated criteria.

In the real world of decision making problems, most criteria have interdependent, interactive or correlative characteristics. The interaction phenomena among criteria or the preference of experts is considered which is making it more feasible and practical than other traditional aggregation operator. In addition, it is not suitable for us to aggregate them by classical weighted arithmetic mean or geometric mean method which based on the additive measure. On the contrary, to approximate the human subjective decision making process, it would be more suitable to apply fuzzy measures, where it is not assuming additivity and independence among decision making criteria

To tackle this problem, Choquet integral is a powerful tool to solve the MCDM problems with correlated criteria. In the Choquet integral model, criteria can be interdependent, a fuzzy measure is used to define a weight on each combination of criteria, thus making it possible to model the interaction existing among the criteria. Besides that, Choquet integral which taking into account for correlated inputs may give a more accurate prediction of the users' rating. Furthermore, it is a very useful tool to measure the expected utility of an uncertain environment.

6   References

[1]   Roy, B. and PH. Vincke, Multicriteria Analysis Survey and New Directions, European Journal of Operational Research, vol. 8, pp. 207-218, 1981.

[2]   E. K. Zavadskas, Z. Turskis and S. Kildiene, *State of Art Surveys of Overviews on MCDM/MADM Methods,* Technological and Economic Development of Economy, vol. 20, no. 1, 165-179, 2014.

[3]   S. D. Pohekar and M. Ramachandran, Application of Multi-Criteria Decision Making to Sustainable Energy Planning-A Review, Renew Sustain Energy Rev, vol. 8, pp. 365-381, 2004.

[4]   C. Diakaki, E. Grigoroudis, N. Kabelis, D. Kolokotsa, K. Kalaitzakis, and G. Stavrakakis, A Multi-Objective Decision Model for the Improvement of Energy Efficiency in Buildings, Energy, vol. 35, pp. 5483-5496, 2010.

[5]   R. R. Yager. On Generalized Bonferroni Mean Operators for Multi-Criteria Aggregation. International Journal of Approximate Reasoning, vol. 50, no. 8, pp. 1279–1286, 2009a.

[6]   A. Sengupta and T. K. Pal, *Fuzzy Preference Ordering of Interval Numbers in Decision Problems.* New York: Springer, Heidelberg, 2009.

[7]   J. L. Marichal, *Aggregation Operator for Multicriteria Decision Aid.* Ph.D. Dissertations, University of Liege, 1999.

[8]   M. Detyniecki, *Mathematical Aggregation Operators and Their Application to Video Querying.* PhD thesis. University of Paris, 2000.

[9]   Z. S. Xu, Fuzzy Harmonic Mean Operators, International Journal of Intelligent Systems, vol. 24, no. 2, pp. 152-172, 2009.

[10] J. Belles-sampera, J. M. Merigó and M. Santolino, Some New Definitions of Indicators for the Choquet Integral, pp. 467–476, 2013.

[11] M. N. Omar and A. R. Fayek, A TOPSIS-based Approach for Prioritized Aggregation in Multi-Criteria Decision-Making Problems, Journal of Multi-Criteria Decision Analysis, 2016.

[12] J. Vanicek, I. Vrana and S. Aly, Fuzzy Aggregation And Averaging for Group Decision Making: A Generalization and Survey. Knowledge-Based Systems, vol. 22, pp. 79-84, 2009

[13] A. K. Madan, M. S. Ranganath, Multiple Criteria Decision Making Techniques in Manufacturing Industries-A Review Study with Application of Fuzzy, International Conference of Advance Research and Innovation (ICARI-2014), pp. 136-144, 2014.

[14] Ogryczak, W. On Multicriteria Optimization with Fair Aggregation of Individual Achievements. In: CSM'06: 20th Workshop on Methodologies and Tools for Complex System Modeling and Integrated Policy Assessment, IIASA, Laxenburg, Austria, 2006.

[15] L. G. Zhou and H. Y. Chen, A Generalization of the Power Aggregation Operators for Linguistic Environment and Its Application In Group Decision Making, Knowl Based Syst, vol. 26, pp. 216-224, 2012.

[16] J. L. Marichal, On Sugeno Integral as an Aggregation Function, Fuzzy Sets and Systems, vol. 114, pp. 347-365, 2000.

[17] R. R. Yager, The Power Average Operator, IEEE Transactions on Systems, Man and Cybernetics, vol. 31, pp. 724-731, 2001.

[18] V. Torra, The Weighted OWA Operator, International Journal Intelligent System, vol. 12, pp. 153-166, 1977.

[19] H. F. Wang and S. Y. Shen, Group Decision Support with MOLP Applications. IEEE Trans Syst Man Cybern, vol. 19, pp. 143-153, 1989.

[20] R. Narasimhan, A Geometric Averaging Procedure for Constructing Supertransitivity Approximation to Binary Comparison Matrices, Fuzzy Sets System, vol. 8, pp. 53-61, 1982.

[21] S. Ovchinnikov, An Analytic Characterization of Some Aggregation Operator, International Journal Intelligent System, vol. 7, pp. 765-786, 1998.

[22] Z. S. Xu and Q. L. Da, An Overview of Operators for Aggregating Information, International Journal of Intelligent Systems, vol. 18, pp. 953-969, 2003.

[23] R. Mesiar, A. Kolesarova, T. Calvo and M. Komornikova, A Review of Aggregation Functions. *Fuzzy Sets and Their Application Extensions: Representation, Aggregation, and Models,* 2008.

[24] D. L. L. R. Martinez and J. C. Acosta, Aggregation Operators Review-Mathematical Properties and Behavioral Measures, International Journal Intelligent Systems and Applications, vol. 10, pp. 63-76, 2015.

[25] M. Grabisch, J. L. Marichal R. Mesiar and E. Pap, Aggregation Functions: Means, Information Science, vol.181, pp. 1-22, 2011.

[26] M. Detyniecki, *Fundamentals on Aggregation Operators*. AGOP, Berkeley, 2001.

[27] R. R. Yager, On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision Making, IEEE Transactions on Systems, Man, and Cybernetics, vol. 18, no. 1, 1988.

[28] T. Calvo, G. Mayor and R. Mesiar, Aggregation Operators: New Trends and Application. Physica-Verlag, New York, 2002.

[29] Z. Xu, Multi-Person Multi-Attribute Decision Making Models Under Intuitionistic Fuzzy Environment, Fuzzy Optim Decis Mak, vol. 6, no. 3, pp. 221-236, 2007.

[30] Z. S. Xu and J. Chen, Approach to Group Decision Making Based on Interval-Valued Intuitionistic Judgment Matrices, Systems Engineering-Theory and Practice, vol. 27, pp. 126-133, 2007.

[31] D. F. Li, Multiattribute Decision Making Method Based on Generalized OWA Operators with Intuitionistic Fuzzy Sets, Expert Systems with Applications, vol. 37, pp. 8673-8678, 2010.

[32] D. F. Li, The GOWA Operator Based Approach to Multiattribute Decision Making Using Intuitionistic Fuzzy Sets, Mathematical and Computer Modelling, vol. 53, pp. 1182-1196, 2011.

[33] J. R. Chang, S. Y. Liao and C. H. Cheng, Situational ME-LOWA Aggregation Model for Evaluating the Best Main Battle Tank, Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, pp. 1866-1870, 2007.

[34] J. M. Merigo and M. Casanovas, The Fuzzy Generalized OWA Operator And Its Application In Strategic Decision Making, Cybernetics and Systems, vol. 41, no. 5, pp. 359-370, 2010.

[35] H. Zhao, Z. S. Xu, M. F. Ni and S. S. Liu, Generalized Aggregation Operators for Intuitionistic Fuzzy Sets. International Journal of Intelligent Systems, vol. 25, pp. 1-30, 2010.

[36] L. Shen, H. Wang and X. Feng, *Some Arithmetic Aggregation Operators Within Intuitionistic Trapezoidal Fuzzy Setting and Their Application to Group Decision Making.* Management Science and Industrial Engineering (MSIE), pp. 1053-1059, 2011.

[37] M. Casanovas and J. M. Merigo, A New Decision Making Method with Uncertain Induced Aggregation Operator, Computational Intelligence in Multicriteria Decision Making (MCDM), pp. 151-158, 2011.

[38] J. M. Merigo, A Unified Model Between The Weighted Average and The Induced OWA Operator, Expert Systems with Applications, vol. 38, no. 9, pp. 11560-11572., 2011.

[39] Y. Xu and H. Wang, Approaches Based on 2-Tuple Linguistic Power Aggregation Operators for Multiple Attribute Group Decision Making Under Linguistic Environment, Applied Soft Computing, vol. 11, pp. 3988-3997, 2011.

[40] D. Yu, Intuitionistic Trapezoidal Fuzzy Information Aggregation Methods and Their

Applications to Teaching Quality Evaluation, Journal of Information & Computational Science, vol. 10, no. 6, pp. 1861-1869, 2013.

[41] M. Xia, Z. Xu, and B. Zhu, Geometric Bonferroni Means with Their Application in Multi-Criteria Decision Making, Knowledge-Based Systems, vol. 40, pp. 88-100, 2013.

[42] L. Zhou, Z. Tao, H. Chen and J. Liu, Continuous Interval-Valued Intuitionistic Fuzzy Aggregation Operators and Their Applications to Group Decision Making, Applied Mathematical Modelling, vol. 38, pp. 2190-2205, 2014.

[43] F. Chiclana, F. Herrera and E. Herrera-Viedma, The Ordered Weighted Geometric Operator: Properties and Applications in MCDM Problems, Physica-Verlag, Heidelberg, pp. 173-183, 2002.

[44] J. Wu and Q. W. Cao, Same Families of Geometric Aggregation Operators with Intuitionistic Trapezoidal Fuzzy Numbers, Applied Mathematical Modelling, vol. 37, no. 1, pp. 318-327, 2013.

[45] Z. Xu and R. R. Yager, Some Geometric Aggregation Operators, IEEE Transact Fuzzy Syst, vol. 15, pp. 1179-1187, 2006.

[46] Z. Xu and J. Chen, On Geometric Aggregation over Interval-Valued Intuitionistic Fuzzy Information. Fourth International Conference on Fuzzy Systems and Knowledge Discovery, IEEE Computer Society Press, pp. 466-471, 2007.

[47] G. U. Wei, Some Geometric Aggregation Functions and Their Application to Dynamic Multiple Attribute Decision Making in the Intuitionistic Fuzzy Setting, International Journal Uncertain Fuzzy Knowledge-Based System, vol. 17, no. 2, pp. 179-196, 2009.

[48] S. Das, S. Karmakar, T. Pal and S. Kar, Decision Making with Geometric Aggregation Operators based on Intuitionistic Fuzzy Sets, 2014 2nd International Conference on Business and Information Management (ICBIM), pp. 86-91, 2014.

[49] C. Tan, Generalized Intuitionistic Fuzzy Geometric Aggregation Operator And Its Application to Multi-Criteria Group Decision Making, Soft Computing, vol. 15, no. 5, pp. 867–876, 2011.

[50] G. W. Wei and X. R. Wang, Some Geometric Aggregation Operators Based on Interval-Valued Intuitionistic Fuzzy Sets and Their Application To Group Decision Making, International Conference on Computational Intelligence and Security, IEEE Computer Society Press, pp. 495-499, 2007.

[51] Z. S. Xu, Approaches to Multiple Attribute Group Decision Making Based on Intuitionistic Fuzzy Power Aggregation Operators, Knowledge-Based Systems, vol. 24, pp. 749-760, 2011.

[52] Z. S. Xu and J. Chen, An Approach to Group Decision Making Based on Interval-Valued Intuitionistic Judgment Matrices, System

Engineering Theory & Practice, vol. 27, no. 4, pp. 126-133, 2007.

[53] R. Verma and B.D. Sharma, Hesitant Fuzzy Geometric Heronian Mean Operators and Their Application to Multi-Criteria Decision Making, Mathematica Japonica, 2015.

[54] Z. S. Xu, Harmonic Mean Operator for Aggregating Linguistic Information, Fourth International Conference on Natural Computing, IEEE Computer Society, pp. 204-208, 2008.

[55] Z. S. Xu, Fuzzy Harmonic Mean Operators, International Journal of Intelligent Systems, vol. 24, pp. 152-172, 2009.

[56] G. Wei and W. Yi, Induced Trapezoidal Fuzzy Ordered Weighted Harmonic Averaging Operator, Journal of Information & Computational Science, vol. 7, no. 3, pp. 625-630, 2010.

[57] G. W. Wei, Some Harmonic Aggregation Operators with 2-Tuple Linguistic Assessment Information And Their Application to Multiple Attribute Group Decision Making, International Journal of Uncertainty, Fuzziness and Knowledge-Based System, vol. 19, no. 6, pp. 977-998, 2011.

[58] H. Zhou, Z. Xu and F. Cui, Generalized Hesitant Fuzzy Harmonic Mean Operators and Their Applications in Group Decision Making. International Journal of Fuzzy Systems, pp. 1-12, 2015.

[59] P. Liu, X. Zhang and F. Jin, A Multi-Attribute Group Decision-Making Method Based on Interval-Valued Trapezoidal Fuzzy Numbers Hybrid Harmonic Averaging Operators, Journal of Intelligent & Fuzzy Systems, vol. 23, no. 5, pp. 159-168, 2012.

[60] C. Bonferroni, Sulle Medie Multiple Di Potenze. Bolletino Matematica Italiana, vol. 5, no. 3, pp. 267-270, 1950.

[61] H. Sun and M. Sun, Generalized Bonferroni Harmonic Mean Operators and Their Application to Multiple Attribute Decision Making, Journal of Computational Information Systems, vol. 8, no. 14, pp. 5717-5724, 2012.

[62] M. Sun and J. Liu, Normalized Geometric Bonferroni Operators of Hesitant Fuzzy Sets and Their Application in Multiple Attribute Decision Making, Journal of Information & Computational Science, vol. 10, no. 9, pp. 2815-2822, 2013.

[63] P. D. Liu and F. Jin, The Trapezoidal Fuzzy Linguistic Bonferroni Mean Operators and Their Application to Multiple Attribute Decision Making, Scientia Iranica, vol. 19, no. 6, pp. 1947-1959, 2012.

[64] R. R. Yager, Prioritized OWA Aggregation. Fuzzy Optimization Decision Making, vol. 8, pp. 245-262, 2009.

[65] Z. Xu and R. R Yager, Intuitionistic Fuzzy Bonferroni Means, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 41, no. 2, pp. 568-578, 2011.

[66] G. Beliakov, S. James, J. Mordelova and T. Ruckschlossova, and R. R. Yager, Generalized Bonferroni Mean Operators in Multi-Criteria Aggregation. Fuzzy Sets and Systems, vol. 161, pp. 2227-2242, 2011.

[67] B. Zhu, Z. S. Xu and M. Xia, Hesitant Fuzzy Geometric Bonferroni Means, Information Science, vol. 205, pp. 72-85, 2012

[68] M. Xia, Z. S. Xu, and N. Chen, Some hesitant fuzzy aggregation operators with their application in group decision making, Group Decis, Negotiat, vol. 22, pp. 259-279, 2013.

[69] G. Wei, X. Zhao, R. Lin and H. Wang, Uncertain Linguistic Bonferroni Mean Operators and Their Application to Multiple Attribute Decision Making, Applied Mathematical Modelling, vol. 37, pp. 5277-5285, 2013.

[70] J. H. Park and E. J. Park, Generalized Fuzzy Bonferroni Harmonic Mean Operators and Their Applications in Group Decision Making, Journal of Applied Mathematics, vol. 2013, 1-14, 2013.

[71] R. Verma, Generalized Bonferroni Mean Operator For Fuzzy Number Intuitionistic Fuzzy Sets And Their Application to Multi Attribute Decision Making," International Journal of Intelligent Systems, vol. 30, no. 5, pp. 499-519, 2015.

[72] Z. S. Xu and R. R. Yager, Power-Geometric Operators and Their Use in Group Decision Making, IEEE Transaction on Fuzzy Systems, vol. 18, pp. 94-105, 2010.

[73] Z. S. Xu and X. Q. Cai, Uncertain Power Average Operators for Aggregating Interval Fuzzy Preference Relations. Group Decision and Negotiation, 2010.

[74] Z. S. Xu, Approach to Multiple Attribute Group Decision Making Based on Intuitionistic Fuzzy Power Aggregation Operator, Knowledge-Based Systems, vol. 24, no. 6, pp. 746-760, 2011.

[75] L. Zhou, H. Chen and J. Liu, Generalized Power aggregation Operators and Their Applications in Group Decision Making, Computers and Industrials Engineering, vol. 62, pp. 989-999, 2012.

[76] S. P. Wan, Power Average Operators of Trapezoidal Intuitionistic Fuzzy Numbers And Application to Multi-Attribute Group Decision Making, Applied Mathematical Modelling, vol. 37, no. 6, pp. 4112-4126, 2013.

[77] Z. Zhang, Hesitant Fuzzy Power Aggregation Operators and Their Application to Multiple Attribute Group Decision Making. Information Science, vol. 234, pp. 150-181, 2013.

[78] X. Qi, C. Liang and J. Zhang, Multiple Attribute Group Decision Making Based on Generalized Power Aggregation Operators under Interval-Valued Dual Hesitant Fuzzy Linguistic Environment, Int. J. Mach. & Cyber, 2015.

[79] T. Y. Chen, Bivariate Models of Optimism and Pessimism in Multi-Criteria Decision-Making Based on Intuitionistic Fuzzy Sets, Information Sciences, vol. 181, pp. 2139-2165, 2011.

[80] K. H. Gou, and W. L. Li, An Attitudinal-Based Method for Constructing Intuitionistic Fuzzy Information in Hybrid MADM Under Uncertainty, Information Sciences, 189, 77-92, 2012.

[81] J. Z. Wu, F. Chen, C. P. Nie and Q. Zhang, Intuitionistic Fuzzy-Valued Choquet Integral and Its Application in Multicriteria Decision Making, Information Sciences, vol. 222, pp. 509-527, 2013.

[82] Z. S. Xu, Induced Uncertain Linguistic OWA Operators Applied to Group Decision Making, Information Fusion, vol. 7, pp. 231-238, 2006.

[83] L. A. Zadeh, The Concept of A Linguistic Variable and Its Application to Approximate Reasoning Part 1, 2 and 3, Information Sciences, vol. 8, pp. 199-249, 1975.

[84] L. A. Zadeh, A computational approach to fuzzy quantifiers in natural languages. Computers & Mathematics with Applications, vol. 9, pp. 149-184, 1983.

[85] J. J. Zhu and K. W. Hipel, Multiple Stages Grey Target Decision Making Method with Incomplete Weight Based on Multi-Granularity Linguistic Label. Information Sciences, vol. 212, pp. 15-32, 2012.

[86] Z. S. Xu, Uncertain Linguistic Aggregation Operators Based Approach to Multiple Attribute Group Decision Making Under Uncertain Linguistic Environment. Information Sciences, vol. 168, pp. 171-184, 2004.

[87] L. Martinez and F. Herrera, An Overview on the 2-Tuple Linguistic Model for Computing with Words in Decision Making: Extensions, Application and Challenges, Information Sciences, vol. 207, pp. 1-18, 2012.

[88] L. Wang, Q. Shen and L. Zhu, L. Dual Hesitant Fuzzy Power Aggregation Operators Based on Archimedean T-Conorm and T-Norm And Their Application to Multiple Attribute Group Decision Making, Applied Soft Computing, vol. 38, pp. 23-50, 2016.

[89] S. Das and D. Guha, Power Harmonic Aggregation Operator With Trapezoidal Intuitionistic Fuzzy Numbers For Solving MAGDM Problems, Iranian Journal of Fuzzy Systems, vol.12, no. 6,pp. 41-74, 2015.

[90] G. Choquet, Theory of capacities. Annales de I'Institut Fourier, vol. 5, pp. 131-295, 1953.

[91] T. Murofushi and M. Sugeno, An Interpretation of Fuzzy Measure and the Choquet Integral as an Integral with Respect to A Fuzzy Measure, Fuzzy Sets and Systems, vol. 29, no. 2, pp. 201-227, 1989.

[92] Z. S. Xu, Choquet Integrals of Weighted Intuitionistic Fuzzy Information, Information Sciences, vol. 180, pp. 726-736, 2010.

[93] R. R. Yager. Induced Aggregation Operators, Fuzzy Sets and Systems, vol. 137, pp. 59-69, 2003.

[94] P. Mayer and M. Roubens, On The Use of the Choquet Integral with Fuzzy Numbers in Multiple Criteria Decision Support, Fuzzy Sets and Systems, vol. 157, pp. 927-938, 2006.

[95] D. Hlinena, M. Kalina, and P. Kral, Choquet Integral with Respect to Lukasiewicz Filters, and Its Modifications, Information Sciences, vol. 179, pp. 2912-2922, 2009.

[96] T. Ming-Lang, J. H. Chiang and L. W. Lan, Selection of Optimal Supplier in Supply Chain Management Strategy With Analytic Network Process And Choquet Integral, Computer & Industrial Engineering, vol. 57, pp. 330-340, 2009.

[97] G. Buyukozkan, and D. Ruan,. Choquet Integral Based Aggregation Approach to Software Development Risk Assessment. Information Science, vol. 180, pp. 441-451, 2010.

[98] S. Angilella,, S. Greco and B. Matarazzo, Non-Additive Robust Ordinal Regression : A Multiple Criteria Decision Model Based on the Choquet Integral, European Journal of Operational Research, vol. 201, no. 1, pp. 277–288, 2010.

[99] K. Huang, J. Shieh, K. Lee and S. Wu, Applying A Generalized Choquet Integral with Signed Fuzzy Measure Based on the Complexity to Evaluate the Overall Satisfaction of the Patients. *Proceeding of the Nineth International Conference on Machine Learning and Cybernetics, Qingdao, 11–14 July 2010.*

[100] T. Demirel, N. C. Demiral and C. Kahraman, Multi-Criteria Warehouse Location Using Choquet Integral, Expert Systems with Application, vol. 37, pp. 3943-3952, 2010.

[101] C. Tan and X. Chen, Intuitionistic Fuzzy Choquet Integral Operator for Multi-Criteria Decision Making, Expert Systems with Application, vol. 37, pp. 149-157, 2010.

[102] C. Tan, A Multi-Criteria Interval-Valued Intuitionistic Fuzzy Group Decision Making with Choquet Integral-Based TOPSIS, Expert Systems with Application, vol. 38, pp. 3023-3033, 2011.

[103] H. Bustince, J. Fernandez, J. Sanz, M. Galar, R. Mesiar and A. Kolesarova, Multicriteria Decision Making by Means of Interval-Valued Choquet Integrals. Advances in Intelligent and Soft Computing, vol. 107, pp. 269-278, 2012.

[104] W. Yang and Z. Chen, New Aggregation Operators Based on the Choquet Integral and 2-Tuple Linguistic Information, Expert Systems with Applications, vol. 39, no. 3, pp. 2662–2668, 2012.

[105] M. A. Islam, D. T. Anderson and T. C. Havens, Multi-Criteria Based Learning of the Choquet Integral using Goal Programming. *Conference on Soft Computing (WConSC), 2015 Annual Conference of the North American,* pp. 1-6, 2015.

[106] D. Yu, Y. Wu and W. Zhou, Multi-Criteria Decision Making Based on Choquet Integral under Hesitant Fuzzy Environment, Journal of Computational Information Systems, vol. 7, no. 12, pp. 4506-4513, 2011.

[107] J. J. Peng, J. Q. Wang, H. Zhou and X. H. Chen, A Multi-Criteria Decision-Making Approach Based on TODIM and Choquet Integral Within Multiset Hesitant Fuzzy Environment, Applied Mathematics & Information Sciences, vol. 9, no. 4, pp. 2087-2097, 2015.

[108] J. Q. Wang, D. D. Wang, Y. H. Zhang and X. H. Chen, Multi-Criteria Group Decision Making Method Based on Interval 2-Tuple Linguistic and Choquet Integral Aggregation Operators, Soft Computing, vol. 19, no. 2, pp. 389-405, 2015.

[109] M. Sugeno, *Theory of Fuzzy Integral and Its Application.* Doctoral Dissertation, Tokyo Institute of Technology, 1974.

[110] O. Mendoza and P. Melin, Extension of the Sugeno Integral with Interval Type-2 Fuzzy Logic, Fuzzy Information Processing Society, 2008. NAFIPS 2008. Annual Meeting of the North American, pp. 1-6, 2008.

[111] Y. Liu, Z. Kong and Y. Liu, Interval Intuitionistic Fuzzy-Valued Sugeno Integral, 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2012), pp. 89-92, 2012.

[112] M. Tabakov and M. Podhorska-Okolow, Using Fuzzy Sugeno Integral as an Aggregation Operator of Ensemble of Fuzzy Decision Trees in the Recognition of HER2 Breast Cancer Histopathology Images, 2013 International Conference on Computer Medical Applications (ICCMA), pp. 1-6, 2013.

[113] D. Dubois, H. Prade and Agnes. Rico, Residuated Variants of Sugeno Integrals: Towards New Weighting Schemes for Qualitative Aggregation Methods, Information Sciences, vol. 329, pp. 765-781, 2016.

[114] W. Jianqiang and Z. Zhong, Aggregation Operators on Intuitionistic Trapezoidal Fuzzy Number And Its Application to Multi-Criteria Decision Making Problem, Journal of Systems Engineering and Electronics, vol. 20, no. 2, pp. 321-326, 2009.

[115] Z. Zhang and P. Liu, Method for Aggregating Triangular Fuzzy Intuitionistic Fuzzy Information and Its Application to Decision Making, Technological and Economic Development of Economy, vol. 16, no. 2, pp. 280-290, 2010.

[116] J. M. Merigo and M. Casanovas, Fuzzy Generalized Hybrid Aggregation Operators and Its Application In Fuzzy Decision Making, International Journal of Fuzzy Systems, vol. 1, no. 1, pp. 15-23, 2010.

[117] M. Xia and Z. S. Xu, Hesitant Fuzzy Information Aggregation in Decision Making,

International Journal of Approximate Reasoning, vol. 52, no. 3, pp. 395-407, 2011.

[118] W. Yu, Y. Wu and T. Lu, Interval-Valued Intuitionistic Fuzzy Prioritized Operators and Their Application in Group Decision Making, Knowledge-Based Systems, vol. 30, pp. 57-66, 2012.

[119] R. Verma and B. D. Sharma, Trapezoid Fuzzy Linguistic Prioritized Weighted Average Operators and Their Application to Multiple Attribute Group Decision Making," Journal of Uncertainty Analysis and Applications, vol. 2, pp. 1-19, 2014.

[120] H. Liao and Z. Xu, Extended Hesitant Fuzzy Hybrid Weighted Aggregation Operators and Their Application in Decision Making. Soft Computing, vol. 19, pp. 2551–2564, 2015.

[121] R. Verma, Multiple Attribute Group Decision Making based on GeneralizedTrapezoid Fuzzy Linguistic Prioritized Weighted Average Operator, International Journal of Machine Learning and Cybernetics, 2016. DOI: 10.1007/s13042-016-0579-y

[122] R. R. Yager. Prioritized Aggregation Operators, International Journal Approximate Reasoning, vol. 48, pp. 263-274, 2008.

[123] G. W. Wei, Hesitant Fuzzy Prioritized Operators and Their Application to Multiple Attribute Group Decision Making, Knowledge-Based System, vol. 31, pp. 176-182, 2012.

[124] J. Q. Wang, J. T. Wu, J. Wang, H. Y. Zhang and X. H. Chen, Interval-Valued Hesitant Fuzzy Linguistic Sets and Their Application in Multi-Criteria Decision-Making Problems, Information Sciences, vol. 288, pp. 55-72, 2014.

[125] T. Y. Chen, A Prioritized Aggregation Operator-Based Approach to Multiple Criteria Decision Making Using Interval-Valued Intuitionistic Fuzzy Sets: A Comparative Perspective, Information Science, vol. 281, pp. 97-112, 2014.

[126] R. Verma and B. D. Sharma, Intuitionistic Fuzzy Einstein Prioritized Weighted Operators and Their Application to Multiple Attribute Group Decision Making, Applied Mathematics and Information Sciences, vol. 9, no. 6, pp. 3095-3107, 2015.

[127] C. Liang, S. Zhao and J. Zhang, Multi-Criteria Group Decision Making Method Based on Generalized Intuitionistic Trapezoidal Fuzzy Prioritized Aggregation Operators, Int. J. Mach. & Cyber, 2015.

[128] J. Dong, D. Y. Yang and S. P. Wan, Trapezoidal Intuitionistic Fuzzy Prioritized Aggregation Operators and Application to Multi-Attribute Decision Making, Iranian Journal of Fuzzy Systems, vol. 12, no. 4, pp. 1-32, 2015.

[129] R. Verma, Prioritized Information Fusion Method for Triangular Fuzzy Information and Its Application to Multiple Attribute Decision Making, International Journal of Uncertainty,

Fuzziness and Knowledge Based Systems, vol. 24, no. 2, pp. 265-289, 2016.

[130] J. M. Merigó, M. Casanovas and L. Martinez, Linguistic Aggregation Operators for Linguistic Decision Making Based on the Dempster–Shafer Theory of Evidence, International Journal Uncertainty, Fuzziness and Knowledge-based Systems, vol. 18, no. 3, pp. 287–304, 2011.

[131] J. H. Wang and J. Hao, A New Version of 2-Tuple Fuzzy Linguistic Representation Model for Computing with Words, IEEE Transaction on Fuzzy Systems, vol. 14, no. 3, pp. 435-445, 2006.

[132] F. Herrera, E. Herrera-Viedma and L. Martinez, A Fuzzy Linguistic Methodology to Deal with Unbalanced Linguistic Term Sets, IEEE Transaction on Fuzzy Systems, vol. 16, no. 2, pp. 354-370, 2008.

[133] G. U. Wei, Extension of TOPSIS Method for 2-Tuple Linguistic Multiple Attribute Group Decision Making with Incomplete Weight Information. Know. Inf. Syst., vol. 25, pp. 623-634, 2010.

[134] G. U. Wei, GRA-based Linear-Programming Methodology for Multiple Attribute Group Decision Making with 2-Tuple Linguistic Assessment Information. International Interdiscipline Journal, vol. 14, no. 4, pp. 1105-1110, 2011.

[135] G. W. Wei, Grey Relational Analysis Method for 2-Tuple Linguistic Multiple Attribute Group Decision Making with Incomplete Weight Information, Expert Systems with Applications, vol. 38, pp. 4824-4828, 2011.

[136] Y. Xu and H. Wang, Approaches based on 2-Tuple Linguistic Power Aggregation Operators for Multiple Attribute Group Decision Making Under Linguistic Environment, Applied Soft Computing, vol. 11, no. 5, pp. 3988-3997, 2011.

[137] G. H. Wei, FIOWHM Operator and Its Application to Multiple Attribute Group Decision Making, Expert Systems with Applications, vol. 38, no. 4, pp. 2984–2989, 2011.

[138] C. Li, S. Zeng, T. Pan and L. Zheng, A Method based on Induced Aggregation Operators and Distance Measures to Multiple Attribute Decision Making Under 2-Tuple Linguistic Environment, Journal of Computer and System Sciences, vol. 80, pp. 1339-1349, 2014.

[139] P. D. Liu and F. Jin, Methods for aggregating intuitionistic uncertain linguistic variables and their application to group decision making, Information Sciences, vol. 205, pp. 58-71, 2012.

Hidden-layer Ensemble Fusion of MLP Neural Networks for Pedestrian Detection

Kyaw Kyaw Htike
School of Information Technology, UCSI University, Kuala Lumpur, Malaysia
E-mail: ali.kyaw@gmail.com

> *Being able to detect pedestrians is a crucial task for intelligent agents especially for autonomous vehicles, robots navigating in cities, machine vision, automatic traffic control in smart cities, and public safety and security. Various sophisticated pedestrian detection systems have been presented in literature and most of the state-of-the-art systems have two main components: feature extraction and classification. Over the past decade, the majority of the attention has been paid to feature extraction. In this paper, we show that much can be gained by having a high-performing classification algorithm, and changing only the classification component of the detection pipeline while fixing the feature extraction mechanism constant, we show reduction in pedestrian detection error (in terms of log-average miss rate) by over 40%. To be specific, we propose a novel algorithm for generating a compact and efficient ensemble of Multi-layer Perceptron neural networks that is well-suited for pedestrian detection both in terms of detection accuracy and speed. We demonstrate the efficacy of our proposed method by comparing with several state-of-the-art pedestrian detection algorithms.*

> *Povzetek: Razvit je nov algoritem z nevronskimi mrežami za zaznavanje pešcev v prometu npr. za avtonomno vozilo.*

1 Introduction

Pedestrian detection is an important problem in Artificial Intelligence and computer vision. Many pedestrian detection systems have been proposed with different feature extraction and classification methods. One of the earliest general machine learning based pedestrian detection systems was proposed by Papageorgiou and Poggio [1]. They use Haar wavelets coefficients measuring (at multiple scales) differences in intensity levels of pixels as the feature representation, and a Support Vector Machine (SVM) with the quadratic kernel as the classifier.

A year later, this motivated a seminal work on object detection, namely the Viola-Jones face detector [2]. Viola and Jones use the idea of *integral images* to speed up the extraction of rectangular Haar basis functions to use as features which then serve as input to an adaboost classifier. The classifier, which is applied to an image in a sliding window fashion, is specifically designed to be an attentional cascade so that most of non-face examples can be rejected with relatively few feature extraction steps. This results in a fast face detector. Although Haar basis functions are well-suited for detecting frontal-faces, it was not very clear whether it is also good for detecting other categories of objects. Indeed, one of the critical cues human beings use to classify or detect objects is shape information (for which the bulding blocks are image gradients or edges); the set of Haar basis functions does not exploit this. In fact, after [2] came out, features that make effective use of image gradient information would soon be proposed in [3].

Leibe *et al.* [4] propose an *Implicit Shape Model* which learns, for each cluster of local image patches, a vote distribution on the centroid of the object class. At test time, Generalized Hough Transform [5] is used to detect objects; to be specific, each local image patch votes with the learnt distributions for the centroid in the hough voting space and then local maxima (or peaks) in the voting space correspond to object centroids in the test image. For each of these local maxima, local image patches that contributed (*i.e.* voted for) are identified through a process known as *backprojection*. From this, a rough segmentation (and consequently, bounding boxes) of the objects can be obtained. Their method requires segmentation of the training images which can be labor-intensive and costly, and identifying the local maxima and performing backprojection can be highly sensitive to many types of noise. Moreover, due to the use of image patches, the system is not robust to illumination and various image geometric transformations.

In 2005, a groundbreaking work on object detection was published by Dalal and Triggs [3] who propose Histograms of Oriented Gradients (HOG) as features for pedestrian detection. In HOG, a histogram of image gradients is constructed in each small local region (termed as a *cell*) in an image and these histograms are concatenated to form a high-dimensional feature vector. They carried out a large number of experiments to explore and evaluate the design space of the HOG feature extraction process. This includes highlighting the importance of *local contrast normaliza-*

tion whereby each histogram corresponding to a cell is redundantly normalized with respect to other nearby cells. It was shown that with HOG features, a linear SVM was sufficient to obtain state-of-the-art results and outperformed several techniques such as generalized haar wavelets [1], PCA-SIFT [6] and Shape Contexts [7].

Moreover, HOG is still highly influential to this day; the majority of current state-of-the-art research is based on either variations of HOG or building upon ideas outlined in HOG [8, 9, 10, 11]. For instance, the well-known part-based object detection system, Deformable Part Models (DPM) [12], use HOG features as building blocks. Many systems have extended DPMs (*e.g.* [13, 14, 15]). Schwartz *et al.* [16] use Partial Least Squares to reduce high dimensional feature vectors that combine edge, texture and color cues, and Quadratic Discriminant Analysis as the classifier.

Dollar *et al.* [17] propose Integral Channel Features (ICF) that can be considered as a generalization of HOG by including not only gradient when computing local histograms, but also other "channels" of information such as sums of grayscale and color pixel values, sums of outputs obtained by convolving the image with linear filters (such as Difference of Gaussian filters) and sums of outputs of nonlinear transformation of the image (*e.g.* gradient magnitude). An attempt was made in [18] to speed up features such as ICF by approximating some of the scales of the feature pyramid by interpolating from corresponding nearby scales during multi-scale sliding window object detection.

Benenson *et al.* [19] propose an alternative to speed up object detection by training a separate model for each of the nearby scales of the image pyramid. Although this increases the object detection speed at test time, it also considerably lengthens the training time.

Benenson *et al.* [20] extend HOG [3] by automatically selecting the HOG cell sizes and locations using boosting. Their approach is very expensive to train. However, their work shows that the plain HOG is powerful and with the right sizes and placements of HOG cells, with a single rigid component, it is possible to outperform various feature extraction methods researchers have proposed over the years after the invention of HOG.

For these reasons, in this work, we adopt HOG [3] as the feature extraction method. Furthermore, we focus on the classification component of the object detection pipeline. In order to isolate the performance of the classifier, we set the feature extraction mechanism constant for all the experiments of our proposed method. For the classification component of the pedestrian detection pipeline, we propose a powerful and efficient non-linear classifier ensemble that significantly increases the pedestrian detection performance (*i.e.* accuracy) while at the same time reducing the computational complexity at test time which is especially important for a task such as pedestrian detection. Although the proposed algorithm could also be potentially applied to other object detection tasks and general machine learning classification problems, we focus on pedestrian detection in this paper.

To the best of our knowledge, neural networks, or more accurately, Multi-layer Perceptrons (MLPs) have not been used for pedestrian detection. This observation is also true for ensembles of MLPs (EoMLPs): there have not been any major work using EoMLPs for object detection, let alone pedestrian detection. Although there can be many reasons behind this dearth of application of EoMLPs in pedestrian detection, one possible reason could be due to the highly computationally expensive nature of EoMLPs at test time and due to the popularity of linear Support Vector Machines (SVMs) and other regularized linear classifiers. We show in this paper that given the same feature extraction mechanism, using our proposed non-linear classifier gives much better pedestrian detection performance than the traditional classifiers commonly used for pedestrian detection.

Works such as [21], which apply EoMLPs to real-world problems, exist (although quite rare to find); however, they are for general pattern tasks rather than pedestrian detection or even object detection. Literature on EoMLPs have been published in the machine learning and Artificial Intelligence community; a few well-known ones are [22, 23, 24, 25]. They demonstrate the effectiveness of EoMLPs compared to individual MLPs on some standard statistical benchmarks; none of them have been applied to any object detection or even computer vision problems. Furthermore, although there are many different ensemble methods of numerous types of classifiers such as bagging [26], boosting [27], Random subspace [28], Random Forest [29] and Rulefit [30], the focus of this paper is on EoMLPs and pedestrian detection. Moreover, our proposed algorithm differs from [22, 23, 24, 25] in numerous ways including being suitable to be applied for pedestrian detection problems and our method outperforms state-of-the-art EoMLPs as will shown in the experimental results in Section 4.

We term our novel algorithm as *Hidden-layer Ensemble Fusion of Multi-layer Perceptrons* (HLEF-MLP).

2   Contribution

The contribution that we make in this paper is five-fold:

1. We propose a novel way, for the purpose of pedestrian detection, of training multiple individual MLPs in an ensemble and fusing the members of the ensemble at the hidden-feature level rather than at the output score level as done in existing EoMLPs [22, 23, 24, 25]. This has the benefit of being able to correct the mistakes of the ensemble in a major way since the final classifier still has access to hidden level features rather than only output scores or classification labels.

2. We use L1-regularization in a stacking fashion to jointly and efficiently select individual hidden feature units of all the members in the ensemble which has the effect of fusing the members of the ensemble. This results in only a few active projection units at test time,

which can be implemented as fast matrix projections on modern hardware for efficient pedestrian (or object) detection.

3. In HLEF-MLP, the decisions of the members are combined or fused in a systematic way using the given supervision labels such that the combination is optimal for the classification task. This is in contrast to ad-hoc class-agnostic nature of most fusion schemes (which turn to use techniques such as averaging).

4. We show that given the same feature extraction mechanism, HLEF-MLP gives much better pedestrian detection performance than the state-of-the-art classifiers commonly used for pedestrian detection.

5. HLEF-MLP is stable and robust to initialization conditions and local optima of individual member MLPs in the ensemble. Therefore, we do not need to be so careful about training each MLP for two main reasons: firstly, we are not relying solely on one MLP, and secondly, we are able to, during fusion, correct mistakes of the individual MLPs at the hidden features level as mentioned previously. In fact, each MLP falling in its own local optima should be seen as achieving diversity in the ensemble which is a desirable goal that can be obtained for "free". At the L1-regularized fusion step, the optimization function is convex, therefore it is guaranteed to achieve the global optimum.

3 Method

3.1 Overview

Let $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)\}$ be the labelled training dataset, where N is the number of training data points, $\mathbf{x}_i \in \mathbb{R}^k$ is the k-dimensional feature vector corresponding to the i-th training data point, $y_i \in \{1, 0\}$ is the supervision label associated with $\mathbf{x}_i$.

HLEF-MLP consists of two stages. $\mathcal{D}$ is split into $\mathcal{D} = \{\mathcal{D}^1, \mathcal{D}^2\}$, where

$$\mathcal{D}^1 = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{\frac{N}{2}}, y_{\frac{N}{2}})\}$$

and

$$\mathcal{D}^2 = \{(\mathbf{x}_{\frac{N}{2}+1}, y_{\frac{N}{2}+1}), \ldots, (\mathbf{x}_N, y_N)\}$$

In the first stage which we call "ensemble learning", we train each individual MLP on $\mathcal{D}^1$ and in the second stage termed "sparse fusion optimization", we use $\mathcal{D}^2$ to automatically learn how to fuse the decisions of the members of the ensemble. We now describe each stage.

3.2 Ensemble learning

The inputs to the first stage are $\mathcal{D}^1$ (obtained as described in Section 3.1) and the number of members M in the ensemble. The ensemble can be written as

$[f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})]$ where each member $f_j(\mathbf{x})$ of the ensemble can be formulated as:

$$f_j(\mathbf{x}) = \text{logsig}(\mathbf{w_j} \tanh(\mathbf{W}_j \mathbf{x} + \mathbf{b}_j) + b_j) \qquad (1)$$

where $\mathbf{x}$ is the input feature vector, and $\mathbf{W}$ and $\mathbf{b}$ are the matrix and vector respectively corresponding to an affine transformation of $\mathbf{x}$. Each column of $\mathbf{W}$ corresponds to the weights of the incoming connections to a neuron. Therefore the number of hidden neurons in $f_j(\mathbf{x})$ is equal of the number of columns in $\mathbf{W}_j$ and the number of rows of $\mathbf{W}_j$ is k (recall that $\mathbf{x} \in \mathbb{R}^k$). Therefore, $\mathbf{W} \in \mathbb{R}^{k \times h}$, where h is the number of neurons in the hidden layer.

The architecture of this first stage of the ensemble is illustrated in Figure 1. In the figure, each ensemble member (*i.e.* a MLP neural network) is shown as a vertical chain of blocks of mathematical operations for a total of M chains. The j-th ensemble (chain) corresponds to the function f_j as defined in Equation 1 and it can be seen that the block chain diagram follows exactly the sequence of operations defined in Equation 1.

For example, for the first ensemble member f_1, the input data vector $\mathbf{x}$ is first matrix-multiplied with $\mathbf{W}_1$ (first block), and the resulting vector is then added with the vector $\mathbf{b}_1$ in the second block. The function $\tanh(\cdot)$ is then applied (third block). This is followed by matrix multiplication of the output of the previous block with $\mathbf{w_1}$ in the fourth block. In the last block, a scalar addition with b_1 is performed.

The functions $\tanh(\cdot)$ and $\text{logsig}(\cdot)$ (visualized in Figure 2) are non-linear activation functions (applied after respective affine transformations) independently acting on each dimension of the vector obtained after the affine transformation and are defined as follows:

$$\tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}} \qquad (2)$$

$$\text{logsig}(a) = \frac{e^a}{1 + e^a} \qquad (3)$$

The *latent* parameters of the members of the first stage of the ensemble need to be *learnt* from the training data $\mathcal{D}^1$ (which is a set of pairs of input feature vectors and output supervision labels) and this training (*i.e.* learning) process is depicted in Figure 3.

We set all $\mathbf{W}_j$ to be the same size (*i.e.* each MLP $f_j(\mathbf{x})$ in the ensemble has the same number of hidden neurons). For each member MLP $f_j(\mathbf{x})$ in the ensemble, the following loss function can be constructed:

$$L(\mathcal{D}^1, \mathbf{W}_j, \mathbf{b}_j, \mathbf{w}_j, b_j) =$$
$$-\frac{2}{N} \sum_{i=1}^{\frac{N}{2}} y_i \, \text{logsig}(\mathbf{w_j} \tanh(\mathbf{W}_j \mathbf{x}_i + \mathbf{b}_j) + b_j) + \qquad (4)$$
$$(1 - y_i) \, \text{logsig}(\mathbf{w_j} \tanh(\mathbf{W}_j \mathbf{x}_i + \mathbf{b}_j) + b_j)$$

where $\{\mathbf{x}_i, y_i\}_{i=1}^{\frac{N}{2}}$ are pairs of input feature vectors and output supervision labels from $\mathcal{D}^1$.
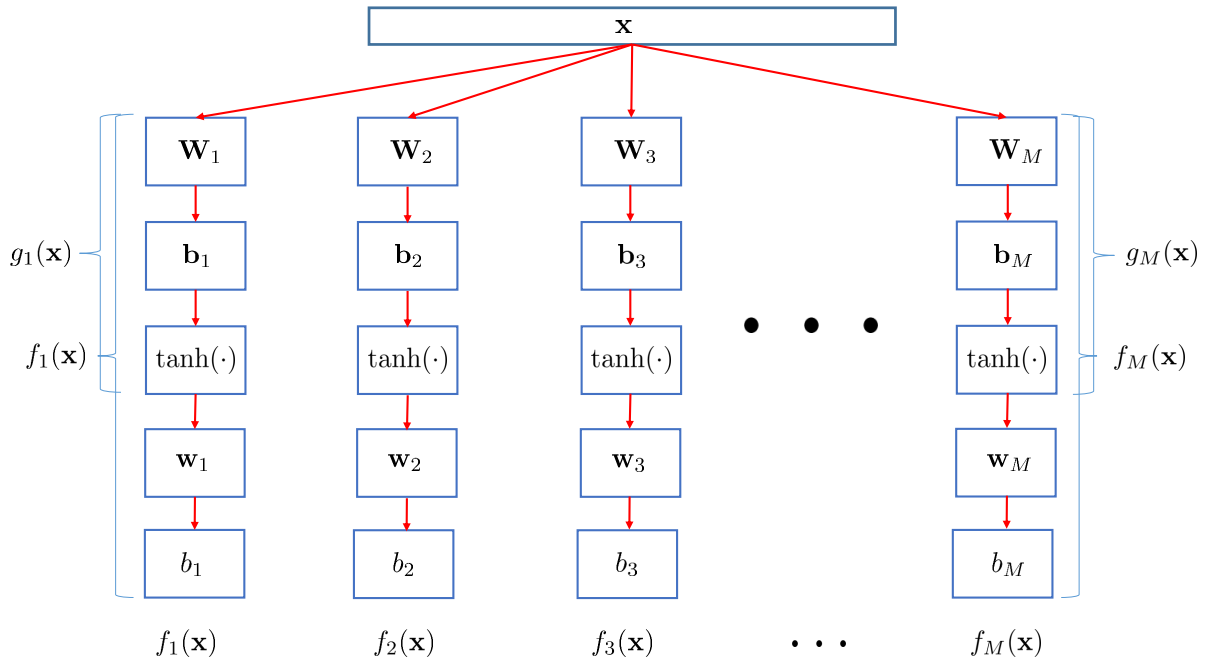
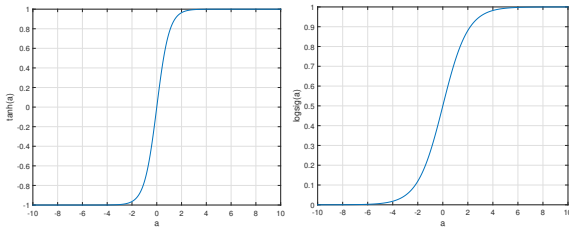Figure 1: Architecture of the first stage of the ensemble.

Figure 2: Non-linear activation functions; on the left is the curve for $\tanh(a)$ and on the right is $\text{logsig}(a)$, where a is the input signal to the activation function.

The loss function given in Equation 4 is a smooth and differential function and we optimize it using L-BFGS [31] since it is a type of efficient semi-newton method that does not require setting sensitive hyperparameters such as learning rate, momentum and mini-batch size. After the optimization, all the weights of each network:

$$\{\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_M, \mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_M, \mathbf{w}_1, \ldots, \\ \ldots, \mathbf{w}_M, b_1, b_2, \ldots, b_M\} \quad (5)$$

are obtained.

3.3 Sparse fusion optimization

In the second stage which is sparse fusion optimization, function $g_j(\mathbf{x})$ is used to project each data point $\mathbf{x}$ as given below:

$$g_j(\mathbf{x}) = \tanh(\mathbf{W}_j\mathbf{x} + \mathbf{b}_j) \quad (6)$$

Each data point $\mathbf{x} \in \mathcal{D}^2$ is projected using $\{g_j(\mathbf{x})\}_{j=1}^M$. That is, a new training dataset $\hat{\mathcal{D}}^2$ is constructed as follows:

$$\begin{bmatrix} g_1(\mathbf{x}_1) & g_2(\mathbf{x}_1) & \ldots & g_M(\mathbf{x}_1) \\ g_1(\mathbf{x}_2) & g_2(\mathbf{x}_2) & \ldots & g_M(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ g_1(\mathbf{x}_N) & g_2(\mathbf{x}_N) & \ldots & g_M(\mathbf{x}_N) \end{bmatrix} \quad (7)$$

where each row corresponds to one new data point in $\hat{\mathcal{D}}^2$. The architecture of this second stage of the ensemble is depicted in Figure 4.

It is important to note that the projection is done using the function $g(\cdot)$, and *not* $f(\cdot)$. In other words, this can be interpreted as projecting to the hidden layers of the individual MLP neural networks (which are members of the ensemble) and then learning to fuse in this hidden layer, hence the name of our proposed algorithm being *Hidden-layer Ensemble Fusion of Multi-layer Perceptrons* (HLEF-MLP).

This improves over the traditional (state-of-the-art) ensemble techniques in terms of both generating a much more compact ensemble (making it available to be applied to very time-critical applications such as pedestrian detection) and the final pedestrian detection performance (measured by log-average miss rate).

After constructing the projected dataset from $\hat{\mathcal{D}}^2$, a L1-regularized logistic regression is then trained on $\hat{\mathcal{D}}^2$ by optimizing:
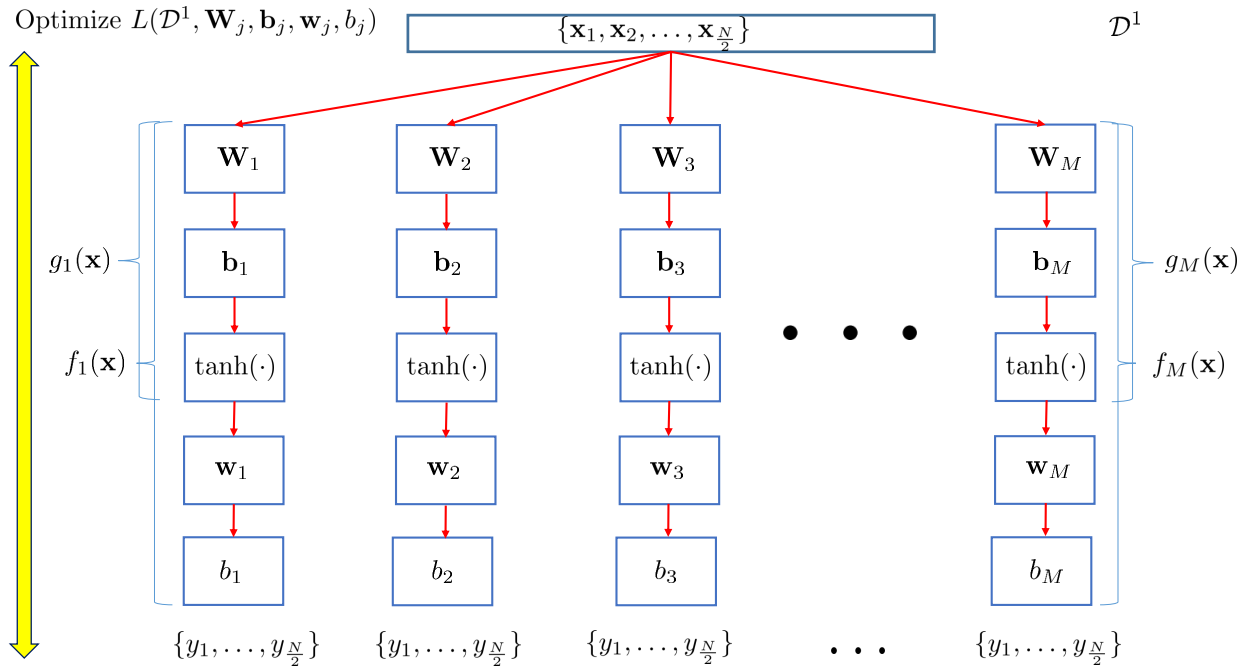
Optimize $L(\mathcal{D}^1, \mathbf{W}_j, \mathbf{b}_j, \mathbf{w}_j, b_j)$ $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{\frac{N}{2}}\}$ $\mathcal{D}^1$

Figure 3: Independent ensemble member learning process.

$$\mathbf{m}_{\text{trained}} =$$

$$\arg \min_{\mathbf{m}} \sum_{i=\frac{N}{2}+1}^{N} f_L(\mathbf{m}, [g_1(\mathbf{x}_i), g_2(\mathbf{x}_i), \ldots, \quad (8)$$

$$g_M(\mathbf{x}_i)], y_i) + \beta f_R(\mathbf{m})$$

where $\mathbf{m}_{\text{trained}} \in \mathbb{R}^k$ is the trained classifier and is in fact a vector of weights. Moreover, f_L is the *loss function*, f_R is the *regularization* to encourage $\mathbf{m}$ to take small values (hence in a way, favoring simpler models), and β balances the regularization term and loss term. A higher value of β would result in a smoother solution (and less fit to the training data $\hat{\mathcal{D}}^2$) whereas lower β would result lower training error. The loss function is given by:

$$f_L(\mathbf{m}, [g_1(\mathbf{x}_i), \ldots, g_M(\mathbf{x}_i)], y_i) =$$
$$\log(1 + \exp(-y_i \mathbf{m}^T [g_1(\mathbf{x}_i), \ldots, g_M(\mathbf{x}_i)])) \quad (9)$$

In order to encourage sparse solutions, we use L1-regularization for which the regularization term is given by:

$$f_R = [1, \ldots, 1]^T \mathbf{m} \quad (10)$$

This effectively sets many of the components of the weight vector $\mathbf{m}$ to zero, resulting in a very compact ensemble (speeding up the pedestrian detection), while at the same time, retaining or even improving the ensemble performance for pedestrian detection. Figure 5 illustrates the learning process for the sparse fusion of the members in the ensemble at the hidden layer.

3.4 Prediction at test time

As illustrated in Figure 6, at test time, given a test feature vector $\mathbf{x}$, the prediction score s_{pred} is obtained by:

$$s_{\text{pred}} = \frac{1}{1 + \exp(-(\mathbf{m}_{\text{trained}})^T [g_1(\mathbf{x}), \ldots, g_M(\mathbf{x})])} \quad (11)$$

where $\{g_1, g_2, \ldots, g_M\}$ are the hidden-layer-projection functions (parts of the members of the ensemble) whose latent parameters have been obtained by the training process described in Section 3.2 and $\mathbf{m}_{\text{trained}}$ is the set of latent sparse weights that have been obtained as explained in Section 3.3.

Equation 11 can also be equivalently written as:

$$s_{\text{pred}} \qquad = \frac{1}{1 + \exp(a)}$$
$$\text{where } a \quad = -(\mathbf{m}_{\text{trained}})^T [\tanh(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1), \ldots,$$
$$\tanh(\mathbf{W}_M \mathbf{x} + \mathbf{b}_M)] \quad (12)$$

Since $\mathbf{m}_{\text{trained}}$ is a sparse vector (*i.e.* a vector where the majority of the elements are zeros), at test time, entire columns of the matrices $\{\mathbf{W}_1, \ldots, \mathbf{W}_M\}$ corresponding to the positions of the zero vector elements in $\mathbf{m}_{\text{trained}}$ can be omitted. This greatly speeds up the pedestrian detection process, while improving the performance (log-average miss rate), due to the sparse fusion technique at the hidden layers as described in Sections 3.2 and 3.3.
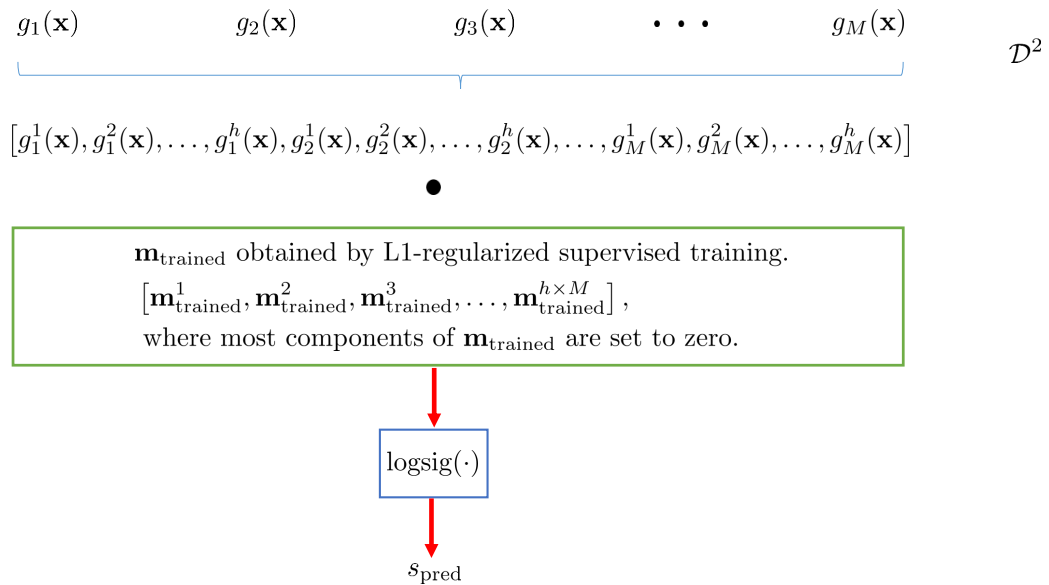
$$g_1(\mathbf{x}) \qquad g_2(\mathbf{x}) \qquad g_3(\mathbf{x}) \qquad \bullet\,\bullet\,\bullet \qquad g_M(\mathbf{x}) \qquad\qquad \mathcal{D}^2$$

$$\left[g_1^1(\mathbf{x}), g_1^2(\mathbf{x}), \ldots, g_1^h(\mathbf{x}), g_2^1(\mathbf{x}), g_2^2(\mathbf{x}), \ldots, g_2^h(\mathbf{x}), \ldots, g_M^1(\mathbf{x}), g_M^2(\mathbf{x}), \ldots, g_M^h(\mathbf{x})\right]$$

$\bullet$

$\mathbf{m}_{\text{trained}}$ obtained by L1-regularized supervised training.
$$\left[\mathbf{m}_{\text{trained}}^1, \mathbf{m}_{\text{trained}}^2, \mathbf{m}_{\text{trained}}^3, \ldots, \mathbf{m}_{\text{trained}}^{h \times M}\right],$$
where most components of $\mathbf{m}_{\text{trained}}$ are set to zero.

$\text{logsig}(\cdot)$

s_{pred}

Figure 4: Architecture of the second stage of the ensemble (fusion).

4 Results and discussion

4.1 Dataset

We use the INRIA Person dataset [3] for evaluating our algorithms. In the dataset, for training, there are 614 images containing pedestrians for which ground truth is available; after cropping and flipping each pedestrian, the total number of pedestrians come up to be 2474. There are also 1218 lage images that do not contain any pedestrians; from these, data corresponding to non-pedestrian class can be generated by a combination of initial sampling and hard negative mining as is common in object detection literature [32, 9].

4.2 Ensemble hyper-parameters

We set the size of the ensemble M (see Section 3.2) to 100 and the number of hidden neurons h in each hidden layer to 10 for all the experiments involving both our proposed method and baselines on various ensemble variations.

4.3 Experiment setup

We perform seven different experiments in order to compare our proposed algorithm with state-of-the-art pedestrian detection systems. The experiments are described below.

4.3.1 `VJ`

The Viola-Jones detector of [2] applied to pedestrian detection.

4.3.2 `HOG`

Pedestrian detector of the seminal work of Dalal and Triggs [3], using Histogram of Oriented Gradients (HOG) for feature extraction and linear SVM as the classifier.

4.3.3 `HikSvm`

The system proposed by Maji *et al.* [33] who use a variation of HOG (concatenation of histograms of oriented gradients for a few different cell sizes) and SVM with an approximation of the intersection kernel.

4.3.4 `Pls`

The pedestrian detection system proposed by Schwartz *et al.* [16] using Partial Least Squares (PLS) analysis to reduce the dimensions of very high dimensional features obtained by concatenating different sets of features obtained from edge, texture and color information derived from the original image. Then Quadratic Discriminant Analysis is used as the classifier.

4.3.5 `OursMLPEnsHidFuse`

Our proposed approach in this paper as described in Section 3. At test time, due to L1-regularized fusion at the hidden layer level, we can expect that the resulting ensemble will be very small, which will be suitable for time-critical applications such as pedestrian detection.
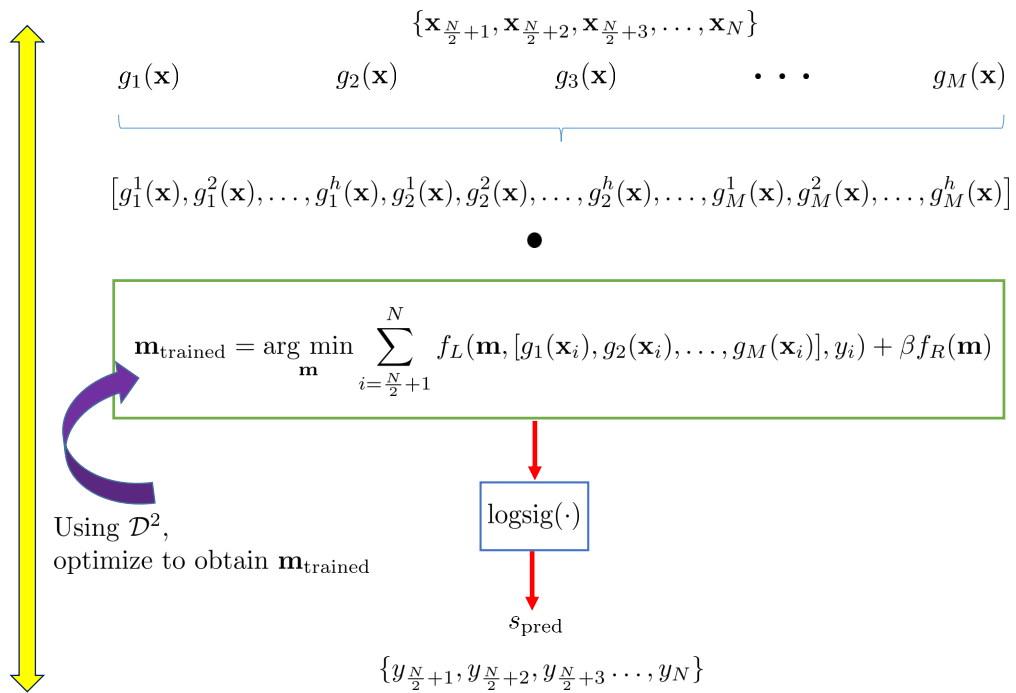
$$\{\mathbf{x}_{\frac{N}{2}+1}, \mathbf{x}_{\frac{N}{2}+2}, \mathbf{x}_{\frac{N}{2}+3}, \dots, \mathbf{x}_N\}$$

$$g_1(\mathbf{x}) \qquad g_2(\mathbf{x}) \qquad g_3(\mathbf{x}) \qquad \cdots \qquad g_M(\mathbf{x})$$

$$\left[g_1^1(\mathbf{x}), g_1^2(\mathbf{x}), \dots, g_1^h(\mathbf{x}), g_2^1(\mathbf{x}), g_2^2(\mathbf{x}), \dots, g_2^h(\mathbf{x}), \dots, g_M^1(\mathbf{x}), g_M^2(\mathbf{x}), \dots, g_M^h(\mathbf{x})\right]$$

$$\mathbf{m}_{\text{trained}} = \arg\min_{\mathbf{m}} \sum_{i=\frac{N}{2}+1}^{N} f_L(\mathbf{m}, [g_1(\mathbf{x}_i), g_2(\mathbf{x}_i), \dots, g_M(\mathbf{x}_i)], y_i) + \beta f_R(\mathbf{m})$$

Using $\mathcal{D}^2$,
optimize to obtain $\mathbf{m}_{\text{trained}}$

$$\text{logsig}(\cdot)$$

$$s_{\text{pred}}$$

$$\{y_{\frac{N}{2}+1}, y_{\frac{N}{2}+2}, y_{\frac{N}{2}+3} \cdots, y_N\}$$

Figure 5: Learning to fuse the members in the ensemble.

$$\mathbf{x}$$

$$\mathbf{W}_1 \qquad \mathbf{W}_2 \qquad \mathbf{W}_3 \qquad \mathbf{W}_M$$

$$\mathbf{b}_1 \qquad \mathbf{b}_2 \qquad \mathbf{b}_3 \qquad \mathbf{b}_M$$

$$g_1(\mathbf{x}) \quad \tanh(\cdot) \qquad \tanh(\cdot) \qquad \tanh(\cdot) \qquad \tanh(\cdot) \quad g_M(\mathbf{x})$$

$$\left[g_1^1(\mathbf{x}), g_1^2(\mathbf{x}), \dots, g_1^h(\mathbf{x}), g_2^1(\mathbf{x}), g_2^2(\mathbf{x}), \dots, g_2^h(\mathbf{x}), \dots, g_M^1(\mathbf{x}), g_M^2(\mathbf{x}), \dots, g_M^h(\mathbf{x})\right]$$

$$\left[\mathbf{m}_{\text{trained}}^1, \mathbf{m}_{\text{trained}}^2, \mathbf{m}_{\text{trained}}^3, \mathbf{m}_{\text{trained}}^4, \mathbf{m}_{\text{trained}}^5, \dots, \mathbf{m}_{\text{trained}}^{h \times M}\right]$$

$$\text{logsig}(\cdot)$$

$$s_{\text{pred}}$$
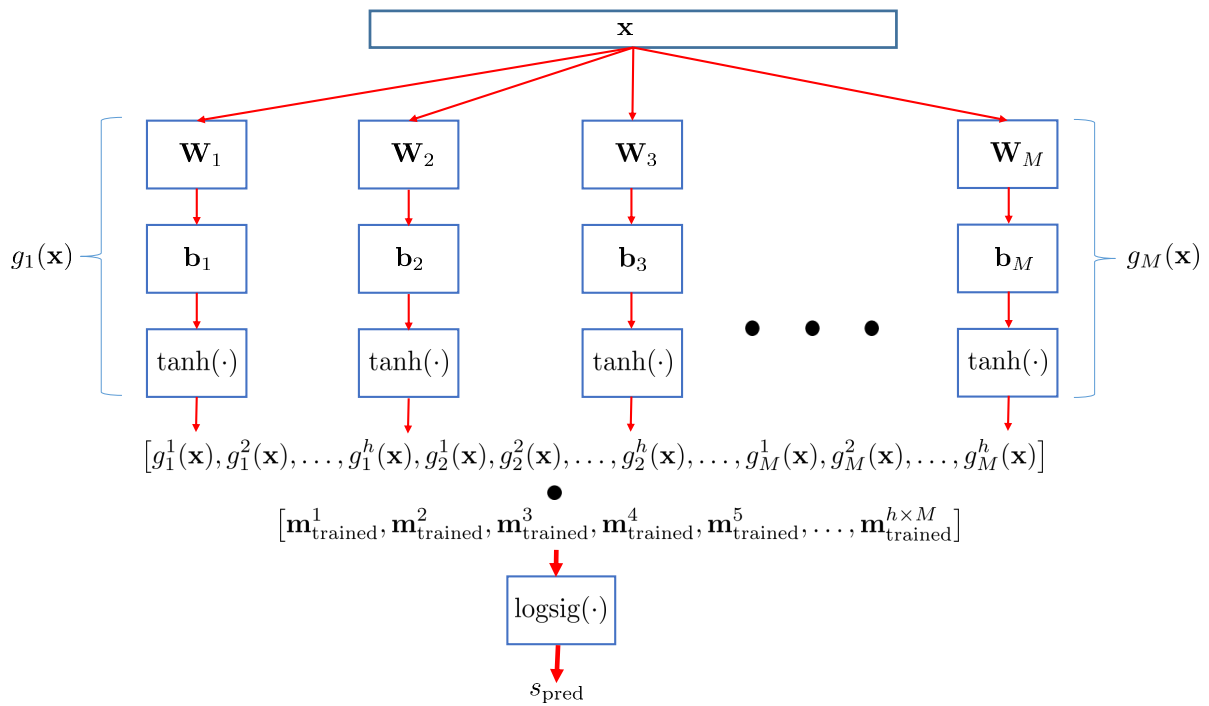
Figure 6: Using the fused ensemble at test time.

4.3.6 `OursMLPEnsScoreFuse`

A variation of our proposed method `OursMLPEnsHidFuse`; instead of fusing at the level of hidden features, L1-regularized fusion takes place at the score level. This is also somewhat similar to classifier stacking [34] in literature; however most stacking ensembles in literature is using simple unregularized classifiers whereas `OursMLPEnsScoreFuse` is able to select the most efficient members of the ensemble jointly using supervised label information.

In terms of efficiency at test time, we can anticipate it to be worse than `OursMLPEnsHidFuse` because the supervised L1-regularized selection is being performed at the score level and also at test time, there is a need to multiple matrix projections (one for each member of the ensemble) and then combine them.

4.3.7 `OursMLPEns`

This is our implementation of the state-of-the-art MLP fusion by bagging. Here unlike `OursMLPEnsHidFuse` and `OursMLPEns`, there are no separation of the training dataset $\mathcal{D}$ into $\mathcal{D}^1$ and $\mathcal{D}^2$; independent training of the members of the ensemble takes place on the entire training data $\mathcal{D}$. Moreover, there is no sparse fusion optimization.

We can expect that compared to `OursMLPEns-HidFuse` and `OursMLPEnsScoreFuse`, this will have the worst efficiency at test time because the entire ensemble needs to be evaluated which is highly expensive.

4.4   Evaluation criteria

To evaluate the pedestrian detections from the aforementioned experiments, firstly there needs to be a measure on what a correct pedestrian detection is. Although this can be defined in many ways, we use the PASCAL 50% overlap criterion [35] which is the most widely-used one in state-of-the-art detection literature. Let a detected bounding box be $\mathbf{r}_d$ and the corresponding groundtruth bounding box be $\mathbf{r}_g$. In PASCAL 50% overlap criterion, in order to establish whether $\mathbf{r}_d$ is a correct detection, the overlap ratio, α_o, defined in Equation 13 is computed and then the detection $\mathbf{r}_d$ is deemed to be correct if $\alpha_o > 0.5$.

$$\alpha_o = \frac{\text{area}(\mathbf{r}_g \cap \mathbf{r}_d)}{\text{area}(\mathbf{r}_g \cup \mathbf{r}_d)} \tag{13}$$

The α_o can be understood as the ratio of the intersection of the detected bounding box with ground truth bounding box and the union of the detected box with ground truth box.

Graphs are an important tool to visualize the performance of pedestrian detectors since they show the performance at various levels of detection thresholds. To this end, we plot curves where miss rate is on the y-axis and false positives per image (FPPI) is on the x-axis.

This has been empirically proven in state-of-the-art pedestrian detection (*e.g.* by Benenson *et al.* [9]) to be
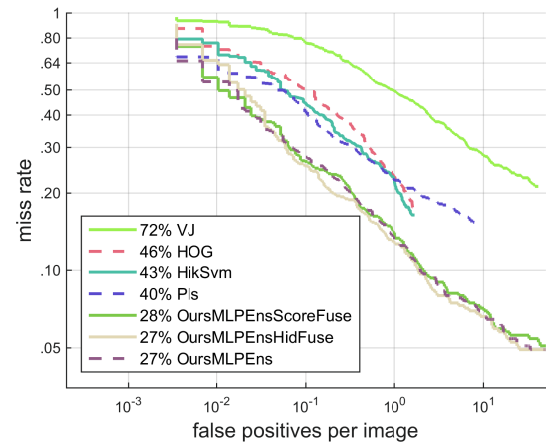
Figure 7: ROC curves for all experiments)
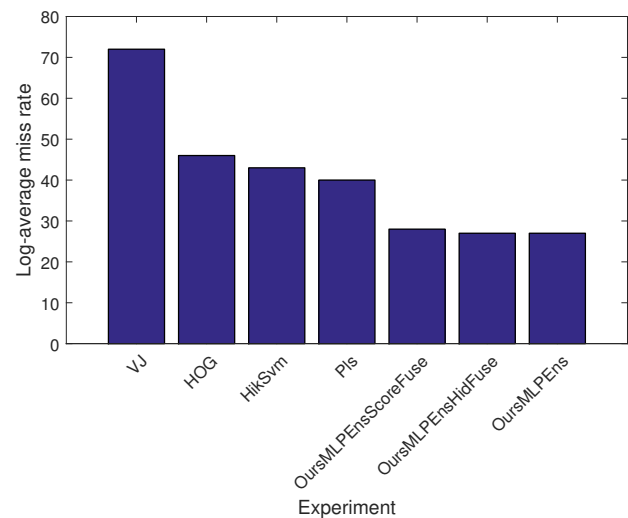
Figure 8: Comparison of log-average miss rate (lower is better).

a more accurate way of measuring the detection performance than the previously used false positives per window (FPPW).

In a miss rate versus FPPI plot, lower curves denote higher detection performance. Moreover, to summarize the performance of a detector with a single value, log-average miss rate is calculated which is approximately equal to the area under the miss rate versus FPPI curve.

4.5   Results

The miss rate versus FPPI plots for all the experiments are shown in Figure 7. In the figure, the curves are ordered in terms of log-average miss rate (LAMR) from worst to best. To better focus on the summarized performance, we also plot the LAMRs of each experiment in terms of a bar graph as illustrated in Figure 8. Finally, we depict in Figure 9 the comparison of average detection time taken by different algorithms for a 640×480 image. We also show in Table 1
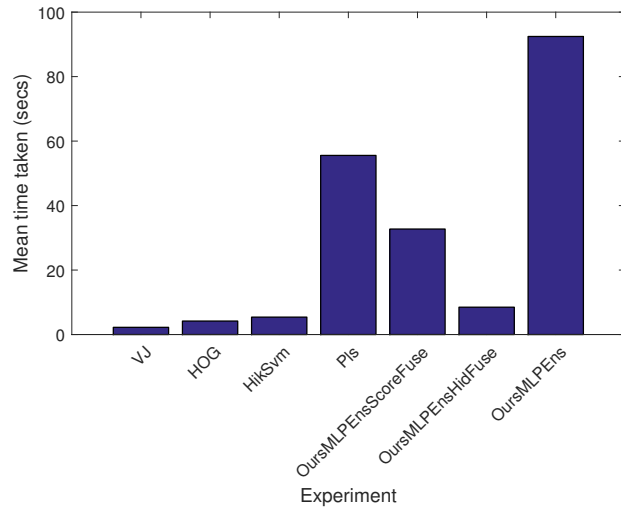
Figure 9: Comparison of average detection time for a 640×480 image.

Experiment	Mean time (secs)
VJ	2.23
HOG	4.18
HikSvm	5.41
Pls	55.56
OursMLPEnsScoreFuse	32.72
OursMLPEnsHidFuse	8.49
OursMLPEns	92.45

Table 1: Comparison in terms of detection time.

the raw detection time values for easier comparison.

From Figures 7, 8 and 9, the following observations can be made:

- VJ has the worst performance among all. This is to be expected since the Haar-features that it uses does not capture the object shape information well. Moreover, the seminal work HOG greatly improves over VJ for generic object detection.

- Although HikSvm requires complex modification in the feature extraction and the classifier compared to HOG, it only results in a modest detection performance gain (*i.e.* decrease in LAMR from 46% to 43%). Similar observation can be made for Pls although the improvement is relatively better.

- Our proposed algorithm OursMLPEnsHidFuse has the best performance among them, tied in the first place with OursMLPEns. Given that OursMLP-EnsHidFuse is using the standard HOG features, we can see that just by using our proposed algorithm, the LAMR goes down from 46% and 27%. This is a significant improvement (corresponding to sover 40% reduction in LAMR). In comparison, HikSvm, despite proposing both new feature extraction and clas-

sification algorithms, only managed less than 0.1% reduction in LAMR.

- OursMLPEnsHidFuse has slightly higher detection performance than OursMLPEnsScoreFuse whereas in terms of speed at test time, OursMLP-EnsHidFuse is almost 4 times faster. This shows the efficacy of our proposed algorithm OursMLP-EnsHidFuse and also proves that fusion at the level of the hidden layers not only results in better detection performance and also very compact matrix projections (and hence much faster speed) due to the L1-regularized sparse fusion learning process having access to the hidden layer features.

- Despite OursMLPEnsHidFuse and OursMLPEns being tied at the first place in terms of detection performance, OursMLPEnsHidFuse is significantly (over 10 times) faster than OursMLPEns. This shows that our proposed OursMLPEnsHidFuse has the same detection accuracy as the very expensive state-of-the-art MLP bagging which is not practical for detection purposes. In other words, the novel method that we have presented in this paper OursMLPEns-HidFuse gives the same (top) performance as a MLP bagging-ensemble but with 10 times faster speed for pedestrian detection.

- OursMLPEnsScoreFuse is faster than OursMLPEns but lower in LAMR than OursMLP-EnsHidFuse. This again shows that our proposed OursMLPEnsHidFuse not only results in a much smaller ensemble model (and consequently, faster detection speed), but is also more effective in bringing out the effectiveness of the ensemble compared to OursMLPEnsScoreFuse.

- OursMLPEnsHidFuse has better detection accuracy than PLS while being significantly faster using only standard HOG features. It shows that we have not saturated the performance of HOG features yet. There is still a lot of improvement that can made in the classification algorithm for pedestrian detection systems.

- Our experiments show for the first time that using the proposed algorithm OursMLPEnsHidFuse, it is possible to apply ensemble techniques for efficient object detection purposes.

4.5.1 Analysis of trained ensemble model complexity

Although the results and the discussion about detection performance and speeds in the previous section should be sufficient, for completeness, we theorize and analyze the model complexities and sizes for the three ensemble methods described in the previous section: OursMLPEns, OursMLPEnsScoreFuse and OursMLPEnsHidFuse. It is to be noted that as mentioned previously, there are $M = 100$ MLPs in the ensemble.

For `OursMLPEns`, given a test feature vector **x**, there are 100 different matrix projections with each matrix having k rows and h columns (which is equal to the number of dimensions of the feature vector and the number of hidden neurons in each MLP respectively). After each projection, tansig($\cdot$) function must be applied, followed by a dot product between the resulting vector and a linear weight vector, which will produce a single score (for each member of the ensemble). Then the logsig($\cdot$) function needs to be performed on this score. Then 100 such scores will be averaged to give the final ensemble score. Therefore the ensemble complexity is quite high, especially for a highly computationally expensive task such as sliding window pedestrian detection.

For `OursMLPEnsScoreFuse`, after training the L1-regularized linear classifier on scores at the end of the model fusion step, 33 networks are discovered to be nonzero. This means that theoretically, it should be $\frac{100}{33} \approx 3$ times faster than `OursMLPEns`. In fact, this theoretical observation tallies with the experimental results presented in Table 1.

With regards to `OursMLPEnsHidFuse`, it was found that performing L1-regularized classifier optimization on hidden features resulted in highly sparse weight vector $\mathbf{m}_{\text{trained}}$ with the number of nonzero weight components equal to a mere 159. Therefore, at test time, there is only a need to perform a *single* matrix projection with a matrix having k rows and 159 columns. After that, $tansig$ function has to be applied followed by a dot product with a linear weight vector and finally a logsig($\cdot$) function. This means that the effective model complexity is much lower than either `OursMLPEns` or `OursMLPEnsScoreFuse`. This again matches with the empirical evidence.

5 Conclusion and future work

In this paper, we propose a novel algorithm to train a compact ensemble of Multi-layer Perceptron neural networks for pedestrian detection. The proposed algorithm integrates the members of the ensemble at the hidden feature layer level, resulting in a very small ensemble size (allowing for fast pedestrian detection) while at the same time outperforming existing neural network ensembling techniques and other pedestrian detection systems. We obtain very encouraging state-of-the-art results, and show for the first time that, using our proposed algorithm, it is indeed possible to apply neural network ensemble techniques for the task for pedestrian detection, something that was previously thought to be too inefficient due to the very large model size of ensembles.

There are several interesting directions of research based upon the work in this paper; firstly, there is an opportunity to apply the method presented in this paper to other object detection tasks and applications such as face and vehicle detection, and other general pattern recognition problems. Secondly, it will be highly beneficial to explore the effect of different feature extraction mechanisms on the resulting ensembles in terms of both detection performance and efficiency.

References

[1] Constantine Papageorgiou and Tomaso Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, 2000.

[2] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1–511. IEEE, 2001.

[3] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Conference on Computer Vision and Pattern Recognition*, pages 886–893. IEEE, 2005.

[4] Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an Implicit Shape Model. In *ECCV Workshop on Statistical Learning in Computer Vision*, volume 2, pages 7–14, 2004.

[5] Dana H Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern recognition*, 13(2):111–122, 1981.

[6] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630, 2005.

[7] Serge Belongie, Jitendra Malik, and Jan Puzicha. Matching shapes. In *International Conference on Computer Vision*, volume 1, pages 454–461. IEEE, 2001.

[8] Kyaw Kyaw Htike and David Hogg. Adapting pedestrian detectors to new domains: a comprehensive review. *Engineering Applications of Artificial Intelligence*, 50:142–158, April 2016.

[9] Rodrigo Benenson, Mohamed Omran, Jan Hosang, and Bernt Schiele. *Ten Years of Pedestrian Detection, What Have We Learned?*, pages 613–627. Springer International Publishing, Cham, 2015.

[10] Kyaw Kyaw Htike and David Hogg. *Unsupervised Detector Adaptation by Joint Dataset Feature Learning*, pages 270–277. Springer International Publishing, Cham, 2014.

[11] Kyaw Kyaw Htike and David Hogg. Weakly supervised pedestrian detector training by unsupervised prior learning and cue fusion in videos. In *International Conference on Image Processing*, pages 2338–2342. IEEE, October 2014.

[12] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, June 2008.

[13] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[14] Pedro F Felzenszwalb, Ross B Girshick, and David McAllester. Cascade object detection with deformable part models. In *Conference on Computer Vision and Pattern Recognition*, pages 2241–2248. IEEE, 2010.

[15] Ross B Girshick, Pedro F Felzenszwalb, and David A Mcallester. Object detection with grammar models. In *Advances in Neural Information Processing Systems*, pages 442–450, 2011.

[16] William Robson Schwartz, Aniruddha Kembhavi, David Harwood, and Larry S Davis. Human detection using Partial Least Squares analysis. In *International Conference on Computer Vision*, pages 24–31. IEEE, 2009.

[17] Piotr Dollar, Zhuowen Tu, Pietro Perona, and Serge Belongie. Integral channel features. In *British Machine Vision Conference*, pages 91.1–91.11. BMVA Press, 2009.

[18] Piotr Dollár, Serge Belongie, and Pietro Perona. The fastest pedestrian detector in the West. In *British Machine Vision Conference*, volume 2, page 7, 2010.

[19] Rodrigo Benenson, Markus Mathias, Radu Timofte, and Luc Van Gool. Pedestrian detection at 100 frames per second. In *Conference on Computer Vision and Pattern Recognition*, pages 2903–2910. IEEE, 2012.

[20] Rodrigo Benenson, Markus Mathias, Tinne Tuytelaars, and Luc Gool. Seeking the strongest rigid detector. In *Conference on Computer Vision and Pattern Recognition*, pages 3666–3673. IEEE, 2013.

[21] E Filippi, M Costa, and E Pasero. Multi-layer perceptron ensembles for increased performance and fault-tolerance in pattern recognition tasks. In *International Conference on Neural Networks: IEEE World Congress on Computational Intelligence*, volume 5, pages 2901–2906. IEEE, 1994.

[22] Pablo M Granitto, Pablo F Verdes, and H Alejandro Ceccatto. Neural network ensembles: evaluation of aggregation algorithms. *Artificial Intelligence*, 163(2):139–162, 2005.

[23] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (10):993–1001, 1990.

[24] Anders Krogh, Jesper Vedelsby, et al. Neural network ensembles, cross validation, and active learning. *Advances in Neural Information Processing Systems*, 7:231–238, 1995.

[25] Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. Ensembling neural networks: many could be better than all. *Artificial intelligence*, 137(1):239–263, 2002.

[26] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[27] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of Boosting. *Annals of Statistics*, 28(2):337–407, 1998.

[28] Tin Kam Ho. The random subspace method for constructing decision forests. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(8):832–844, 1998.

[29] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[30] Jerome H Friedman and Bogdan E Popescu. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, pages 916–954, 2008.

[31] Dong C. Liu, Jorge Nocedal, and Dong C. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

[32] Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):743–761, 2012.

[33] Subhransu Maji, Alexander C Berg, and Jitendra Malik. Classification using intersection kernel support vector machines is efficient. In *Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[34] Saso Džeroski and Bernard Ženko. Is combining classifiers with stacking better than selecting the best one? *Machine learning*, 54(3):255–273, 2004.

[35] Mark Everingham, Luc J. Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

Weighted Majority Voting Based Ensemble of Classifiers Using Different Machine Learning Techniques for Classification of EEG Signal to Detect Epileptic Seizure

Sandeep Kumar Satapathy
Department of Computer Science and Engineering
Siksha 'O' Anusandhan University, Khandagiri, Bhubaneswar-751030, Odisha, India
E-mail: sandeepkumar04@gmail.com

Alok Kumar Jagadev
School of Computer Engineering
KIIT University, Bhubaneswar-751024, Odisha, India
E-mail: alok.jagadev@gmail.com

Satchidananda Dehuri
Department of Information and Communication Technology
Fakir Mohan University, Vyasa Vihar-756019, Balasore, Odisha, India
E-mail: satchi.lapa@gmail.com

Electroencephalogram (EEG) signal is a miniature amount of electrical flow in a human brain that holds and controls the entire body. It is very difficult to understand these non-linear and non-stationary electrical flows through naked eye in the time domain. In specific, epilepsy seizures occur irregularly and un-predictively while recording EEG signal. Therefore, it demands a semi-automatic tool in the framework of machine learning to understand these signals in general and to predict epilepsy seizure in specific. With this motivation, for wide and in-depth understanding of the EEG signal to detect epileptic seizure, this paper focus on the study of EEG signal through machine learning approaches. Neural networks and support vector machines (SVM) basically two fundamental components of machine learning techniques are the primary focus of this paper for classification of EEG signals to label epilepsy patients. The neural networks like multi-layer perceptron, probabilistic neural network, radial basis function neural networks, and recurrent neural networks are taken into consideration for empirical analysis on EEG signal to detect epilepsy seizure. Furthermore, for multi-layer neural networks different propagation training algorithms have been studied such as back-propagation, resilient-propagation, and quick-propagation. For SVM, several kernel methods were studied such as linear, polynomial, and RBF during empirical analysis. Finally, the study confirms with the present setting that, in all cases recurrent neural network performs poorly for the prepared epilepsy data. However, SVM and probabilistic neural networks are quite effective and competitive. Hence to strengthen the poorly performing classifier, this work makes an extension over individual learners by ensembling classifier models based on weighted majority voting.

Povzetek: Sistem s pomočjo strojnega učenja iz EEG signalov zazna epileptični napad.

1 Introduction

Epilepsy is a persistent disorder of mental ability that has an abnormal EEG signal flow [1], which manifests in the disoriented human behaviour. In this world, around 40 to 50 million people are mostly affected by this disease [2]. Many people also call it as *fits* that causes loss of memory and interruption in consciousness, strange sensations, and significant alteration in emotions and behaviour. Research related to the epilepsy disease is basically used for differentiating between Ictal (seizure period) and Interictal (period between seizures) EEG signals.

Hence, the transition from preictal to ictal state for an epileptic seizure contains a gradual change from a chaotic to ordered wave forms. Moreover, the amplitude of the spikes does not necessarily signify the harshness of seizures [2].

The difference between the seizure and the common artifact is quite easy to recognize where generally the seizures within EEG measurement [3] have a prominent spiky, repetitive, transient, or noise-like pattern. Hence, unlike other general signals, for an untrained observer, the EEG signal is quite difficult for

understanding and analysis. The recording of these signals is mostly done by the help of a set of electrodes placed on the scalp using 10 to 20 electrode placement systems. The system incorporates the electrodes, which are placed with a specific name based on specific parts of the brain, e.g., Frontal Lobe (F), Temporal Lobe (T), etc. These naming and placement schemes have been discussed in more details in [3].

For the facilitation and effective diagnosis of the epilepsy, several neuro-imaging techniques such as functional magnetic resonance imaging (fMRI) and position emission tomography (PET) are used. An epileptic seizure can be characterized by paroxysmal occurrence of synchronous oscillation. This impersonation can be separated into two categories depending on the extent of involvement in different brain regions such as focal or partial and generalized seizures [4]. Focal seizure also known as *epileptic foci* are generated at specific sphere in the brain. In contrast to this, generalized seizures occur in most parts of the brain.

A careful analysis and diagnosis of EEG signals for detecting epileptic seizure in the human brain usually contribute to a substantial insight and support to the medical science. Thus, EEG is quite a beneficial as well as a cost effective way for the study of epilepsy disease. For generalized seizure, the duration of seizure can be easily detected by naked eyes whereas it is very difficult to recognize intervals during focal epilepsy.

Classification is the most useful and functional technique [5] for properly detecting the epileptic seizures in EEG signals. Classification being a data mining technique is generally used for pattern recognition [6]. Other than that, it is used to predict a group membership for unknown data instances. Hence, by designing a classifier model using different machine learning approaches, we can identify epileptic seizures in EEG brain signal. Pre-processing is considered as one of the necessary task to get it into a proper feature set format, even before considering the classification of raw EEG signal. Generally, the data sample of EEG is not linearly separable. Thus, to obtain non-linear discriminating function for classification, we are using machine learning techniques. Moreover, the case of limiting our focus on machine learning approaches is because of their capability and efficiency for smooth approximation and pattern recognition. However, there are learners who are performing very poorly; hence this work makes an extension over individual learners by ensembling classifier models based on weighted majority voting.

In this analytical study, a publicly available EEG dataset have been considered that is related to epilepsy for all experimental evaluations. Based on this there are mainly two phases of the epileptic seizure detection process that are carried out. The first phase is to analyse the EEG signal and convert it into a set of samples with set of features. The second phase is to classify the already processed data into different classes such as epilepsy or normal.

The rest of the subdivisions of this paper are organized as follows. Section 2 describes the recording and pre-processing of EEG signals through discrete wavelet transform. Some classification methods based on machine learning techniques are described in Section 3. Section 4 discusses the ensemble of classifiers. In Section 5, the detail of empirical work and analysis of results obtained by different machine learning models and ensemble of classifiers. Section 6 draws the conclusions and suggests possibilities for future work.

2   Methods for dataset preparation

In the present work we have collected data from [7] which is a publicly available database [8] related to diagnosis of epilepsy. This resource provides five sets of EEG signals. Each set contains reading of 100 single channel EEG segments of 23.6 seconds duration each. These five sets are described as follows. Datasets A and B are considered from five healthy subjects using a standardized electrode placement system. Set A contains signals from subjects in a slowed down state with eyes open. Set B also contains signal same as A but ones with the eyes closed. The data sets C, D and E are recorded from epileptic subjects through intracranial electrodes for interictal and ictal epileptic activities. Set D contains segments recorded from within the epileptogenic zone during seizure free interval. Set C also contains segments recorded during a seizure free interval from the hippocampal formation of the opposite hemisphere of the brain. The set E only contains segments that are recorded during seizure activity. All signals are recorded through the 128 channel amplifier system. Each set contains 100 single channel EEG data. In all there are 500 different single channel EEG data. In Subsection 2.1, we illustrate how to crack these signals using discrete wavelet transform [9] and prepare several statistical features to form a proper sample feature dataset.

2.1   Wavelet transform

This is a modern signal analysis technique which overcomes the limitations of other transformation techniques. Other transformation methods may include Fast Fourier Transform (FFT), Short Time Fourier Transform (STFT), etc. The major restrictions of these techniques are the analysis limits to stationary signals. These are not effective for analysis of transient signals such as EEG signal. Transient in the sense the frequency is changing rapidly with respect to time. Then, with the help of wavelet coefficients [10] we can analyse transient signals easily and also efficiently. Wavelet transform can be of two types: Continuous Wavelet Transform (CWT) [11] and Discrete Wavelet Transform (DWT) [12, 13].

2.1.1   Continuous wavelet transform

It is defined as:

$$CWT(a,b) = \int_{-\infty}^{\infty} x(t).\varphi_{a,b}^*(t)dt, \qquad (1)$$

where, $x(t)$ represents the original signal, a and b represents the scaling factor and translation along the time axis, respectively. The * symbol denotes the complex conjugation and $\varphi_{a,b}^*$ is computed by scaling the wavelet at time band scale a.

$$\varphi_{a,b}^*(t) = \frac{1}{\sqrt{|a|}} \varphi\left(\frac{t-b}{a}\right), \qquad (2)$$

where, $\varphi_{a,b}^*(t)$ stands for the mother wavelet. In CWT it is presumed that the scaling and translation parameters a and b changes continuously. But the main disadvantage of CWT is the calculation of wavelet coefficients for every possible scale can result in a large amount of data. It can surmount with the help of DWT.

2.1.2   Discrete wavelet transform

It is almost same as CWT except that the value of a and b does not change continuously. It can be defined as:

$$DWT = \frac{1}{\sqrt{|2^p|}} \int_{-\infty}^{\infty} x(t)\varphi\left(\frac{t-2^p q}{2^p}\right) dt, \qquad (3)$$

where a and b of CWT are replaced in DWT by $2^p \& 2^q$ respectively.
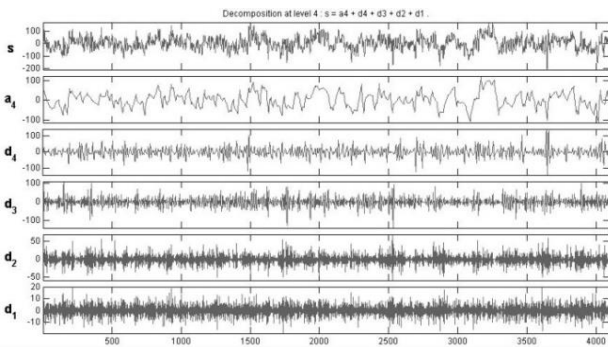
Figure 1: Single channel EEG signal decomposition of set A using db-2 up to level 4.
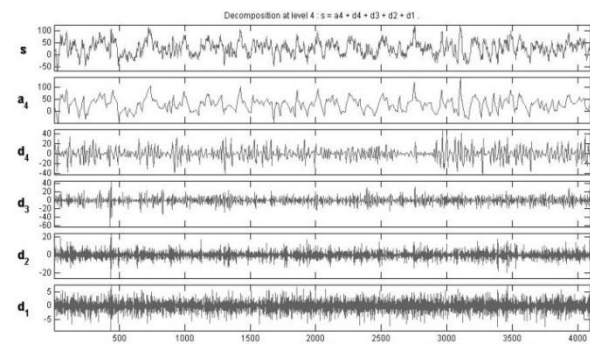
Figure 2: Single channel EEG signal decomposition of set D using db-2 up to level 4.

It is a transformation technique that provides a new data representation which can spread to multiple scales. Therefore, the analysis of transforming signal can be performed at a multiple resolution scale. DWT is performed by successively passing the signal through a series of high pass and low pass filters producing a set of detail and approximation coefficients. This generates a decomposing tree known

as Mallat's decomposition tree. In this analytical work, the raw EEG signals that have been picked up from web resources is decomposed using DWT [13] available as a toolbox in MATLAB. This signal is decomposed using the Daubechis Wavelet function of order 2 up to 4 levels [12]. Thus, it produces a series of wavelet coefficient like four detailed coefficients (D1, D2, D3, and D4) and an approximation signal (A4). Figures 1, 2, and 3 provides a snapshot of this decomposition of a single channel EEG recording from set A, D, and E respectively.
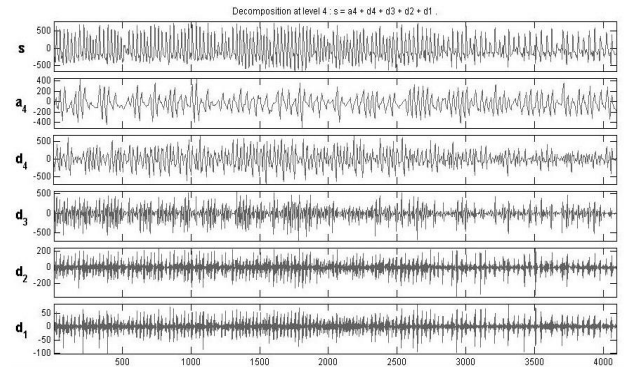
Figure 3: Single channel EEG signal decomposition of set E using db-2 up to level 4.

Later, on this decomposition some of the statistical features of many have been extracted from the signals such as Minimum (MIN), Maximum (MAX), MEAN, and Standard Deviation (SD). Figure 4 is a sample output of the MATLAB toolbox showing different features of a single channel EEG recording from set A and set E. The same procedure can be followed for all other EEG recordings to make a perfect set. So, after this level, we are ready with a sample feature dataset of order 500 by 20 matrixes as shown in Table 1. Then it can be further used for classification tasks.
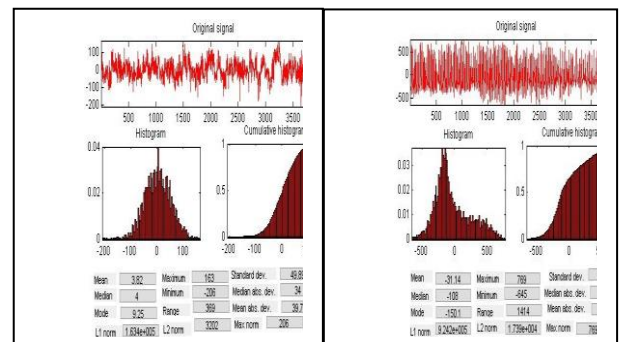
Figure 4: Statistical features extraction from signals after decomposition.

In addition to DWT, there are other feature extraction techniques [5] that can also be used successfully to extract features from the raw EEG signal. These techniques may include Wavelet Packet Decomposition (WPD), Principal Component Analysis (PCA), Lyapunov Exponent, ANNOVA test, etc.

Table 1: Structure and dimension of dataset for EEG signal classification.

Seizure Detection Sets	Size of Sample	Class 0	Class 1
Set1- (A & E)	200x20	100x20	100x20
Set 2- (D & E)	200x20	100x20	100x20
Set 3- (A+D & E)	300x20	200x20	100x20

3   Machine learning classifiers

Machine learning (ML) is a set of computerized techniques, which focus to automatically learn to recognize complex patterns and make intelligent decisions based on data. ML has proven its ability to uncover hidden information present in large complex datasets. Using ML, it is possible to cluster similar data, classify, or to find association among various features [14, 15]. In the context of EEG signal analysis, ML is the application of algorithms for extracting patterns from EEG signals [16]. However, there are other steps also carried out e.g., data cleaning &pre-processing, data reduction & projection, incorporation of prior knowledge, proper validation and interpretation of results while analysing EEG signals. EEG analysis has number of challenges which make it suitable for machine learning techniques [16].

- EEG comes in large databases.
- EEG recordings are very noisy.
- EEG signals have large temporal variance.

Some popular machine learning approaches are neural networks, evolutionary algorithms, fuzzy theory, and probabilistic learning. In this analytical work, our focus is restricted with neural networks, its variants and support vector machines for classification EEG signals.

3.1   Multilayer perceptron neural network (MLPNN)

Artificial neural network simulates the operation of a neural network of the human brain and solves a problem. Generally, single layer Perceptron neural networks are sufficient for solving linear problems, but nowadays the most commonly employed technique for solving nonlinear problems is Multilayer Perceptron Neural Network (MLPNN) [17]. It can hold various layers such as one input and one output layer along with at least one hidden layer. There are connections between different layers for data transmission. The connections are generally weighted edges to add some extra information's to the data and it can be propagated through different activation functions.

The heart of designing an MLPNN is the training of network for learning the behaviour of input-output patterns. In this work, we have designed an MLPNN with the help of a Java Encog framework. This network is trained with the help of three popular training algorithms such as Back- propagation (BP)

[18], Resilient Propagation (RPROP) [19], and Manhattan Update Rule (MUR).

Back-propagation training algorithm [5, 19, and 20] is different from other algorithms in terms of the weight updating strategies. In back propagation [21, 22, 23], generally weight is updated by the equation (4) [24, 25, 26].

$$w_{ij}(k+1) = w_{ij}(k) + \Delta w_{ij}(k), \qquad (4)$$

where in regular gradient decent

$$\Delta w_{ij}(k) = -\eta \frac{\partial E}{\partial w_{ij}}(k) \qquad (5)$$

with a momentum term

$$\Delta w_{ij}(k) = -\eta \frac{\partial E}{\partial w_{ij}}(k) + \mu \Delta w_{ij}(k-1) \qquad (6)$$

Resilient propagation [19] is a supervised training algorithm for feed forward neural network. Instead of magnitude, it takes into account only the sign of the partial derivative, or gradient decent and acts independently on each weight. The advantage of RPROP algorithm is that it needs no setting of parameters before applying it. The weight updating is done according to the equation (7). Equation 4 is same for the RPROP for weight update.

$$\Delta w_{ij} = \begin{cases} +\Delta_{ij}, if \frac{\partial E}{\partial w_{ij}}(k) > 0, \\ +\Delta_{ij}, if \frac{\partial E}{\partial w_{ij}}(k) < 0, \\ \quad 0, Otherwise \end{cases} \qquad (7)$$

$$\Delta_{ij} = \begin{cases} \eta_+ * \Delta_{ij}(k-1), S_{ij} > 0, \\ \eta_- * \Delta_{ij}(k-1), S_{ij} < 0, \\ \Delta_{ij}(k-1), Otherwise. \end{cases}$$

$$(8)$$

where, $S_{ij} = \frac{\partial E}{\partial w_{ij}}(k-1) * \frac{\partial E}{\partial w_{ij}}(k)$ and $\eta_+ = 1.2$ and $\eta_- = 0.5$.

Manhattan update rule also works similar to RPROP and only uses the sign of the gradient and magnitude is discarded. If the magnitude is zero, then no change is made to the weight or threshold value. If the sign is positive, then the weight or threshold value is increased by a specific amount defined by a constant. If the sign is negative, then the weight or the threshold value decreases by a specific amount defined by a constant. This constant must be provided to the training algorithm as a parameter.

3.2   Variants of neural network

In addition to MLPNN, many different types of neural networks have been developed over the year for solving problems with varying complexities of pattern classification. Some of these includes: Recurrent Neural Network (RNN) [41], Probabilistic Neural Network (PNN) [42], and Radial Basis Function Neural Network (RBFNN) [43].

3.2.1 Recurrent neural network

RNN [44] is a special type of artificial neural network having a fundamental feature is that the network contains at least one feedback connection [45], so that activation can flow round in a loop. This feature enables the network to do temporal processing and learn the patterns. The most important common features shared by all types of RNN [46, 47] are, they incorporate some form of Multilayer Perceptron as sub-system. They implement the non-linear capability of MLPNN [48, 49] with some form of memory. In this research work the ANN architecture, we have implemented for modelling and classifying is the Elman Recurrent Neural Network (ERNN). It was originally developed by Jeffrey Elman in 1990. The Back-Propagation through time (BPTT) learning algorithm is used for training [50, 51], which is an extension of Back-propagation that performs gradient decent on a complete unfolded network.

If a network training sequence starts on time t_0 and ends at time t_1, the total cost function can be calculated as:

$$E_{total}(t_0, t_1) = \sum_{t=t_0}^{t_1} E_{\frac{sse}{ce}}(t), \tag{9}$$

and the gradient decent weight update can be calculated as:

$$\Delta w_{ij} = -\eta \sum_{t=t_0}^{t_1} \frac{\partial E_{\frac{sse}{ce}}(t)}{\partial w_{ij}}. \tag{10}$$

3.2.2 Probabilistic neural network

PNN was first proposed by Specht in 1990. It is a classifier that maps input patterns in a number of class levels. It can be forced into a more general function approximator. This network is organized into a multilayer feed forward network with input layer, pattern layer, summation layer, and the output layer. PNN [52] is an implementation of a statistical algorithm called kernel discriminant analysis. The advantages of PNN are like; it has a faster training process as compared to Back-Propagation. Also, there are no local minima issues. It has a guaranteed coverage to an optimal classifier as the size of the training set increases. But it has few disadvantages like slow execution of the network because of several layers and heavy memory requirements, etc.

In PNN [52] a Probability Distribution Function (PDF) is computed for each population. An unknown sample s belongs to a class p if,

$$PDF_p(s) > PDF_q(s) \forall p \neq q, \tag{11}$$

where, $\mathrm{PDF}_k(s)$ is the PDF for class k.

Other parameters used are Prior Probability - h, Misclassification Cost $- c$, so the classification decision becomes,

$$h_p c_p PDF_p(s) > h_q c_q PDF_q(s) \forall p \neq q \tag{12}$$

PDF for a single sample can be calculated by using the formula,

$$PDF_k(s) = \frac{1}{\sigma} W\left(\frac{s-s_k}{\sigma}\right), \tag{13}$$

where s – Input (unknown), s_k - k^{th} sample, W- weighting function, σ - smoothing parameter. PDF for a single population can be calculated by taking the average of PDF of n samples.

$$PDF_k^n(s) = \frac{1}{n\sigma} \sum_1^n W\left(\frac{s-s_k}{\sigma}\right) \tag{14}$$

From the result table, it is experimentally proved that for epilepsy identification in EEG signal, PNN gives the most accurate result by taking minimum amount of time.

3.2.3 Radial basis function neural network

RBF networks are also a type of feed-forward network, trained by using a supervised training algorithm. The main advantage of RBF network is, it has only one hidden layer. The RBF network, usually trains much faster than back-propagation networks. This kind of network is less susceptible to problems with non-stationary inputs because of the behaviour of radial basis function hidden units. The general formula for the output of RBF network [53] can be represented as follows, if we consider the Gaussian function as the basis function.

$$y(x) = \sum_{i=1}^{M} w_i e^{\left(\frac{-(\|x-c_i\|)^2}{2\sigma^2}\right)} \tag{15}$$

where x, $y(x)$, c_i, σ, and M denotes input, output, center, width, and number of basis function centered at c_i, similarly w_i denotes weights.

For this work, we have constructed a Radial Basis Function Network by taking into consideration of the Gaussian function as the basis function with a pre-fixing of randomized centres and widths.

3.3 Support vector machine (SVM)

SVM is the most widely used machine learning technique based pattern classification technique nowadays. It is based on statistical learning theory and was developed by Vapnik in the year 1995. The primary aim of this technique is to project nonlinear separable samples onto another higher dimensional space by using different types of kernel functions. In late years, kernel methods have received major attention, especially due to the increased popularity of Support Vector Machines [27]. Kernel functions play a significant role in SVM [28, 29] to bridge from linearity to nonlinearity. Least square SVM [30] is also an important SVM technique that can be applied for classification task [31]. Extreme learning Machine and Fuzzy SVM [32, 33, 34] and Genetic algorithm tuned expert model [32] can also be applied for the purpose of classification.

In this analytical work, we have evaluated three different types of kernel functions [35], i.e., Linear, Polynomial, and RBF kernel [36]. Linear kernel is the simplest kernel function available. Kernel algorithm using a linear kernel is often equivalent to their non-kernel counterparts [37]. From the result table it can be clearly understood that for a classification problem consisting of only sets A & E or D & E, is providing

100% accuracy. But it is not able to classify properly by considering sets A+D & E.

Polynomial kernel is a non-stationary kernel. This kernel function can be represented as given in equation:

$$K(x, y) = (\alpha x^T y + c)^d,   \quad (16)$$

where α, c and d denotes slope, any constant, and degree of polynomial, respectively.

Somehow this kernel function [38, 39] is better as compared to linear kernel function. However, the RBF kernel function [40] has been proven as the best kernel function used for this application, which can classify different groups with 100% accuracy with a minimum time interval.

4   Ensemble of machine learning classifiers

From the empirical analysis we conclude that there some classifier models e.g., SVM and PNN outperforms all other techniques such as MLPNN, RNN, and RBFNN. Hence, to boost the poorly performer as compared to SVM and PNN, we have proposed a model for an ensemble based classifier that combines the above three techniques and improves the accuracy of classification for epileptic seizure detection. This proposed ensemble technique uses a weighted majority vote for classification. The main goal of an ensemble method is to combine the efficiency of several basic classifier models and build a learning algorithm to improve the robustness over a single classification technique. Here we have combined the classification results of MLPNN, RNN, and RBFNN and constructed an ensemble classifier. This classifier uses a weighted majority based vote for classification. Generally in majority based vote the class label of a sample is decided by the class label that is classified by maximum number of classifiers. Let there are two classes (1 and 2) and three classifiers (clf1, clf2, and clf3). Let for a sample clf1 and clf2 classifies to class 2 whereas clf3 classifies to class 1 then the ensemble of classifiers classifies the sample to class 2.
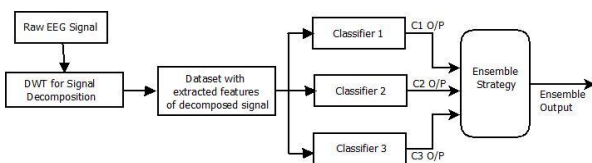
Figure 5: Proposed framework for ensemble based classifier for detection of epileptic seizure.

Figure 5 describes the architecture of ensemble of classifiers to detect epileptic seizure. It combines the output of three different classifiers such as classifier 1 (MLPNN), classifier 2 (RNN), and classifier 3 (RBFNN) based on a weighted majority voting mechanism. Weight parameter has been added to give different weightage to different classifiers based on their performance. For this we have collected the predicted class probabilities for each classifier and

multiplied it with classifier weight and the average is taken. Based on this weighted average probabilities, the class label has been assigned. Here we have taken simple weighted majority technique where same weight is assigned to each class label which is 1/k, where k is the number of class labels.

5   Empirical study

This section gives an empirical study on different classification techniques based on machine learning approach for detection of epilepsy in EEG brain signal. Various experiments are done to validate this empirical study. The Machine Learning based classifiers are proved as the most efficient way for pattern recognition. It aids to design models that can learn from some previous experience (known as training) and further it can be able to recognize appropriate patterns for unknown samples (known as testing).

All experiments for this research work are performed using a powerful Java Framework known as Encog [54] developed by Jeff Heaton and his team. Currently we are using Encog 3.2 Java framework for all experimental result evaluation. This is the latest version and it supports almost all the features of machine learning techniques. Along with this framework, there are a lot of packages, classes and methods that have been defined to support the experimental evaluations. Java is the most potent and efficient language nowadays. The rightness of the experimental works can be verified easily using this language. There are almost nine different machine learning algorithms that have been implemented for EEG signal classification for epileptic seizure detection.

5.1   Environment and parameter setup

The Encog Java framework provides a vast circle of library classes, interfaces, and methods that can be utilized for designing different machine learning based classifier models. There are lists of parameters (as shown in Table 2) required to be set for smooth and accurate execution of models.

5.2   Performance measures and validation techniques

Here, we have hashed out about the performance of all machine learning based classifiers for classifying EEG signal. The different measures used for performance estimation are: Specificity (SPE), Sensitivity (SEN), Accuracy (ACC), and Time elapsed for execution of models. From the evaluation result given in Table 3, it is clear that MLPNN with resilient propagation is the most efficient training algorithm both in considerations of accuracy as well as the amount of time needed to execute the programs in all different setting such as A&E, D&E, and A+D & E. This MLPNN technique can be compared with other machine learning techniques. In this work all the experimental evaluations are validated using k- fold cross validation

Table 2: Lists of parameters for models execution.

Classification Techniques	Required Parameters and Values
MLPNN/BP	Activation Function - Sigmoid Learning Rate = 0.7 Momentum Coefficient = 0.8 Input Bias – Yes
MLPNN/RPROP	Activation Function - Sigmoid Learning Rate = NA Momentum Coefficient = NA Input Bias – Yes
MLPNN/MUR	Activation Function - Sigmoid Learning Rate = 0.001 Momentum Coefficient = NA Input Bias – Yes
SVM/Linear	Kernel Type – Linear Penalty Factor = 1.0
SVM/Polynomial	Kernel Type – Polynomial Penalty Factor = 1.0
SVM/RBF	Kernel Type – Radial Basis Function Penalty Factor = 1.0
PNN	Kernel Type – Gaussian Sigma low – 0.0001 (Smoothing Parameter) Sigma high – 10.0 (Smoothing Parameter) Number of Sigma - 10
RNN	Pattern Type – Elman Primary Training Type – Resilient Propagation Secondary Training Type – Simulated Annealing **Parameters for SA** Start Temperature – 10.0 Stop Temperature – 2.0 Number of Cycles - 100
RBFNN	Basis Function – Inverse Multiquadric Center& Spread Selection – Random Training Type– SVD (Singular Value Decomposition)

where value of k is taken as 10. So the total dataset has been divided into 10 folds. Each fold is constructed with samples having almost same number from each class labels. In each iteration one fold is considered for testing the classifier and rest of the folds are taken for training the classifier. This is a very efficient validation technique as it rules out all possibilities of misclassification and gives an accurate efficiency measure.

Table 4 shows a comparison of different kernel types used for classification using Support Vector Machine (SVM). It is the most powerful and efficient machine learning tool for designing classifier model. This table clearly shows a very good result for SVM with RBF kernel.

Table 5 defines a list of experiments led by studying different forms of Neural Network, such as Radial Basis Function Neural Network, Probabilistic Neural Network, and Recurrent Neural Network. It suggests that the effectiveness of using PNN for classification of EEG signal for detecting epileptic seizures is promising.

5.3   Comparative analysis

Table 6 gives a detail empirical analysis of the performance of different classification techniques based on machine learning approaches. As discussed above in this experimental evaluation we have used 10-fold cross validation to validate the results of classification.

Table 7 gives the result of experimental evaluation for the proposed ensemble technique and results for individual classification techniques. Figure 6 gives a graphical representation of comparison of different individual machine learning techniques with ensemble based classification technique. These experimental result shows there is a remarkable increase in the accuracy for case 3 (A+D & E) along with other two cases.

Table 3: Experimental evaluation result of MLPNN with different training algorithms.

Cases for Seizure Types	Multi-Layer Perceptron Neural Network with different Propagation Training Algorithms											
	Back-Propagation				Resilient-Propagation				Manhattan-Update Rule			
	SPE	SEN	ACC	TIME	SPE	SEN	ACC	TIME	SPE	SEN	ACC	TIME
Case1 (A,E)	100	90.09	94.5	16.52	99.009	100	99.5	2.846	97.29	77.77	85	7.541
Case2 (D,E)	100	83.33	90	22.22	99.009	100	99.5	2.547	55.68	78.78	60	7.181
Case3 (A+D,E)	100	86.95	92.5	23.12	95.85	85.98	92.33	14.79	93.78	82.24	89.66	14.85

Table 4: Experimental evaluation result of SVM with different kernel types.

Cases for Seizure Types	Support Vector Machine with different Kernel Types											
	Linear				Polynomial				RBF			
	SPE	SEN	ACC	TIME	SPE	SEN	ACC	TIME	SPE	SEN	ACC	TIME
Case1 (A,E)	100	100	100	2.127	100	100	100	2.101	100	100	100	2.002
Case2 (D,E)	100	100	100	1.904	100	100	100	1.902	100	100	100	2.021
Case3 (A+D,E)	90.67	76.63	85.66	11.61	100	99.009	99.66	7.24	100	100	100	2.511

Table 5: Experimental evaluation result of RBFNN, RNN, PNN with different training algorithms.

Cases for Seizure Types	Other Types of Neural Network											
	RBF Neural Network				Probabilistic Neural Network				Recurrent Neural Network			
	SPE	SEN	ACC	TIME	SPE	SEN	ACC	TIME	SPE	SEN	ACC	TIME
Case1 (A,E)	83.076	65.925	71.5	2.051	100	100	100	0.967	77.173	73.148	75	10.31
Case2 (D,E)	100	97.08	98.5	1.828	100	100	100	0.977	64.705	71.604	67.5	13.29
Case3 (A+D,E)	92.30	66.41	81	2.928	100	100	100	1.616	67.346	66.666	67.333	19.58

Table 6: Comparative analysis of different machine learning classification techniques.

Machine Learning Classification Technique	Case-1 (set A & E)		Case-2 (set D & E)		Case-3 (set A+D & E)	
	Overall Accuracy in %age	Approximate Time taken in seconds	Overall Accuracy in %age	Approximate Time taken in seconds	Overall Accuracy in %age	Approximate Time taken in seconds
MLPNN/BP	94.5	16.527	90	22.226	92.5	23.127
MLPNN/RP	99.5	2.846	99.5	2.547	92.33	14.798
MLPNN/MUR	85	7.541	60	7.181	89.66	14.85
SVM/Linear	100	2.127	100	1.904	85.66	11.61
SVM/Ploy	100	2.101	100	1.902	99.66	7.24
SVM/RBF	100	2.002	100	2.021	100	2.511
PNN	100	0.967	100	0.977	100	1.616
RNN	75	10.31	67.5	13.29	67.33	19.58
RBFNN	71.5	2.051	98.5	1.828	81	2.928

Table 7: Comparative analysis of different machine learning classification techniques with ensemble based classifier.

Machine Learning Classification Technique	Case-1 (set A & E)		Case-2 (set D & E)		Case-3 (set A+D & E)	
	Overall Accuracy in %age	Approximate Time taken in seconds	Overall Accuracy in %age	Approximate Time taken in seconds	Overall Accuracy in %age	Approximate Time taken in seconds
MLPNN/RP	99.5	2.846	99.5	2.547	92.33	14.798
RNN	75	10.31	67.5	13.29	67.33	19.58
RBFNN	71.5	2.051	98.5	1.828	81	2.928
ENSEBLE CLASSIFIER	99.5	3.745	99.5	3.876	98.3	5.475
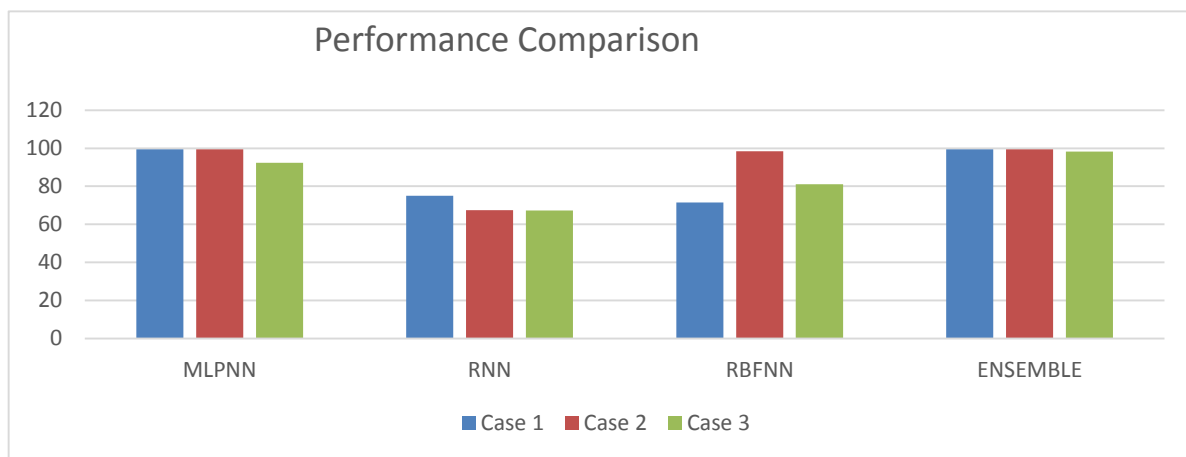
Figure 6: Performance comparison of different machine learning techniques with ensemble based classifier for detection of epileptic seizure.

6   Conclusions and future study

By classifying the EEG signals collected from different patients in different situations, detection of the epileptic seizure in EEG signal can be performed. Thus classification can be accomplished by using different machine learning techniques. In this work, the efficiency and functioning pattern of different machine learning techniques like MLPNN, RBFNN, RNN, PNN, and SVM for classification of EEG signal for epilepsy identification have been compared. Further, the tool MLPNN uses three training algorithms like BACKPROP, RPROP, and Manhattan Update Rule. Similarly, three kernels such as Linear, Polynomial, and RBF kernels are used in SVM. Hence, this comparative study clearly shows the differences in the efficiency of different machine algorithms with respect to the task of classification. Moreover, from the experimental study, it can be concluded that SVM is the most efficient and powerful machine learning technique for the purpose of classification of the EEG signal. Also, SVM with RBF kernel provides the utmost accuracy in all settings of the classification task. Besides this, PNN is a good contender for SVM for this specific application. But compared to SVM, PNN requires some extra overhead in setting the parameters. Also our proposed ensemble of classifiers based on weighted majority voting that combines the efforts of three different poorly performer classifiers such as MLPNN, RNN and RBFNN is enhancing the performance in different cases. Our continuous efforts in this area of research (both theoretical and experimental)is marching with lots of issues and will go ahead in future by considering the real cases with state-of-the-art meta-heuristic optimization techniques.

7   References

[1]   Niedermeyer, E. and Lopesda Silva, F. (2005) *Electroencephalography: Basic Principles, Clinical Applications, and Related Field*s, 5th edition Lippincott Williams and Wilkins, London.

[2]   Sanei, S. and Chambers, J. A. (2007) *EEG Signal Processing,* Wiley, New York.

[3]   Lehnertz, K. (1999) 'Non-linear time series analysis of intracranial EEG recordings in patients with epilepsy--an overview', *International Journal of Psychophysiology,* Vol. 34, No. 1, pp.45-52.

[4]   Alicata, F.M., Stefanini, C., Elia, M., Ferri, R., Del Gracco, S., and Musumeci, S.A. (1996) 'Chaotic behavior of EEG slow-wave activity during sleep', *Electroencephalography and Clinical Neurophysiology*, Vol. 99, No. 6, pp. 539–543.

[5]   Acharya, U. R., Sree, S. V., Chuan Alvin, A. P., and Suri, J. S. (2012) 'Use of principal component analysis for automatic classification of epileptic EEG activities in wavelet framework', *Expert Systems with Applications*, Vol. 39, No. 10, pp.9072-9078.

[6]   Acharya, U. R., Sree, S. V., Swapna, G., Joy Martis, R., and Suri, J. S. (2013) 'Automated EEG analysis of epilepsy: A review', *Knowledge Based System*, Vol. 45, pp. 147-165.

[7]   EEG Data. [Online] *http://www.meb.unibonn.de/science/physik/eegdata.html*, 2001.

[8]   Andrzejak, R. G., Lehnertz, K., Mormann, F., Rieke, C., David, P. and Elger, C. E.(2001) 'Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state', *Physical Review* E, Vol. 64, No. 6, pp. 1-6.

[9]   Gandhi, T., Panigrahi, B. K., and Anand, S. (2011) 'A comparative study of wavelet families for EEG signal classification', *Neurocomputing*, Vol. 74, No. 17, pp. 3051–3057.

[10]   Yong, L. and Shenxun, Z. (1998) 'The application of wavelet transformation in the analysis of EEG', *Chinese Journal of Biomedical Engineering*, pp. 333-338.

[11] Ocak, H. (2009) 'Automatic detection of epileptic seizures in EEG using discrete wavelet transform and approximate entropy', *Expert Systems with Applications,* Vol. 36, No.2, pp. 2027–2036.

[12] Adelia, H., Zhoub, Z., and Dadmehrc, N. (2003) 'Analysis of EEG records in an epileptic patient using wavelet transform', *Journal of Neuroscience Methods*, Vol. 123, No.1, pp. 69–87.

[13] Parvez, M. Z. and Paul, M. (2014) 'Epileptic seizure detection by analyzing EEG signals using different transformation Techniques', *Neurocomputing*, Vol. 145 (Part A), pp.190-200.

[14] Majumdar, K. (2011) 'Human scalp EEG processing: various soft computing approaches, *Applied Soft Computing*, Vol. 11, No. 8, pp. 4433–4447.

[15] Teixeira, C. A., Direito, B., Bandarabadi, M., Quyen, M. L. V., Valderrama, M., Schelter, B., Schulze-Bonhage, A., Navarro, V., Sales, F., and Dourado, A. (2014) 'Epileptic seizure predictors based on computational intelligence techniques: A comparative study with 278 patients', *Computer Methods and Programs in Biomedicine,* Vol. 114, No. 3,pp. 324–336.

[16] Siuly and Li, Y. (2014) 'A novel statistical algorithm for multiclass EEG signal classification', *Engineering Applications of Artificial Intelligence*, Vol. 34, pp. 154–167.

[17] Jahankhani, P., Kodogiannis, V., and Revett, K. (2006) 'EEG signal classification using wavelet feature extraction and neural networks', Proc. of *IEEE John Vincent Atanasoff International Symposium on Modern Computing* (JVA 2006), pp.120-124.

[18] Guler, I.D. (2005) 'Adaptive neuro-fuzzy inference system for classification of EEG signals using wavelet coefficients', *Neuroscience Methods*, Vol. 148, pp. 113–121.

[19] Riedmiller, M. and Braun, H. (1993) 'A direct adaptive method for Faster Back-propagation Learning: The RPROP Algorithm', *Proc. of IEEE International Conference on Neural Networks*, Vol. 1, pp. 586 – 591.

[20] Mirowski, P., Madhavan, D., LeCun, Y., Kuzniecky, R. (2009) 'Classification of patterns of EEG synchronization for seizure prediction', *Clinical Neurophysiology*, Vol. 120, pp. 1927–1940.

[21] Subasi, A. and Ercelebi, E. (2005) 'Classification of EEG signals using neural network and logistic regression', *Computer Methods Programs in Biomedicine*, Vol. 78, No. 2,pp. 87–99.

[22] Subasi, A., Alkan, A., Kolukaya, E. and Kiymik, M. K. (2005) 'Wavelet neural network classification of EEG signals by using AR model with MLE pre-processing', *Neural Networks*, Vol. 18, No. 7, pp. 985–997.

[23] Ubeyli, E.D. (2009) 'Combined neural network model employing wavelet coefficients for EEG signals classification', *Digital Signal Processing*, Vol. 19, No. 2, pp. 297–308.

[24] Kalayci, T. and Ozdamar, O. (1995) 'Wavelet pre-processing for automated neural network detection of EEG spikes', *IEEE Engineering in Medicine and Biology Magazine*, Vol. 14, No. 2, pp. 160–166.

[25] Orhan, U., Hekim, M., and Ozer, M. (2011) 'EEG signals classification using the K-means clustering and a multilayer perceptron neural network model', *Expert Systems with Applications*, Vol. 38, No. 10, pp. 13475–13481.

[26] Pradhan, N., Sadasivan, P.K., and Arunodaya, G.R.(1996) 'Detection of seizure activity in EEG by an artificial Neural Network: A Preliminary study', *Computers and Biomedical Research*, Vol. 29, No. 4, pp. 303-313.

[27] Lima, C. A. M., Coelho, A. L. V., and Eisencraft, M. (2010) 'Tackling EEG signal classification with least squares support vector machines: A sensitivity analysisstudy', *Computers in Biology and Medicine*, Vol. 40, No. 8, pp. 705–714.

[28] Kumar, Y., Dewal, M.L. and Anand, R.S. (2014) 'Epileptic seizure detection using DWT based fuzzy approximate entropy and support vector machine', *Neurocomputing,* Vol.133, No. 7, pp. 271–279.

[29] Limaa, C. A.M. and Coelhob, A. L.V. (2011) 'Kernel machines for epilepsy diagnosis via EEG signal classification: A comparative study', *Artificial Intelligence in Medicine*, Vol. 53, No. 2, pp. 83–95.

[30] Siuly, Li, Y. and Wen, P. (2009) 'Classification of EEG signals using sampling techniques and least square support vector machines', *Rough Sets and Knowledge Technology*, Vol. 5589, pp. 375–382.

[31] Übeyli, E.D. (2010) 'Least squares support vector machine employing model-based methods coefficients for analysis of EEG signals', *Expert Systems with Applications,* Vol. 37, No. 1, pp. 233–239.

[32] Dhiman, R., Saini, J.S., and Priyanka (2014) 'Genetic algorithms tuned expert model for detection of epileptic seizures from EEG signatures', *Applied Soft Computing*, Vol. 19, pp. 8–17, 2014.

[33] Xu, Q, Zhou, H., Wang, Y., and Huang, J. (2009) 'Fuzzy support vector machine for classification of EEG signals using wavelet-based features', *Medical Engineering &Physics,* Vol. 31, No. 7, pp. 858–865.

[34] Yuan, Q, Zhou, W., Li, S., and Cai, D. (2011) 'Epileptic EEG classification based on extreme learning machine and nonlinear features', *Epilepsy Research*, Vol. 96, No. 1-2, pp. 29-38.

[35] Liu, Y., Zhou, W., Yuan, Q. and Chen, S. (2012) 'Automatic Seizure Detection Using Wavelet Transform and SVM in Long-Term Intracranial EEG', *IEEE Transactions on Neural Systems and*

Rehabilitation Engineering, Vol. 20, No. 6, pp. 749-755.

[36] Shoeb, A., Kharbouch, A., Soegaard, J., Schachter, S., and Guttag, J. (2011) 'A machine learning algorithm for detecting seizure termination in scalp EEG', *Epilepsy &Behavior*, Vol. 22, No. 1, pp. 36–43.

[37] Subasi, A. and Gursoy, M. I. (2010) 'EEG signal classification using PCA, ICA, LDA and support vector machines', *Expert Systems with Applications*, Vol. 37, No. 12, pp.8659–8666.

[38] Cristianini, N. and Taylor, J. S. (2001) *Support Vector and Kernel Machines*, Cambridge University Press, London.

[39] Taylor, J.S. and Cristianini, N. (2000) *Support Vector Machines and Other Kernel-Based Learning* Methods, Cambridge University Press, London.

[40] Vatankhaha, M., Asadpourb, V., and FazelRezaic, R. (2013) 'Perceptual pain classification using ANFIS adapted RBF kernel support vector machine for therapeutic usage', *Applied Soft Computing*, Vol. 13, No. 5, pp. 2537–2546.

[41] Pineda, F.J. (1987) 'Generalization of back-propagation to recurrent neural networks', *Physical Review Letters*, Vol.59, No. 19, pp. 2229–2232.

[42] Adeli, H. and Panakkat, A. (2009) 'A probabilistic neural network for earthquake magnitude prediction', *Neural Networks*, Vol. 22, No. 7, pp. 1018-1024.

[43] Ghosh-Dastidar, S., Adeli, H., and Dadmehr, N. (2008) 'Principal Component Analysis-Enhanced Cosine Radial Basis Function Neural Network for Robust Epilepsy and Seizure Detection', *IEEE Transactions on Biomedical Engineering*, Vol. 55, No. 2, pp.512-518.

[44] Guler, N. F., Ubeyli, E. D., and Guler, I. (2005) 'Recurrent neural networks employing Lyapunov exponents for EEG signals classification', *Expert Systems with Applications*, Vol. 29, No. 3, pp. 506–514.

[45] Petrosian, A., Prokhorov, D., Homan, R., Dascheiff, R., and Wunsch, D. (2000) 'Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG', *Neurocomputing,* Vol. 30, No. 1-4, pp. 201–218.

[46] Petrosian, A., Prokhorov, D.V., Lajara-Nanson, W., and Schiffer, R. B. (2001) 'Recurrent neural network-based approach for early recognition of Alzheimer's disease in EEG', *Clinical Neurophysiology*, Vol. 112, No. 8, pp.1378–1387.

[47] Übeyli, E. D. (2009) 'Analysis of EEG signals by implementing Eigen vector methods/recurrent neural networks', *Digital Signal Processing*, Vol.19, No. 1, pp.134–143.

[48] Derya, E. (2010) 'Recurrent neural networks employing Lyapunov exponents for analysis of ECG signals', *Expert Systems with Applications*, Vol. 37, No. 2, pp. 1192–1199.

[49] Saad, E.W., Prokhorov, D.V. and Wunsch, D.C. (1998) 'Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks', *IEEE Transaction on Neural Networks*,Vol.9, No. 6, pp. 1456–1470.

[50] Gupta, L., McAvoy, M., and Phegley, J. (2010) 'Classification of temporal sequences via prediction using the simple recurrent neural network', *Pattern Recognition*, Vol. 33,No. 10, pp. 1759–1770.

[51] Gupta, L. and McAvoy, M. (2000) 'Investigating the prediction capabilities of the simple recurrent neural network on real temporal sequences', *Pattern Recognition*, Vol. 33, No. 12, pp. 2075–2081.

[52] Derya, E. (2010) 'Lyapunov exponents/probabilistic neural networks for analysis of EEG signals', *Expert Systems with Applications*, Vol. 37, No. 2, pp. 985–992.

[53] Saastamoinen, A., Pietila, T., Varri, A., Lehtokangas, M. and Saarinen, J. (1998) Waveform detection with RBF network application to automated EEG analysis', *Neurocomputing*, Vol. 20, No. 1-3, pp. 1–13.

[54] Heaton, J. (2011) *Programming Neural Networks with Encog 3 in Java*, 2nd Edition, Heaton Research.

Software Architectures Evolution Based Merging

Zine-Eddine Bouras
LISCO Laboratory, Department of Mathematics and Computer Sciences
P.O. Box 218, EPST Annaba Algeria
E-mail: z.bouras@epst-annaba.dz

Mourad Maouche
Department of Software Engineering, Faculty of Information Technology
P.O. Box 1 Philadelphia University 19392, Jordan
E-mail: mmaouch@philadelphia.edu.jo

During the last two decades the software evolution community has intensively tackled the software merging issue. The main objective is to compare and merge, in a consistent way, different versions of software in order to obtain a new version. Well established approaches, mainly based on the dependence analysis techniques on the source code, have been used to bring suitable solutions. However the fact that we compare and merge a lot of lines of code is very expensive. In this paper we overcome this problem by operating at a high level of abstraction. The objective is to investigate the software merging at the level of software architecture, which is less expensive than merging source code. The purpose is to compare and merge software architectures instead of source code. The proposed approach, based on dependence analysis techniques, is illustrated through an appropriate case study.

Povzetek: Prispevek se ukvarja z ustvarjanjem nove verzije programskega sistema iz prejšnjih na nivoju abstraktne arhitekture.

1 Introduction

Software evolution is the response to software systems that are constantly changing in response to changes in user needs and the operating environment. This arises, often, when new requirements are introduced into an existing system, specified requirements are not correctly implemented, or the system is to be moved into a new operating environment [1]. One way to cope with evolution is to carry out the software from the scratch, but this solution is very expensive. Another way, that is less expensive, is to proceed by merging changes.

Software practitioners are used first to manage individually each change in a separate and independent way leading to a new version, then to check that all resulting individual versions do not exhibit incompatible behaviors (non-interference), and finally to merge them into a single version that incorporates all changes (if they do not interfere) [2].

Such techniques, known as program merging, have been widely used at the level of source code [2-4]. However comparing and merging a huge number of lines of codes is very expensive. Our main motivation is to overcome this problem by going up at the level of software architecture where the number of comparison and merge is smaller than in the source code.

In this way, we must address some problems like (1) understanding what an existing architecture does and how it works (dependency analysis), (2) how to capture the differences between several versions of a given architecture, and (3) how to create new architecture. The first problem was resolved by Kim et al. [5].

The objective of this paper is to suggest an approach to deal with the rest of problems, namely finding an approach to compare and merge software architectures. More precisely, we suggest reusing the well-known and efficient program merging algorithm due to Horwitz [6]. This paper will show the applicability of this algorithm through an appropriate example.

The rest of the paper is organized as follows. Section 2 is dedicated to related works. Section 3 presents the notion of software architecture description and a running example to be used throughout this paper. Section 4 introduces software merging in general and software architecture merging in particular. Section 5 is dedicated to the needed concepts in our approach. In section 6 we present, detail, and illustrate our approach of software architecture description merging.

2 Related Works

Besides differencing programs done by Horwitz [6], there are other works that investigate differencing hierarchical information for a large code such that Apiwattanapong et al. in [7] and Raghavan et al. in [8].

In the context of design differencing Xing and Stroulia in [9] use the assumption that the entities they are differencing are uniquely named and many nodes match exactly. A basic change due to designers is to rename entities in order to become more expressive. In this way the proposed approach fails.

Abi-Antoun et al. in [10] propose an algorithm based on empirical evaluation to cope with architectural merging issue. Empirical Evaluation losses information in some cases, merging architecture needs the study of dependencies, formally, between components.

Finally there is an approach that copes with software architectures evolution based merging. Bouras and Maouche [11] use an internal form to represent software architecture and proceed by a syntactic differentiation. They, also, detect some type of conflicts that can fail the process.

Our approach is more formal and precise in term of dependency analysis. It uses the technique of slicing that is a formal filter. Slicing permits dependency analysis of software architecture by allowing us to find matching's and differences between elements of Software Architecture Descriptions (SAD) during merging process, and then merge components (if they are compatibles) to obtain a new version of SAD.

3   Software architecture description

Understanding all aspects of complex system is very hard. It therefore makes sense to be able to look at only those aspects of a system that are of interest at a given time. The concept of architecture views exists for this purpose. According to IEEE 2007, a view is a representation of a whole system from the perspective of a set of concerns. Each view addresses a set of system concerns, following the conventions of its viewpoint, where a viewpoint is a specification that describes the notations and modeling techniques to be used in a view to express the architecture in question from a given perspective [12]. Examples of viewpoints include: Functional viewpoint, Logical viewpoint, Component-and-connector viewpoint, etc. This paper is based on Component-and-connector viewpoint which specify structural properties of component and connector models in an expressive and intuitive way. They provide means to abstract away direct hierarchy, direct connectivity, port names and types, and thus can crosscut the traditional boundaries of the implementation-oriented hierarchical decomposition of systems and sub-systems [13, 14].

3.1   The example: Electronic Commerce

We introduce the running example, inspired from [5], to be used throughout this paper.

An order entry form is entered, electronically by a clerk. This form is taken by the Electronic Order Processing System (EOPS) and transformed on several actions through its five components: Ordering, Order_Entry, Inventory, Shipping, and Accounting. Components are distributed over different platforms, have a number of connectors between them, are independent processes, and communicate with each other through parameterized events. EOPS is depicted in figure 1.

EOPS stores the order information through CGI, and triggers Ordering which is the front-end of the whole system. This triggering is done through an I_order event. I_order generates a place_order event (internal action depicted by a dotted arrow) at the place_order port. The payment results from a payment_req event of Ordering which takes place when Ordering gets notified from the Order_Entry (implicit invocation depicted by a bold arrow).

When the payment gets approval, Ordering gets an order_success event and generates I_ship_info event to notify the customer of a successful order (internal action). Otherwise Ordering gets an order_fail event and notifies customer of unsuccessful order through I_order_rej event (internal action).

Order_Entry gets a take_order event from Ordering whenever customer places an order (external communication depicted by an arrow). An order is broken down into several items and each information of them is sent to Inventory through a ship_item event along with the customer information. ship_item events are generated whenever each ordered item is processed by Inventory to pass next item information until all the items for an order are processed. The done event results from a next_item event when there is no more items to be processed and this event triggers the payment information request payment_req of Ordering. Inventory generates a get_next event whenever it gets a find_item event to get the other item information for the order. Inventory generates two events, a ship event to Shipping and add_item to Accounting if an item is in the inventory it generates a back_order event to Inventory in order to get and ship the out-of-stock item, otherwise (concurrency). A restock_items event occurs when a customer cancels an order, this event does not cause any further event generation and is represented by special symbol called internal sink.

Shipping takes care of gathering the items of an order through recv_item events from Inventory. When it gets a shipping approval through a recv_receipt event from Accounting, it generates a shipping_info event to Ordering and it ships ordered items (synchronization). When it receives a cancel event (due to canceled order), it generates a restock event along with the item information it received. Accounting accumulates the total amount for an order whenever it receives an items event and verifies resources by communicating with outside components when it receives a checking request and sends the result (e.g., good/bad). Upon receiving payment_res (i.e., good or bad), it issues either an issue_receipt event as an approval for shipping when successful or fail and restock events to inform the failure of the order process to the customer.

4   Software architecture merging

Merging approaches take a form when concurrent modifications of the same system are done by several developers. They are able to merge changes in order to obtain ultimately one consolidated version of a system

again. However, they are faced to two challenges: the representation and how to find out differentiation.

The first one concerns the representation of software artifact on which the merge approach operates. It may either be text-based or graph-based. The second challenge concerns how differences are identified, represented, and merged [14, 15].
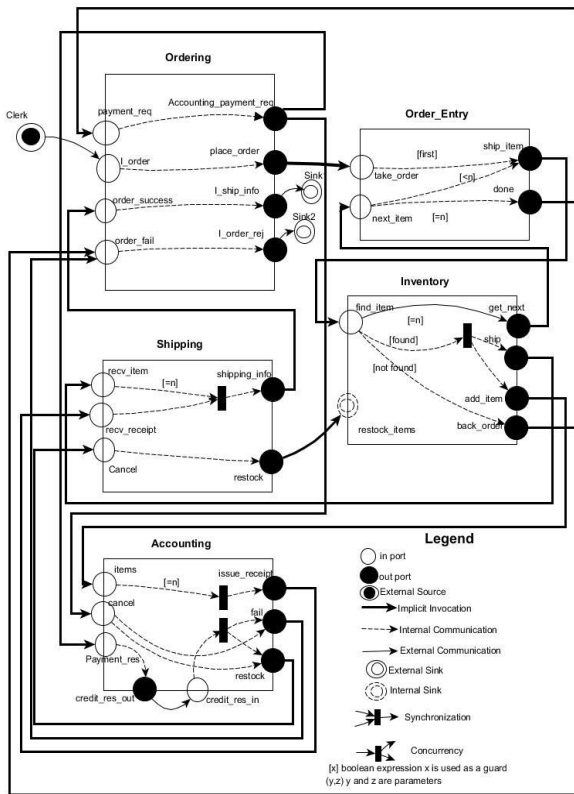

Figure 1: Software architecture of electronic order processing system.

Text-based merge approaches operate solely on the textual representation of a software artifact in terms of text files. The unit element of the text file may either be a paragraph, a line, a word, or even an arbitrary set of characters. Unit element of given version is compared to the original unit element in order to create the new one. The major advantage of such approaches is their independence of the programming languages used in the versioned artifacts. However, the major problem when merging flat files, syntax and semantics of a programming language are losses [14, 15].

Graph-based approaches overcome these problems; they operate on a graph-based representation of a software artifact for achieving more precise merging. Such approaches translate the versioned software artifact into a specific structure (graph) before merging. The unit elements (e.g. components) are represented by nodes and their relationships (e.g. connectors) by arcs. Changes consist of adding/deleting/updating unit elements [16]. However it requires a preliminary and primordial step which is known as software architecture understanding [16, 17]. It is very important to understand component's context and its running environment in order to

efficiently manage all kinds of dependencies. In general, as soon as a new component is installed, removed, or updated in a given software architecture, it has an impact on a part of the system. The new component may refer to certain components, and also be used by other components [16-20].

5 Software architecture merging concepts

Before starting our software architecture merging approach, it is useful to introduce some preliminary concepts related to software architectures and their understanding. These concepts concern how to represent software architecture as a graph (Software Architectural Description Graph) and how to find matching's and differences between components (slicing), and finally merging them.

5.1 Software Architectural Description Graph

Understanding a software addresses some problems like what it does and how it works. This is due to the implicit relationships between lines of codes. An explicit representation is needed.

Kim et al. in [5] propose a suitable dependence graph to support SAD named Software Architectural Description Graph (SADG). It consists of representing explicitly, dependencies between architecture elements i.e. component-connector, connector-component, and additional dependences. Informally, SADG is an arc-classified digraph whose vertices represent either the components or connectors in the description, and arcs represent dependencies between architectural design elements. Formal definitions and illustrations of SADG are detailed in [5].

In this paper we distinguish between two kinds of software architectural description: Base and variants. Base represents the original software architecture for which changes are requested. Variants represent a family of related and independent versions resulting from changes done on Base by independent developers. Also we point out that merge conflicts may occur. They take place if one change invalidates another change, or if two changes do not commute. Then, it is not decidable where to integrate changes [3]. For example if software architect of Variant A decides to update boolean expression [=n] to [n=10] in component Ordering_Entry (between in port next_item and out port done), while software architect of Variant B states that the same n will be n= 20, we are in the front of a conflict between architects, and merging process fails. In this case conflict is resolved manually. In this paper, we consider merging architectures without conflicts.

5.2 Architectural slicing

When a maintenance programmer wants to modify a component in order to satisfy new requirements, the programmer must first investigate which components

will affect the modified component and which components will be affected by the modified component. By using a slicing method, the programmer can extract the parts of a software architecture containing those components that might affect, or be affected by, the modified component. This can assist the programmer greatly by providing such change impact information.

Using architectural slicing to support change impact analysis of software architectures promises benefits for architectural evolution. Slicing is a particular application of dependence graphs. Together they have come to be widely recognized as a centrally important technology in software engineering. This due to the fact they operate on the deep rather than surface structures, they enable much more sophisticated and useful analysis capabilities than conventional tools [6].

Traditional slicing techniques cannot be directly used to slice software architectures. Therefore, to perform slicing at the architecture level, appropriate slicing notions for software architectures must be defined with new types of dependence relationships using components and connectors. Some works have investigated the issue of adapting the definition PDG to the level of software architecture. Between them, we can cite works of Rodrigues and Barbosa in [17] which propose the use of software *slicing* techniques to support a component's identification process via a specific dependence graph structure, the FDG (Functional Dependency Graph).

Zhao's technique, in [21] is based on analyzing the architecture of a software system given in Acme ADL. He captures various types of dependencies that exist in an architectural description. The considered dependencies arise as a result of dependence relationships existing among ports and/or roles of components and/or connectors. Architecture slicing technique operates by removing unrelated components and connectors, and ensures that the behavior of a sliced system remains unaltered.

Kim introduced an architectural slicing technique called dynamic software architecture slicing (DSAS) in [5]. A dynamic software architecture slice represents the run-time behavior of those parts of the software architecture that are selected according to the particular slicing criterion of interest to the software architect such as a set of resources and events.

An important distinction between a *static* and a *dynamic* slice is that static slices is computed without making assumptions regarding inputs, whereas the computation of dynamic slice relies on a specific test case. In other words, the difference between static and dynamic slicing is that dynamic slicing assumes *fixed* input, whereas static slicing does not make assumptions regarding the input, hence smaller in size than its static counterpart.

In order to illustrate the concept of dynamic slicing consider the fact that, we are interested by the run-time behavior of those parts of the software architecture of EOPS that are selected according to the particular slicing criterion when a customer wants to sell only one item that is in the inventory. The dynamic slicing concept is dedicated to find all implied parts of EOPS.

This triggering is done through an I_order event. I_order generates a place_order event at the place_order port. The payment results from a payment_req event of Ordering which takes place when Ordering gets notified from the Order_Entry.

Order_Entry gets a take_order event from Ordering whenever customer places the order (wiyh n=1). The item is sent to Inventory through a ship_item event along with the customer information. ship_item event is generated whenever the ordered item is processed by Inventory. Inventory generates a ship event to Shipping.

Shipping takes care of gathering the items of an order through recv_item events from Inventory. It generates a shipping_info event to Ordering and it ships ordered items. Finally Ordering gets an order_success event and generates CGI_ship_info event to the CGI program to notify the customer of a *successful* order. The run-time behavior is depicted in Figure 2.

Our graph representation is inspired from Kim's researches [5] where the process of architectural slice extraction from the software architectural description is based on the concept of Software Architectural Description Graph (SADG) and is a graph traversal.

Finally, comparing the behavior of Base with the behavior of a given variant consists of comparing static architectural slices of Base with static architectural slices of the given variant.
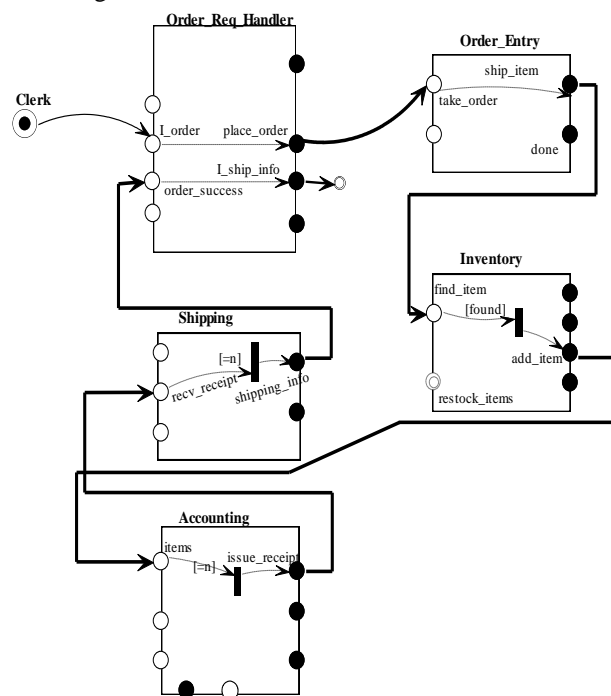

Figure 2: Example of Architecture Dynamic Slice.

5.3   Graph similarities

Comparing two graphs needs at first to find, for a given node (or edge) in a given graph, its corresponding node (or edge) in the other. An efficient way to find out similarity is the use of signature and structural matching [16].

A signature is defined as a pair of corresponding elements needs to share a set of properties such as type

information, which can be a subset of their syntactical information. Type information can be used to select the elements of the same type from the candidates to be matched because only elements with the same type need to be compared. Signature is used as the first criterion to match elements as proposed by [16]. If there is more than one candidate that has been found, the signature cannot identify a node uniquely. It is, therefore, to do further analysis by structural matching.

Structural matching is based on calculation of Graph Similarity using Maximum Common Edge Subgraphs [16]. The first algorithm to find the candidate node with maximal edge similarity for a given host node takes the host node and a set of candidate nodes of graph 2 as input, computes the edge similarity of every candidate node and returns a candidate with maximal edge similarity. The second algorithm for computing edge similarity between a candidate node and a host node takes two maps as, input, stores all the incoming and outgoing edges of the host and candidate nodes indexed by their edge signature. By examining the mapped edge pairs between these two maps, the algorithm computes the edge similarity as output. Graph similarities algorithm can be summarized as the following:

Let Base and a Variant SADG's
1. For each variant node

    1.1 Use signature matching to find candidate node
        If there is more than one candidate use structural matching
        Compare each node and its associated edges of Base with its variant peer (similar).

    1.2 Determine and collect sets of changed elements
    If no candidate, host node belongs to Delete set
        // exists in Base and not in Variant
    Remaining nodes in variant belongs to New set
        // exists in Variant and not in Base
    Compare names of each pair of nodes mapping
    If values are different, name belongs to Update set
        // all node mapping and differences are found

 2. Edges connecting to delete nodes are Delete edges
    Edges connecting to new nodes are New edges
    Apply signature matching to find out the edge mapping
    Remaining edges in Base belongs to Delete edges set
    Remaining edges in variant belongs to New edges set

All nodes in N1 have been examined by signature and structural matching; all possible node mappings between N1 and N2 are found.

6 Software architecture merging process

In this section we show how to reuse and adapt the Horwitz algorithm [6] to the context of software architecture merging. We show that this approach solves the issue of architecture merging because both, program

and architecture merging may be brought to a graph theory problem.

6.1 Software merging algorithm

Figure 3 resumes merging process. It starts from (1) a Base Software Architecture Description, (2) build a set of variants (resulting from Base changes), (3) build Software Architectural Description Graph for each SADG, (4) compare each variant with Base to determine sets of changed and preserved elements, and (5) combine these sets to form a single integrated new version (if changes don't interfere). Steps (1) and (2) are done concurrently by developers, in step (3) we construct SADG's according to Kim's approach.
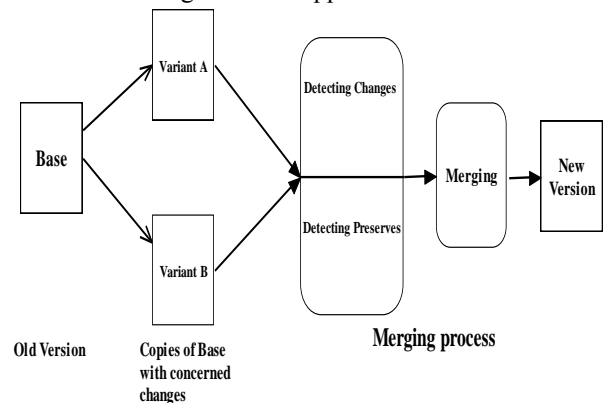
Figure 3: Merging Process.

Step 4: Compare each variant with the base to determine sets of changed and preserved elements
    For each variant
        4.1. Determine peer nodes and edges with Base by signature and structural matching.

        4.2. Extract from each SADG the associated slices.

        4.3. Determine sets of changed and preserved elements

            4.3.1. Map and compare each slice of the base software with its peer in variant.
            4.3.2. Determine and collect changed and preserved slices.

Step 5: Combine changed and preserved slices to form a new SADG.
    5.1. Merge preserved of Base and changed slices of variants.
    5.2. Check that variants do not interfere
    5.3. Derive the resulting dependency graph.
    5.4. Generate the SADG of the new version of software architecture description from the resulting SADG.

Our contribution in this paper is to develop steps (4) and (5) in order to merge software architecture. In the following we formalize these sub-steps.

6.2   Formalization

Given SADGs *SADG$_{Base}$*, *SADG$_A$*, and *SADG$_B$*, of Base, and variants A and B respectively. The algorithm performs three steps.

The first step identifies three subgraphs that represent the changed behavior of A with respect to Base(ΔA, Base), the changed behavior of B with respect to Base (ΔB, Base) and the preserved behavior that is the same in all architectures (PreA,B,Base) by using the set of vertices whose slices in SADGBase, SADGA, and SADGB are identical (i.e. . PPA,B,Base).

The second step unifies these subgraphs to form a merged dependence graph SADGM.

In the third step, a merged architecture GM is generated from graph SADGM.

6.2.1   Construction of a slice

First, we show how to compute an architecture slice. In this section we use the notation Component_name: inport/outport_name in order to represent components and connectors in an internal form. For example, Order_Req_Handler:I_order is the input port I_order of component Order_Req_Handler.

Each SADG is transformed in an internal form. The internal form is a set of triplets (a, b, c) which reflects the fact that there is an edge of type c from a to b. c can be an implicit invocation (ii), an internal action (ia) or an external communication (ec) while a and b are components and connectors using the previous notation.

For example (Ordering:I_order,Ordering:place_order,ia) means that: there is an internal action (ia) between in port I_order of component Ordering (Ordering:I_order) and out port place_order of Ordering (Ordering:place_order)

Table 1 represents a sample of internal form of Base SADG.

Arch itect ure	Internal Form
Base	((External_Source_Clerk, Ordering:I_order,ec), (Ordering:I_order,Ordering:place_order,ia), (Ordering:payment_req, Ordering:I_payment_req, ia), (Ordering:order_success, Ordering:I_ship_info,ia), (Ordering:order_fail, Ordering:Iorder_rej,ia), (Ordering: I_payment_req, Ordering:payment_req,ec), (Ordering: I_ship_info, sink1, ec), (Ordering:I_payment_req,Accounting:items,ii), (Ordering:place_order,Order_Entry:take_order, ii), ……)

Table 1: A sample of internal form of Base SADG.

Because of we are interested by static dependency analysis of SADG, we extract all slices starting from external source entry (e.g. clerck) to component that is in the front-end of the whole system until the end of the process (e.g. external sink). A static slice is a graph traversal by transitive closure from external source node to a final node from where we cannot continue the traversal (e.g. external sink).

In our example of Figure 1 there are more than fifteen slices that represent the complete behavior of EOPS. Table 2 represents one of them.

Slice	Internal form
	((External_Source_Clerk, Ordering:I_order), (Ordering:I_order,Ordering:place_order,ia), (Ordering:place_order,Order_Entry:take_order,ii), (Order_Entry:take_order, Order_Entry:ship_item,ia), (Order_Entry:ship_item, Inventory:find_item,ii), (Inventory:find_item, Inventory:get_next,ia), (Inventory:get_next, Order_Entry:next_item,ii), (Order_Entry:next_item, Order_Entry:done,ia), (Order_Entry:done, Ordering:payment_req,ii), (Ordering:payment_req, Ordering:Accounting_payment_req,ia), (Ordering, Accounting _payment_req, Accounting:cancel,ii), (Accounting:cancel, Accounting:fail,ia), (Accounting:fail, Ordering:order_fail,ii), (Ordering:order_fail, Ordering:I_order_rej, ia), (Ordering:I_order_rej, Sink2,ec))

Table 2: Internal form of a static slice.

Note that this slice reflects the behavior of canceling an order.

At the end of this step, each one (Base and variants) SADG's is transformed into a set of slices and the process of comparison can starts.

6.2.2   Changed slices

Let Δ$_{X, Base}$ the set of changed slices between variant X and Base. Changed slices are computed as the following:

$$AP_{A, Base} = \{v \in V(SADG_A) \mid (SADG_{Base}/v) \neq (SADG_A/v)\}$$

$$AP_{B, Base} = \{v \in V(SADG_B) \mid (SADG_{Base}/v) \neq (SADG_B/v)\}$$

$$\Delta_{A, Base} = b(SADG_A, AP_{A, Base})$$

$$\Delta_{B, Base} = b(SADG_B, AP_{B, Base}).$$

Where

V(SADG$_x$) denotes the set of vertices in SADG of variant X.

SADG$_{X}$/v is a vertex in the SADG of X from where we want to inspect its impact in the overall SADG of X.

b(SADG$_X$, AP$_{X, Base}$) is the set of peer changed slices in SADG$_{Base}$ and SADG$_X$.

In other words, internal forms of peer slices are compared. As a result they haven't the same graph traversal (different internal forms).

An example of changed slices is introduced in the section dedicated to application (6.3).

6.2.3   Preserved slices

Preserved architectural slices (*Pre$_{A, Base, B}$*) are computed as the following:

$$PP_{A, Base, B} = \{v \in V (SADG_{Base}) \mid (SADG_A/v) = (SADG_{Base}/v) = (SADG_B/v)\}.$$

$$Pre_{A, Base, B} = (SADG_{Base}, PP_{A, Base, B}).$$

The same graph traversal exists in both Base and variants.

We find an example of preserved slices in the section of application.

6.2.4   Forming the merged SADG

The merged graph G_M characterizes the SADG of the new version of the software architecture. G_M is computed as the following:

$$GM = \Delta_{A, Base} \cup \Delta_{B, Base} \cup Pre_{A, Base, B}$$

Informally, GM is composed of slices that are changed in SADG's of variants A and B with respect to Base and those that are unchanged.

6.3   Application

In this section we illustrate and validate the suggested merging approach through the running example of figure 1.

Starting from an initial software architecture description (Base) we introduce two independent requirement changes that are expected to be compatible. For this purpose two independent copies of Base are first created and modified concurrently (Variant A and Variant B). We will proceed as follows:

a.   Generate the SADG of Base, Variant A, and Variant B.
b.   Extract slices from these SADG's.
c.   Determine the set of changed slices and the set of preserved slices.
d.   Show that Variant A and Variant B do not interfere.
e.   Merge the set of changed slices and the set of preserved slices in order to get the SADG of the new version.
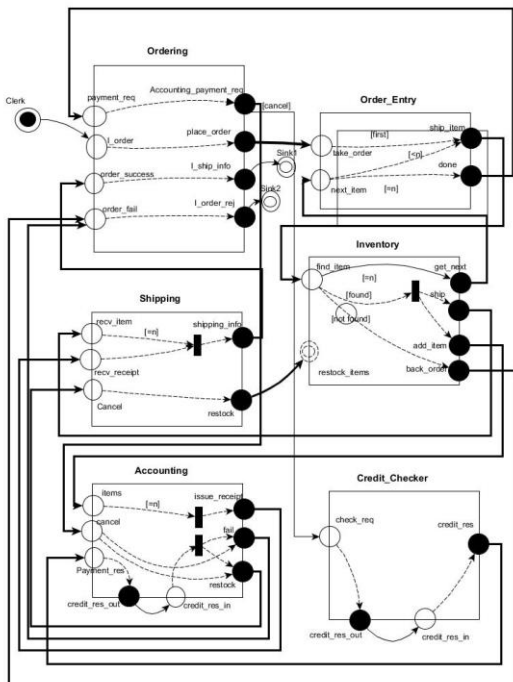
Figure 4: SADG of variant A.

6.4   Building SADG's of variants A and B

Two non-interfering variants are considered. In Variant A, a credit card payment option is added while in Variant B and in case stocks are empty at the order time, the request is handled through a back order mechanism.

6.4.1   Variant A SADG

In Variant A, Software Architect A inserts a new component that will take in charge the credit card payment option. This leads to the following changes in the architectural description of the software: (1) adding a new component (Credit_Checker), (2) creation of new connectors (from Ordering to Credit_Checker, and from Credit_Checker to Accounting), and (3) removing external connection (from Credit_res_out to Credit_res_in). Figure 4 represents the SADG of variant A.

6.4.2   Variant B SADG

In Variant B, Software Architect B inserts a new component that will take in charge the back order mechanism. This leads to the following changes in the architectural description of the software: (1) adding a new component (Back_order), (2) creation of new connectors (from Inventory to Back_order, from Back_order to Accounting, from Back_order to Shipping). Figure 5 depicts the SADG of variant B.
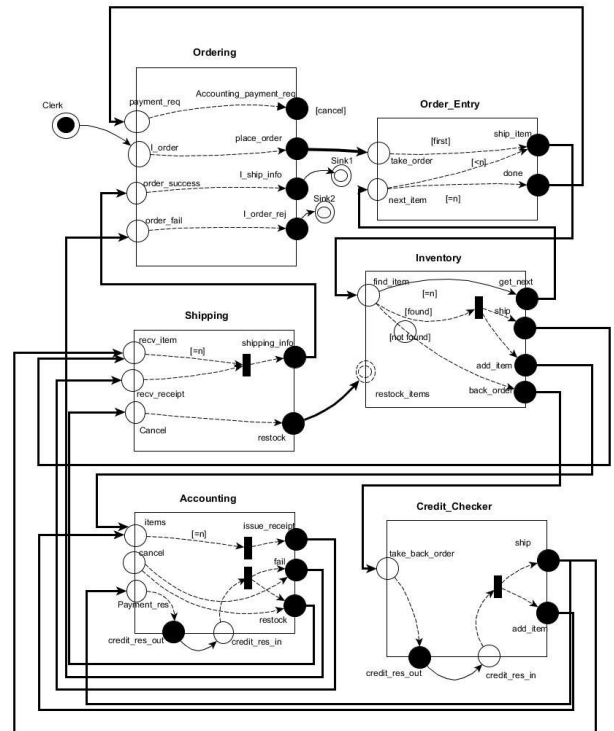
Figure 5: SADG of variant B.

6.5   Slice Extractions

Slice extraction process outcomes more than fifteen slices per SADG. For lack of space, only a pertinent sample of computed slices is presented in this paper. Selected sample involves an example of *changed slices*
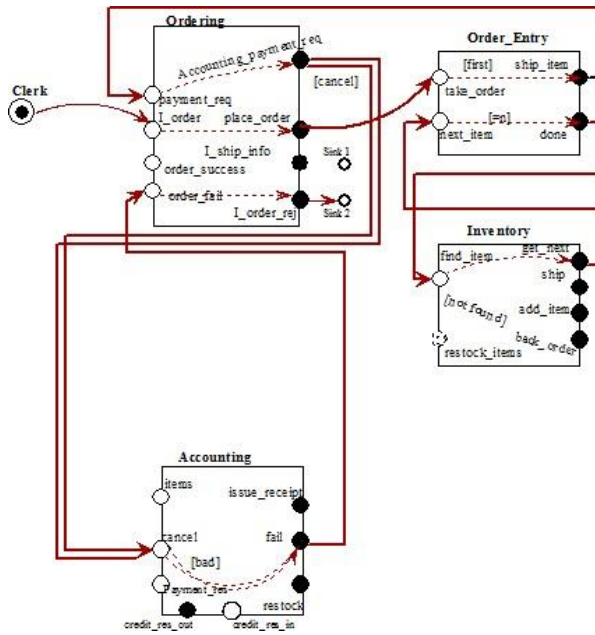
Figure 6: Slice of canceling order of Base.
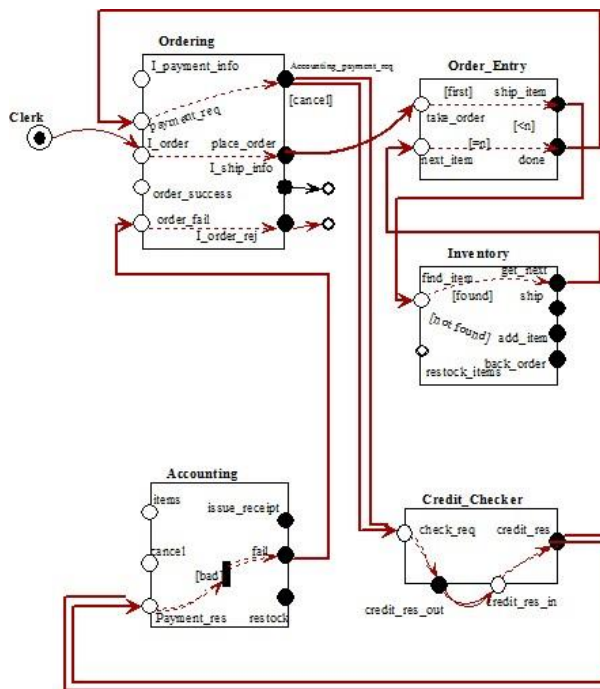
Figure 7: Slice of canceling order of Variant A.

.case ($\Delta_{A,\ Base} = b(SADG_A,\ AP_{A,\ Base})$ and an example of *preserved slices* case ($Pre_{A,\ Base,\ B} = (G_{Base},\ PP_{A,\ Base,\ B})$). These examples focus on the following two behaviors of interest: (1) slice traversals that leads to the canceling orders (payment unspecified and credit payment), and (2) slice traversal that leads to a successful ordering.

Figures 6 and 7 illustrate changed slice of canceling order in Base and Variant A respectively. Differences between these peer slices are depicted with double arrows in figures 6 and 7. In this case the slice of Variant A will belong to $\Delta A,\ Base$ set and is one of slices forming the SADG of the new version of Software architecture.
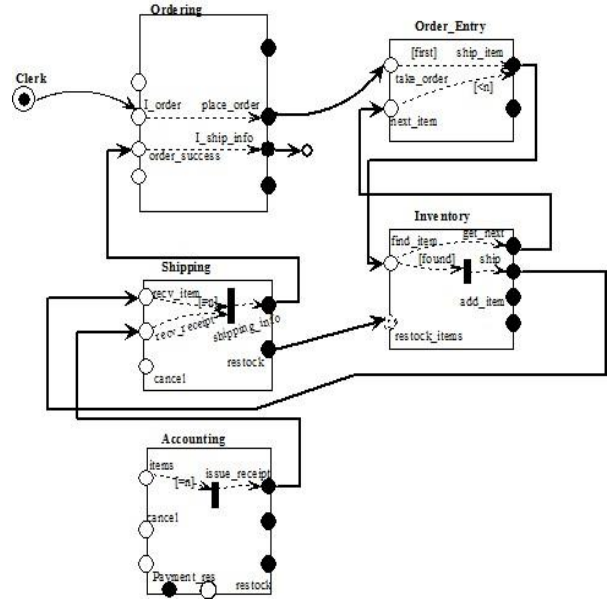
Figure 8: Slice of successful ordering in Base, Variants A and B.

Figure 8 reflects the same behavior in the three SADG. The graph traversal of successful ordering slice is the same in Base and variants.

Thus they will be classified in the category of preserved slices. They belong to:

$$PP_{A,\ Base,\ B} = \{v \in V\ (SADG_{Base}) \mid (SADG_A/v)$$

$$= (SADG_{Base}/v) = (SADG_B/v)\}$$

Intersection of changed slices between Base and Variant A and changed slices between Base and Variant B gives an empty set, consequently there is no interference between changes. We can continue the process.

6.6   Forming the merged SADG

This step involves forming a new SADG by using the result of previous steps. It consists of merging all changed architectural slices between SADG of Base and variants, and thus preserved in these SADGs. The union of changed and preserved slices forms the SADG of the new version of software architectural description.

$$GM = \Delta_{A,\ Base} \cup \Delta_{B,\ Base} \cup Pre_{A,\ Base,\ B}$$

Figure 9 depicts the SADG of the new version of software architectural description.

7   Conclusion

First, it is important to situate our works according to Horwitz's works.

The main contribution of Horwitz's work was to propose a new process of software evolution. This process was formalized and implemented at the program level. So Horwitz opened a new way for future research in evolution through the life cycle of software. This way involves mainly the facts (1) make explicit the
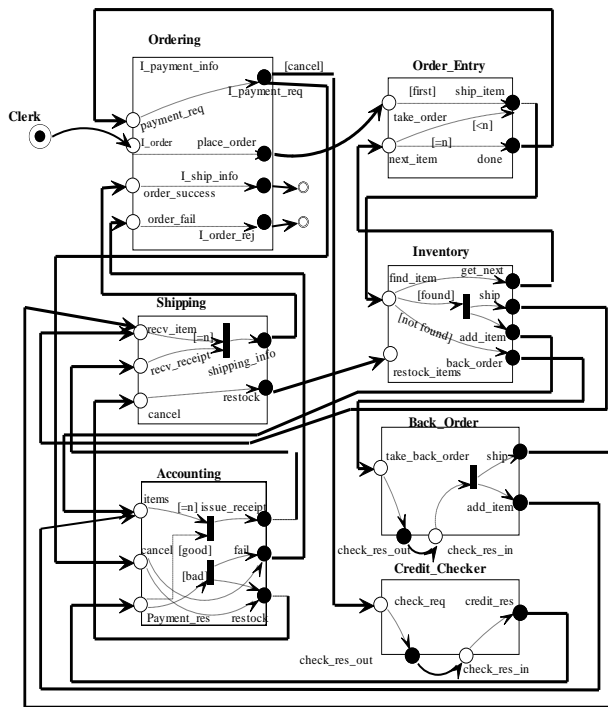
Figure 9: SADG of the new version.

dependencies between elements (e.g. data and control) which are usually implicit, (2) extract all behaviors (influence of one element over the other) for each version (Variants and Base), and (3) compare the behavior of each variant according to the Base program and finally form the new version that consists of the elements that remained preserved in all versions and those that have created differences in the variants.

Since, several studies have been made by exploiting this process. We also followed this process, but at the level of software architectures. We solved the problem of the similarity of graphs, ignored by Horwitz. We investigate and found the best way to represent the dependencies between elements of architectures that are different than those of programs. From there we followed the process. So, we showed that software evolution based merging at the level of software architecture is possible. Consequently this will lessen the cost of evolution.

Nowadays we continue in the theoretical aspects of this approach. Particularly, we are planning to investigate, consolidate and implementing this approach by involving conflicts. Another promising investigation consists of tackling the Software Architecture Merging where software architectures are described by well-known Architecture Description Languages (ADLs). Indeed in some cases architectures are provided in terms of ADLs. The question is "is it possible to merge architectures from ADLS or passing, first, by the graph transformation?"

References

[1]   Tom Mens (2008). "Introduction and Roadmap: History and Challenges of Software Evolution". Eds Tom Mens · Serge Demeyer, Springer-Verlag Berlin Heidelberg, pp. 1-14.

[2]   T. Mens (2002), "A State-of-the-Art Survey on Software Merging", IEEE Transactions on Software Engineering, vol 28, no 5, pp. 449–462.

[3]   D. Binkley, S. Horwitz, and T. Reps (1995). "Program Integration for Languages with Procedure Calls", ACM Transactions on Software Engineering and Methodology, vol 4, no 1, pp. 3-35.

[4]   T. Khammaci, and Z. Bouras (2002). "Versions of Program Integration", Handbook of Software Engineering and Knowledge Engineering, vol 2, World Scientific Publishing: Singapore, pp. 465-486.

[5]   T. Kim, Y. Song, and L. Chung (2000). "Software architecture analysis: a dynamic slicing approach", International Journal of Computer & Information Science, vol 1, no 2, pp. 91-103, 2000.

[6]   S. Horwitz, and T. Reps, "The use of dependence graph in software engineering", Proceedings of the 14th International on software engineering, Melbourne, Australia, 1992, pp. 392-411.

[7]   T. Apiwattanapong, A. Orso, and M. Harrold (2004), "A Differencing Algorithm for Object-oriented Programs", Automated Software Engineering, vol. 14, no 1, pp. 3-36.

[8]   S. Raghavan, R. Rohana, D. Leon, A. Podgurski, and V. Augustine (2004). "A semantic-graph differencing tool for studying changes in large code bases" Proceedings of 20th IEEE International Conference on Software Maintenance, pp. 188-197.

[9]   Z. Xing, and E. Stroulia (2005), "UMLDiff: An Algorithm for Object–Oriented Design Differencing", Proceedings of the 20th IEEE/ACM International Conference on Automated Software Engineering (ASE 2005), pp. 54-65.

[10] M. Abi-Antoun, J. Aldrich, N. Nahas, B. Schmerl and D. Garlan (2006). "Differencing and Merging of Architectural Views", proceedings of the 21st IEEE International Conference on Automated Software Engineering (ASE'06), pp. 47-58.

[11] Z. Bouras, M. Maouche (2015). "Merging software architectures with conflicts detections", International Journal of Information Systems and Change Management, Eds. Inderscience Publisher, Vol 7, No 3, pp. 242-260.

[12] K. Kobayashi, M. Kamimura; K. Yano; and K. Kato (2013). "SArF map: Visualizing software architecture from feature and layer viewpoints", in Proceedings of International Conference on Program Comprehension (ICPC'2013), San Fransisco USA, 2013, pp. 43 – 52.

[13] S. Maoz, J. Ringert, and B. Rumpe (2013). "Synthesis of Component and Connector Models from Crosscutting Structural Views", Proceedings of ACM SIGSOFT Symposium on the Foundations of Software Engineering (ESEC/FSE'13), Eds. B. Meyer, pp. 444-454.

[14] B. Westfechtel (2010). "A Formal Approach to Three-Way Merging of EMF Models", Proceedings of the 1st International Workshop on Model Comparison in Practice, IWMCP '10, Malaga, Spain, pp. 31-41.

[15] P. Brosch, M. Seidl, M. Wimmer and G. Kappel (2012). "Conflict Visualization for Evolving UML Models", Journal of Object Technology, vol 11, no 3, pp. 1–30.

[16] P. Langer, K. Wieland, M. Wimmer, and J. Cabot (2011), "From UML Profiles to EMF Profiles and Beyond", eds. TOOLS. LNCS, vol. 6705, Springer, Heidelberg, pp. 52–67.

[17] F. Rodrigues, and S. Barbosa(2006), **"**Component Identification Through Program. Slicing", Proceedings of the International Workshop on Formal Aspects of Component Software (FACS 2005), Electronic Notes in Theoretical Computer Science, pp. 291-304.

[18] J. Guo, Y. Liao, and R. Pamula (2006). "Static Analysis Based Software Architecture Recovery", Computational Science and Its Applications Lecture Notes in Computer Science, 3982, pp. 974-983.

[19] B. Li, Y. Zhou, Y. Wang, and J. Mo (2005). "Matrix-based component dependence representation and its applications in software quality assurance" SIGPLAN Notices, vol 40, no 11, pp. 29–36.

[20] J. Lalchandani (2009). "Static Slicing of UML Architectural Models", Journal of Object Technology, vol 8, no 1, pp. 159-188.

[21] J. Zhao (2000). "A Slicing-Based Approach to Extracting Reusable Software Architectures," Proceedings of the. 4th European Conference on Software Maintenance and Reengineering, IEEE Computer Society Press, Zurich, Switzerland . pp.215-223.

Dynamic Protocol for the Demand Management of Heterogeneous Resources with Convex Cost Functions

Jernej Zupančič
International Postgraduate School Jozef Stefan and Jozef Stefan Institute
Jamova cesta 39, 1000 Ljubljana
E-mail: jernej.zupancic@ijs.si

Matjaž Gams
Jozef Stefan Institute
Jamova cesta 39, 1000 Ljubljana
E-mail: matjaz.gams@ijs.si

Increasing demand of resources has driven research towards design of mechanisms that address resource-demand management, which usually focus on balancing peak demand and optimal device scheduling. While balancing homogeneous sources is well known, the goal of this paper is to balance the demand of heterogeneous resources with a convex cost function as well as to encourage consumers to reduce their demand. We propose a novel dynamic protocol achieving both goals using a serial cost-sharing pricing mechanism. The pricing-mechanism design is strategy proof and budget balanced, whereas the protocol does not require the consumers to reveal their utility functions, and supports hierarchical organization of consumers. We initially applied the proposed approach to a multi-agent system, where the consumers are smart house devices controlled by a corresponding agent, which in turns negotiates the electricity price with a smart city. Experimental results show that the proposed protocol successfully reduces the total consumption and lowers the peaks while all the consumers in equilibrium are given a fair price. We further provide experimental results indicating advantages of serial cost-sharing pricing and tariff pricing mechanisms over the average cost-pricing mechanism that is commonly used in the resource-demand management.

Povzetek: V prispevku predstavimo mehanizem za uravnoteženje porabe heterogenih virov, katerih cenovne funkcije so konveksne.

1 Introduction

The rising demand of resources such as electricity, water, natural gas and oil along with a limited amount of natural resources increases the costs of resource supply [1]. Consumers are typically priced the same per-unit price for a resource, regardless of whether they increase or decrease their consumption.

Mechanisms that stimulate the use of electricity when its generation is cheap and depreciate its consumption when its generation is expensive, have already been proposed and implemented in the energy sector [9].

Similar mechanisms can be implemented for any other resource [13], although this is rarely done, since electricity demand management is recognised as the main resource-demand management problem. The main reason for this is that it is inefficient to build new power generation plants in order to meet peak demands, while there are parts of the day, when there is barely any demand at all. While other reports have focused on shifting the load from one part of the day to another [9, 12], we propose a mechanism to encourage consumers to reduce their consumption in general.

A resource-demand management mechanism that sets the same price for every consumer in advance does not fully exploit demand-management capabilities. Dynamic-pricing mechanisms were proposed in [3], [10] and [4]. In those papers, the price for the next time unit changes dynamically and is the same for every consumer. Additionally, such mechanisms require the consumers either to report their utility function (which raises privacy concerns) or to leave the decision about consumption to the consumers themselves (which is not as efficient as it could be). A dynamic-pricing mechanism that negotiates prices with every individual was proposed in [2]. However, their approach assumes that the consumers will report their consumption truthfully, which is not always the case. Distributed real-time electricity pricing mechanism was introduced in [8], however, due to intense communications requirements between the suppliers and consumers it takes time to calculate the optimum solution. Truth incentive demand management was proposed in [5]. The mechanism described in [5] has several desirable properties besides be-

ing truth incentive: it is proved to converge to a Nash equilibrium, it is budget balanced (the total cost of resource provision equals the total cost the consumers pay), and it reaches the global optimum (minimal peak-to-average ratio). However, it still raises some issues: the resource consumption of every consumer is known to everyone, consumers are only encouraged to shift their load to different parts of the day and not to reduce consumption, smaller consumption does not result in smaller per-unit price, and real-time pricing requires price prediction capabilities.

Further, the only known general technique for designing truthful and approximately budget-balanced cost-sharing mechanisms with good efficiency or computational complexity properties is due to Moulin [6].

In some ways similar to our approach is the trading market for the smart electricity grid described in [11]. The authors proposed continuous double auction for the smart electricity grid. However, their mechanism is not scalable nor it is dynamic. Further, [14] addresses similar problem – dynamic resource-demand management and it uses serial cost sharing mechanism. However, no comparison with other mechanism were made.

To be successfully implemented in practice, a mechanism has to address all the above mentioned shortcomings. It has to be easily understood by consumers, it should change the prices dynamically and adapt them fairly to each individual according to the amount of resource consumed. Further, the mechanism should be truth incentive and should encourage lower resource consumption without asking the consumers to publicly reveal their demand or utility function. We propose a novel protocol that possesses those properties. It is based on the following three key ideas:

1. It ensures consumer satisfaction, by using a fair per-unit pricing of the resource;

2. It has built-in truth incentive and budget-balancing;

3. To preserve privacy, consumers do not have to reveal their utility functions.

The rest of the paper is structured as follows. In Section 2, we describe the problem domain to which our mechanism can be applied. In Section 3, we describe the proposed mechanism and its properties. Results from applying the mechanism to electricity demand management domain are presented in Section 4, while Section 5 concludes the paper with summary and discussion.

2 Problem domain

Our mechanism can be applied to a general resource-demand management problem (when there are many consumers that consume convexly priced resource), but for illustrative purposes we will make use of smart city domain to provide examples. The proposed protocol applies to a multi-agent system, where devices and appliances that

consume a resource in one consumer unit (e.g., a smart house) are controlled by one agent – house representative agent. House representative agents interact with the negotiator agent forming a multi-agent system (Figure 1).

Agents participating in the proposed protocol are defined in Definition 1.

Definition 1. *Let $\mathcal{A}$ be a set of all agents in the system.*

Let d_{ij} be a particular device j in a house i. Let $hr_i \in \mathcal{A}$ be a house representative agent that is responsible for coordinating all devices d_{ij} in the house. Therefore, house i consists of devices and a representative agent and is denoted as H_i.

Let $n_r \in \mathcal{A}$ be a negotiator agent that is responsible for fair distribution of resource r per-unit prices.

Every house representative agent interacts only with the resource negotiator agent. House representative agents do not interact among each other.
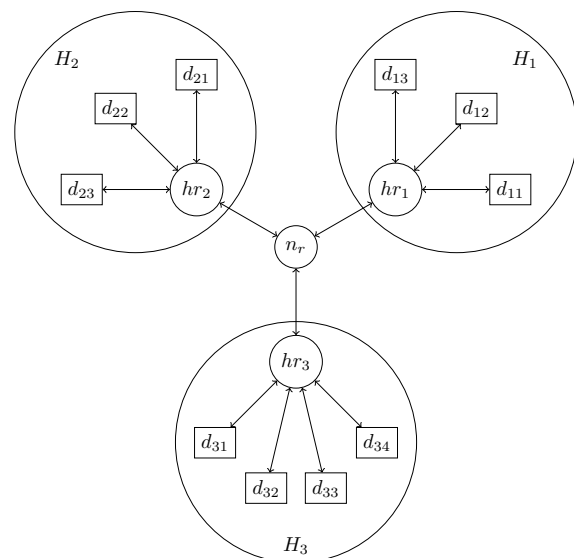
Figure 1: Multi-agent system example to which the protocol applies

We make the following assumptions for our problem domain.

House representative agents can control the devices in the house. House representative agents have the ability to switch the devices on and off. Further, agents can lower the power consumption of the device in finite number of discrete steps if gradual decrease of power consumption is supported by the device.

House representative agent strives to maximise consumer satisfaction. It is a primary goal of the house representative agent to consider users preferences, when negotiating with the resource negotiator. It is unacceptable for the inhabitant to pay a higher per-unit price for a resource than he was prepared to.

Cost function of a resource is convex, strictly increasing and non-constant. Examples of functions of this kind

are frequently used quadratic cost function and piecewise linear cost function [5].

3 The Protocol

The protocol defines a resource-demand management mechanism that enables the matching of resource demand with resource supply. Supply varies because resource provision usually depends on some independent variables (weather, fuel costs, taxes etc.). We assume that for one protocol run the supply and per-unit prices can be accurately defined in advance. Demand varies due to the supply and the consumer preferences. We assume that consumer preferences can be accurately defined in advance before each protocol run. Demand then depends only on the supply and using the proposed protocol the equilibrium can be reached.

The protocol is designed by the following requirements:

- negotiation finishes in a finite number of steps,

- consumers are motivated to reduce resource consumption by the protocol itself,

- the protocol is truth incentive,

- the pricing is fair,

- the total cost of resource provision equals the total cost the consumers pay, i.e., the protocol is budget-balanced.

According to those requirements and design the proposed protocol is a variant of a Moulin mechanism [6] - iterative fair sharing procedure.

3.1 Informal Protocol Description

Informal description of the protocol follows. First the smart house owners define the maximum per-unit price they are prepared to pay for every operating state of every device. For instance, smart house owner can have a dish washer, a washing machine and a heat pump installed in the house. For the dishwasher the owner is prepared to pay the per-unit price of 0.12 EUR/kWh, for the washing machine 0.10 EUR/kWh, for the heat pump at maximum power 0.30 EUR/kWh since keeping the home warm is more important than washing the clothes, and for the heat pump at half of the maximum power 1.0 EUR/kWh since in the case of high resource demand the per-unit price will be higher but the owner is prepared to pay the high per-unit price in order to maintain the border level of comfort. Other resources, e.g. water or gas can be added in the same way. By stating all the per-unit prices the user defines the strategy of her house representative agent and the agent can participate in the mechanism defined by our protocol on behalf of the house owner. Since the house representative agent resides in the house and does not share the information about

prices, the character of the home owner or her representative agent remains private information.

Our protocol is dynamic, meaning that it is repeated after some period of time has elapsed from the previous run. For some resources the per-unit price is expected to be more or less stable over time and does not change very often. Time elapsed between protocol runs in these cases can therefore be longer. However, per-unit prices of some resources can fluctuate rapidly. For instance, the per-unit price of electricity can change rapidly with the increasing use of renewable resources for its generation depends on the wind speed, solar radiation, water flow rate etc. Since those variables are hard to predict a day in advance, an hour or half an hour time intervals are assumed and the protocol must be able to support a large number of houses and perform all the calculations in a short period of time.

When a new time interval in which the decision about consumption has not been reached yet approaches, the negotiator agent triggers the protocol. First, it gathers the information about resource supply from the suppliers in order to construct a resource cost function. Then it asks every house representative agent it has connection to as to what is her preferred consumption. Since this is at the start of the protocol and no price has been set yet, every house representative agent responds with the highest required consumption. The negotiator agent then computes the per-unit prices individually for every house using some kind of a pricing mechanism and informs the house representative agents of the per-unit prices. It then again asks every house representative agent about their required consumption. The house representative agent can then either agree with the per-unit price and it responds with the same consumption as before or it can reduces the required consumption in the hope of a lower per-unit price. The proposed protocol guarantees that the per-unit price for lower consumption is either lower or in the worst case equal to the per-unit price of higher consumption. The cycle *ask for consumption - respond with consumption - compute per-unit prices* is repeated until every house representative agent is satisfied with the per-unit price and at that point the protocol run ends and an agreement is made stating that each house will consume the specified amount of the resource in the approaching time interval at the per-unit price that was set by the negotiator agent and agreed upon by the house representative agent.

When new time interval starts the house representative agents turns the devices and appliances in the home ON or OFF, depending on the agreements made. Every additional consumption of the resource, beyond the agreed one, is paid for at uniform maximal per-unit price that covers the costs of resource providers for additional amount of resource provision.

3.2 The Protocol Definition

During the run of the protocol many rounds take place with house representative agents reporting their desired con-

sumption to the negotiator agent and negotiator agent computing per-unit prices for every house representative agent.

The per-unit prices are computed individually for every consumer using Serial cost sharing mechanism [7]. The Algorithm 1 determines fair per-unit price for every consumer – lower consumption results in lower costs. Since the resource is convexly priced, it also results in lower per-unit pricing.

Algorithm 1: Serial cost sharing function

```
Serial_Cost_Sharing(f, c):
  N = length(c)
  sort c in increasing order
  p = []
  for i in range(N):
    p[i] = cost(i, f, c)/c[i]
  sort p in order to
    match original order of c

  return p
```

Algorithm 1 takes two inputs: resource cost function, denoted as f, and desired consumption vector, denoted as c, and returns a vector of per-unit prices p. Individual cost is computed by a function $cost(i, f, c)$ that takes as inputs the following parameters: consumer i, a sorted consumption vector c, and resource cost function f. Function $cost$ is defined recursively in Eq. (1).

$$cost(0, f, c) = 0$$
$$cost(i, f, c) =$$
$$\frac{f\left(\sum_{j=1}^{i-1} c[j] + (N+1-i) \cdot c[i]\right)}{N+1-i} -$$
$$- \frac{\sum_{j=1}^{i-1} cost(j, f, c)}{N+1-i} \qquad (1)$$

where N is the number of consumers participating in negotiation.

The cost function just needs to be convex, whereas any source, homogeneous or heterogeneous (water, gas, security and similar), can be added with the corresponding weights:

$$cost = \sum_{resource} \omega_{resource} \cdot cost_{resource}, \qquad (2)$$

where $resource$ is a specific resource type, $cost_{resource}$ is the cost of that resource type and $\omega_{resource}$ is the weight of that resource in the resource portfolio.

If any of the consumers does not agree with the per-unit price it receives and wants to further reduce its consumption anticipating a lower per-unit price, another round of negotiation takes place. In the following round, the desired consumption of individual consumer can be the same or lower. Negotiation stage ends when the demand is the same for every consumer in two consecutive rounds.

It is a requirement of the protocol that a consumer does not respond with strictly higher desired consumption than in the previous round.

3.3 The Protocol Properties

In this subsection we evaluate the proposed protocol according to the requirements stated at the beginning of the Section 3.

Lemma 1. *Assume there is a finite number of consumers and each consumer is in possession of finite number of devices or appliances and every device or appliance has finite number of resource consumption levels. Then the protocol finishes in a finite number of steps.*

Proof 1. *Assume that the protocol does not finish in finite number of steps. That means that consumption vector (a vector of consumptions of all the consumers) is different in each two consecutive rounds. That means that in every round at least one of the consumers reported strictly lower consumption than before. Since we assumed infinite number of steps at least one consumer can strictly lower her consumption infinite times. This contradicts the premise from the lemma. Therefore, our assumption is wrong and the protocol finishes in finite number of steps.* □

Definition 2. *Let f be a resource cost function, so that $f(x)$ is the total cost for the provision of x amount of resource. Let cost be a pricing mechanism used to determine the cost for every resource consumer. Let $\mathbf{c}$ be a consumption vector, denoting consumptions for all the N-consumers participating in the mechanism. The pricing mechanism is* budget-balanced *iff*

$$f\left(\sum_{i=1}^{N} c_i\right) = \sum_{i=1}^{N} cost(c_i).$$

The following two lemmas are a corollary of [7].

Lemma 2. *The pricing mechanism used in the proposed protocol is budget-balanced.*

Proof 2. *Let N be a number of consumers, let c be a consumption vector where c_i-s are increasingly ordered, let f be a resource cost function and let the cost be a serial cost sharing mechanism that is used in the proposed protocol. The costs of the largest consumer can be computed as*

$$cost(N, f, c) =$$

$$\frac{f\left(\sum_{j=1}^{N-1} c[j] + (N+1-N) \cdot c[N]\right)}{N+1-N} -$$

$$- \frac{\sum_{j=1}^{N-1} cost(j, f, c)}{N+1-N}$$

$$= f\left(\sum_{j=1}^{N} c[j]\right) - \sum_{j=1}^{N-1} cost(j, f, c).$$

Therefore,

$$\sum_{j=1}^{N} cost(j, f, c) = f\left(\sum_{j=1}^{N} c[j]\right).$$

$\square$

Definition 3. *Let per-unit price for the consumption x and using pricing mechanism p be denoted by $p(x)$. Let consumers A and B consume a_A and a_B amount of resource, respectively. A pricing mechanism p is* fair *iff for each a_A and a_B so that $a_A \leq a_B$, $p(a_A) \leq p(a_B)$ also applies.*

Lemma 3. *The pricing used by the proposed protocol is fair.*

Proof 3 (sketch). *Since resource cost function is convex and increasing, its derivative is also increasing. Since the derivative of the cost function represents the marginal price (per-unit price) of the resource provision, the per-unit price of the resource is higher when more of it is used. Serial cost sharing mechanism divides the cost in such a way that for the same amount of resource the consumers pay the same per-unit price and any additional consumption is paid at a different price. Since the per-unit price of additional consumed resource is higher, due to the increasing and convex cost function, so is the final per-unit price of a larger consumer. Therefore if $a_A \leq a_B$ then $p(a_A) \leq p(a_B)$.* $\square$

Definition 4. *Let the pairs $\{(c_i, p_i)\}_i$ represent the (consumption, price) pairs by which the consumer can state his preferences: for the consumption of c_i amount of resource the consumer is prepared to pay p_i per-unit price. Let $A = (c_A, p_A)$ and $B = (c_B, p_B)$ be two possible outcomes of the protocol. The consumer preferres A over B iff*

1. *$c_A > c_B$ and $p_A \leq p_i$ where $c_i = c_A$, or*

2. *$c_A = c_B$ and $p_A < p_B \leq p_i$ where $c_i = c_A$.*

In other words, the consumer prefers to consume the maximum possible amount of resource she can afford. In this case the following lemma applies.

Lemma 4. *Assume the definition of preference as in Definition 4. Further, assume the consumers are infinitely risk-averse. Then the proposed protocol is truth incentive and strategy-proof.*

Proof 4 (sketch). *During the protocol the consumer can strategize by responding with the same level of consumption in two consecutive rounds of the protocol although she can not afford the per-unit price (according to her real preferences). That is because the per-unit price of the resource can lower when all other consumers lower their consumption. If the per-unit price lowers enough the per-unit price could become affordable for the strategic consumer. However, since no consumer has the full knowledge of all the consumers preferences or the resource prices, there is a positive probability that the per-unit price will not become affordable. If the consumer is infinitely risk-averse, she wants to avoid the worst possible results and therefore, she will not strategize. In every round the consumer will respond according to her preferences, which will guarantee a good outcome for her. Therefore, the proposed protocol is truth incentive and strategy-proof.* $\square$

4 Experimental Results

Application of the protocol in electrical energy consumption domain is presented in this section. We carried out the experiment to examine the amount of electrical energy consumed and the average per-unit price of the electrical energy when using the proposed protocol.

4.1 Model description

While there are many variants for house representative agent implementation, the important properties of the house representative agent are:

1. It can reason about desired resource consumption in every round of the protocol, and

2. it can decide whether the offered per-unit price is satisfiable for the house.

In our experiment house representative agent possess the information about the electrical energy consumption of the devices and the information about consumers settings – maximal per-unit price for operating each device. This information is private to each house representative agent and sent neither to other house representative agents nor the resource negotiator.

The parameters for each house in our experiments were the following:

– Number of consumption levels: uniformly drawn integer number from the interval $[3, 15]$,

– Consumption level size: uniformly drawn real value from the interval $[0.1\mathrm{kWh}, 1.5\mathrm{kWh}]$,

– Acceptable per-unit price for every consumption level: drawn from normal distribution with mean $0.15\mathrm{EUR/kWh}$ and standard deviation of $0.055\mathrm{EUR/kWh}$.

A convex, strictly increasing piece-wise linear cost function was used for electrical energy pricing (suggested in [5]). It can be represented by a list of tuples (B_i, pup_i), where B_i is the amount of energy available for the per-unit price of pup_i. To obtain a convex function, we assume that lower priced energy is supplied first.

The parameters for cost function were the following:

– Number of tuples (B_i, pup_i): 50 tuples were used;

– Block sizes: on average each block was four time the size of total minimal level consumption;

– Block per-unit prices: drawn from normal distribution with mean 0.3EUR/kWh and standard deviation of 0.055EUR/kWh.

4.2 Electricity Consumption Experiment

In order to make a comparison of the serial cost sharing mechanism used in our protocol to standard cost sharing mechanism, the average cost sharing mechanism was also implemented [5].

Definition 5. *Let f be a resource cost function and let c be a consumption vector. The average cost sharing mechanism for consumer i is defined as:*

$$averageCost(i, f, c) = \frac{f\left(\sum_{k=1}^{N} c\,[k]\right)}{\sum_{k=1}^{N} c\,[k]} \cdot c\,[i]\,.$$

The protocol was run 2000 times with serial cost sharing mechanism and average cost sharing mechanism for every number of houses we specified – specifically 100, 500, 1000, 1500, 2000. The main variables we observed were the percentage of total possible consumption that was realized after the protocol run and the average per-unit price the consumers have to pay.

Since there was a large difference between the two pricing mechanisms a third mechanism was implemented that combines the average and serial cost sharing mechanisms.

Definition 6. *Let T be a number of possible different per-unit prices of the resource. Let c be a consumption vector, where consumptions are orderer from smallest to largest, and let N be a number of consumers. Now group the consumers into T equal sized groups where in t-th group there are all consumers with index from $(t-1)\cdot\left\lfloor\frac{N}{T}\right\rfloor+1$ to $t\cdot\left\lfloor\frac{N}{T}\right\rfloor$. Total consumption for every group can be computed and new consumption vector for groups can be assembled*

$$c_T =$$

$$\left(\sum_{i=1}^{\left\lfloor\frac{N}{T}\right\rfloor} c\,[i], \sum_{i=\left\lfloor\frac{N}{T}\right\rfloor+1}^{2\cdot\left\lfloor\frac{N}{T}\right\rfloor} c\,[i], \ldots, \sum_{i=(T-1)\cdot\left\lfloor\frac{N}{T}\right\rfloor+1}^{N} c\,[i]\right).$$

For each group the costs can be calculated using the serial cost sharing mechanism and a cost vector can be obtained $cost_T = (cost_1, cost_2, \ldots, cost_T)$. The per-unit price for consumer i in group t can be calculated using the average cost sharing mechanism

$$price_i = \frac{cost_t}{\sum_{k=(t-1)\cdot\left\lfloor\frac{N}{T}\right\rfloor+1}^{t\cdot\left\lfloor\frac{N}{T}\right\rfloor} c\,[k]} \cdot c\,[i]\,.$$

We call this mechanism a tariff pricing mechanism.

We have made the same experiments as before using the additional pricing mechanism, where we grouped the consumer into two equal sized groups. The final results are
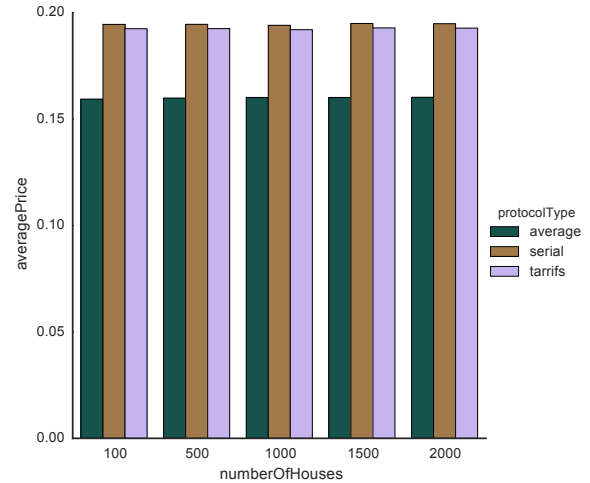
Figure 2: Bar plot with number of houses and average per-unit prices on axes
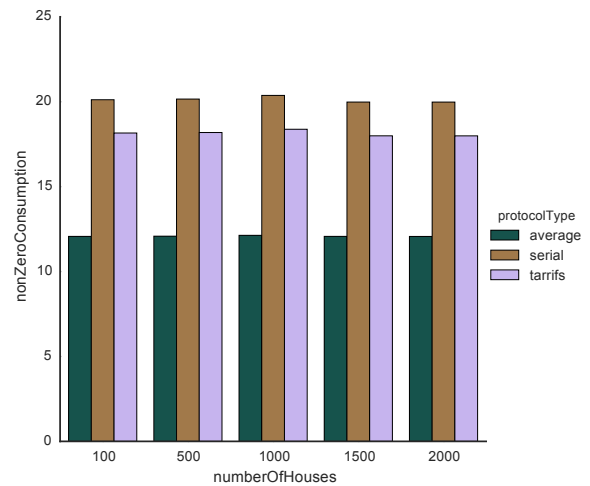
Figure 3: Bar plot with number of houses and percentage of possible total consumption on axes

presented in Figures 2, 3 and 4. Every point or a bar represents the average of 2000 runs of the protocol.

From the experiments we concluded that a number of houses has a negligible effect on the observed variables when model like ours is used. The largest effect has the type of the protocol used: protocol with serial or average cost sharing mechanism. Since the consumers are different and some are prepared to pay more for the same amount of resource the consumption increases when using the protocol with serial cost sharing. It is the advantage of the serial cost sharing to produce personalized per-unit prices according to the amount of resource used. For that reason also, the average per-unit price increases when using serial cost sharing. However, every consumer still receives the amount of the resource she is prepared to pay for. And according to definition 4 maximal consumption within the limits is the preferred result. Therefore, serial cost sharing
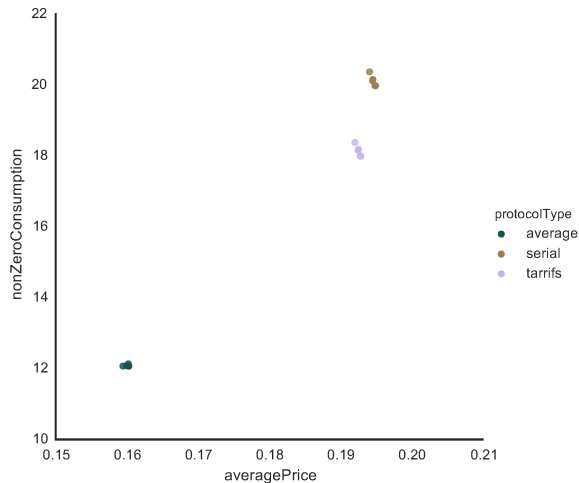
Figure 4: Scatter plot with average per-unit prices and percentage of possible total consumption on axes

mechanism is the preferred mechanism.

By using only two groups the results of the protocol using tariff pricing are much more similar to the protocol using serial cost sharing than the average pricing. Consumption and average per-unit price increase due to the flexible pricing mechanism used.

5   Discussion & Conclusion

We have proposed a truth incentive mechanism that encourages the reduction of consumption of convexly priced resources and can be easily understood by consumers. Further, the mechanism scales linearly with the number of consumers, since the per-unit prices can be easily computed.

The novel protocol ensures consumer satisfaction when the consumer is truthful and deals also with heterogeneous resources. But there are also some problems associated with our approach. First, since one can not predict the consumption of others and does not know the exact resource cost function, one can not predict the possible outcome of the protocol. Therefore, it is not rational to agree to a higher per-unit price than one is willing to pay. Second, the algorithms that deals with homogeneous resources [6] find optimal solutions and have several desired properties. Our algorithm meets some of the desired requirements but needs several rounds to achieve that, however the number of cycles is still less than in general Moulin mechanisms.

Since serial cost sharing was used the proposed mechanism is budget-balanced. The stopping of the protocol was discussed in Subsection 3.3. However, one of the main assumptions used to prove the stopping was that after every round of the negotiation the consumer does not demand strictly higher desired consumption. Although not addressed in the proposed version of the protocol, additional mechanism could be implemented to release this assumption and still maintain the the stopping property.

Further work will include the analysis of the consumer behaviour when the protocol is applied repeatedly with the same consumers and the same resource negotiator. Additionally, other agent organization structures will be considered in order to enable the protocol implementation in real-life scenarios.

The proposed protocol demonstrates various desired properties beside being truth incentive and scalable. Due to its efficient resource reduction capabilities it is considered to be applicable to the sustainable smart city domain.

6   Acknowledgements

References

[1] Edward B Barbier. *Economics, natural-resource scarcity and development: conventional and alternative views.* Routledge, 2013.

[2] Frances Brazier, Frank Cornelissen, Rune Gustavsson, Catholijn M Jonker, Olle Lindeberg, Bianca Polak, and Jan Treur. Agents negotiating for load balancing of electricity use. In *Distributed Computing Systems, 1998. Proceedings. 18th International Conference on*, pages 622–629. IEEE, 1998.

[3] Ahmad Faruqui and Sanem Sergici. Household response to dynamic pricing of electricity: a survey of 15 experiments. *Journal of Regulatory Economics*, 38(2):193–225, 2010.

[4] Koen Kok. Multi-agent coordination in the electricity grid, from concept towards market introduction. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Industry track*, pages 1681–1688. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[5] A-H Mohsenian-Rad, Vincent WS Wong, Juri Jatskevich, Robert Schober, and Alberto Leon-Garcia. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. *Smart Grid, IEEE Transactions on*, 1(3):320–331, 2010.

[6] Hervé Moulin. Incremental cost sharing: Characterization by coalition strategy-proofness. *Social Choice and Welfare*, 16(2):279–320, 1999.

[7] Herve Moulin and Scott Shenker. Serial cost sharing. *Econometrica: Journal of the Econometric Society*, pages 1009–1037, 1992.

[8] Toru Namerikawa, Norio Okubo, Ryutaro Sato, Yoshihiro Okawa, and Masahiro Ono. Real-time pricing mechanism for electricity market with built-in incentive for participation. *IEEE Transactions on Smart Grid*, 6(6):2714–2724, 2015.

[9] Sarvapali D Ramchurn, Perukrishnen Vytelingum, Alex Rogers, and Nick Jennings. Agent-based control for decentralised demand side management in the smart grid. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 5–12. International Foundation for Autonomous Agents and Multiagent Systems, 2011.

[10] Pedram Samadi, Hamed Mohsenian-Rad, Robert Schober, and Vincent WS Wong. Advanced demand side management for the future smart grid using mechanism design. *Smart Grid, IEEE Transactions on*, 3(3):1170–1180, 2012.

[11] Perukrishnen Vytelingum, Sarvapali D Ramchurn, Thomas D Voice, Alex Rogers, and Nicholas R Jennings. Trading agents for the smart electricity grid. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pages 897–904. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[12] Perukrishnen Vytelingum, Thomas D Voice, Sarvapali D Ramchurn, Alex Rogers, and Nicholas R Jennings. Agent-based micro-storage management for the smart grid. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pages 39–46. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[13] Fredrik Wernstedt, Paul Davidsson, and Christian Johansson. Demand side management in district heating systems. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, page 272. ACM, 2007.

[14] Jernej Zupančič, Damjan Kužnar, Boštjan Kaluža, and Matjaž Gams. Two-stage negotiation protocol for lowering the consumption of convexly priced resources. In *Proceedings of the 2014 Workshop on Intelligent Agents and Technologies for Socially Interconnected Systems*, page 5. ACM, 2014.

JOŽEF STEFAN INSTITUTE

Jožef Stefan (1835-1893) was one of the most prominent physicists of the 19th century. Born to Slovene parents, he obtained his Ph.D. at Vienna University, where he was later Director of the Physics Institute, Vice-President of the Vienna Academy of Sciences and a member of several scientific institutions in Europe. Stefan explored many areas in hydrodynamics, optics, acoustics, electricity, magnetism and the kinetic theory of gases. Among other things, he originated the law that the total radiation from a black body is proportional to the 4th power of its absolute temperature, known as the Stefan–Boltzmann law.

The Jožef Stefan Institute (JSI) is the leading independent scientific research institution in Slovenia, covering a broad spectrum of fundamental and applied research in the fields of physics, chemistry and biochemistry, electronics and information science, nuclear science technology, energy research and environmental science.

The Jožef Stefan Institute (JSI) is a research organisation for pure and applied research in the natural sciences and technology. Both are closely interconnected in research departments composed of different task teams. Emphasis in basic research is given to the development and education of young scientists, while applied research and development serve for the transfer of advanced knowledge, contributing to the development of the national economy and society in general.

At present the Institute, with a total of about 900 staff, has 700 researchers, about 250 of whom are postgraduates, around 500 of whom have doctorates (Ph.D.), and around 200 of whom have permanent professorships or temporary teaching assignments at the Universities.

In view of its activities and status, the JSI plays the role of a national institute, complementing the role of the universities and bridging the gap between basic science and applications.

Research at the JSI includes the following major fields: physics; chemistry; electronics, informatics and computer sciences; biochemistry; ecology; reactor technology; applied mathematics. Most of the activities are more or less closely connected to information sciences, in particular computer sciences, artificial intelligence, language and speech technologies, computer-aided design, computer architectures, biocybernetics and robotics, computer automation and control, professional electronics, digital communications and networks, and applied mathematics.

The Institute is located in Ljubljana, the capital of the independent state of **Slove**nia (or S♡nia). The capital today is considered a crossroad between East, West and Mediterranean Europe, offering excellent productive capabilities and solid business opportunities, with strong international connections. Ljubljana is connected to important centers such as Prague, Budapest, Vienna, Zagreb, Milan, Rome, Monaco, Nice, Bern and Munich, all within a radius of 600 km.

From the Jožef Stefan Institute, the Technology park "Ljubljana" has been proposed as part of the national strategy for technological development to foster synergies between research and industry, to promote joint ventures between university bodies, research institutes and innovative industry, to act as an incubator for high-tech initiatives and to accelerate the development cycle of innovative products.

Part of the Institute was reorganized into several high-tech units supported by and connected within the Technology park at the Jožef Stefan Institute, established as the beginning of a regional Technology park "Ljubljana". The project was developed at a particularly historical moment, characterized by the process of state reorganisation, privatisation and private initiative. The national Technology Park is a shareholding company hosting an independent venture-capital institution.

The promoters and operational entities of the project are the Republic of Slovenia, Ministry of Higher Education, Science and Technology and the Jožef Stefan Institute. The framework of the operation also includes the University of Ljubljana, the National Institute of Chemistry, the Institute for Electronics and Vacuum Technology and the Institute for Materials and Construction Research among others. In addition, the project is supported by the Ministry of the Economy, the National Chamber of Economy and the City of Ljubljana.

Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
Tel.:+386 1 4773 900, Fax.:+386 1 251 93 85
WWW: http://www.ijs.si
E-mail: matjaz.gams@ijs.si
Public relations: Polona Strnad

INFORMATICA

AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS

INVITATION, COOPERATION

Submissions and Refereeing

Please register as an author and submit a manuscript at: http://www.informatica.si. At least two referees outside the author's country will examine it, and they are invited to make as many remarks as possible from typing errors to global philosophical disagreements. The chosen editor will send the author the obtained reviews. If the paper is accepted, the editor will also send an email to the managing editor. The executive board will inform the author that the paper has been accepted, and the author will send the paper to the managing editor. The paper will be published within one year of receipt of email with the text in Informatica MS Word format or Informatica LaTeX format and figures in .eps format. Style and examples of papers can be obtained from http://www.informatica.si. Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the managing editor.

SUBSCRIPTION

Please, complete the order form and send it to Dr. Drago Torkar, Informatica, Institut Jožef Stefan, Jamova 39, 1000 Ljubljana, Slovenia. E-mail: drago.torkar@ijs.si

Since 1977, Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automation and other related areas. In its 16th year (more than twentythree years ago) it became truly international, although it still remains connected to Central Europe. The basic aim of Informatica is to impose intellectual values (science, engineering) in a distributed organisation.

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board.

Informatica web edition is free of charge and accessible at http://www.informatica.si.

Informatica print edition is free of charge for major scientific, educational and governmental institutions. Others should subscribe.

Informatica WWW:

http://www.informatica.si/

Referees from 2008 on:

A. Abraham, S. Abraham, R. Accornero, A. Adhikari, R. Ahmad, G. Alvarez, N. Anciaux, R. Arora, I. Awan, J. Azimi, C. Badica, Z. Balogh, S. Banerjee, G. Barbier, A. Baruzzo, B. Batagelj, T. Beaubouef, N. Beaulieu, M. ter Beek, P. Bellavista, K. Bilal, S. Bishop, J. Bodlaj, M. Bohanec, D. Bolme, Z. Bonikowski, B. Bošković, M. Botta, P. Brazdil, J. Brest, J. Brichau, A. Brodnik, D. Brown, I. Bruha, M. Bruynooghe, W. Buntine, D.D. Burdescu, J. Buys, X. Cai, Y. Cai, J.C. Cano, T. Cao, J.-V. Capella-Hernández, N. Carver, M. Cavazza, R. Ceylan, A. Chebotko, I. Chekalov, J. Chen, L.-M. Cheng, G. Chiola, Y.-C. Chiou, I. Chorbev, S.R. Choudhary, S.S.M. Chow, K.R. Chowdhury, V. Christlein, W. Chu, L. Chung, M. Ciglarič, J.-N. Colin, V. Cortellessa, J. Cui, P. Cui, Z. Cui, D. Cutting, A. Cuzzocrea, V. Cvjetkovic, J. Cypryjanski, L. Čehovin, D. Čerepnalkoski, I. Čosić, G. Daniele, G. Danoy, M. Dash, S. Datt, A. Datta, M.-Y. Day, F. Debili, C.J. Debono, J. Dedič, P. Degano, A. Dekdouk, H. Demirel, B. Demoen, S. Dendamrongvit, T. Deng, A. Derezinska, J. Dezert, G. Dias, I. Dimitrovski, S. Dobrišek, Q. Dou, J. Doumen, E. Dovgan, B. Dragovich, D. Drajic, O. Drbohlav, M. Drole, J. Dujmović, O. Ebers, J. Eder, S. Elaluf-Calderwood, E. Engström, U. riza Erturk, A. Farago, C. Fei, L. Feng, Y.X. Feng, B. Filipič, I. Fister, I. Fister Jr., D. Fišer, A. Flores, V.A. Fomichov, S. Forli, A. Freitas, J. Fridrich, S. Friedman, C. Fu, X. Fu, T. Fujimoto, G. Fung, S. Gabrielli, D. Galindo, A. Gambarara, M. Gams, M. Ganzha, J. Garbajosa, R. Gennari, G. Georgeson, N. Gligorić, S. Goel, G.H. Gonnet, D.S. Goodsell, S. Gordillo, J. Gore, M. Grčar, M. Grgurović, D. Grosse, Z.-H. Guan, D. Gubiani, M. Guid, C. Guo, B. Gupta, M. Gusev, M. Hahsler, Z. Haiping, A. Hameed, C. Hamzaçebi, Q.-L. Han, H. Hanping, T. Härder, J.N. Hatzopoulos, S. Hazelhurst, K. Hempstalk, J.M.G. Hidalgo, J. Hodgson, M. Holbl, M.P. Hong, G. Howells, M. Hu, J. Hyvärinen, D. Ienco, B. Ionescu, R. Irfan, N. Jaisankar, D. Jakobović, K. Jassem, I. Jawhar, Y. Jia, T. Jin, I. Jureta, Đ. Juričić, S. K, S. Kalajdziski, Y. Kalantidis, B. Kaluža, D. Kanellopoulos, R. Kapoor, D. Karapetyan, A. Kassler, D.S. Katz, A. Kaveh, S.U. Khan, M. Khattak, V. Khomenko, E.S. Khorasani, I. Kitanovski, D. Kocev, J. Kocijan, J. Kollár, A. Kontostathis, P. Korošec, A. Koschmider, D. Košir, J. Kovač, A. Krajnc, M. Krevs, J. Krogstie, P. Krsek, M. Kubat, M. Kukar, A. Kulis, A.P.S. Kumar, H. Kwaśnicka, W.K. Lai, C.-S. Laih, K.-Y. Lam, N. Landwehr, J. Lanir, A. Lavrov, M. Layouni, G. Leban, A. Lee, Y.-C. Lee, U. Legat, A. Leonardis, G. Li, G.-Z. Li, J. Li, X. Li, X. Li, Y. Li, Y. Li, S. Lian, L. Liao, C. Lim, J.-C. Lin, H. Liu, J. Liu, P. Liu, X. Liu, X. Liu, F. Logist, S. Loskovska, H. Lu, Z. Lu, X. Luo, M. Luštrek, I.V. Lyustig, S.A. Madani, M. Mahoney, S.U.R. Malik, Y. Marinakis, D. Marinčič, J. Marques-Silva, A. Martin, D. Marwede, M. Matijašević, T. Matsui, L. McMillan, A. McPherson, A. McPherson, Z. Meng, M.C. Mihaescu, V. Milea, N. Min-Allah, E. Minisci, V. Mišić, A.-H. Mogos, P. Mohapatra, D.D. Monica, A. Montanari, A. Moroni, J. Mosegaard, M. Moškon, L. de M. Mourelle, H. Moustafa, M. Možina, M. Mrak, Y. Mu, J. Mula, D. Nagamalai, M. Di Natale, A. Navarra, P. Navrat, N. Nedjah, R. Nejabati, W. Ng, Z. Ni, E.S. Nielsen, O. Nouali, F. Novak, B. Novikov, P. Nurmi, D. Obrul, B. Oliboni, X. Pan, M. Pančur, W. Pang, G. Papa, M. Paprzycki, M. Paralič, B.-K. Park, P. Patel, T.B. Pedersen, Z. Peng, R.G. Pensa, J. Perš, D. Petcu, B. Petelin, M. Petkovšek, D. Pevec, M. Pičulin, R. Piltaver, E. Pirogova, V. Podpečan, M. Polo, V. Pomponiu, E. Popescu, D. Poshyvanyk, B. Potočnik, R.J. Povinelli, S.R.M. Prasanna, K. Pripužić, G. Puppis, H. Qian, Y. Qian, L. Qiao, C. Qin, J. Que, J.-J. Quisquater, C. Rafe, S. Rahimi, V. Rajkovič, D. Raković, J. Ramaekers, J. Ramon, R. Ravnik, Y. Reddy, W. Reimche, H. Rezankova, D. Rispoli, B. Ristevski, B. Robič, J.A. Rodriguez-Aguilar, P. Rohatgi, W. Rossak, I. Rožanc, J. Rupnik, S.B. Sadkhan, K. Saeed, M. Saeki, K.S.M. Sahari, C. Sakharwade, E. Sakkopoulos, P. Sala, M.H. Samadzadeh, J.S. Sandhu, P. Scaglioso, V. Schau, W. Schempp, J. Seberry, A. Senanayake, M. Senobari, T.C. Seong, S. Shamala, c. shi, Z. Shi, L. Shiguo, N. Shilov, Z.-E.H. Slimane, F. Smith, H. Sneed, P. Sokolowski, T. Song, A. Soppera, A. Sorniotti, M. Stajdohar, L. Stanescu, D. Strnad, X. Sun, L. Šajn, R. Šenkeřík, M.R. Šikonja, J. Šilc, I. Škrjanc, T. Štajner, B. Šter, V. Štruc, H. Takizawa, C. Talcott, N. Tomasev, D. Torkar, S. Torrente, M. Trampuš, C. Tranoris, K. Trojacanec, M. Tschierschke, F. De Turck, J. Twycross, N. Tziritas, W. Vanhoof, P. Vateekul, L.A. Vese, A. Visconti, B. Vlaovič, V. Vojisavljević, M. Vozalis, P. Vračar, V. Vranić, C.-H. Wang, H. Wang, H. Wang, H. Wang, S. Wang, X.-F. Wang, X. Wang, Y. Wang, A. Wasilewska, S. Wenzel, V. Wickramasinghe, J. Wong, S. Wrobel, K. Wrona, B. Wu, L. Xiang, Y. Xiang, D. Xiao, F. Xie, L. Xie, Z. Xing, H. Yang, X. Yang, N.Y. Yen, C. Yong-Sheng, J.J. You, G. Yu, X. Zabulis, A. Zainal, A. Zamuda, M. Zand, Z. Zhang, Z. Zhao, D. Zheng, J. Zheng, X. Zheng, Z.-H. Zhou, F. Zhuang, A. Zimmermann, M.J. Zuo, B. Zupan, M. Zuqiang, B. Žalik, J. Žižka,

Informatica

An International Journal of Computing and Informatics

Web edition of Informatica may be accessed at: http://www.informatica.si.

Informatica

An International Journal of Computing and Informatics