# Recent Innovations and Comparison of Deep Learning Techniques in Malware Classification : A Review

Balram Yadav, Sanjiv Tokekar

Institute of Engineering &Technology, DAVV, Indore-452017, Madhya Pradesh, India
e-mail: *balram.dreamsworld@gmail.com*

ORCID iD: 0000-0001-5182-8324 0000-0002-0845-0300

Research Paper Received: 28.10.2020 Revised: 17.11.2020 Accepted: 03.12.2020

**Abstract**—The internet made an individuals life very easy and more productive, but there are some associated threats linked to the internet and devices. Malware is considered the most severe threat for decades to the digital world and malware variants identification and classification is the most vital and critical research problem. It is an invasive malicious code that accesses devices, information, and services without the permission, knowledge of the user. Researchers, analysts and antivirus companies are incessantly inventing and implementing new strategies to fight back malware and its variants. In the last decade, one of the strategies is extensively used in the field of malware detection and classification is the deep learning methods using malware visualization. Results revealed that using visualization; malware can be identified, classified more promptly, efficiently, and accurately. Deep learning algorithms vary according to applications, architecture, and uses, so it is required to review and inspect the work based on deep learning to use malware visualization to know the recent approaches and innovations that have been established, to identify problems, current issues, challenges, and of course at the same time to motivate potential research directions. In this effort, an extensive survey of works that utilized deep learning methods using malware image representation, for malware classification is reviewed with a detailed discussion on key methods such as data sets description, malware image representation strategies, and deep learning architectures of parameters, contributions, and limitations. A comparison of the reviewed work is presented based on various key factors.

**Keywords**—Deep learning, Deep learning architectures, Image representation, Malware, Malware classification, Malware visualization.

## 1. Introduction

Today, the internet has become an essential and inevitable part of everybodys life. People use the internet for various services like social networking, banking, communication, shopping, and for other purposes. Malware is an acronym for malicious software that intends to harm, it is defined as specially written programs to perform malicious activities. In recent years the astounding growth [1], [2] of the internet has designated a giant barrier in front of analysts, researchers, and antivirus companies. The researchers and analysts frequently recommend defenses and designing novel methods to fight mal-

ware and its variant attacks.

The number of malware found continuous to increase day by day because malware and its variants can be generated using automated tools [5] and reusing code modules. The daily increase in malware and its variants is a severe problem [13] and has become one of the most adverse issues in the field of cyber security and the digital world. The infected files submitted to antivirus companies for the analysis purpose is enormous [1], [2] and it is almost difficult to analyze each file manually, so there is a need for some automation [4] to analyze these files effectively with less intervention of humans and efforts. The attacks of malware and its variants are not only limited to the internet, but it is also affecting the internet of things (IoT) networks and devices, mobile networks, cloud, and researchers [22]–[25] started to explore malware analysis using DL and visualization techniques in the above-mentioned areas.

The Figure 1, depicts the enormous growth in the number of attacks. A report by the antivirus company revealed that in the period 2011 - November 2020, 1113.73 million malware was recorded and 267.23 million new malware reported in the last two years Figure 2. One more report from McAfee antivirus company Figure 3, portrays the present scenario, millions of malware and variants are discovered in the first quarter of 2020 in various categories and much more malware is undiscovered and unreported. McAfee Labs observed an increase in the growth of new malware, IoT malware, ransomware, and new mobile malware.

In the last decade, DL based approaches for malware analysis has become the primary approach among data mining, ML, and traditional approaches. DL is a subset of ML and the functionality and

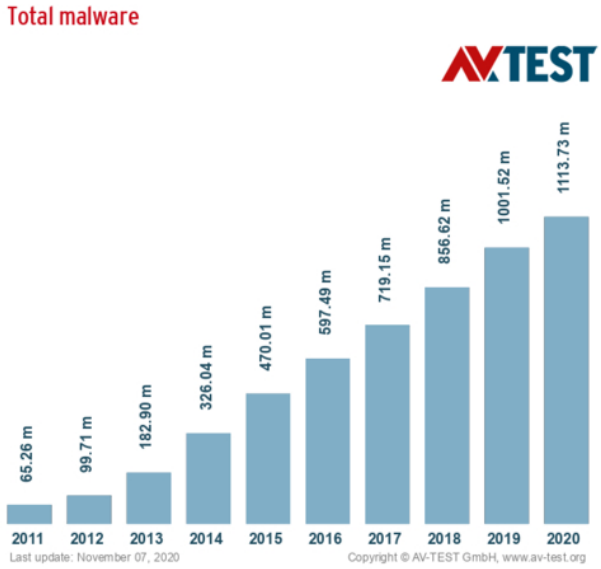


Fig. 1. Malware Statistics

architecture of DL is inspired by the human brain. DL refers to the set of techniques used for automatic feature extraction and to discover and identify hidden patterns in images, sound, text, and video, and other signal based processing. DL is a layered-based structure where the first few layers are responsible

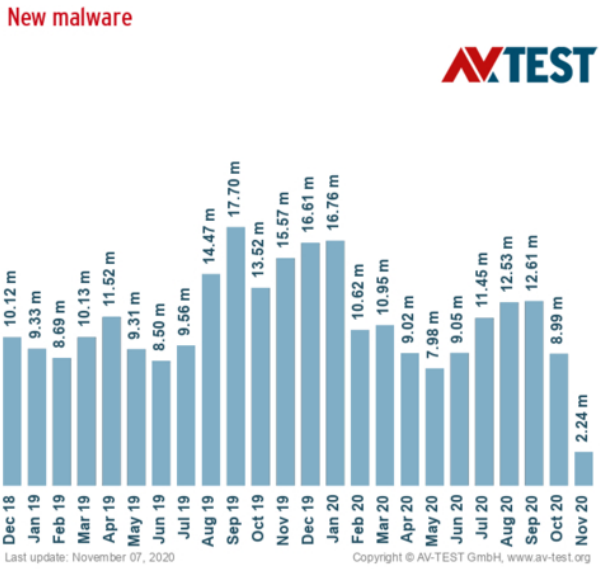


Fig. 2. New Malware Evolution Statistics

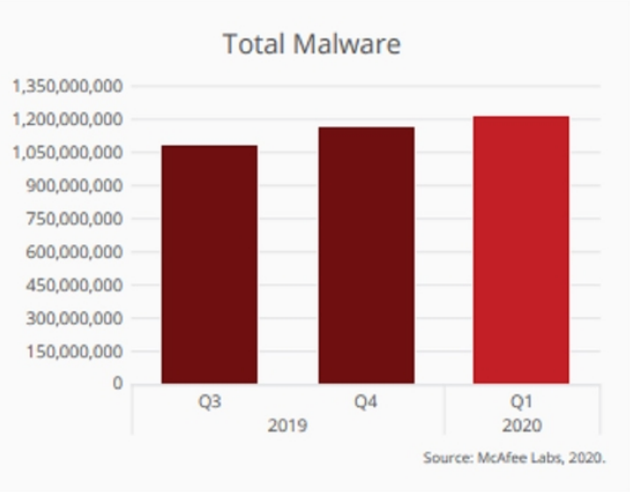


Fig. 3. Malware Evolution Quarterly Statistics

for feature extraction and the last layers are for classification, detection by using supervised, unsupervised, and hybrid DL architectures. Traditional approaches such as static, dynamic, or hybrid analysis extract separate levels of features from malicious samples for identification and classification, which cannot perform efficiently and accurately.

The paper is organized as follows. Section 2 discusses and compares (Table 2) the recent innovations and approaches used and also the comparison made on utilized DL models (Table 1) in malware classification and identification. Section 3 discusses the challenges and issues of malware classification and DL based methods. Finally, the review work is concluded in Section 4.

## 2. Recent Malware Classification Techniques and Innovations

The investigation into malicious code is done by tradition mainly with static, dynamic, and hybrid analysis, but since 2011 when Nataraj et al. [3] introduced an unconventional method of malicious code identification and classification. They proposed and experimented with malware visualization and the results shown greater improvement over traditional approaches and are very effective. So this review work primarily focused on the recent approaches and innovations that have been established, based on the classification of malware and its variants using visualization techniques and DL. All the methods either data mining or ML approach dependent on the extraction of features, applying more cleverer frameworks or classifiers for classification purposes. The major downside of ML is manual, feature extraction [24]. Malware visualization is related to translating malware code in the form of image [6] and the main advantage of doing this is that different code sections of malware in the form of images can be differentiated and compared easily [8].

In the year 2011, the research related to malware visualization by Nataraj et al. [3] is considered the first malware visualization based solution. They implemented a technique to represent malware binaries in the form of gray images and afterward utilized a global image-based feature GIST (this feature provides a high level, low dimensional representation of some vital information of an image) to quantify texture-based similarity. The converted gray images are classified by utilizing the k-nearest neighbor (KNN) classifier, and the results they obtained confirms that visualization of malware is very effective, accurate, and quickly classify the malware. The proposed solution is successful because visualization promotes to detect of visual similarity between malware images by extracting texture features, through visualization a small change in the images can be measured, strategy also proved that there exist structural similarities between the malware of the same family and execution of malware binary is not required.

In 2013, inspired by the concept [3] the authors Han et al. [5] proposed a novel way of translating

malware binaries into RGB image matrices. In their proposed solution first, malware binaries are disassembled and then opcode sequences are extracted and stored in blocks, every block of these sequences is processed by two hash functions (Simhash and djb2) to generate coordinate and RGB pixel information of the image matrix. To compare RGB image matrices selective area matching is used. The achieved classification accuracy is not promising besides the author utilized a higher dimension, color image that helps to extract more optimized patterns into it and the color image contains more data per pixel over gray images and it avails to classified malware successfully and also manual feature extraction limits the proposed method and high computational cost method is not scalable.

In 2015, Makandar and Patrot [6] in their proposed solution converted malware into grayscale images and classified these converted malware images based on texture features. They extracted a total of 512 feature vectors (based on GWT and GIST to observe the malware behavior) and among them, 320 two-dimensional features are selected, and classified by applying an ANN. The author has applied a feed-forward artificial neural network for classification because the artificial neural network is more compatible to detect hidden patterns from perplexed data like images, sound, signals, etc., and at the same time flexible and easy to implement. The applied ANN utilized a feed-forward backpropagation algorithm and achieved accuracy is not good due to high dimensional feature vector processing so proposed malware classification using ANN does not prove the effective solution.

The research by Pal and Sudeep [7] in 2016, stated that classification accuracy can be increased by using pre-processing techniques on the raw data and results evident in the importance of the pre-processing techniques. The authors exposed that by pre-processed raw data; the accuracy of classification improved using CNN and also verified that raw data applied to train any DL model does not produce good results always. The authors applied three types of pre-processing techniques, mean normalization makes the brightness of the training data is normalized to each image dimension, standardization organized raw data mean and variance normalized, and zero component analysis of the raw data transforms the edges of the image more profound and shown how accuracy varies by experimenting on three different CNN models which differs in architecture and settings of different hyperparameters. The reason behind the success of the proposed method is that all the three pre-processing techniques are mathematically based and transform the raw data and edges of the training images more profound and organized and CNN detects more optimized features from these edges and availed in accuracy increment.

Dong et al. [19] in the year 2016, presented a comparison of DL methods and traditional methods for intrusion detection. The authors discussed various traditional approaches used for detection includes port identification, protocols failing to perform its intended operations, signature-based methods, data mining methods, and presented a hybrid method of using SVM and RBM. They concluded that DL methods proved useful in network analysis for detection and classification because of its capability to evaluate similar patterns in the data. The challenge in DL is that if malicious data is hidden within normal data in the network the whole process defeat. At last, the authors combined two techniques to form a hybrid detection system that is AEs and DBN, SMOTE technique applied to the data imbalance problem and a sizable voluminous dataset is utilized for the experiment.

Ding et al. [16] in the year 2016, has successfully applied a deep belief network (DBN), one of the

unsupervised DL algorithms for malware analysis. The authors extracted opcode sequences from malware codes as features and then utilized DBN to detect malware. An opcode implies the operation to be performed and specified in the machine language instruction. The neural network is trained by these extracted opcode sequences many times so that the hidden patterns can be learned effectively and the malicious program can be detected. The achieved results are not very promising, but successfully applied DBN for malware detection, the reason for low accuracy can be that because DBN is unsupervised learning so the amount of unlabeled data affects the performance but when adequate data are not available for supervised learning, DBN is capable to perform.

The research by Hardy et al. [17] in 2016, build an intelligent DL framework to detect malware. They utilized a stacked autoencoder (AE), which is one of the unsupervised DL algorithms, to extract generic features they utilized API call sequences from malicious programs (the portable executable files) followed by the supervised parameter tuning to detect unknown malware. The results of the proposed method are not very promising; they experimented only with different settings (number of hidden layers, number of neurons in hidden layers) in the proposed architecture and finally chosen three hidden layers and 100 neurons in each hidden layer but successfully applied AE for malware detection.

In the year 2016, Tobiyama et al. [18] proposed and applied DNN in combination for malware analysis. The proposed malware detection framework is mainly using the extraction of API call sequences and DL techniques for classification. The authors in their proposed architecture first applied a RNN to extract the API call sequences (to record malware behavior) as features and then these features are converted into images afterward applied CNN to classify feature images. A process behavior is defined as activities/actions and to perform each activity various operations are carried out and to record these process behaviors API call sequences are generated. The proposed method is successful because features are extracted utilizing RNN which remembers previous layer data to produce next layer data and then classified by CNN.

Meng et al. [14] in the year 2017, proposed a malware classification algorithm that computes features using static analysis after that applied the DL model for classification. The model first extracts the malware gene sequences as features and then a CNN is applied for classification. In malware analysis, malware genes is a block of codes that carry functional, i.e. API calls, so the author focused on call sequences of API to identify malware. The authors first processes the portable executable files and then files are converted into assembly codes using the ida tool and extract the API call sequences. These malware gene features are depicted in n*128, a two-dimensional matrix where n is the length of the API call sequence, and each line in the matrix is treated as a row of pixels in an image. The proposed model is successful because it cumulated the static analysis feature extraction with DL and the proposed CNN architecture gains the benefit of 128 filters with different filter sizes and 3000 epochs were carried out for training.

Kabanga and Kim [26] in 2017, proposed a CNN based model to classify malware images. The authors utilized CNN based classifier because it is reliable can apply to a whole image at a time, and best for automatic feature extraction. The author concluded that only image features used for the classification task are not sufficient. The authors achieved high accuracy only due to the big image size, a 3 layer CNN architecture configuration, and settings of hyperparameters for training.

Wang et al. [33] in 2017, presented malicious traffic classification using representation learning based on a CNN. The proposed strategy directly took raw traffic data and by preprocessing converts it into images, and feed the images into CNN for classification. To prove the usefulness of the proposed method author experimented with two different scenarios for detection and classification separately using three different classifiers. In the first scenario, experiments were performed for detection, and in the second scenario performed for classification purposes. The achieved accuracy is excellent, besides they utilized low image size; the author also tested 5*5 filter size with 32 and 63 numbers of filters in their CNN architecture.

Chandra and Sharma [40] in 2017, proposed a method to improve the performance of image classification by using RNN. The authors successfully applied LSTM and Identity initialized RNN (IRNN), to execute parallelly and independently of each other and the final output is the mean of the output generated by these two RNNs. The authors achieved a very low classification error and improved the performance of the basic RNN. The author proved RNN is also a promising a DL model for image-based classification by overcoming the limitation of IRNN.

Kumari et al. [34] in 2017, presented CNN based model to classify malware using image processing. The authors followed the malware visualization based classification as used by [3]. They implemented 3 CNN based architectures for classifying malware. The proposed method is not very effective as achievable accuracy is not very promising besides, they utilized a very big image size and additionally utilized a high-performance pre-trained VGG16 model with transfer learning and fine-tuning, in their experiment they used 256 batch sizes which is very high.

Kim et al. [43] in 2017, proposed the latest DL model for unknown malware classification and detection. They applied a transferred generative adversarial network (tGAN). They overcame the limitation of basic GAN by pre-train GAN with an AE structure. To address the data imbalance issue and to generate new samples they proposed and applied the tGAN model predicated on GAN. The proposed architecture consists of three modules for training, creation, and detection purpose. The author achieved good accuracy with the improved GAN model by surmounting the limitation of the basic GAN model by using AE to train the generator, as the generator generates the samples similar to original data and discriminator learns patterns in it and discriminates between original and generate data and this process becomes unstable when the generator generates non-sensible data.

Kalash et al. [8] in 2018, build a deep CNN based classifier for malware classification. By translating malware binaries to grayscale images, the malware classification is converted into a malware image classification inspired by the concept that malware variants of the same family share the common texture and visual features as used by [3]. The proposed CNN model architecture is based on a high performance pre-trained VGG-16 model to classify the malware images; they fine-tuned the hyperparameters and got remarkable accuracy on both the well-known malware datasets.

Zhihua et al. [11] in 2018, proposed a novel DL strategy to improve the detection of malware variants and also addressed the data imbalance problem. The author's followed the visualization of the malicious code into images as used by [3] then classification is done by using a CNN. The author's experimented with different sizes (24*24,48*48,96*96,192*192) of malware images using different architectures of CNN. The main

advantage of the proposed method is that they addressed data imbalance issues, good detection speed and automation of feature extraction. The author's method is successful because they applied very deep CNN architecture and additionally resolved the data imbalance issue with the augmentation technique (with rotation,width and height shift, rescale and other methods), the experiment is conducted on different CNN architectures as image size changes.

Ni et al. [12] in 2018, proposed a novel malware classification method that extracts the opcode sequences from the malware codes, and then these opcode sequences are translated into images based on simhash (a hash function to generate hash values) afterward CNN classifies the malware images. The authors encode the extracted features by simhash and these encoded values are converted into images to compare similarity among sequences. Experimental results proved that malware variants of the same family share common visual features and differ from other variants. The proposed method is successful and achieved very excellent accuracy as they utilized static analysis for feature extraction and learning and classification is done by DL, the proposed architecture was not very deep and the image size was very low beside they achieve very high accuracy. One important thing in the architecture was that they utilized 3 fully connected layers and 32 numbers of filters in the architecture.

Kornish et al. [15] in 2018, the proposed method for malware classification utilizing deep CNN. They expressed malware samples as image features that can absorb deep CNN for classification. The data set utilized by the author contains assembly code and raw binary samples. To convert assembly code into an image is a time-consuming process so the author uses raw binary samples; these binary samples have values (raw binary instructions) in hexadecimal format. Hexadecimal values can easily convert into decimal values and represented as an image using a hamming distance. They used 3 pre-trained deep CNN models Alexnet, VGG16, and VGG19. The method is only for classification not for detection and not able to classify unknown malware. Author method is different in the sense that malware binaries consist of hexadecimal values (representing instructions), these values are converted to decimal values and organized into an image and also they used 3 pre-trained deep CNN models Alexnet, VGG16, and VGG19 and for classification, SVM is used with high dimensional images.

Kumar et al. [20] in 2018, presented their work towards malware detection using image processing and DL. They used image similarity measurement to detect zero-day malware using CNN and the reason behind utilizing CNN is that there is no need for manual, feature extraction, directly feed malware images into it and CNN works on the whole image at a time which makes it flexible and intelligent DL model. They tested their methodologies in three different types of data sets and to convert malware binaries into images they used a method applied by [21]. The authors have applied 3-layers deep CNN model-based detection with 32, 32, and 64 numbers of filters followed by two fully connected layers and achieved excellent accuracy by setting different hyperparameters of the proposed architecture. They achieved remarkable accuracy for malware detection by combining the benign files with malimg malware data files.

Wang et al. [32] in 2018, presented malicious traffic classification using representation learning based on a DL model. Representation learning is the newly developed ML approach that directly processes raw traffic data; it is used to automatically learn features from raw data [33]. Many of the proposed methods directly remove the network traffic load for feature extraction and this skipped

information may lower down the performance of the model. The achieved accuracy is not very good because of the low image size; the author also tested 5*5 filter size with 16 and 32 filters in their CNN architecture configuration.

The research by Kim et al. [39] in 2018, proposed a framework which is a fusion of CNN with a gated recurrent unit (GRU) for malware classification. The proposed model architecture consists of four distinct levels, in the first level CNN's utilized, in second level GRU units, a layer of DNNs in the third level, and a sigmoid layer at the last level. All CNN at the first level produces an output and all these outputs feed as time-series data to GRU layers then each GRU units produce a single output value using a gate mechanism equal to the number of CNN's in the first level, then the output of GRU is input to the third level and each DNN produces a single output corresponding to each GRU, the final level which is the sigmoid layer produces classification result. The author has applied three variants of DL models and achieved low accuracy, but they successfully applied state of the art architecture for malware classification.

The research by Mourtaji et al. [9] in 2019, inspired by [3] and they proposed a DL based model to classify malware grayscale images. The authors first converted malware binaries into grayscale images and afterward applied a CNN based classifier to classify these images. They used part of the CNN architecture defined by S. Karen [10] and experimented with two benchmark datasets which proves the feasibility of the proposed method. The reason behind the success of the method is that they implemented a CNN model with initial weights same as the VGG-16 model, also utilized 64 and 128 numbers of filters and fine-tuned that model with a gray image of size 32*32 and experimented with different settings of hyperparameters.

Recently in the year 2019, Singh et al. [13] presented a novel strategy to represent malware binaries as color image matrix and then classify malware families. They experimented with RGB images over grayscale images and utilized a very big dataset and classification is performed by applying DNN architectures ResNet-50 (Residual Network) including a CNN. The authors proposed a novel approach to convert the malware binaries (a string of zeros and ones) into RGB values. They proved that with color image representation of malware, higher accuracy can achieve. Higher accuracy was achieved by using the ResNet-50 model as compared to CNN. The author's DL model is successful because the architecture proposed by them is very deep (15 layers CNN, 5 convolutional layers, 2 dense layers with 32,50,80,100,120 numbers of filters) and also with color image over gray image upgraded the accuracy, additionally with low dimensional image size achieved results are too good.

Naeem [25] in the year 2019, introduced the malware detection problem in the area of IoT networks and implemented a DL model to detect malware on the IoT networks. Many DL-based solutions have been proposed for malware detection and classification, but they suffer from high computational cost, complexity in the perspective of IoT networks because of the low configuration of IoT devices, so there is a need for accurate, speedy models for malware analysis in its environment. The author proposed the detection of malware by converting malware codes and benign binaries files into color images and then applied ML and DL separately to detect malicious activities in the IoT network. A deep CNN model is proposed for efficient malware detection and experimented on different image sizes, achieved accuracy is remarkable, but the proposed method is not effective for large scale dataset because of low run time. The achieved higher accuracy

proves the successfulness of the proposed solution, the reason behind that they utilized a color image of a very high dimension and in their proposed 4 layered architecture 128, 256, 512, 512 numbers of filters are utilized.

Hsiao et al. [31] in 2019, presented an advanced CNN that is siamese CNN (SCNN). DL models require a huge amount of data (many training samples) for high performance. It is not advisable to train DL models with an insufficient amount of training data it lowers down the performance of the model. There is one more problem with DL models is that for newly found malware whole training process is executed including the newly found malware sample. The authors adopted one-shot learning to address the issue. For visualizing the malware image authors used the approach applied by [3]. The proposed method is successful because of the big image size and the proposed siamese CNN architecture configuration has a sequence of twin CNN, both have 4 convolution layers, 3 max-pooling layers, a final fully connected layer with sigmoid as an activation function, a tailored layer is additionally integrated into the architecture measure the similarity between the feature vectors, batch size of 6, learning rate 0.00006, and 2000 epochs were used.

Research by Yin et al. [41] in 2019, proposed a combination of CNN and RNN to classify color images (not malware images). In their proposed architecture CNN is applied first for feature extraction, then again, they applied CNN and RNN (to extract continuous dependency features) separately into the extracted features from first CNN. The results are better than their original CNN model implemented previously, but the achieved accuracy is very low as compare to others besides, they utilized color images over gray images and experimented with CNN with RNN and intermediate CNN for feature extraction.

The research by Lu and Li [42] in 2019, implemented the latest DL models that are generative adversarial network (GAN) for malware classification and also addressed data imbalance issues. They applied GAN by using an 18-layer deep CNN and referred to this model as a deep convolutional generative adversarial network (DCGAN). Experimental results evident a 6% increment in the classification accuracy after utilizing GAN. The achieved accuracy (average testing) of the proposed model is incremented and the other performance measures are additionally incremented but the results are not promising besides they utilized the most recent DL model that is GAN for training and classification, more training samples are generated utilizing GAN and an 18-layers deep residual network is utilized as the malware classifier and trained the GAN network for 10000 epochs.

Recently in the year 2020, Jain et al. [35] applied extreme learning machines (ELM) for malware classification and also compared CNN and ELM model-based classification. Results confirmed that training time for ELMs is less than to train a CNN, achieve higher accuracy on 1d data processing, and the authors also underlined that for 2d data processing ELMs are faster than CNN. They experimented with the number of convolutional layers and other hyper-parameters settings but achieved promising results with a 2 layer CNN model with training images of dimension 128  128 and 32, 64 numbers of filters, with ELMs, they experiment with very little settings, they only tuned the number of neurons (128, 256, 512, 1024, 2048 and 4096) in the hidden layer and activation functions (tanh, relu, softlim, hardlim and multiquadric), finally chosen 4096 neurons and 50 ELMs for the experiment. The proposed method proved that ELM slightly performed well compared to CNN. Experiments proved that you always do not need to perform classification on 2d data, 1d data

proved to be very effective.

Naeem et al. [36] in 2020, introduced and proposed malware detection and classification problem into IoT networks. The IoT is an advancement of the traditional internet, it allows a massive amount of smart and small devices to connect for sharing information, communication, smart controlling, etc. They translated a Class.DEX file into a color image and then input it to a very deep CNN network for detection. The authors developed and tested a deep CNN to handle the IoT malware and achieve higher accuracy. The results are excellent due to color images (as color images are more feature enrich as compare to gray images) and very big image sizes, the author applied 64, 128, 256, and 512 numbers of filters in their very deep four-layer architecture.

In January 2020, Kumar and Bagane [37] build a hybrid DL based classifier, they utilized CNN with long short-term memory (LSTM) for the malware classification. First, they applied CNN to extract optimized features from the malware images, and afterward in the last layer after flattening the output they applied the bi-directional LSTM for classification. They successfully applied CNN with LSTM for malware classification and captured recurrent patterns in malware samples, big image size is utilized and settings of hyperparameters availed to achieve good accuracy.

Research by Vasan et al. [38] recently in 2020, applied a new approach based on the accumulation of CNNs architecture for detection of packed and unpacked malware, and the reason behind this accumulation is different deeper architectures extracts higher representation of features by combining these higher optimized features detection might improve. The authors utilized the deep pre-trained VGG16 and ResNet-50 models for feature extraction and combined the extracted features of both models in a 6144-dimensional feature vector then trained SVM on these feature vectors and predicted the malware. The authors developed and tested a novel ensemble of pre-trained CNN models and performed very well as compared to traditional DL models, they achieved very excellent accuracy by utilizing the color image, additionally, the image size is too sizable voluminous and also ensemble the high-performance pre-trained DL models simultaneously.

### *2.1. DL models for malware classification*

In our study, we have seen many malware classification systems which are developed on the DL such as DNN, CNN, RNN and LSTM, AE and DBN based models and fusion of different DL models (hybrid and ensemble) are also applied to the task. It is very complicated to conclude any specific DL models for malware classification because authors utilized different architectures, datasets, and performance measures to evaluate the proposed DL model. Table 1. below compares various DL models utilized for the malware classification and identification task based on key factors.

### *2.2. DL and visualization based malware classification*

A plethora of research has been evolved using various ML and DL algorithms, some systems are based on coalescing different learning techniques. In this section, we are going to compare the recent works that have been established in the literature for malware classification and identification using DL and malware visualization. DL provides new approaches for malware analysis, especially for malware classification issue and at the same time shown improvements over traditional approaches. Table 2. below enlists various research papers reviewed, on many key factors and it is a detailed comparison of our review work.

TABLE 1: Comparison of Utilized DL Models in Malware Classification

| **Used DL models** | **Learning** | **Model type** | **Advantages** | **Disadvantages** |
|---|---|---|---|---|
| DNN: A DNN is composed of many the input, output, and many hidden layers and used to model general, nonlinear problems | Supervised | Discriminative | DNNs perform better than traditional techniques. Utilized for general classification problems | High computing requirements, training time and computational costs |
| CNN: The convolutional, pooling, and fully connected layers are stacked to form CNN architecture | Supervised | Discriminative | Found effective and best suited for malware classification. The learning process is fast. CNN is very flexible to design. Deep CNN implementation higher accuracy can achieve than a DNN model. Fewer neuron connections are needed. Various variations to CNN implementation possible | It needs a hierarchy of layers to extract optimized features. Large and labeled data set is required for processing. CNN is not effective for non spatial data processing |
| RNN: An RNN looks similar to a traditional ANN except that it has a memory-state and is added to the neurons and with an RNN, the output is sent back to the previous layer number of times and this loop structure allows RNN to alter the current data based on previously processed data and current data | Supervised | Can be both | RNNs found effective for optimized feature extraction because an image can be represented as time-series data. RNNs can process variable-length input sequences. The memory of the RNN can be adjusted for longer and shorter sequences so learning of variable length sequences is possible. Overcome the memory limitation of DNNs and able to recognize previous events | Suffers from vanishing gradient problem and very hard to train due to its loop structure design |
| AE: AE, by design, transforms data into a hidden representation and then reconstructs data from that hidden representation inputs are high dimensional data. It is compressed by a hidden layer and the output layer reconstructs the inputs | Semi or Unsupervised | Generative | AE has been found useful in dimensionality reduction of feature vectors and found effective in pre training tasks. The encoder and decoder are trained simultaneously so achieves minimum loss | To train AE lots of data, processing time and fine-tuning of hyperparameters are required |
| DBN: It consists of multiple layers of hidden units(hidden units used for detecting features and correlations of the data). DBN composed of RBM | Semi or Unsupervised | Generative | Effective in pre training tasks. Layer by layer learning. It can process unlabeled data | DBN is not better for computer vision. Unlabelled data affects the performance |
| ELM: It is a noniterative learning algorithm for feedforward NN that has only one hidden layer | Can be both | Discriminative | It is a fast learning algorithm as all the parameters are tuned only once. High training speed. | It has shallow network architecture, considered well for basic classification problems but not for complex tasks |
| GAN: GANs are DNN architectures comprised of generator and discriminator module. The generator module creates new samples of identical data, while the discriminator distinguishes between original samples and fake generated samples | Unsupervised | Generative | Novel DL model, GAN is useful in the cyber security field, and it is proving very promising to handle data imbalance issue. GANs are effective and shown acceptability to images. Faster sample generation | To train a GAN is very hard because GAN training considered unstable due to gradient descent |

TABLE 2: Comparison of Reviewed DL based Approaches

| **Authors** | **Approach &Features** | **Data set & Image description** | **Train & Test ratio** | **Outcomes** | **Pros** | **Cons** |
|---|---|---|---|---|---|---|
| Nataraj et al. [3] 2011 | Data mining and k-NN,GIST | Malimg dataset, gray image (64*64) | 90%-10% | Accuracy 97.18% | Texture feature extraction & comparison is faster with visualization and proved efficient. | Manual feature extraction, computational cost of texture analysis is high |
| Han et al. [5] 2013 | Static analysis,Opcode | color image (256 * 256) generated from visualization tool | NA | Accuracy 95% | RGB image matrices translation proved effective(RGB matrices of same family had higher similarity). Method is not scalable | Disassemblies of binary files are required. Length of opcode varies so pixels information differs in matrices |
| Makandar et al. [6] 2015 | Machine learning, ANN,GIST, GWT | Mahenhur dataset,gray image(64*64) | training on 480& testing on 3131 samples | Accuracy 96.35% TPR 3017 TFR 114 | ANN proved effective for malware classification | Evaluated on small dataset and 320, 2d features are too much for analysis |
| Pal et al. [7] 2016 | CNN, automatic | CIFAR 10 dataset,color image (32*32) | 84%-16% | Accuracy 64-68% | preprocessing proved effective | low accuracy |
| Ding et al. [16] 2016 | DBN,n-grams | 3000 benign and 3000 malicious files | 67%-33% | Accuracy 96.1% | dimensiality reduction of feature vector | the amount of unlabeled data affects the performance |
| Hardy et al. [17] 2016 | Stacked AE,API calls | Comodo cloud dataset, feature vector*class label | 90%-10% | Accuracy 96.85% | unknown malware detection | results are not very promising |
| Tobiyama et al. [18] 2016 | RNN, LSTM, CNN, process behavior | NTT secure dataset, gray image (30*30) | 55%-45% | Accuracy 96% | RNNs proved effective in feature extraction for malware classification as API call sequences are fed into it as time-series data | Evaluated on small dataset &RNN's back propagation training time on variable length sequences is the major limitation |
| Dong et al. [19] 2016 | RBM,SVM, HTTP response code,request type etc | KDD-96 dataset, gray image (128*128) | 90% - 10% | Accuracy 81% | RBM based model proved applicable but not effective and they conclude that effective detection of attacks is highly dependent on the type of the attack | results are worst |
| Meng et al. [14] 2017 | CNN,gene sequences | VX heaven dataset, n*128 image size | 90% - 10% | Accuracy 98% | Number of filters are large and used 5*5 filter size | disassembling of windows executables are required before proposed CNN implementation |
| Kabanga et al. [26] 2017 | CNN, automatic | Malimg dataset, gray image (128*128) | 90% - 10% | Accuracy 98% | achieved good accuracy. proved without preprocessing & augmentation techniques better accuracy can be achieved | experimented with settings of hyperparameters |

*Continued on next page*

TABLE 2 – *Continued from previous page*

| **Authors** | **Approach &Features** | **Data set & Image description** | **Train & Test ratio** | **Outcomes** | **Pros** | **Cons** |
|---|---|---|---|---|---|---|
| Wang et al. [33] 2017 | CNN, automatic | USTC-TFC2016 dataset,gray image (28*28) | 90%-10% | Accuracy 99.41% | achieved remarkable accuracy with a low image size | traffic size is not fixed in real scenarios & only used spatial features not design for unknown malware traffic |
| Chandra et al. [40] 2017 | RNN, automatic | different data sets, image size varies with datasets | varies with datasets | NA | reduced classification error and implemented improved RNN model | Gradient vanishing. Training an RNN is a very difficult task |
| Kumari et al. [34] 2017 | CNN, automatic | Microsoft dataset,gray image (150*150) | 80% -10% &10% validation | Accuracy 96.98% 97.07% 96.79% | utilized &evaluated one of the largest &novel datasets & highly scalable for multi-class classification | accuracy is not very promising and utilized big image size |
| Kim et al. [43] 2017 | tGAN, automatic | Microsoft dataset,color image (C*R) | 90% - 10% | Accuracy 96.39% | takes less time to detect zero-day malware | only for samples generation |
| Kalash et al. [8] 2018 | CNN, automatic | Malimgand Microsoft dataset, gray image (224*224) | 90%-10% ,50%-50% | Accuracy 98.52% 98.99% 99.97% | only 3 samples are misclassified also achieved higher accuracy | used pre-trained model. method is not data set specific.very large image size |
| Zhihua et al. [11] 2018 | CNN, automatic | Malimg dataset,gray image (96*96) | NA | Accuracy 94.5% , Precision 94.6 , Recall 94.5 | addressed data imbalane issue. low detection time | low performance measures |
| Ni et al. [12] 2018 | CNN, automatic | Microsoft dataset,gray image (24*24) | 80% - 20% | Accuracy 99.260% F1-Score 98.07 FPR 2.34% | reached high accuracy | Disassembly of malware is required to compute features |
| Kumar et al. [20] 2018 | CNN, automatic | Malimg dataset +3000 benign files,gray image(128*128) | 90% - 10% | Accuracy 98% | detect unknown malware | adversarial attacks and decompilation is required to convert .exe into binary and assembly |
| Kornish et al. [15]2018 | CNN, automatic | Microsoft dataset ,color image (57*57,227*227) | 60% - 40% | Accuracy 97-98% | proposed method directly processes raw network traffic data | accuracy is not very good.large image sizes |
| Wang et al. [32] 2018 | CNN, automatic | UNSW-NB15 dataset,gray image (28*28) | 90%-10% | Accuracy 97.3 % F1-score 98.5% Precision 98.65% Recall 81.37% | Results proved the feasibility of the method. Processes raw traffic data directly (without any pre processing) and converts into images | 28*28 image size is too low for optimal feature extraction |

*Continued on next page*

TABLE 2 – *Continued from previous page*

| **Authors** | **Approach &Features** | **Data set & Image description** | **Train & Test ratio** | **Outcomes** | **Pros** | **Cons** |
|---|---|---|---|---|---|---|
| Kim et al. [39] 2018 | CNN,GRU, DNN, automatic | Microsoft dataset, color image (224*224) | 50% - 50% | Accuracy 92.6% | GRU is utilized to treat feature maps from the CNN as time-series data | low accuracy |
| Mourtaji et al. [9] 2019 | CNN, automatic | Malimg and Microsoft dataset,gray image (32*32) | 85%-15%, 50%-50% | Accuracy 98.72% 99.88% | Effective solution. Method is not data set specific | obfuscation techniques |
| Singh et al. [13] 2019 | CNN, ResNet-50, automatic | Malshare, Virusshare, VirusTotal, color image (32*32) | 70% - 30% | Accuracy 98.98% 99.40% | with color image, higher accuracy achieved. requires no code extraction, execution & decompilation | processing of fixed image sizes limits the method |
| Naeem [25] 2019 | DCNN, automatic | Malimg & leopard mobile dataset,color image (192*192) | 55%-45% ,34%-66% | Accuracy 98.18% 97.34% | achieved better accuracy and less time in detection | IoT devices are limited in size and resources, so lightweight DL is required |
| Hsiao et al. [31] 2019 | SCNN, automatic | Virusshare dataset,gray image (105*105) | training on 35 & testing on 17 malware families | Accuracy 92% | Early-stage malware detection, small set of training data can be used to train a model | low accuracy& very high dimensional image processing |
| Yin et al. [41] 2019 | CNN,RNN, automatic | CIFAR 10 dataset, color image (32*32) | NA | Accuracy 80% | successfully applied CNN with RNN | low accuracy |
| Lu et al. [42] 2019 | GAN, automatic | Malimg dataset, color image (32*32) | 90%-10% | Accuracy 84% | successfully applied GAN & accuracy increased | low accuracy |
| Jain et al. [35] 2020 | CNN,ELM, automatic | Malimg dataset,gray image (128*128, 64*64) | 80% -10% & 10% for validation | Accuracy 96.3% 97.7% | 1d data analysis proved to be very effective.ELM is faster | shallow network architecture |
| Naeem et al. [36] 2020 | DCNN, automatic | Leopard mobile and & windows data set, color image (224*224, 229*229) | 70%-30% | Accuracy 97.81% 98.47% | achieved higher accuracy. Android malware issue addressed | image dimension is very big, training time is higher. |
| Kumar et al. [37] 2020 | CNN,LSTM, automatic | Malimg dataset, color image (224*224) | NA | NA | successfully applied CNN with LSTM | experimental data is not specified & the result is missing |
| Vasan et al. [38] 2020 | VGG16, ResNet-50, automatic | Malimg dataset, color image (224*224) | 70%-30% | Accuracy 99.50% | an ensemble of CNNs is effective with handcrafted features | Time-consuming due to complex & utilized deep pre-trained models |

## 3. Challenges and open issues

During the review of the recent studies, many issues and challenges of malware classification were found, and these research findings lead to myriad problems in malware classification. For effective classification, a classifier must be consistent and effective, and to build an effective classifier, it is required to consider all the issues and difficulties. A closer look at the above studies on malware classification using DL reveals some gaps and shortcomings.

**Dataset issues:**

a.) Solutions proposed by many researchers in literature are specific to the particular malware dataset not generic in nature [8].
b.) Dataset imbalance problem is another challenge in effective malware classification [11].
c.) Dataset samples to differ by specifications and formats.
d.) Dataset contains malware images of different sizes but a DL model requires square (n*n) images which limits the models.
e.) Many researchers used large image sizes which also increases training time. No specification of the image dimension.
f.) The size of the dataset affects the model's performance.
g.) Studies shows that the preprocessing of raw data increases the performance but also consumes time [7].
h.) The dataset used for training the different DL models cannot reflect real-world malware.

**DL model issues:**

a.) Malware classification is very efficiently handled by DL and visualization techniques but falls short in some aspects.
b.) DL model architectures mainly differ in the number of hidden layers, size and number of kernels/filters, size of strides, and other hyper-parameters, DL models need to be used more intelligently.
c.) DL models are influenced by the adversarial attacks.
d.) Studies shows that CNN is widely used to address the malware classification but other DL models or to combine the different DL models may improve the performance.

Although over the decade a lot of work has been already been manifested using malware visualization and DL techniques, still there is the scope of upgrading the classification task in terms of performance measures, training, and testing time, transform malware binaries into color images, to address data imbalance issue for classifying malware into their respective families, reduction in the size of the feature vector, still, images have a region that is not required in the analysis [15], we can eliminate those regions for classification and many more new DL models established [35] and yet to establish.

We have found that CNN's are the most prevailing DL models used [7], [8] for malware image classification and results so far have been very promising. One potential application of utilizing CNN with other DL models like RNN, LSTM, and Autoencoder, incorporated to get higher accuracy and performance. One of the emerging threats in malware analysis is the file-less malware [44]. It does not utilize the file system for its execution (to carry out its malicious activities), thereby eschewing traditional approaches and became one of the hurdles in malware analysis. This malware persists in the system through memory and registry files. If no code or file is available for analysis no detection is possible.

## 4. Conclusion

In this paper, we delve into an exploration of a detailed review of the recent innovations and comparison of malware visualization techniques and DL models for the detection and classification of malware and its variant. Table 2. compares and summarizes the established work concerning certain key criteria. Our review work has led us to conclude that visual analysis of malware binaries helps analysts to identify patterns in it and the evidences from the review imply that it is reasonable to continue with malwares visualization and DL approach to design a more intelligent framework to achieve accuracy, efficiency, and better performance. Future studies on the topic are therefore recommended in the following area, a large malware dataset must be used for the assessment and validation of the performance measures, more efficient techniques to convert malware binaries into color images, image sizes vary as per the dataset, and results, the dimension of feature vector should be reduced, data imbalance problem should be solved using mathematical methods and detection of unknown malware based methods should be taken into consideration. The review unveiled knowledge gaps in the existing work, major challenges, and open issues that will direct future research efforts. The findings of this study can aid to promote research in android malware detection as well as in IoT, and cloud based environments using DL methods.

## References

[1] Malware statistics and Trends Report [online] by AV-test institute, "https://www.av-test.org/en/statistics/malware/".

[2] McAfee Labs Threats Report [online] November 2020, "https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-nov-2020.pdf".

[3] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: visualization and automatic classification," In Proceedings of the 8th International Symposium on Visualization for Cyber Security (VizSec '11), Association for Computing Machinery, New York, NY, USA, Article 4, pp. 17, 2011. DOI:https://doi.org/10.1145/2016904.2016908.

[4] L. Nataraj, V. Yegneswaran, P. Porras, and J. Zhang, "A comparative assessment of malware classification using binary texture analysis and dynamic analysis," In Proceedings of the 4th ACM workshop on Security and artificial intelligence (AISec '11), Association for Computing Machinery, New York, NY, USA, pp. 2130, 2011. DOI:https://doi.org/10.1145/2046684.2046689.

[5] K. Han, J. H. Lim, and E. G. Im, "Malware analysis method using visualization of binary files," In Proceedings of the 2013 Research in Adaptive and Convergent Systems (RACS '13), Association for Computing Machinery, New York, NY, USA, pp. 317321, 2013. DOI:https://doi.org/10.1145/2513228.2513294.

[6] A. Makandar and A. Patrot, "Malware analysis and classification using Artificial Neural Network," International Conference on Trends in Automation, Communications and Computing Technology (I-TACT-15), Bangalore, pp. 1-6, 2015. DOI:https://doi.org/10.1109/ITACT.2015.7492653.

[7] K. K. Pal and K. S. Sudeep, "Preprocessing for image classification by convolutional neural networks," IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, pp. 1778-1781, 2016. DOI:https://doi.org/10.1109/RTEICT.2016.7808140.

[8] M. Kalash, M. Rochan, N. Mohammed, N. D. B. Bruce, Y. Wang and F. Iqbal, "Malware Classification with Deep Convolutional Neural Networks," 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, pp. 1-5, 2018. DOI:https://doi.org/10.1109/NTMS.2018.8328749.

[9] Y. Mourtaji, M. Bouhorma, and D. Alghazzawi, "Intelligent Framework for Malware Detection with Convolutional Neural Network," In Proceedings of the 2nd International Conference on Networking, Information Systems & Security (NISS19), Association for Computing Machinery, New York, NY, USA, Article 7, pp. 16, 2019. DOI:https://doi.org/10.1145/3320326.3320333.

[10] S. Karen and Z. Andrew, "Very deep convolutional networks for large-scale image recognition," arXiv preprint, arXiv, 2014. DOI:https://arxiv.org/abs/1409.1556.

[11] Z. Cui, F. Xue, X. Cai, Y. Cao, G. Wang and J. Chen, "Detection of Malicious Code Variants Based on Deep Learning," In IEEE Transactions on Industrial Informatics, vol. 14, no. 7, pp. 3187-3196, July 2018. DOI:https://doi.org/10.1109/TII.2018.2822680.

[12] S. Ni, Q. Qian and R. Zhang, "Malware identification using visualization images and deep learning," Computers & Security, vol. 77, pp. 871-885, 2018. DOI:https://doi.org/10.1016/j.cose.2018.04.005.

[13] A. Singh, A. Handa, N. Kumar, S.K. Shukla, "Malware Classification Using Image Representation," In: Dolev S., Hendler D., Lodha S., Yung M. (eds) Cyber Security Cryptography and Machine Learning, CSCML 2019, Lecture Notes in Computer

Science, vol. 11527, 2019. DOI:https://doi.org/10.1007/978-3-030-20951-3_6.

[14] X. Meng et al., "MCSMGS: Malware Classification Model Based on Deep Learning," 2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Nanjing, pp. 272-275, 2017. DOI:https://doi.org/10.1109/CyberC.2017.21.

[15] D. Kornish, J. Geary, V. Sansing, S. Ezekiel, L. Pearlstein and L. Njilla, "Malware Classification using Deep Convolutional Neural Networks," 2018 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), Washington, DC, USA, pp. 1-6, 2018. DOI:https://doi.org/10.1109/AIPR.2018.8707429.

[16] Y. Ding, S. Chen and J. Xu, "Application of Deep Belief Networks for opcode based malware detection," 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, pp. 3901-3908, 2016. DOI:https://doi.org/10.1109/IJCNN.2016.7727705.

[17] W. Hardy, L. Chen, S. Hou, Y. Ye and X. Li, "DL 4 MD: A Deep Learning Framework for Intelligent Malware Detection," 2016.

[18] S. Tobiyama, Y. Yamaguchi, H. Shimada, T. Ikuse and T. Yagi, "Malware Detection with Deep Neural Network Using Process Behavior," 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), Atlanta, GA, pp. 577-582, 2016. DOI:https://doi.org/10.1109/COMPSAC.2016.151.

[19] B. Dong and X. Wang, "Comparison deep learning method to traditional methods using for network intrusion detection," 2016 8th IEEE International Conference on Communication Software and Networks (ICCSN), Beijing, pp. 581-585, 2016. DOI:https://doi.org/10.1109/ICCSN.2016.7586590.

[20] R. Kumar, Z. Xiaosong, R. U. Khan, I. Ahad, and J. Kumar, "Malicious Code Detection based on Image Processing Using Deep Learning," In Proceedings of the 2018 International Conference on Computing and Artificial Intelligence (ICCAI 2018), Association for Computing Machinery, New York, NY, USA, pp. 8185. 2018. DOI:https://doi.org/10.1145/3194452.3194459.

[21] D. Gavrilu, M. Cimpoeu, D. Anton and L. Ciortuz, "Malware detection using machine learning," 2009 International Multiconference on Computer Science and Information Technology, Mragowo, 2009, pp. 735-741, 2009. DOI:https://doi.org/10.1109/IMCSIT.2009.5352759.

[22] J. Su, D. V. Vasconcellos, S. Prasad, D. Sgandurra, Y. Feng and K. Sakurai, "Lightweight Classification of IoT Malware Based on Image Recognition," 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, pp. 664-669, 2018. DOI:https://doi.org/10.1109/COMPSAC.2018.10315.

[23] K. D. T. Nguyen, T. M. Tuan, S. H. Le, A. P. Viet, M. Ogawa and N. L. Minh, "Comparison of Three Deep Learning-based Approaches for IoT Malware Detection," 2018 10th International Conference on Knowledge and Systems Engineering (KSE), Ho Chi Minh City, pp. 382-388, 2018. DOI:https://doi.org/10.1109/KSE.2018.8573374.

[24] M. Hasan, Md. M. Islam, Md. I. I. Zarif, M.M.A. Hashem, "Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches," Internet of Things, vol. 7, 2019. DOI:https://doi.org/10.1016/j.iot.2019.100059.

[25] H. Naeem, "Detection of Malicious Activities in Internet of Things Environment Based on Binary Visualization and Machine Intelligence," Wireless Pers Communication 108, pp. 26092629, 2019. DOI:https://doi.org/10.1007/s11277-019-06540-6.

[26] E. Kabanga and C. Kim, "Malware Images Classification Using Convolutional Neural Network", Journal of Computer and Communications, vol. 6, pp. 153-158, 2018. DOI:https://doi.org/10.4236/jcc.2018.61016.

[27] D. Li and D. Yu, "Deep Learning: Methods and Applications," Foundations and Trends R in Signal Processing, 7(34), pp. 197387, June 2014. DOI:https://doi.org/10.1561/2000000039.

[28] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F. E. Alsaadi, "A survey of deep neural network architectures and their applications," Neurocomputing, vol. 234, pp. 11-26, 2017. DOI:https://doi.org/10.1016/j.neucom.2016.12.038.

[29] K. Donghwoon, K. Hyunjoo, K. Jinoh, S. Sang, K. Ikkyun and K. Kuinam, "A survey of deep learning-based network anomaly detection," Cluster Computing, vol. 22, pp. 949961, 2019. DOI:https://doi.org/10.1007/s10586-017-1117-8.

[30] J. Yang , J. Deng , S. Li et al., "Improved traffic detection with support vector machine based on restricted Boltzmann machine," Soft Computing, vol. 21, pp. 31013112, 2017. DOI:https://doi.org/10.1007/s00500-015-1994-9.

[31] S. C. Hsiao, D. Y. Kao, Z. Liu, R. Tso, "Malware Image Classification Using One-Shot Learning with Siamese Networks," Procedia Computer Science, vol. 159, pp. 1863-1871, 2019. DOI:https://doi.org/10.1016/j.procs.2019.09.358.

[32] Y. Wang, J. An and W. Huang, "Using CNN-based Representation Learning Method for Malicious Traffic Identification," 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS), Singapore, pp. 400-404, 2018. DOI:https://doi.org/10.1109/ICIS.2018.8466404.

[33] W. Wang, M. Zhu, X. Zeng, X. Ye and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," 2017 International Conference on Information Networking (ICOIN), Da Nang, pp. 712-717, 2017. DOI:https://doi.org/10.1109/ICOIN.2017.7899588.

[34] M. Kumari, G. Hsieh and C. A. Okonkwo, "Deep Learning Approach to Malware Multi-class Classification Using Image Processing Techniques," 2017 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, pp. 13-18, 2017. DOI:https://doi.org/10.1109/CSCI.2017.3.

[35] M. Jain, W. Andreopoulos and M. Stamp, "Convolutional neural networks and extreme learning machines for malware classification," J Comput Virol Hack Tech 16, pp. 229244, 2020. DOI:https://doi.org/10.1007/s11416-020-00354-y.

[36] H. Naeem, U. Farhan, N. M. Rashid, K. Shehzad, V. Dan-

ish, J. Sohail and S. Saqib, "Malware Detection in Industrial Internet of Things based on Hybrid Image Visualization and Deep Learning Model," Ad Hoc Networks 105, 2020. DOI:https://doi.org/10.1016/j.adhoc.2020.102154.

[37] G. S. Kumar, P. Bagane, "Detection Of Malware Using Deep Learning Techniques," International journal of scientific & technology research, vol. 9, issue 01, pp. 1688-1691, January 2020.

[38] D. Vasan, M. Alazab, S. Wassan, B. Safaei, Q. Zheng, "Image-Based malware classification using ensemble of CNN architectures (IMCEC)," Computers & Security, vol. 92, 2020. DOI:https://doi.org/10.1016/j.cose.2020.101748.

[39] C. H. Kim, E. K. Kabanga and S. Kang, "Classifying malware using convolutional gated neural network," 2018 20th International Conference on Advanced Communication Technology (ICACT), Chuncheon-si Gangwon-do, Korea (South), pp. 1-1, 2018. DOI:https://doi.org/10.23919/ICACT.2018.8323639.

[40] B. Chandra and R. K. Sharma, "On improving recurrent neural network for image classification," 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, pp. 1904-1907, 2017. DOI:https://doi.org/10.1109/IJCNN.2017.7966083.

[41] Y. Qiwei, Z. Ruixun and S. XiuLi, "CNN and RNN mixed model for image classification," MATEC Web of Conferences, 277, 2019. DOI:https://doi.org/10.1051/matecconf/201927702001.

[42] Y. Lu and J. Li, "Generative Adversarial Network for Improving Deep Learning Based Malware Classification," 2019 Winter Simulation Conference (WSC), National Harbor, MD, USA, pp. 584-593, 2019. DOI:https://doi.org/10.1109/WSC40007.2019.9004932.

[43] JY. Kim, SJ. Bu, SB. Cho, "Malware Detection Using Deep Transferred Generative Adversarial Networks," In: Liu D., Xie S., Li Y., Zhao D., El-Alfy ES. (eds) Neural Information Processing, - 24th International Conference, ICONIP 2017, vol. 10634 , pp. 556-564, 2017. DOI:https://doi.org/10.1007/978-3-319-70087-8_58.

[44] K. Sudhakar and K. Sushil, "An emerging threat Fileless malware: a survey and research challenges," Cybersecurity, vol. 3, no. 1, 2020. DOI:https://doi.org/10.1186/s42400-019-0043-x.