*Sensors and Materials*, Vol. 34, No. 6 (2022) 2391–2401
MYU Tokyo



S & M 2979

# Deep-learning-based Intrusion Detection with Enhanced Preprocesses

Chia-Ju Lin,[1] Yueh-Min Huang,[1] and Ruey-Maw Chen[2*]

[1]Department of Engineering Science, National Cheng Kung University,
No. 1, University Road, Tainan City 701, Taiwan (R.O.C.)
[2]Department of Computer Science and Information Engineering, National Chin-Yi University of Technology,
No. 57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung 411030, Taiwan (R.O.C.)

(Received December 27, 2021; accepted June 6, 2022)

***Keywords:*** intrusion detection, KDDCUP'99, data preprocessing, standard deviation standardization, deep learning, convolutional neural network

Intrusion detection has become a crucial issue due to an increase in cyberattacks. In most studies on this topic, intrusion detection performance has been found to be strongly related to the feature extraction and selection preprocess. However, there has been less research on problems or solutions related to the attributes of unequal metrics. Recently, deep-learning-based schemes have shown strong performance in image classification tasks without feature preprocessing. Therefore, in this study, we discuss the conversion of packet data into images for use in deep learning schemes with effective data preprocesses used to process the attributes of unequal metrics. A standard deviation standardization process is proposed to process the attributes of unequal metrics, which is followed by a data quantization process. Then, zigzag coding and the inverse discrete cosine transform are employed to convert the data into attribute images, which are used as the inputs for a convolutional neural network model. Intrusion detection is then achieved using the trained model. The experimental results demonstrate that the proposed scheme has reliable and efficient intrusion detection capability with a recall rate exceeding 94%. Meanwhile, packet attributes represented by 16 × 16 images provide about the same intrusion detection performance as that for 32 × 32 images. In summary, computational complexity can be reduced and performance can be maintained when using small images.

## 1. Introduction

The internet has now permeated every level of society, which has made life more convenient by providing services such as online shopping, instant messaging, and web blogs. However, a high level of dependence on the internet may increase exposure to problems such as spam, malicious attacks, and illegal software. Various types of cyberattacks have become more prevalent in recent years, so internet security has become a popular research topic.

*Corresponding author: e-mail: raymond@ncut.edu.tw
https://doi.org/10.18494/SAM3786

ISSN 0914-4935 

Illegally acquiring data is a frequent purpose of cyberattacks, and the tactics used for malicious attacks have become increasingly diverse and radical. SonicWall Capture Advanced Threat Protection Service reported hundreds of thousands of unexpected cyberattacks in 2018 and an increase of 118% as compared with the previous year.[1] The growth of the internet through the use of portable equipment in recent years has further increased the number of malicious software attacks. Intrusion detection technologies have been widely used for resolving network security issues to effectively prevent malicious attacks on information systems. Some institutions have placed great importance on applying robust intrusion detection systems to avoid such attempts or unauthorized access.

Intrusion detection systems can be divided into host-based intrusion detection systems (HIDS)[2] and network-based intrusion detection systems (NIDS) according to the location where they are set up. They can also be categorized into two types according to the technologies used for analysis: misuse intrusion detection (also named signature-based detection or rule-based detection) and anomaly intrusion detection.[3,4]

Numerous studies have proposed the use of machine learning for intrusion detection. The classification performance of an intrusion detection system is strongly related to the feature extraction and selection results.[5] Machine learning comprises feature extraction or selection, followed by classification.[6,7] Initially, feature extraction and selection processes were aimed at reducing the number of redundant or irrelevant features to improve the detection speed and classification process during subsequent procedures.[8]

Machine learning is a supervised learning scheme, where each packet of data corresponds to a category, with the category assigned during the training process. Common classifiers such as rule-based classifiers as well as Bayesian networks (BNs), decision trees (DTs), the k-nearest neighbor algorithm (KNN), genetic algorithms (GAs), support vector machines (SVMs), and neural networks such as artificial neural networks (ANNs) are also commonly applied in the intrusion detection classification process.[9,10] However, inadequate feature selection or extraction affects the classification performance of the models in the above schemes.

Deep learning has been widely examined in various fields including image recognition and speech recognition using convolutional neural networks (CNNs), which have demonstrated good performance in image classification tasks,[11] and recurrent neural networks (RNNs), which are used for strongly time-related features.[12] Deep-learning-based methods exhibit high prediction and classification performances since they can automatically learn nonlinearly correlated features from the original data without advance feature extraction.[13] Most traditional machine learning methods used in intrusion detection systems rely on feature selection rather than the original data.

It has been revealed that a suitable standardization process can effectively improve classification accuracy.[14] However, although all features are usually applied for standardization and the subsequent quantization process, not all features have the same metrics and ranges, i.e., some features are discrete values of 0 and 1 and some have continuous values with different ranges. Standardization and quantization of all features of data cause data distortion. Nevertheless, this problem has seldom been raised, nor have solutions to the problem of unequal metrics in each data attribute been reported. Therefore, to solve this problem, in this study, we

propose a standardization approach based on each feature rather than all features to avoid data distortion.

Although CNNs have been successfully applied to a variety of classification problems, there have been few studies on their use in intrusion detection. On the basis of the successful use of CNNs in image classification, in this study, packet attributes are converted into feature images, and image classification using a CNN-based model is performed for intrusion detection. In previous research, importance has been attached to preprocessing of the network packet data. Hence, in this study, data preprocesses, including the standardization of each feature, quantization, and the inverse discrete cosine transform (IDCT), are applied before converting packet data into images.

The rest of this paper is organized as follows. Section 2 presents the dataset used in this work, the proposed preprocess, and the image conversion. Section 3 introduces the deep learning model utilized for intrusion detection. Experimental results and evaluations are provided in Sect. 4. Finally, the conclusions of this study are given in Sect. 5.

## 2. Data Preprocessing

Data preprocessing is an indispensable process in research because the way in which data are processed directly affects the classification performance.[15] In this study, standard deviation standardization is performed on individual features to avoid feature distortion, after which the standardized feature is quantized in the range from 0 to 255. Subsequently, zigzag coding is applied to the quantized data. The zigzag codes are regarded as the spatial frequency distribution of the packet data feature images. The IDCT is then utilized to transform the spatial frequency spectrum into the spatial domain image. The spatial domain image then becomes the input for the subsequent CNN model.

In this study, the KDDCUP'99 dataset is used for intrusion detection. This dataset includes 4898431 packet records, each packet containing 41 attributes (distributions) and a target class feature. The network packet has been divided into four main types of attacks: probing, denial of service (DoS), user to root (U2R), and remote to local (R2L).[16] DoS, probing, U2R, and R2L attacks are classified as abnormal network packets.[17]

Data preprocessing is used to convert the original data into an appropriate format for subsequent analysis and use.[18] The KDDCUP'99 dataset contains both continuous and discrete attributes; among the continuous attributes, every measurement metric is different. If the import data are not subjected to appropriate processes or schedules in advance, the feature extraction process may be ineffective when using the CNN training model.

The preprocessing is divided into four steps: conversion to numerical data, data standardization, data quantization, and zigzag coding and image conversion. The flow of the proposed preprocess is shown in Fig. 1.

- Conversion to numerical data: One-hot encoding is used in machine learning as a method to quantify categorical data.[19] It is applied to transform literal data into numerical data in this study, for example, the protocol type "TCP" is transformed to "3".

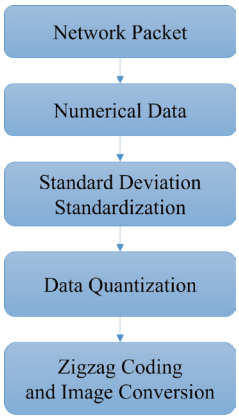



Fig. 1. (Color online) Flow of the enhanced data preprocess.

- Data standardization: The standard deviation is standardized on the basis of each packet attribute using Eqs. (1) and (2), where $\overline{X_k}$ is the average value of the $k^{th}$ attribute, $S_k$ is the mean square error of the $k^{th}$ attribute, and $X_{ik}$ is the $k^{th}$ attribute in the $i^{th}$ packet data record. The standardization of each feature of the packet data record is expressed by Eq. (3).
- Data quantization: The data quantization process is expressed by Eq. (4), where $X$ and $X^*$ are the values to be mapped and the mapped attribute values, and *min* and *max* are the minimum and maximum values in each attribute, respectively. Each attribute is quantized to values ranging from 0 to $N$, where $N$ is set to 255 in this work.

$$\overline{X_k} = \frac{1}{n}\sum_{i=1}^{n} X_{ik} \tag{1}$$

$$S_k = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(X_{ik} - \overline{X_k}\right)^2} \tag{2}$$

$$Z_{ik} = \frac{X_{ik} - \overline{X_k}}{S_k} \tag{3}$$

$$X^* = \left(\frac{X - min}{max - min}\right) \times N \tag{4}$$

- Image conversion: Although 41 packet attributes can be converted into a 7 × 7 image, the input image should be large enough to s*et* all attributes in the lower-frequency part, such as a 16 × 16 or 32 × 32 image. This conversion of 41 packet attributes into a larger image requires data padding, i.e., in addition to the 41 characteristic attributes of each packet, the other data elements are padded with zeros, indicating no data. The zigzag arrangement method is used as the data encoding scheme. From the point of the image domain, the changes in the lower-frequency part are much clearer than those in the higher-frequency part in the spatial frequency domain of an image. Therefore, the zigzag arrangement operation concentrates

data in the upper left corner, which has a relatively low spatial frequency in the frequency domain. Figure 2 gives an example of the coding of data in the form of a zigzag. Moreover, the frequency domain map is obtained by the discrete cosine transform (DCT) of an image.[20] The frequency domain map of the zigzag arrangement in Fig. 2 is shown in Fig. 3. An IDCT is required to convert the frequency domain map back to a grayscale image. The image yielded from the IDCT is shown in Fig. 4, which is the input image for the CNN model.

## 3. Architecture of Deep Learning

In this work, a deep-learning-based network is applied as an intrusion detection system, the architecture of which is shown in Fig. 5. The architecture consists of a CNN and a fully connected neural network. The CNN includes two convolution layers, and each layer is associated with a max-pooling layer. The kernel size of both convolutional layers is 3 × 3 and the stride is 2. The fully connected network includes an input layer, two hidden layers, and an output layer. The input layer has 256 or 1024 neurons depending on the size of the packet attribute image, and the hidden layers have 128 and 32 neurons. The output layer includes two neurons, which indicate whether the input packets are classified as normal or abnormal packets.

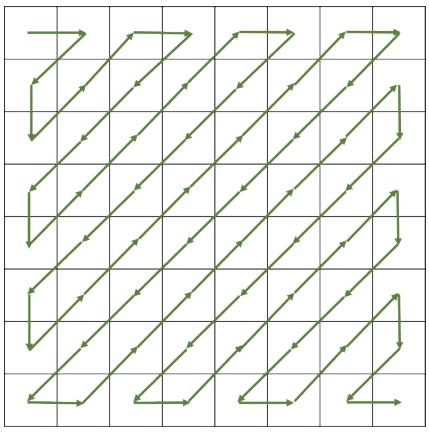


Fig. 2. (Color online) Schematic diagram of zigzag data arrangement.

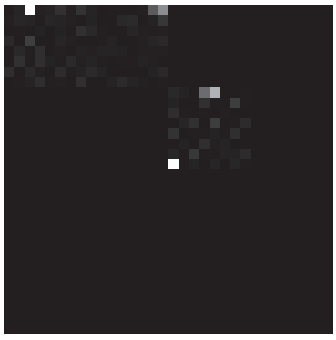


Fig. 3. (Color online) Example of a resulting map of the zigzag arrangement.

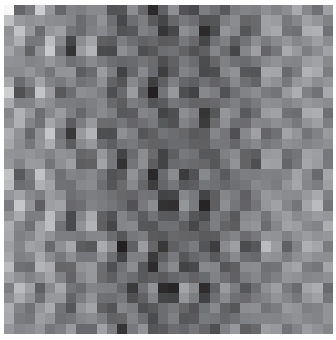


Fig. 4. (Color online) Example of a packet attribute image.

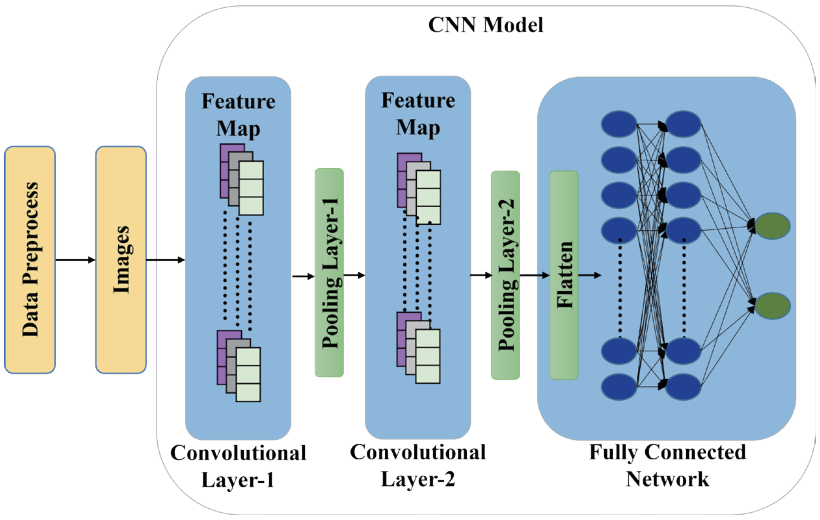



Fig. 5. (Color online) Architecture used for deep learning.

## 4. Experimental Results and Evaluations

The kddcup.data_10_percent.gz file was used in our experiment as the experimental training and test samples. This file contained 10% of the KDDCUP'99 data (494,021 connection records in total), where 70% of the records were training samples and 30% of the records were test samples. Intrusion detection was performed to predict whether the packets were associated with abnormal activities. A true positive was defined as a network packet predicted by the classifier to be abnormal when it was indeed an abnormal network packet, as shown in Table 1.

Additionally, the precision rate, recall rate, and accuracy were used to evaluate the detection performance of the proposed scheme. The precision rate was used to evaluate the ability to detect abnormal network packets. The recall rate represents the model's ability to identify abnormal packets. The accuracy was used to evaluate the performance of the classifier as a whole. These three performance metrics are given by Eqs. (5)–(7). Among these metrics, the recall rate was shown to be crucial for network security. In other words, a high recall rate indicates that abnormal packets can be blocked to prevent intrusion into the system.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{7}$$

The number of epochs was set to 2000, the learning rate was set to 0.001, and the batch size was set to 256 and 16000 for the training and testing phases, respectively. Two sizes of the packet

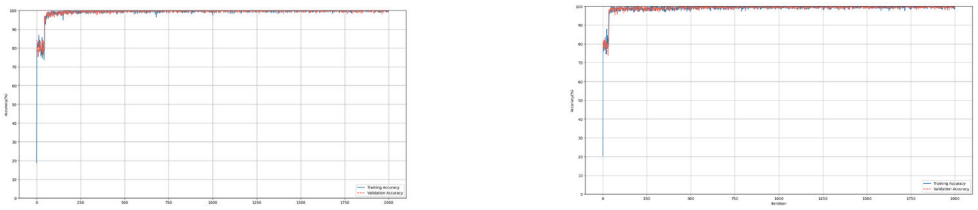
Table 1
Evaluation in intrusion detection.

| | | Prediction | |
|---|---|---|---|
| | | Abnormal | Normal |
| Actual | Abnormal | True positive (TP) | False negative (FN) |
| | Normal | False positive (FP) | True negative (TN) |

attribute image were tested (16 × 16 and 32 × 32 pixels), and for both sizes, each image was either subjected to a standardization process based on each packet attribute (CASE_1) or not subjected to the process (CASE_2). Accordingly, the input layer of the fully connected neural network had 256 and 1024 neurons for the smaller and larger images, respectively, followed by a layer with 128 neurons and a hidden layer with 32 neurons.

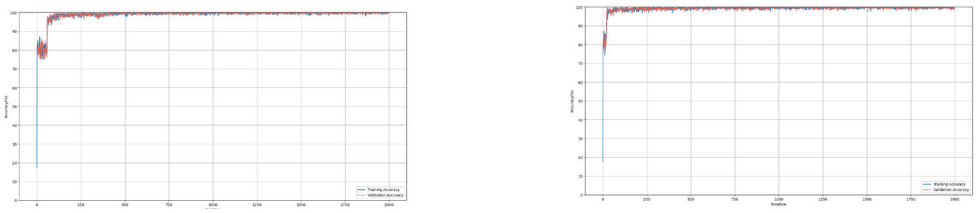
Figures 6 and 7 show the accuracy for both cases with the 16 × 16 and 32 × 32 images, respectively, during the training phase. Figures 8 and 9 show the loss values for both cases with the 16 × 16 and 32 × 32 images, respectively, during the training phase.

The confusion matrices of the testing results based on the evaluation metrics shown in Table 1 are shown in Tables 2–5. The values in each table are the average results of 10 tests. The test results for the precision rate, recall rate, and accuracy of the 16 × 16 and 32 × 32 images are listed in Tables 6 and 7, respectively. The average, maximum, and minimum values are the results of 10 tests. As shown in Tables 6 and 7, recall rates of approximately 94% were obtained, indicating that 94% of the abnormal network packets were detected by the proposed classifier, implying that network security was maintained. Packet attributes represented by the 16 × 16 and 32 × 32 images yielded almost the same results, indicating that the abnormal packet detection performance was maintained using the 16 × 16 attribute images. In other words, detection performance could be maintained by using small images, demonstrating that the amount of packet data and the computational complexity can also be reduced. Furthermore, using the proposed standard deviation standardization in the preprocessing yielded higher recall rates than those when using conventional standardization.



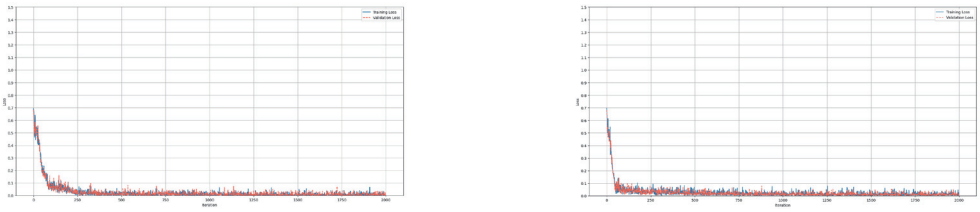
(a) (b)

Fig. 6. (Color online) Accuracy of (a) CASE_1 and (b) CASE_2 for 16 × 16 images.



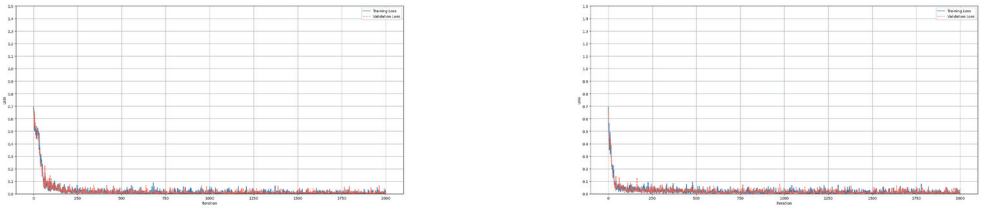
(a) (b)

Fig. 7. (Color online) Accuracy of (a) CASE_1 and (b) CASE_2 for 32 × 32 images.



(a) (b)

Fig. 8. (Color online) Loss values of (a) CASE_1 and (b) CASE_2 for 16 × 16 images.



(a) (b)

Fig. 9. (Color online) Loss values of (a) CASE_1 and (b) CASE_2 for 32 × 32 images.

Table 2
(Color online) Confusion matrix for 16 × 16 images (CASE_1).

| **16 × 16 CASE_1** | | Prediction | |
|---|---|---|---|
| | | Abnormal | Normal |
| Actual | Abnormal | **10366 (TP)** | 623 (FN) |
| | Normal | 2539 (FP) | 2472 (TN) |

Table 3
Confusion matrix for 16 × 16 images (CASE_2).

| 16 × 16 CASE_2 | | Prediction | |
|---|---|---|---|
| | | Abnormal | Normal |
| Actual | Abnormal | 10159 (TP) | 621 (FN) |
| | Normal | 2513 (FP) | 2707 (TN) |

Table 4
(Color online) Confusion matrix for 32 × 32 images (CASE_1).

| **32 × 32 CASE_1** | | Prediction | |
|---|---|---|---|
| | | Abnormal | Normal |
| Actual | Abnormal | **10382 (TP)** | 617 (FN) |
| | Normal | 2528 (FP) | 2473 (TN) |

Table 5
Confusion matrix for 32 × 32 images (CASE_2).

| 32 × 32 CASE_2 | | Prediction | |
|---|---|---|---|
| | | Abnormal | Normal |
| Actual | Abnormal | 10119 (TP) | 619 (FN) |
| | Normal | 2509 (FP) | 2753 (TN) |

Table 6
(Color online) Experimental results for 16 × 16 images.

| | Precision rate (%) | Recall rate (%) | Accuracy (%) |
|---|---|---|---|
| | | **CASE_1** | |
| **Average** | **80.325** | **94.331** | **80.238** |
| Minimum | 79.901 | 94.124 | 79.438 |
| Maximum | 81.140 | 94.642 | 81.300 |
| | | CASE_2 | |
| Average | 80.169 | 94.239 | 80.413 |
| Minimum | 79.762 | 94.039 | 79.025 |
| Maximum | 80.361 | 94.430 | 80.956 |

Table 7
(Color online) Experimental results for 32 × 32 images.

| | Precision rate (%) | Recall rate (%) | Accuracy (%) |
|---|---|---|---|
| | | **CASE_1** | |
| **Average** | **80.418** | **94.390** | **80.344** |
| Minimum | 79.749 | 94.183 | 79.695 |
| Maximum | 81.039 | 94.526 | 81.327 |
| | | CASE_2 | |
| Average | 80.131 | 94.235 | 80.450 |
| Minimum | 79.506 | 93.910 | 79.706 |
| Maximum | 80.548 | 94.486 | 80.950 |

## 5. Conclusions

Feature extraction and selection are difficult to achieve in the field of intrusion detection. Deep-learning-based schemes have been shown to have good performance in image classification tasks without prior feature extraction. In this study, we adopted a CNN-based deep learning scheme for intrusion detection, in which network packet data is are converted into images by using the proposed effective data preprocesses to improve intrusion detection efficiency. We also proposed a standard deviation standardization process based on each feature rather than all features to avoid distortion, which is followed by a data quantization process. Then, zigzag coding and the IDCT are applied to convert the data into feature images, which are input into the CNN for classification.

The experimental results showed that the proposed scheme with the data preprocessing method is effective in identifying abnormal network packets, and recall rates exceeding 94% were obtained. In other words, more than 94% of abnormal packets can be detected, and security is maintained when using the proposed scheme. Moreover, the proposed standard deviation standardization method (CASE_1) yielded higher recall rates than the conventional standardization process (CASE_2). Packet attributes were represented by feature images of two sizes, 16 × 16 and 32 × 32. These images yielded approximately the same intrusion detection performance. Therefore, computation complexity can be reduced by using small images without adversely affecting the detection performance

## References


1 2019 Sonicwall Cyber Threat Report, https://www.sonicwall.com/resources/white-papers/2019-sonicwall-cyber-threat-report/ (Accessed May 2021).
2 H. Debar, M. Dacier, and A. Wespi: Comput. Networks **31** (1999) 805. https://doi.org/10.1016/S1389-1286(98)00017-6
3 N. S. Sulaiman, A. Nasir, W. R. W. Othman, S. F. A. Wahab, N. S. Aziz, A. Yacob, and N. Samsudin: J. Physics: Conference Series **1874** (2021) 012042. https://doi.org/10.1088/1742-6596/1874/1/012042
4 H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung: J. Network and Comput. Appl. **36** (2013) 16. https://doi.org/10.1016/j.jnca.2012.09.004
5 H. Yang and F. Wang: IEEE Access **7** (2019) 64366. https://doi.org/10.1109/ACCESS.2019.2917299
6 C.-F. Tsai, Y.-F. Hsu, C.-Y. Lin, and W.-Y. Lin: Expert Syst. Appl. **36** (2009) 11994. https://doi.org/10.1016/j.eswa.2009.05.029
7 H. T. Nguyen, K. Franke, and S. Petrovic: Feature Extraction Methods for Intrusion Detection Systems: In Threats, Countermeasures, and Advances in Applied Information Security (IGI Global, Hershey, PA, USA, 2012) p. 23. https://doi.org/10.4018/978-1-4666-0978-5.ch002
8 J. Yan, B. Zhang, N. Liu, S. Yan, Q. Cheng, W. Fan, Q. Yang, W. Xi, and Z. Chen: IEEE Trans. Knowl. Data Eng. **18** (2006) 320. https://doi.org/10.1109/TKDE.2006.45
9 O. A. Alimi, K. Ouahada, A. M. Abu-Mahfouz, S. Rimer, and K. O. A. Alimi: Sustainability **13** (2021) 9597. https://doi.org/10.3390/su13179597
10 M. Qiu, Y. Zhang, T. Ma, Q. Wu, and F. Jin: Sens. Mater **32** (2020) 2659. https://doi.org/10.18494/SAM.2020.2794
11 A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi: Artif. Intell. Rev. **53** (2020) 5455. https://doi.org/10.1007/s10462-020-09825-6
12 J. Kim, J. Kim, H. L. T. Thu, and H. Kim: 2016 Int. Conf. Platform Technol. and Serv. (PlatCon) (2016) 1. https://doi.org/10.1109/PlatCon.2016.7456805
13 S. Rezaei and X. Liu: IEEE Commun. Mag. **57** (2019) 76. https://doi.org/10.1109/MCOM.2019.1800819
14 A. R. Al Shorman, H. Faris, P. A. Castillo, J. J. M. Guervós, and N. Al-Madi: IJCCI (2018) 79. https://doi.org/10.5220/0006959000790085
15 S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas: Int. J. Comput. Sci. **1** (2006) 111. https://doi.org/10.5281/zenodo.1082415
16 M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani: 2009 IEEE Symp. Comput. Intell. in Secur. and Def. Appl. (2009) 1. https://doi.org/10.1109/CISDA.2009.5356528
17 V. Bolon-Canedo, N. Sanchez-Marono, and A. Alonso-Betanzos: Expert Syst. Appl. **38** (2011) 5947. https://doi.org/10.1016/j.eswa.2010.11.028
18 H. Alazzam, A. Sharieh, and K. E. Sabri: Expert Syst. Appl. **148** (2020) 113249. https://doi.org/10.1016/j.eswa.2020.113249
19 C. M. Maxfield: Chapter 4 - FPGA Vs. ASIC Designs: In Fpgas: Instant Access (Newnes, Burlington, 2008) p. 61. https://doi.org/10.1016/B978-0-7506-8974-8.00004-1
20 N. Ahmed, T. Natarajan, and K. R. Rao: IEEE Trans. Comput. **100** (1974) 90. https://doi.org/10.1109/T-C.1974.223784


## About the Authors

**Chia-Ju Lin** received her B.S. and M.S. degrees from the Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taiwan, R.O.C., in 2019 and 2021, respectively. She is currently pursuing her Ph.D. degree in engineering science at National Cheng Kung University, Taiwan. Her research interests include computer networks and artificial intelligence. (wasjulie0905@gmail.com)

**Yueh-Min Huang** (Senior Member, IEEE) received his M.S. and Ph.D. degrees in electrical engineering from the University of Arizona in 1988 and 1991, respectively. He is currently a chair professor with the Department of Engineering Science and Institute of Education, National Cheng Kung University, Taiwan. He has supervised over 60 Ph.D. and 300 M.S. students. He has co-authored three books and published more than 280 refereed journal research articles. His research interests include e-learning, multimedia communications, and artificial intelligence. He became a fellow of the British Computer Society in 2011. He is also the funding chair of the International Symposium of Emerging Technologies for Education (SETE) and the International Conference of Innovative Technologies and Learning (ICITL). He has received many research awards, such as Taiwan's National Outstanding Research Award in 2011/2014 and the 2017 Taiwan Outstanding IT Elite Award. He is on the editorial board of several international journals in the areas of educational technology, computer communications, and web intelligence. (huang@mail.ncku.edu.tw)

**Ruey-Maw Chen** received his M.S. and Ph.D. degrees in engineering science from National Cheng Kung University, Taiwan, R.O.C., in 1985 and 2000, respectively. From 1985 to 1994, he was a senior engineer in avionics system design at Chung Shan Institute of Science and Technology. From 1994 to 2001, he was an engineer at Computer Center, Chin-Yi Institute of Technology. Since 2002, he has been with the Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, where he is a full professor. His research interests include scheduling, neural networks, meta-heuristic algorithms, image processing, and computer networks. (raymond@ncut.edu.tw)