Deep Learning Based Classification of Time Series of Chaotic Systems over Graphic Images

Süleyman UZUN (suleymanuzun@subu.edu.tr)
Sakarya Uygulamali Bilimler Universitesi https://orcid.org/0000-0001-8246-6733
Sezgin KAÇAR
Sakarya Uygulamali Bilimler Universitesi
Burak ARICIOĞLU
Sakarya Uygulamali Bilimler Universitesi

Research Article

Keywords: Chaotic systems, time series, classification, deep learning

Posted Date: December 10th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1138927/v1

Abstract

In this study, for the first time in the literature, identification of different chaotic systems by classifying graphic images of their time series with deep learning methods is aimed. For this purpose, a data set is generated that consists of the graphic images of time series of the most known three chaotic systems: Lorenz, Chen, and Rossler systems. The time series are obtained for different parameter values, initial conditions, step size and time lengths. After generating the data set, a high-accuracy classification is performed by using transfer learning method. In the study, the most accepted deep learning models of the transfer learning methods are employed. These models are SqueezeNet, VGG-19, AlexNet, ResNet50, ResNet101, DenseNet201, ShuffleNet and GoogLeNet. As a result of the study, classification accuracy is found between 96% and 97% depending on the problem. Thus, this study makes association of real time random signals with a mathematical system possible.

1. Introduction

Chaotic systems are nonlinear mathematical models that are used to define chaotic behaviour. In other words, chaotic systems are nonlinear systems that exhibit chaotic behaviour. In the recent years, chaotic systems are used in many different areas of engineering such as cryptography, image and audio encryption, secure communication, data security, random number generation, digital signature applications, weak signal detection and DC-DC converters. Another hot topic in the recent years is deep learning. There are a great number of studies on deep learning, however most of these studies focus on classification processes in different areas. In this study, two of the most topical subjects of the literature, namely chaos and deep learning, are focused on and time series of the chaotic systems are classified with deep learning.

Deep learning is a subbranch of machine learning [1] that employs multi-layer artificial neural network on different areas such as image processing [2], [3], voice recognition [4], [5], natural language processing [6]–[8], and object recognition [9]. The difference of deep learning from the traditional machine learning algorithms is that deep learning makes automatic learning from a database (image, audio, video etc.) possible instead of using predetermined rules [10]–[12]. To the best of the author knowledge, there is no deep learning-based classification study on the images of chaotic systems. However, in the literature, there are studies in which deep learning based classification of the chaotic systems from signals is carried out.

In their study, Boulle et al. [13], have used ShallowNet, Multilayer perceptrons – MLP, the Fully Convolutional Neural Network – FCN, Residual Network – ResNet, Large Kernel Convolutional Neural Network – LKCNN methods to classify time series of the discrete and continuous time dynamic systems. They stated that the highest classification performance is achieved with LKCNN method. Yeo [14], used the Long Short Term Memory Network – LSTM to predict a chaotic dynamical system from observed noise. He concluded that LSTM predicts the nonlinear dynamics with high accuracy by filtering out the noise effectively. Kuremoto et al. [15], proposed generation of Deep Belief Nets – DBNs using Restricted

Boltzmann Machine – RBM and Multi-Layer Perceptron – MLP to predict time series of Lorenz and Henon map chaotic systems. They showed that the prediction sensitivity of their proposed DBNs is higher than traditional DBNs. Sangiorgio et al. [16], compared the prediction accuracy and fidelity of their four proposed models (namely, FF-recursive, FF-multi-output, LSTM-TF and LSTM-no-TF). These models are based on Feed-Forward - FF and Teacher Forcing – TF education model and Recurrent Neural Networks of the artificial, noise free time series of the chaotic oscillator. They show that LSTM predictor gives much better results than FF-recursive and FF-multi-output methods.

As it can be seen in the mentioned studies, while there is very little study on deep learning-based classification of the chaotic signals, there is no classification of chaotic systems over graphic images. Yet, classification of chaotic behaviours or random signals is seen as a necessity. In this article, a method to classify time series of chaotic systems based on SqueezeNet, VGG-19, AlexNet, ResNet50, ResNet101, DenseNet201, ShuffleNet and GoogLeNet of transfer learning methods is presented to address the mentioned necessity. An original study is presented by classifying time series of two different chaotic system with high accuracy, for the first time in the literature.

In the second section of the study, the used chaotic systems and data set obtains from these chaotic systems are presented. In the third section of the study, the deep learning methods used in the classification is briefly presented. In the fourth section of the study, simulation of the classification processes and their performance results are presented. Finally, in the last section the conclusion is presented.

2. Used Caotic Systems And Obtaining Of Data Set

In this section, the three different chaotic systems (namely, Lorenz, Chen and Rossler) whose time series to be classified and data set constructed from time series of the three chaotic systems are mentioned. In the literature, there are a very large number of chaotic systems. Hence, some criteria are considered for the selection of the chaotic systems that are used in the study. These Lorenz, Chen and Rossler systems are selected because of they are the most known systems in the literature, 3 dimensional and their mathematical models contains similar nonlinear terms. In addition to this, they can be used to model real life physical systems such as atmospheric, electrical, or chemical systems. Moreover, the time series and phase portraits of the Lorenz and Chen systems are similar and the time series and phase portraits of Rossler systems is different from that of the other two systems makes selection of these systems is logical in terms of evaluation of classification performance.

2.1 Lorenz System

Lorenz system was developed by Edward Lorenz in 1963 as a simplified mathematical model of atmospheric convection [17], [18]. Lorenz system consists of three ordinary differential equations as seen in Equation 1. Here α, β and γ are the system parameters and x, y and z are the state variables of the system.

$$\dot{x} = \alpha(y - x)$$
$$\dot{y} = x(\gamma - z) - y \quad (1)$$
$$\dot{z} = xy - \beta z$$

Lorenz systems is also used as a simplified mathematical model of thermosiphons [19], lasers [20], electrical circuits [21], brushless DC motors [22], dynamos [23], and chemical reactions [21]. In Figure 1, the time series and phase portraits of the Lorenz system is given for the system parameters $\alpha = 10$, $\beta = 8/3$, $\gamma = 28$ and the initial values $x_0 = 5$, $y_0 = -5$ and $z_0 = 7.5$. As it is seen in Figure 1, the time series has random variations and the phase portraits which show the relationship between two state variables have their own order and orbit. By utilizing this feature of the chaotic systems, the chaotic systems can be used in areas like encryption and data security.

$$\dot{x} = \alpha(y - x)$$
$$\dot{y} = (\gamma - \alpha)x - xz + \gamma y \quad (2)$$
$$\dot{z} = xy - \beta z$$

2.2 Chen System

Guanrong Chen and Ueta were proposed a double scroll chaotic system in 1999. This system is called Chen System or Chen Chaotic Attractor which resembles the Lorenz System. However, the Chen System is not equivalent to the Lorenz System [24], [25]. Chen System also consists of three ordinary differential equations as shown in Equation 2. Here α, β and γ are the system parameters and x, y and z are the state variables of the system.

2.3 Rossler System

The Rossler system, developed by Otto Rossler in 1976, is used in modelling of chemical reactions [26], [27]. Rossler system also consists of three ordinary differential equations as the previous two systems. The Rossler system is given in Equation 3. Here α, β and γ are the system parameters and x, y and z are the state variables of the system.

$$\dot{x} = -y - z$$
$$\dot{y} = x + \alpha y \quad (3)$$
$$\dot{z} = \beta + z(x - \gamma)$$

Rossler system resembles Lorenz systems equation-wise, but Rossler system is easier to analyse, and it is a single scroll system. In Figure 3, the time series and phase portraits of the Rossler system is given for the system parameters $\alpha = 0.1$, $\beta = 0.1$, $\gamma = 14$ and the initial values $x_0 = 11$, $y_0 = 11$ and $z_0 = 0$. Since the time series and phase portraits of the Rossler system is different than those of the other two systems, the

time series of the Rossler system will be easily classified. Hence, the performance of the deep learning-based classification method on an easy problem will be evaluated.

2.4 Creating the Data Set

In this section, the information regarding constructed data set is mentioned. To calculate the state variables of all systems and to obtain the time series which constitute the data set, Runge-Kutta 4 (RK4) algorithm is used. Moreover, the values of the calculation and system parameters are varied to diversify time series data. For every time series, 750 different results are obtained by changing the length of time series (the number of total calculated points), the step size of RK4 algorithm, the values of the system parameters, or the values of the initial conditions. Since all the systems are a 3-dimensional system, 2250 different time series are generated for every system. Hence, the data set is generated by calculating 6750 different time series. In Table 1, the values of calculation and system parameters are given.

Table 1. The used calculation and system parameters.

System	System parameters (Only parameters are changed for every system)	the step size of RK4 algorithm	the length of time series (the number of total calculated points)	Initial conditions
Lorenz	β=2.5, 3, 3.5, 4, 4.5, 5	0.01 0.02 0.05 0.1 0.2	10000, 12500, 15000, 17500, 20000	X_0=0.1, 0.2, 0.3, 0.4, 0.5 Y_0=0.3, 0.4, 0.5, 0.6,0.7 Z_0=0.4, 0.5, 0.6, 0.7, 0.8
Chen	β=0.05, 0.1, 0.15, 0.2, 0.25, 0.3		20000, 25000, 30000, 35000, 40000	X_0=8, 9, 10, 11, 12 Y_0=8, 9, 10, 11, 12 Z_0=0, 1, 2, 3, 4
Rossler	β=2.5, 2.55, 2.6, 2.65, 2.7, 2.75		2000, 4000, 6000, 8000, 1000	X_0=8, 9, 10, 11, 12 Y_0=-8, -9, -10, -11, -12 Z_0=13, 14, 15, 16, 17

Since the transfer learning method is very efficient at the classification of images, all the graph of the obtained time series are saved as a 128x128 pixel image. The data set contains 6750 different graphic images of the time series.

3. The Used Deep Learnİng Models

In this study, a deep learning-based algorithm to classify the *x*, *y*, and *z* time series of the three chaotic systems is presented. The data set contains images of the time series of the three chaotic systems. The time series are obtained with RK4 algorithm, then the graph of the time series is saved as an image. Experimental works were performed utilizing eight different pre-trained deep learning nets (namely, *VGG-19, AlexNet, ResNet50, ResNet-101, DenseNet-201, ShuffleNet and GoogLeNet*) on the images obtained from the chaotic systems.

3.1 Deep Learning and Convolutional Neural Networks – CNN

Deep learning is a subbranch of machine learning algorithms and tries to learn differences on data by employing different architectures [28]. Deep learning methods based on artificial neural network perform learning process by employing multilayer neural networks [29], [30].

Machine learning algorithms are traditionally designed for analysing data sets with very few features. On the other hand, convolutional neural network (CNN) is developed to analyse data sets with a great number of features where machine learning algorithms are not suitable to analyse such data sets. Since the size of image data is high and every image contains hundreds or thousands of pixels, the CNNs perform much better than traditional machine learning algorithms for processing such data sets [31].

Lecun et al. proposed LeNet networks to analyse large size images in 1988 and they had developed this network until 1998. LeNet networks accepted as the first of the CNN models [32], [33]. LeNet networks consists of sublayers of successive convolutional and maximum pooling layers. The layers corresponding to Multi-Layer Perceptron-MLP make the top layers. The weights of the network are trained to minimize the average error between obtained and predicted results [31].

In the literature, there are many proposed CNN network models. Some of these are *SqueezeNet, VGG-19, AlexNet, ResNet50, ResNet-101, DenseNet-201, ShuffleNet*, and *GoogLeNet* [34]. With these pre-trained networks, the operations of CNN models can be carried out faster, the prediction performance can be improved, and training processes can be performed faster. The performance of the networks depends on the problem they applied. Hence, these networks applied to the problem then the pre-trained network with best performance is selected. In this study, many different networks have been tested and the ones with highest performance are selected.

3.1 GoogLeNet

Szegedy et al. proposed GoogLeNet which contains 22 layers and is a pre-trained CNN network architecture. In ILSVRC2014 contest, GoogLeNet was selected as the most successful network. [35]–[37]. As shown in Figure 5, the architecture of the network employs parallel connected layers to minimize the possibility of memorization.

The inception modules that enable the realization of multi-core convolution and maximum pooling at one layer simultaneously provide training of the network with optimum weights and extraction of more useful features [39]. To achieve this, every starting layer has 1×1, 3×3 and 5×5 variable sized convolutional core and every layer uses an extra 3×3 maximum pooling layer to extract more distinctive features with respect to the ones obtained from the previous layer [38].

3.1 VGG-19

VGG-19 network architecture consists of total 24 layers. 16 of them are convolutional layers which have filter size of 3×3. These layers are used to decrease number of filter parameters. Moreover, VGG-19 network has 5 pooling layers and 3 fully connected layers [40], [41].

3.2 SqueezeNet

SqueezeNet CNN network was proposed by Iandola et al. in 2016. This network was developed by improving AlexNet architecture. The difference between these two networks is that AlexNet CNN network has 240MB of parameters while SqueezeNet has 5MB of parameters. Moreover, the performance of SqueezeNet is almost good as that of AlexNet. SqueezeNet network contains Fire layers. In this layer, the number of features to be calculated is decreased by reducing the filter size to 1×1 [43]–[45]. Hence, SqueezeNet performs faster than AlexNet by decreasing the workload of neural network [46].

3.3 ResNet50 and ResNet101

ResNet50 network architecture consists of 152 layers and won the ImageNet contest in 2015 [47]. The architecture contains convolution layer, activation layer, pooling layer, and fully connected layer [48].

There are 5 convolutional blocks that contains 1×1, 3×3 and 1×1 convolutional layers in the structure of the architecture [49]. The global averaging pooling layer and the two-step sampling process in the architecture shrink the size of images [49]. In the fully connected layer, softmax is used as an activation function.

Table 2. Differences between ResNet architectures [47].

Layer	Output size	18-Layer	34-Layer	50-Layer	101-Layer	152-Layer
Conv1	112x112	7x7, 64, stride 2				
		3x3, max pool, stride 2				
Conv2_x	56x56	$\begin{bmatrix}3\times3 & 64\\ 3\times3 & 64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3 & 64\\ 3\times3 & 64\end{bmatrix}\times3$	$\begin{bmatrix}1\times1 & 64\\ 3\times3 & 64\\ 1\times1 & 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1 & 64\\ 3\times3 & 64\\ 1\times1 & 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1 & 64\\ 3\times3 & 64\\ 1\times1 & 256\end{bmatrix}\times3$
Conv3_x	28x28	$\begin{bmatrix}3\times3 & 128\\ 3\times3 & 128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3 & 128\\ 3\times3 & 128\end{bmatrix}\times4$	$\begin{bmatrix}1\times1 & 128\\ 3\times3 & 128\\ 1\times1 & 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1 & 128\\ 3\times3 & 128\\ 1\times1 & 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1 & 128\\ 3\times3 & 128\\ 1\times1 & 512\end{bmatrix}\times8$
Conv4_x	14x14	$\begin{bmatrix}3\times3 & 256\\ 3\times3 & 256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3 & 256\\ 3\times3 & 256\end{bmatrix}\times6$	$\begin{bmatrix}1\times1 & 256\\ 3\times3 & 256\\ 1\times1 & 1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1 & 256\\ 3\times3 & 256\\ 1\times1 & 1024\end{bmatrix}\times23$	$\begin{bmatrix}1\times1 & 256\\ 3\times3 & 256\\ 1\times1 & 1024\end{bmatrix}\times36$
Conv5_x	7x7	$\begin{bmatrix}3\times3 & 512\\ 3\times3 & 512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3 & 512\\ 3\times3 & 512\end{bmatrix}\times3$	$\begin{bmatrix}1\times1 & 512\\ 3\times3 & 512\\ 1\times1 & 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1 & 512\\ 3\times3 & 512\\ 1\times1 & 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1 & 512\\ 3\times3 & 512\\ 1\times1 & 2048\end{bmatrix}\times3$
	1x1	avg pool, 1,000-d fully connected, Softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

3.4 DenseNet-201

DenseNet is a CNN model that is used for classification problem and is a dense convolutional network with its dense connection model. The input of the architecture of DenseNet is 224×224×3 sized images as shown in Figure 10.

The dense blocks of the architecture consist of normalization layer, ReLU layer and 3×3 convolutional layer [51]. DenseNet architecture connects the layers instead of adding up as in the previous architectures like ResNet. The connection layers combine all the features at the previous layers and transfer these combined features to the next layers. While, in the other architectures, the extracted feature maps are transferred to the next layers, in DenseNet architecture, the excess feature maps are removed. Hence, DenseNet architecture prevents the retraining of these excess features in the next layers [52].

3.5 ShuffleNet

ShuffleNet was proposed by Zhang et al. for mobile devices with low processing power and it is a very well performed CNN model [53]. While this network architecture decreases computation cost, it preserves accuracy. Moreover, it achieves a superior performance than other architectures on ImageNet classification and MS COCO object recognition studies [53]. The architecture of ShuffleNet is given in Table 3.

Table 3
ShuffleNet architecture.

Layer	Output size	K size	Stride	Repeat	Output Channels (g groups)				
					g=1	g=2	g=3	g=4	g=8
Image	224x224				3	3	3	3	3
Conv1	112x112	3x3	2	1	24	24	24	24	24
MaxPool	56x56	3x3	2						
Stage2	28x28		2	1	144	200	240	272	384
	28x28		1	3	144	200	240	272	384
Stage3	14x14		2	1	288	400	480	544	768
	14x14		1	7	288	400	480	544	768
Stage4	7x7		2	1	576	800	960	1088	1536
	7x7		1	3	576	800	960	1088	1536
GlobalPool	1x1	7x7							
FC					1000	1000	1000	1000	1000
Complexity					143M	140M	137M	133M	137M

As shown in Table 3, the network model contains a stack of ShuffleNet units that is grouped in three stages. In every stage, the structure of every first block starts with 2 steps. While some parameters in one stage remain the same, the output channels are double for the next stage [54], [55]. The group number g given in Table 3, controls the connection sparsity of pointwise convolutions

3.6 AlexNet

It is a CNN network proposed by Krizhevsky et al. who won ImageNet contest in 2012 [56], [57]. AlexNet architecture consists of successive convolution layers, max-pooling layers, and fully connected layers and it uses ReLU (Rectified Linear Unit) as the activation function [58], [59]. This architecture is usually preferred for image classification [37]. In Figure 11, the architecture of AlexNet is shown.

4. Simulation Results And Evaluation Of Performance

In the experimental study, two different study groups were generated, and three different experimental works were performed for every group. Moreover, two different data sets were generated for every study group. In the first study group, the images obtained from Lorenz and Rossler chaotic systems were used. In the second group, the images obtained from Lorenz and Chen chaotic systems were used. In every

group, the time series images of the *x, y* and *z* state variables of the chaotic systems were classified separately, and their classification performance was evaluated.

In the data set which contains the images of the time series of Lorenz and Rossler chaotic system, 600 images of the x state variable of Lorenz chaotic system are created for the first class and 600 images of the x state variable of Rossler chaotic system are created for the second class. By using various pre-trained networks, the classification tests were performed on these total 1200 images. In the data set which contains the images of the time series of Lorenz and Chen chaotic system, 720 images of the *x* state variable of Chen chaotic system are created for the first class and 600 images of the *x* state variable of Lorenz chaotic system are created for the second class. By using various pre-trained networks, the classification tests were performed on these total 1300 images. In the study, tests were conducted with numerous pre-trained networks and the 8 pre-trained networks with highest performance were used. These networks are Squeezenet, VGG-19, AlexNet, ResNet50, ResNet101, DenseNet201, ShuffleNet, and GoogLeNet.

In the study, the classification of the images of the time series of totally different systems like Lorenz-Rossler and the classification of the images of the time series of similar systems like Lorenz – Chen is performed by utilizing deep neural networks. The purpose here is evaluation of classification performance of different time series images and similar time series images and compare these performance results.

5.1 Evaluation of results and performance of Lorenz - Rossler systems

In the first experimental study of this group, the images obtained from the time series of x state variable of Lorenz and Rossler chaotic systems are classified. In Figure 12, the sample images of the time series of x state variable of Lorenz chaotic system are given.

In Figure 13, the sample images of the time series of x state variable of Rossler chaotic system are given.

As it is seen in Figures 12 and 13, the series of the two systems are very different from each other and this makes the performance of the classification to be very high. Thanks to the preferred pre-trained networks and the optimizations performed on these networks, higher classification performance results are observed in the study. In Table 4, the opposition matrices of the images shown in Figures 12 and 13 are given. From Table 4, the classification performance of the pre-trained networks is seen clearly.

Table 4
x-coordinate time series opposition matrices of the Lorenz and Rossler chaotic systems.

Network	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
SqueezeNet	180	169	0	11
VGG-19	153	179	27	1
AlexNet	170	180	10	0
ResNet50	179	163	1	17
ResNet101	166	177	14	3
DenseNet201	167	180	13	0
ShuffleNet	176	151	4	29
GoogLeNet	173	168	7	12

In Table 5, the performance of the pre-trained networks on the classification of images shown in Figure 12 and 13. As it is seen in Table 5, the highest classification performance is obtained with SqueezeNet and DenseNet201.

Table 5
x-coordinate time series performance metrics of the Lorenz and Rosler chaotic systems.

Network	Accuracy	Precision	Sensitivity	Specificity
SqueezeNet	0.969444	**1.000000**	0.942408	**1.000000**
VGG-19	0.922222	0.850000	0.993506	0.868932
AlexNet	**0.972222**	0.944444	**1.000000**	0.947368
ResNet50	0.950000	0.994444	0.913265	0.993902
ResNet101	0.952778	0.922222	0.982249	0.926702
DenseNet201	0.963889	0.927778	**1.000000**	0.932642
ShuffleNet	0.908333	0.977778	0.858537	0.974194
GoogLeNet	0.947222	0.961111	0.935135	0.960000

In the second experimental study of this group, the images obtained from the time series of *y* state variable of Lorenz and Rossler chaotic systems are classified. In Figure 14, the sample images of the time series of y state variable of Lorenz chaotic system are given.

In Figure 15, the sample images of the time series of y state variable of Rossler chaotic system are given.

As it is seen in Figures 14 and 15, the series of the two systems are very different from each other and this makes the performance of the classification to be very high. Thanks to the preferred pre-trained networks and the optimizations performed on these networks, higher classification performance results are observed in the study. In Table 6, the opposition matrices of the images shown in Figures 14 and 15 are given. From Table 6, the classification performance of the pre-trained networks is seen clearly.

Table 6
y-coordinate time series opposition matrices of the Lorenz and Rosler chaotic systems.

Network	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
SqueezeNet	178	164	2	16
VGG-19	177	180	3	0
AlexNet	166	167	14	13
ResNet50	176	180	4	0
ResNet101	164	180	16	0
DenseNet201	178	173	2	7
ShuffleNet	160	180	20	0
GoogLeNet	176	179	4	1

In Table 7, the performance of the pre-trained networks on the classification of images shown in Figure 14 and 15. As it is seen in Table 7, the highest classification performance is obtained with SqueezeNet and DenseNet201.

Table 7
y-coordinate time series performance metrics of the Lorenz and Rosler chaotic systems.

Network	Accuracy	Precision	Sensitivity	Specificity
SqueezeNet	0.950000	0.988889	0.917526	0.987952
VGG-19	0.991667	0.983333	**1.000000**	0.983607
AlexNet	0.925000	0.922222	0.927374	0.922652
ResNet50	0.988889	0.977778	1.000000	0.978261
ResNet101	0.955556	**0.911111**	1.000000	**0.918367**
DenseNet201	**0.975000**	0.988889	0.962162	0.988571
ShuffleNet	0.944444	0.888889	1.000000	0.900000
GoogLeNet	0.986111	0.977778	0.994350	0.978142

In the last experimental study of this group, the images obtained from the time series of z state variable of Lorenz and Rossler chaotic systems are classified. In Figure 16, the sample images of the time series of z state variable of Lorenz chaotic system are given.

In Figure 17, the sample images of the time series of z state variable of Rossler chaotic system are given.

As it is seen in Figures 16 and 17, the series of the two systems are very different from each other and this makes the performance of the classification to be very high. Thanks to the preferred pre-trained networks and the optimizations performed on these networks, higher classification performance results are observed in the study. In Table 8, the opposition matrices of the images shown in Figures 16 and 17 are given. From Table 8, the classification performance of the pre-trained networks is seen clearly.

Table 8
z-coordinate time series opposition matrices of the Chen and Rossler chaotic systems.

Network	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
SqueezeNet	180	176	0	4
VGG-19	173	180	7	0
AlexNet	180	163	0	17
ResNet50	162	180	18	0
ResNet101	176	180	4	0
DenseNet201	179	180	1	0
ShuffleNet	172	180	8	0
GoogLeNet	179	180	1	0

In Table 9, the performance of the pre-trained networks on the classification of images shown in Figure 16 and 17. As it is seen in Table 9, the highest classification performance is obtained with SqueezeNet and DenseNet201.

Tablo 9. *z*-coordinate time series performance metrics of the Lorenz and Rossler chaotic systems.

Network	Accuracy	Precision	Sensitivity	Specificity
SqueezeNet	0.988889	**1.000000**	0.978261	**1.000000**
VGG-19	0.980556	0.961111	**1.000000**	0.962567
AlexNet	0.952778	**1.000000**	0.913706	**1.000000**
ResNet50	0.950000	0.900000	**1.000000**	0.909091
ResNet101	0.988889	0.977778	**1.000000**	0.978261
DenseNet201	**0.997222**	0.994444	**1.000000**	0.994475
ShuffleNet	0.977778	0.955556	**1.000000**	0.957447
GoogLeNet	**0.997222**	0.994444	**1.000000**	0.994475

5.2 Evaluation of results and performance of Lorenz - Chen systems

In the first experimental study of this group, the images obtained from the time series of x state variable of Lorenz and Chen chaotic systems are classified. In Figure 12, the sample images of the time series of *x*

state variable of Lorenz chaotic system are given. In Figure 18, the sample images of the time series of x state variable of Chen chaotic system are given.

The time series given in Figure 12 and 18 of Lorenz and Chen systems, respectively, are close to each other since Lorenz and Chen systems are similar systems. This makes the classification of these signals difficult. However, high classification performance results are observed in the study with the help of the preferred pre-trained networks and the optimizations performed on these networks. In Table 10, the opposition matrices of the images shown in Figures 12 and 18 are given. From Table 10, the classification performance of the pre-trained networks is seen clearly.

Table 10
x-coordinate time series opposition matrices of the Lorenz and Chen chaotic systems.

Network	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
SqueezeNet	175	209	5	7
VGG-19	178	186	2	30
AlexNet	144	213	36	3
ResNet50	173	200	7	16
ResNet101	176	216	4	0
DenseNet201	158	216	22	0
ShuffleNet	174	176	6	40
GoogLeNet	164	215	16	1

In Table 11, the performance of the pre-trained networks on the classification of images shown in Figure 12 and 18. As it is seen in Table 11, the highest classification performance is obtained with SqueezeNet and DenseNet201.

Table 11
x-coordinate time series performance metrics of the Lorenz and Chen chaotic systems.

Network	Accuracy	Precision	Sensitivity	Specificity
SqueezeNet	0.969697	0.972222	0.961538	0.976636
VGG-19	0.919192	**0.988889**	0.855769	**0.989362**
AlexNet	0.901515	0.800000	0.979592	0.855422
ResNet50	0.941919	0.961111	0.915344	0.966184
ResNet101	**0.989899**	0.977778	**1.000000**	0.981818
DenseNet201	0.944444	0.877778	**1.000000**	0.907563
ShuffleNet	0.883838	0.966667	0.813084	0.967033
GoogLeNet	0.957071	0.911111	0.993939	0.930736

In the second experimental study of this group, the images obtained from the time series of *y* state variable of Lorenz and Chen chaotic systems are classified. In Figure 14, the sample images of the time series of y state variable of Lorenz chaotic system are given. In Figure 19, the sample images of the time series of y state variable of Chen chaotic system are given.

The time series given in Figure 14 and 19 of Lorenz and Chen systems, respectively, are close to each other since Lorenz and Chen systems are similar systems. This makes the classification of these signals difficult. However, high classification performance results are observed in the study with the help of the preferred pre-trained networks and the optimizations performed on these networks. In Table 12, the opposition matrices of the images shown in Figures 14 and 19 are given. From Table 12, the classification performance of the pre-trained networks is seen clearly.

Table 12
y-coordinate time series opposition matrices of the Chen and Rossler chaotic systems.

Network	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
SqueezeNet	177	202	3	14
VGG-19	180	191	0	25
AlexNet	167	191	13	25
ResNet50	180	202	0	14
ResNet101	153	216	27	0
DenseNet201	165	215	15	1
ShuffleNet	176	176	4	40
GoogLeNet	175	210	5	6

In Table 13, the performance of the pre-trained networks on the classification of images shown in Figure 14 and 19. As it is seen in Table 13, the highest classification performance is obtained with SqueezeNet and DenseNet201.

Table 13
y-coordinate time series performance metrics of the Lorenz and Chen chaotic systems.

Network	Accuracy	Precision	Sensitivity	Specificity
SqueezeNet	**0.957071**	0.983333	0.926702	0.985366
VGG-19	0.936869	1.000000	**0.878049**	1.000000
AlexNet	0.904040	0.927778	0.869792	0.936275
ResNet50	0.964646	1.000000	0.927835	1.000000
ResNet101	0.931818	**0.850000**	1.000000	**0.888889**
DenseNet201	**0.959596**	0.916667	0.993976	0.934783
ShuffleNet	0.888889	0.977778	0.814815	0.977778
GoogLeNet	0.972222	0.972222	0.966851	0.976744

In the last experimental study of this group, the images obtained from the time series of z state variable of Lorenz and Chen chaotic systems are classified. In Figure 16, the sample images of the time series of z state variable of Lorenz chaotic system are given In Figure 20, the sample images of the time series of z state variable of Chen chaotic system are given.

The time series given in Figure 16 and 20 of Lorenz and Chen systems, respectively, are close to each other since Lorenz and Chen systems are similar systems. This makes the classification of these signals difficult. However, high classification performance results are observed in the study with the help of the preferred pre-trained networks and the optimizations performed on these networks. In Table 14, the opposition matrices of the images shown in Figures 16 and 20 are given. From Table 14, the classification performance of the pre-trained networks is seen clearly.

Table 14
z-coordinate time series opposition matrices of the Lorenz and Chen chaotic systems.

Network	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
SqueezeNet	170	209	10	7
VGG-19	167	194	13	22
AlexNet	150	212	30	4
ResNet50	175	205	5	11
ResNet101	171	202	9	14
DenseNet201	173	190	7	26
ShuffleNet	153	194	27	22
GoogLeNet	160	213	20	3

In Table 15, the performance of the pre-trained networks on the classification of images shown in Figure 16 and 20. As it is seen in Table 15, the highest classification performance is obtained with SqueezeNet and DenseNet201.

Table 15
z-coordinate time series performance metrics of the Lorenz and Chen chaotic systems.

Network	Accuracy	Precision	Sensitivity	Specificity
SqueezeNet	**0.957071**	0.944444	0.960452	0.954338
VGG-19	0.911616	0.927778	**0.883598**	0.937198
AlexNet	0.914141	0.833333	0.974026	0.876033
ResNet50	0.959596	0.972222	0.940860	0.976190
ResNet101	0.941919	**0.950000**	0.924324	**0.957346**
DenseNet201	0.916667	**0.961111**	0.869347	**0.964467**
ShuffleNet	0.876263	0.850000	0.874286	0.877828
GoogLeNet	0.941919	0.888889	0.981595	0.914163

5. Conclusions

In this study, for the first time in the literature, classification of time series of three different chaotic system is performed with high accuracy. For the classification, the time series of the most common three chaotic systems in the literature are used. These systems are Lorenz, Rossler and Chen chaotic systems. The time series are obtained by numerically solving these systems with RK4 algorithm. To increase data variety in the dataset, the chaotic systems are solved for different step size, initial values, system parameters' value and time length. The dataset consists of images of 6750 different time series. The high-performance classification is performed using SqueezeNet, VGG-19, AlexNet, ResNet50, ResNet101, DenseNet201, ShuffleNet, and GoogLeNet methods. High accuracy classification is performed especially for the systems with similar dynamical properties and time series. In addition to observing relatively high classification performance with all the 8 networks, very high accuracy rate between 95% and 98% is achieved with SqueezeNet, ResNet101, and DenseNet201 networks. The classification of the systems with different dynamical properties and time series like Lorenz and Rossler chaotic systems has very high performance as expected. For the classification of these two systems, in addition to observing relatively high classification performance with all the 8 networks, very high accuracy rate between 97% and 99% is achieved with AlexNet, DenseNet201, and GoogLeNet networks. The performance results of the classifications show that the classification that the time series of the chaotic systems can be performed with very high accuracy using deep learning method. Hence, this study shows that the classification of real life chaotic or random varying signals/data or associating them with a mathematical model over their images is possible.

Declarations

The Potential Conflict of Interest Disclosure

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

Data Availability Statements

Data Availability: No Data Availability.

References

1. Kaya, U., Yılmaz, A., Dikmen, Y., "Deep Learning Methods used in the field of Health," *Eur. J. Sci. Technol.*, no. 16, pp. 792–808, Aug. 2019
2. Barros, P., Parisi, G.I., Weber, C., Wermter, S., "Emotion-modulated attention improves expression recognition: A deep learning model," *Neurocomputing*, vol. 253, pp. 104–114, Aug. 2017
3. Rakhlin, A., Shvets, A., Iglovikov, V., Kalinin, A.A., "Deep Convolutional Neural Networks for Breast Cancer Histology Image Analysis," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, vol. 10882 LNCS, pp. 737–744
4. Zhang, Y., et al., "Towards End-to-End Speech Recognition with Deep Convolutional Neural Networks," in *Interspeech 2016*, 2016, vol. 08-12-Sept, pp. 410–414
5. Qian, Y., Bi, M., Tan, T., Yu, K.: Very Deep Convolutional Neural Networks for Noise Robust Speech Recognition. IEEE/ACM Trans. Audio Speech Lang. Process. **24**(12), 2263–2276 (Dec. 2016)
6. Young, T., Hazarika, D., Poria, S., Cambria, E.: "Recent trends in deep learning based natural language processing [Review Article]," *IEEE Computational Intelligence Magazine*, vol. 13, no. 3. Institute of Electrical and Electronics Engineers Inc., pp. 55–75, 01-Aug-2018
7. Lecun, Y., et al.: PERSPECTIVES Special Topic: Machine Learning Deep learning for natural language processing: advantages and challenges. Natl. Sci. Rev. **5**(1), 22–24 (2018)
8. Deng, L., Liu, Y.: Deep Learning in Natural Language Processing. Springer Singapore, Singapore (2018)
9. Toraman, S., "Pedestrian Detection with Deep Learning from Unmanned Aerial Imagery," *J. Aviat.*, vol. 2, no. 2, pp. 64–69, Dec. 2018
10. Altan, G., "DeepGraphNet: Deep Learning Models in the Classification of Graphs," *Eur. J. Sci. Technol.*, pp. 319–327, Oct. 2019
11. Arena, P., Calí, M., Patané, L., Portera, A., Spinosa, A.G.: "A CNN-based neuromorphic model for classification and decision control," *Nonlinear Dyn.*, vol. 95, no. 3, pp. 1999–2017, Feb. 2019
12. Dang, W., Gao, Z., Sun, X., Li, R., Cai, Q., Grebogi, C., "Multilayer brain network combined with deep convolutional neural network for detecting major depressive disorder," *Nonlinear Dyn.*, vol. 102, no. 2, pp. 667–677, Oct. 2020
13. Boullé, N., Dallas, V., Nakatsukasa, Y., Samaddar, D., "Classification of chaotic time series with deep learning," *Phys. D Nonlinear Phenom.*, vol. 403, p. 132261, Feb. 2020
14. Yeo, K., "Model-free prediction of noisy chaotic time series by deep learning," *arXiv*, Sep. 2017

15. Kuremoto, T., Obayashi, M., Kobayashi, K., Hirata, T., Mabu, S., “Forecast chaotic time series data by DBNs,” in *Proceedings - 2014 7th International Congress on Image and Signal Processing, CISP 2014*, 2014, pp. 1130–1135
16. Sangiorgio, M., Dercole, F., “Robustness of LSTM neural networks for multi-step forecasting of chaotic time series,” *Chaos, Solitons and Fractals*, vol. 139, p. 110045, Oct. 2020
17. Lorenz, E.N., “Deterministic Nonperiodic Flow,” *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, Mar. 1963
18. Pchelintsev, A.N., “Numerical and physical modeling of the dynamics of the Lorenz system,” *Numer. Anal. Appl.*, vol. 7, no. 2, pp. 159–167, Apr. 2014
19. Gorman, M., Widmann, P.J., Robbins, K.A.: Nonlinear dynamics of a convection loop: A quantitative comparison of experiment with theory. Phys. D Nonlinear Phenom. **19**(2), 255–267 (1986)
20. Haken, H.: Analogy between higher instabilities in fluids and lasers. Phys. Lett. A **53**(1), 77–78 (1975)
21. Cuomo, K.M., Oppenheim, A.V., “Circuit implementation of synchronized chaos with applications to communications,” *Phys. Rev. Lett.*, vol. 71, no. 1, pp. 65–68, Jul. 1993
22. Hemati, N.: Strange Attractors in Brushless DC Motors. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **41**(1), 40–45 (1994)
23. Knobloch, E., “Chaos in the segmented disc dynamo,” *Phys. Lett. A*, vol. 82, no. 9, pp. 439–440, Apr. 1981
24. Chen, G., Ueta, T.: Yet another chaotic attractor. Int. J. Bifurc. Chaos **9**(7), 1465–1466 (1999)
25. Liang, X., Qi, G.: Mechanical analysis of Chen chaotic system. Chaos, Solitons and Fractals **98**, 173–177 (May 2017)
26. Rössler, O.E.: An equation for continuous chaos. Phys. Lett. A **57**(5), 397–398 (1976)
27. Letellier, C., Dutertre, P., Maheu, B., “Unstable periodic orbits and templates of the Rössler system: Toward a systematic topological characterization,” *Chaos*, vol. 5, no. 1, pp. 271–282, Mar. 1995
28. Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., Lew, M.S.: Deep learning for visual understanding: A review. Neurocomputing **187**, 27–48 (Apr. 2016)
29. Haskins, G., Kruger, U., Yan, P., “Deep Learning in Medical Image Registration: A Survey,” *Mach. Vis. Appl.*, vol. 31, no. 1, Mar. 2019
30. Küçük, D., et al.: “A Literature Study on Deep Learning Applications in Natural Language Processing. ” in International Journal of Management Information Systems and Computer Science **2**(2), 76–86 (2018)
31. Keleş, A., “Deep Learning and Applications in Health,” *J. Turkish Stud.*, vol. 13, no. Volume 13 Issue 21, pp. 113–127, Jan. 2018
32. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998
33. Le Cun, Y., et al.: “Handwritten Digit Recognition: Applications of Neural Net Chips and Automatic Learning. In: ” in Neurocomputing, pp. 303–318. Berlin Heidelberg, Springer (1990)

34. Ravi, D., et al., “Deep Learning for Health Informatics,” *IEEE J. Biomed. Heal. Informatics*, vol. 21, no. 1, pp. 4–21, Jan. 2017
35. Szegedy, C., et al., “Going deeper with convolutions,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015, vol. 07-12-June, pp. 1–9
36. Russakovsky, O., et al., “ImageNet Large Scale Visual Recognition Challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015
37. Ballester, P., Araujo, R.M., “On the performance of googlenet and alexnet applied to sketches,” in *30th AAAI Conference on Artificial Intelligence, AAAI*: 2016, 2016, pp. 1124–1128
38. Balagourouchetty, L., Pragatheeswaran, J.K., Pottakkat, B., Ramkumar, G., “GoogLeNet-Based Ensemble FCNet Classifier for Focal Liver Lesion Diagnosis,” *IEEE J. Biomed. Heal. Informatics*, vol. 24, no. 6, pp. 1686–1694, Jun. 2020
39. Zhu, Z., Li, J., Zhuo, L., Zhang, J., “Extreme Weather Recognition Using a Novel Fine-Tuning Strategy and Optimized GoogLeNet,” in *DICTA 2017 - 2017 International Conference on Digital Image Computing: Techniques and Applications*, 2017, vol. 2017-December, pp. 1–7
40. Toğaçar, M., Ergen, B., Özyurt, F., “Classification of Flower Species by Using Feature Selection Methods in Convolutional Neural Network Models,” *Fırat Üniversitesi Mühendislik Bilim. Derg.*, pp. 37–45, Mar. 2020
41. Karadağ, B., Arı, A., Karadağ, M., “Derin Öğrenme Modellerinin Sinirsel Stil Aktarımı Performanslarının Karşılaştırılması,” *J. Polytech.*, vol. 0900, pp. 2–14, Mar. 2021
42. Wen, L., Li, X., Li, X., Gao, L., “A new transfer learning based on VGG-19 network for fault diagnosis,” in *Proceedings of the*: 2019 *IEEE 23rd International Conference on Computer Supported Cooperative Work in Design, CSCWD 2019*, 2019, pp. 205–209
43. Dhungel, N., Carneiro, G., Bradley, A.P., “Under review as a conference paper at ICLR 2017 SqueezeNet: AlexNet-level accuracy with 50X fewer parameters and <0.5MB model size,” in *Advances in Computer Vision and Pattern Recognition*, no. 9783319429984, 2017, pp. 225–240
44. Zhi, W., Gao, L., Zhu, Z., “Garbage classification and recognition based on SqueezeNet,” in *Proceedings - 2020 3rd World Conference on Mechanical Engineering and Intelligent Manufacturing, WCMEIM 2020*, 2020, pp. 122–125
45. Gaikwad, A.S., El-Sharkawy, M., “Pruning convolution neural network (squeezenet) using taylor expansion-based criterion,” in *2018 IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2018*, 2018, vol. 2019-Janua, pp. 1–5
46. Aksoy, B., Halis, H.D., Salman, O.K.M.: Identification of Diseases in Apple Plants with Artificial Intelligence Methods and Comparison of the Performance of Artificial Intelligence Methods. Int. J. Eng. Innov. Res. **2**(3), 194–210 (2020)
47. He, K., Zhang, X., Ren, S., Sun, J., “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, vol. 2016-Decem, pp. 770–778

48. Zhang, Z., “ResNet-Based Model for Autonomous Vehicles Trajectory Prediction,” in: 2021 *IEEE International Conference on Consumer Electronics and Computer Engineering, ICCECE 2021*, 2021, pp. 565–568
49. Talo, M., “Classification of Histopathological Breast Cancer Images using Convolutional Neural Networks,” *Fırat Üniversitesi Mühendislik Bilim. Derg.*, vol. 31, no. 2, pp. 391–398, Sep. 2019
50. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., “Densely Connected Convolutional Networks,” in: 2017 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, vol. 2017-Janua, pp. 2261–2269
51. Yilmaz, F., Kose, O., Demir, A., “Comparison of two different deep learning architectures on breast cancer,” in *TIPTEKNO 2019 - Tip Teknolojileri Kongresi*, 2019, vol. 2019-January
52. Zhang, K., Guo, Y., Wang, X., Yuan, J., Ding, Q.: Multiple feature reweight DenseNet for image classification. IEEE Access **7**, 9872–9880 (2019)
53. Zhang, X., Zhou, X., Lin, M., Sun, J., “ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6848–6856
54. Wang, L., Hung, K.-W., “Contention resolution in the loop-augmented ShuffleNet multihop lightwave network,” in *1994 IEEE GLOBECOM. Communications: The Global Bridge*, 2002, pp. 186–190
55. Li, Y., Lv, C., “SS-YOLO: An Object Detection Algorithm based on YOLOv3 and ShuffleNet,” in *Proceedings of 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference, ITNEC 2020*, 2020, pp. 769–772
56. Lu, S., Lu, Z., Zhang, Y.-D., “Pathological brain detection based on AlexNet and transfer learning,” *J. Comput. Sci.*, vol. 30, pp. 41–47, Jan. 2019
57. Krizhevsky, A., Sutskever, I., Hinton, G.E., “ImageNet Classification with Deep Convolutional Neural Networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, Jun. 2017
58. Gonzalez, T.F.: Handbook of Approximation Algorithms and Metaheuristics. Chapman and Hall/CRC (2007)
59. Agarwal, A., Patni, K., Rajeswari, D., “Lung Cancer Detection and Classification Based on Alexnet CNN,” in *Proceedings of the 6th International Conference on Communication and Electronics Systems, ICCES 2021*, 2021, pp. 1390–1397
60. Han, X., Zhong, Y., Cao, L., Zhang, L.: Pre-Trained AlexNet Architecture with Pyramid Pooling and Supervision for High Spatial Resolution Remote Sensing Image Scene Classification. Remote Sens. **9**(8), 848 (Aug. 2017)

Figures

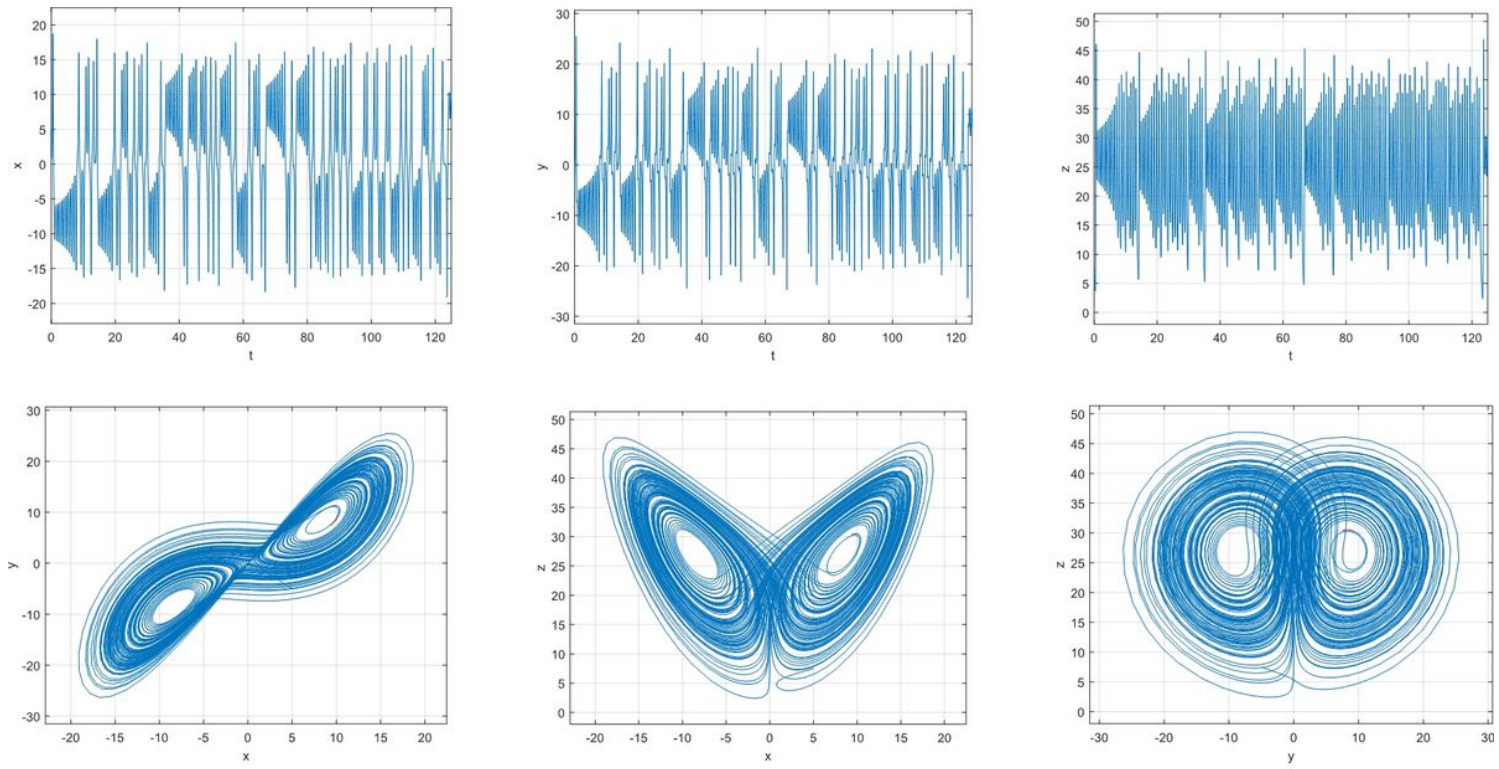

Figure 1

Time series and phase portraits of the Lorenz system for system parameters α = 10, β= 8/3, γ= 28 and {x0, y0, z0} = {5, -5, 7.5}.

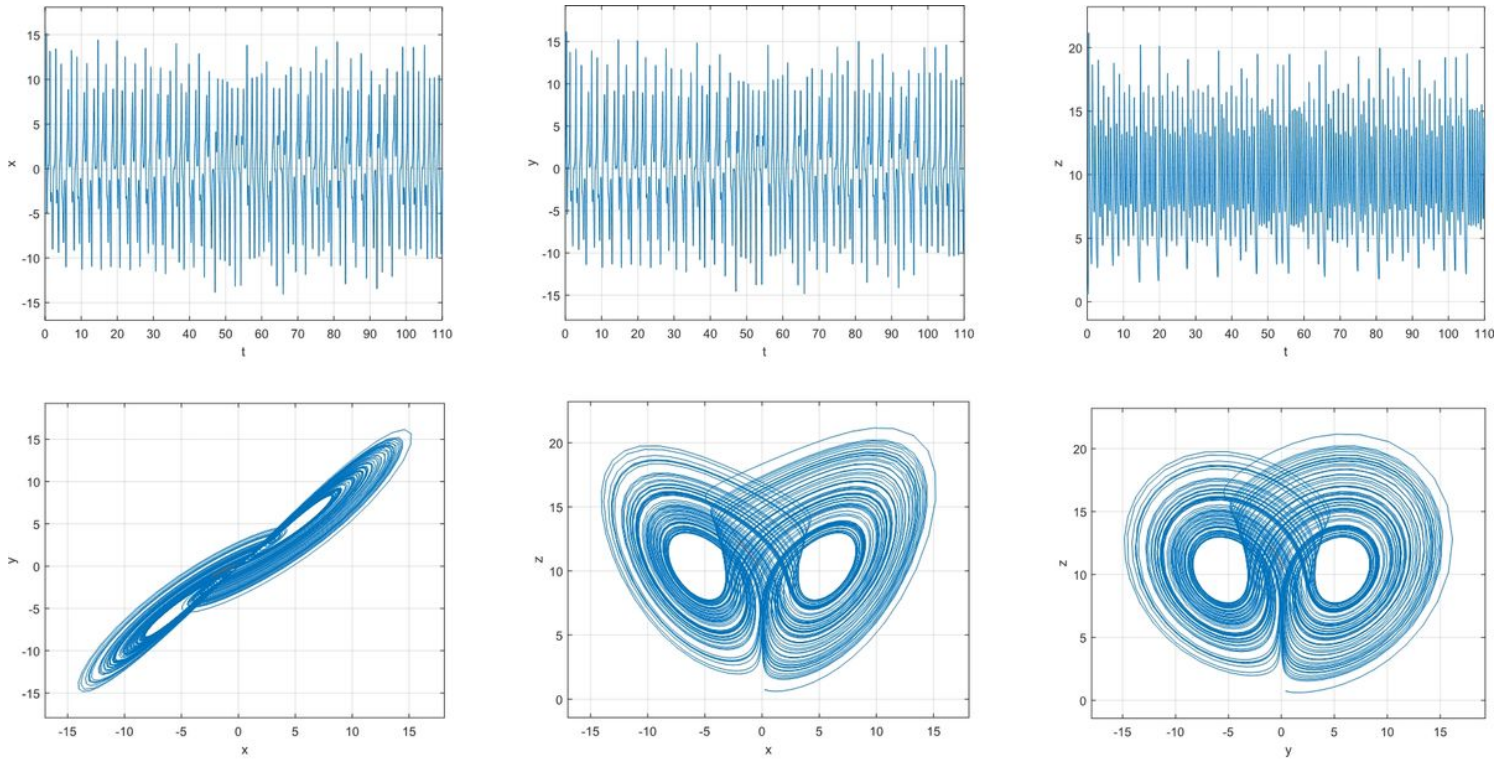

Figure 2

Time series and phase portraits of the Chen system for system parameters α = 40, β = 3, γ = 25 and {x0, y0, z0} = {0.25, 0.4, 0.75}.

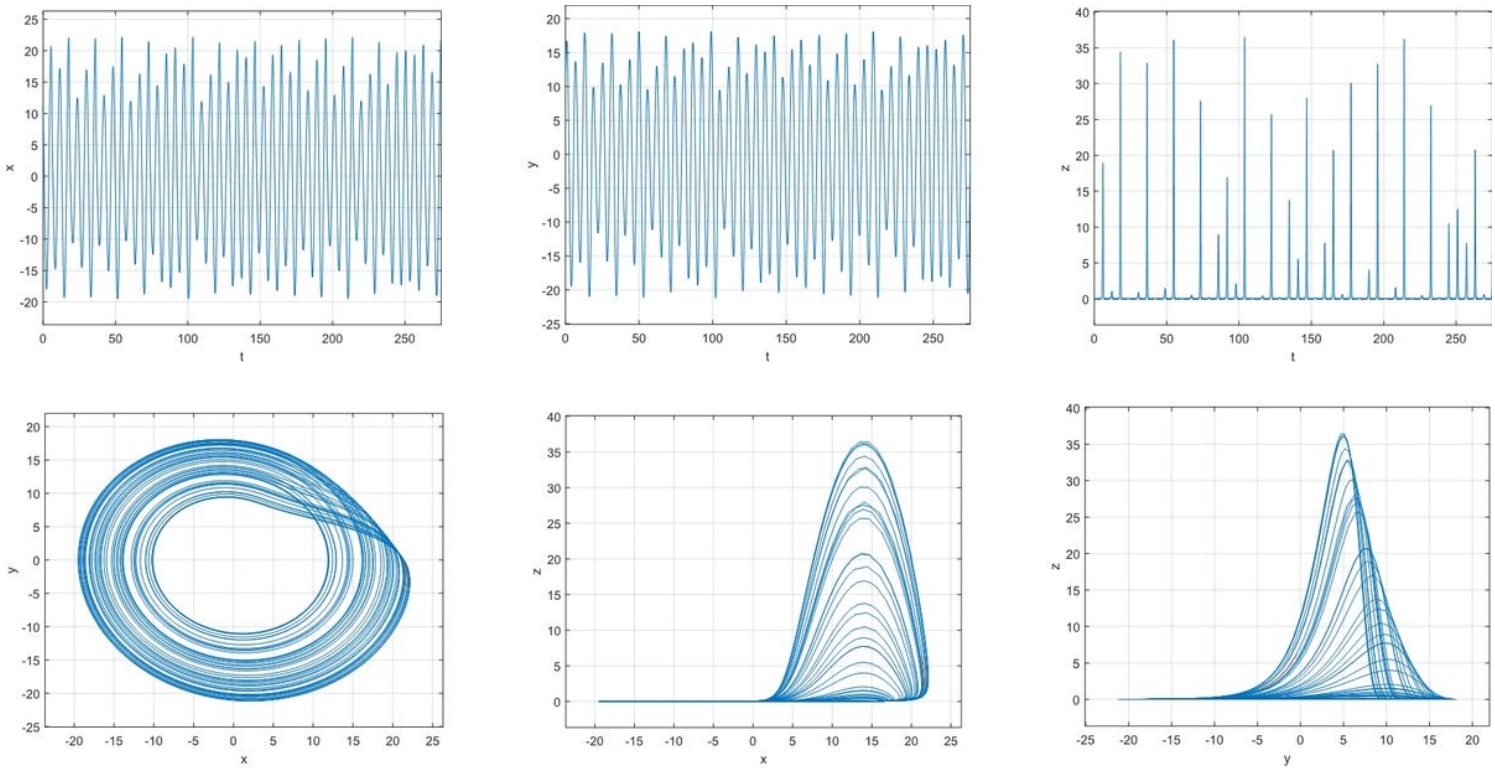

Figure 3

Time series and phase portraits of the Rossler system for system parameters α = 0.1, β= 0.1, γ= 14 and {x0, y0, z0} = {11, 11, 0}.

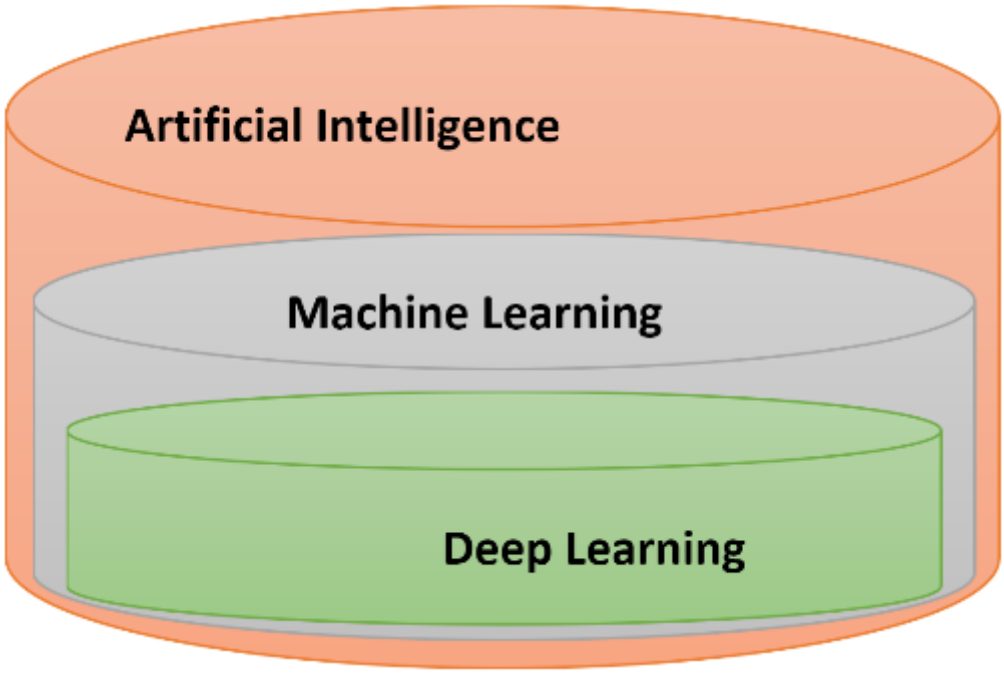

Figure 4

The relationship between deep learning, machine learning and artificial intelligence.

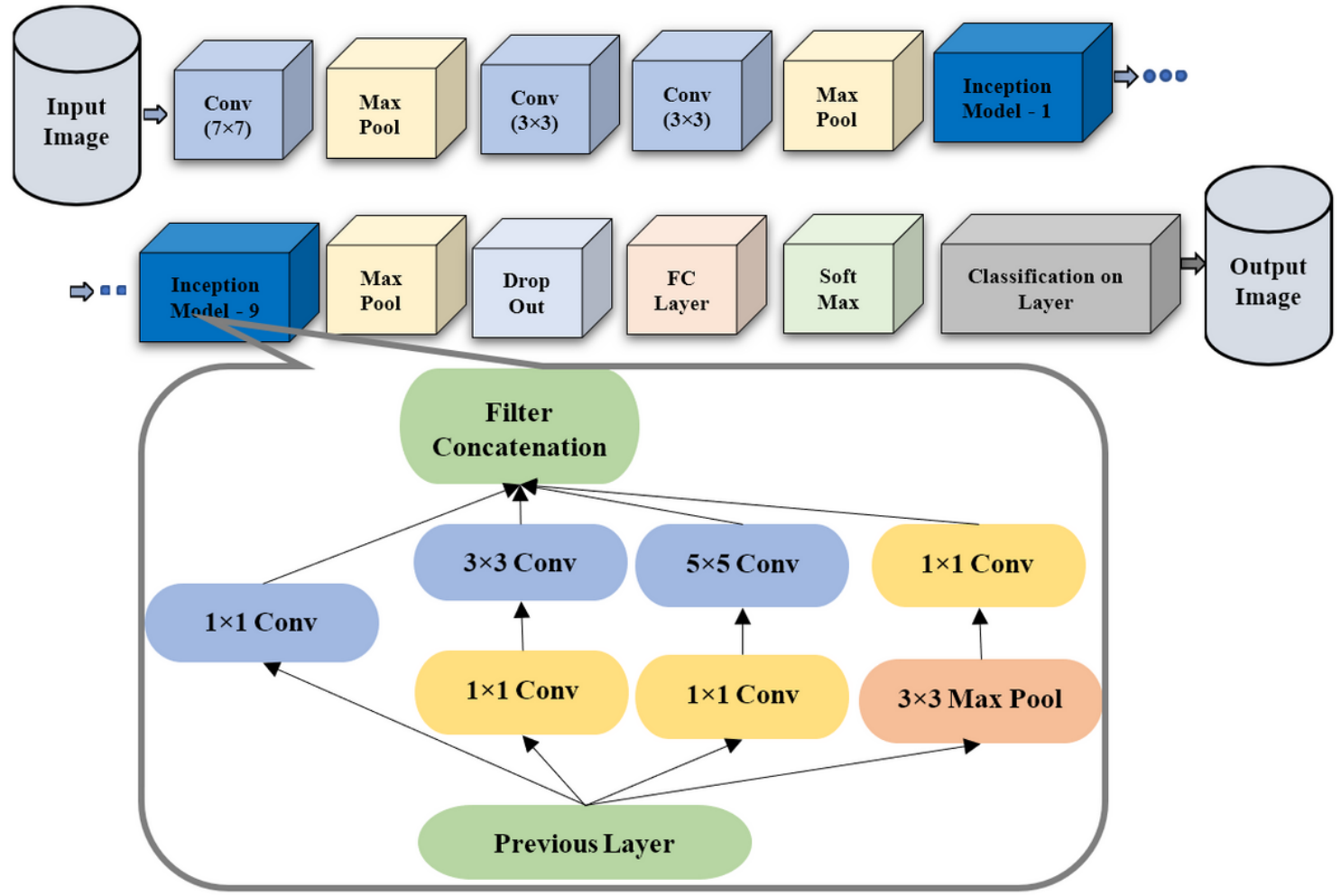

Figure 5

The architecture of GoogLeNet [38].

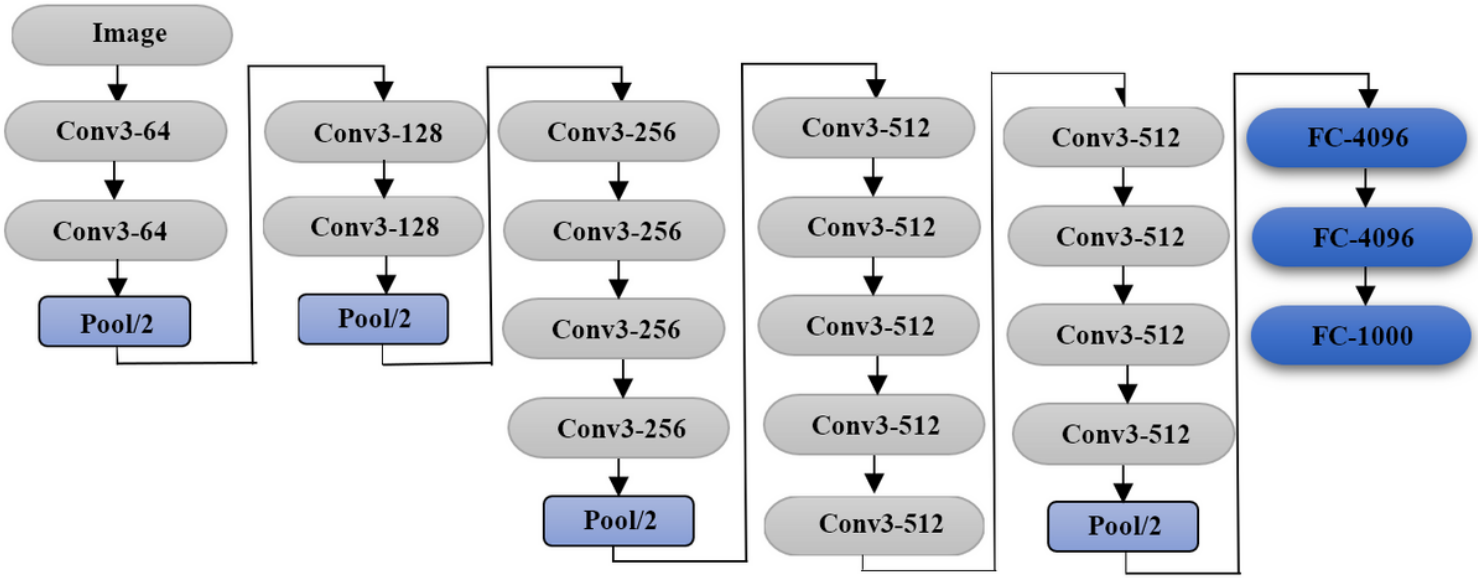

Figure 6

The structure of VGG-19 network [42].

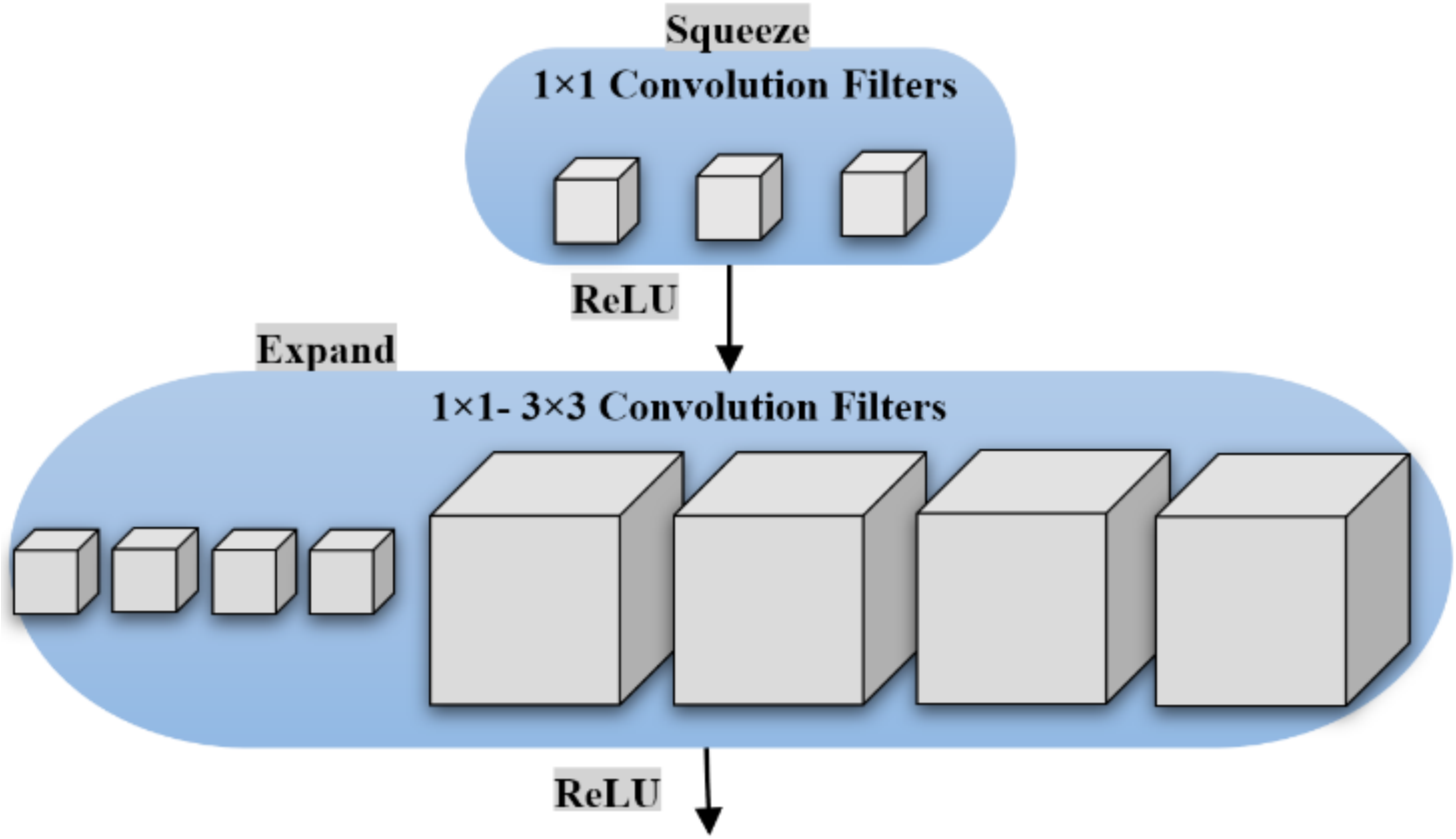

Figure 7

SqueezeNet network fire module [43].

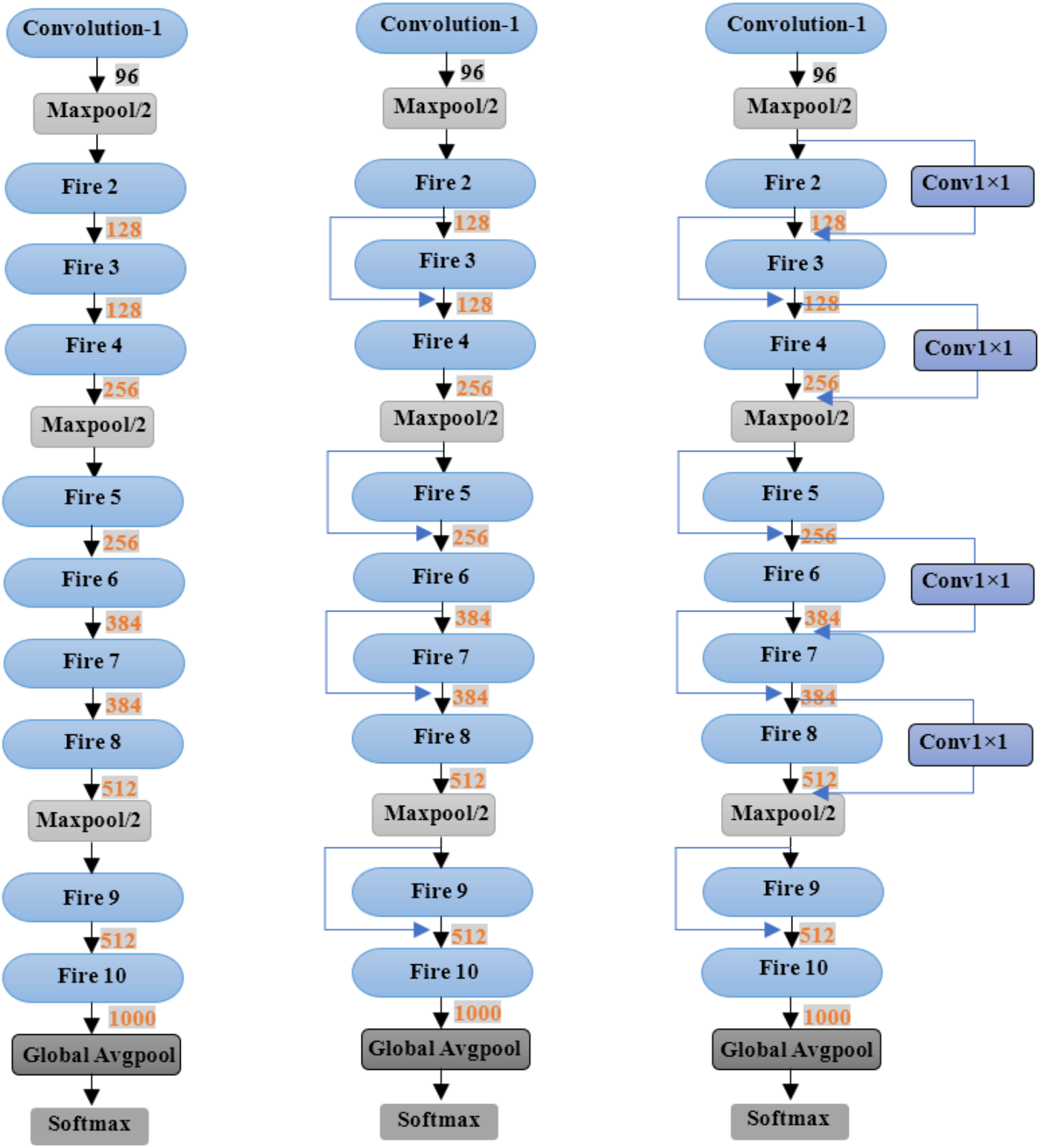

Figure 8

The structure of squeezeNet network [43].

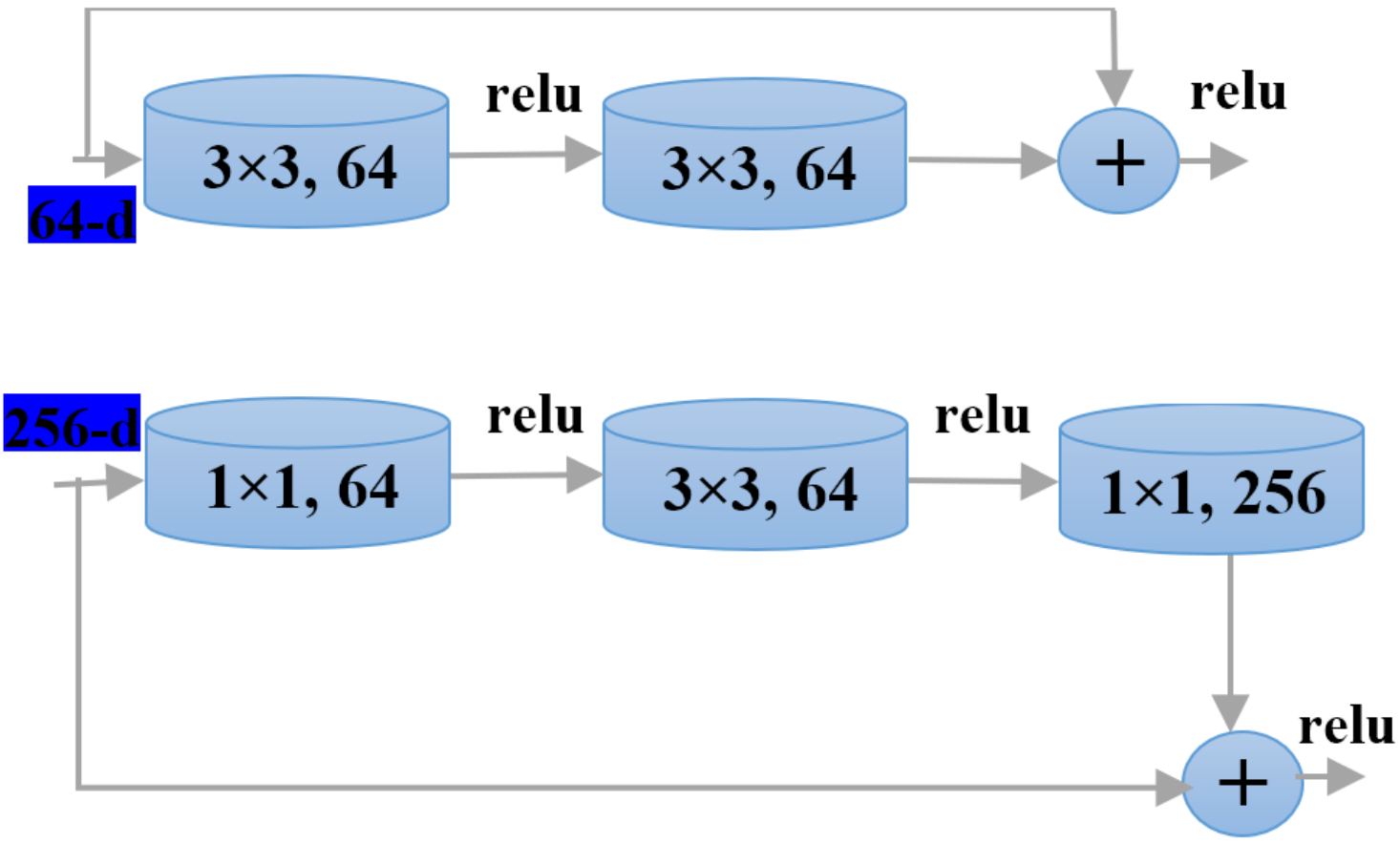

Figure 9

Training sample of the ResNet architecture with residual layers [47].

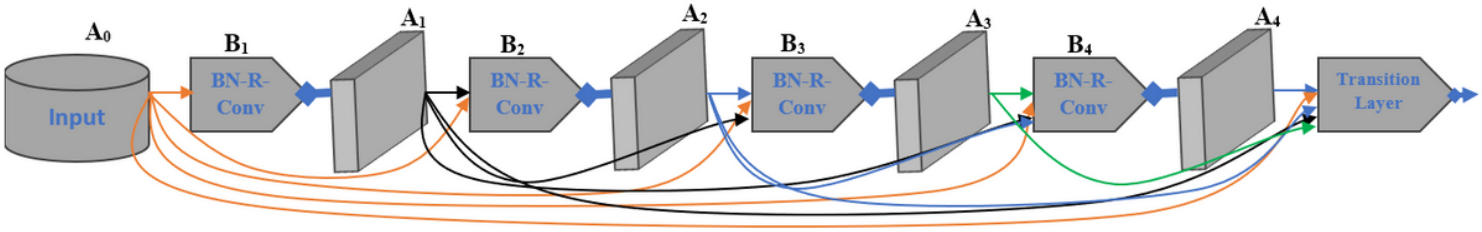

Figure 10

The structure of DenseNet network [50].

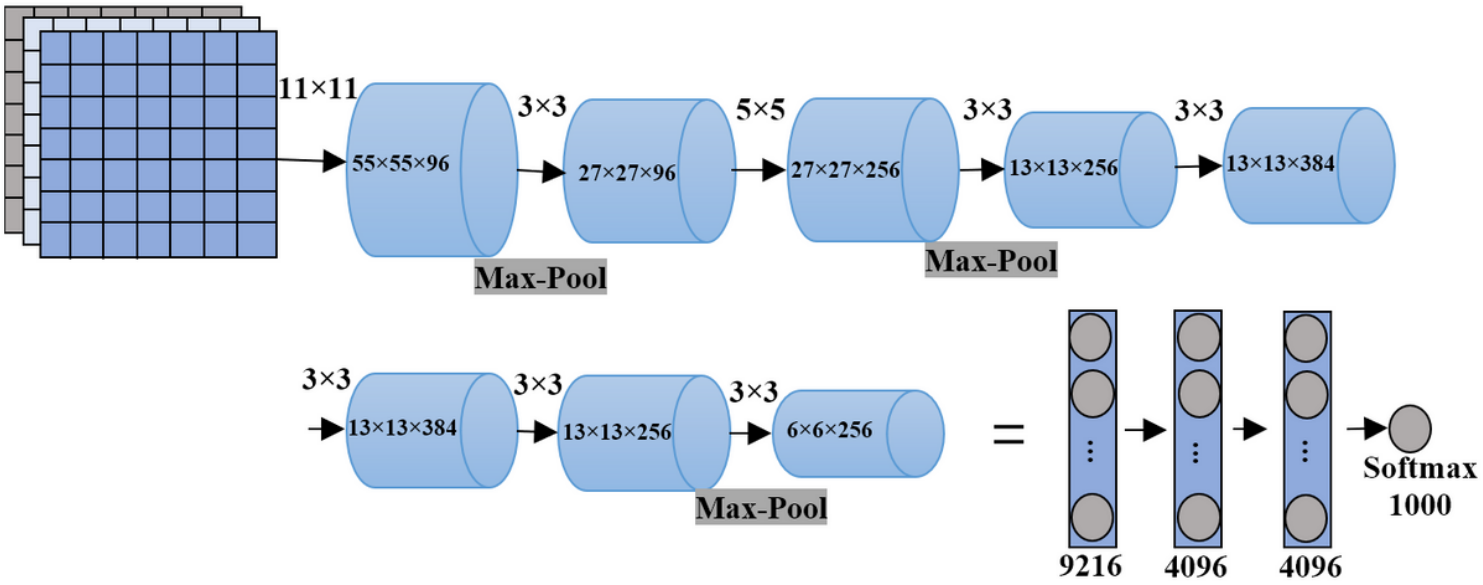

Figure 11

The architecture of AlexNet network [60].

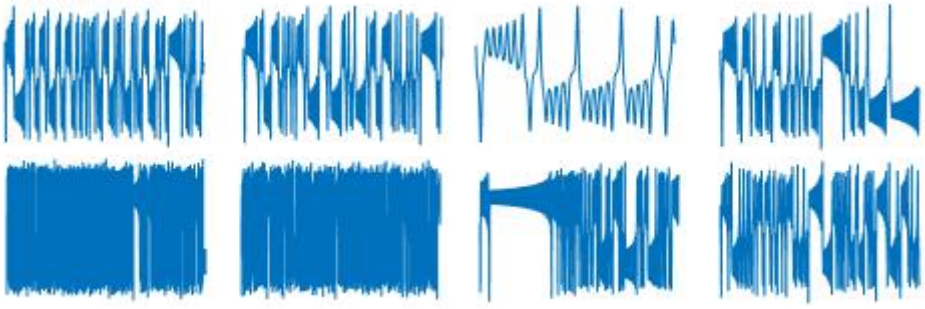

Figure 12

Sample images obtained from x coordinates of time series via Lorenz Chaotic System.

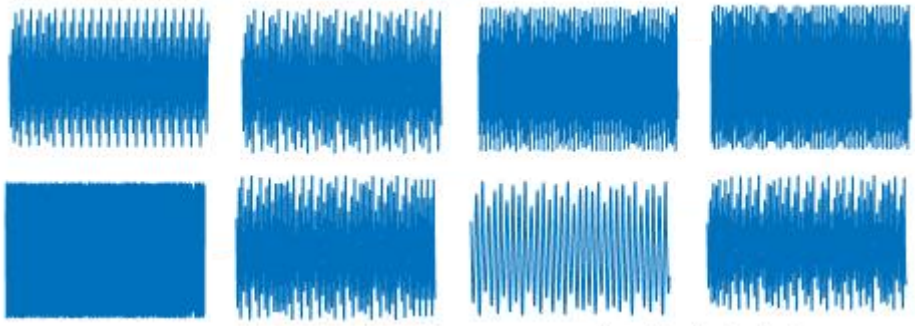

Figure 13

Sample images obtained from the x coordinates of the time series over the Rossler chaotic system.

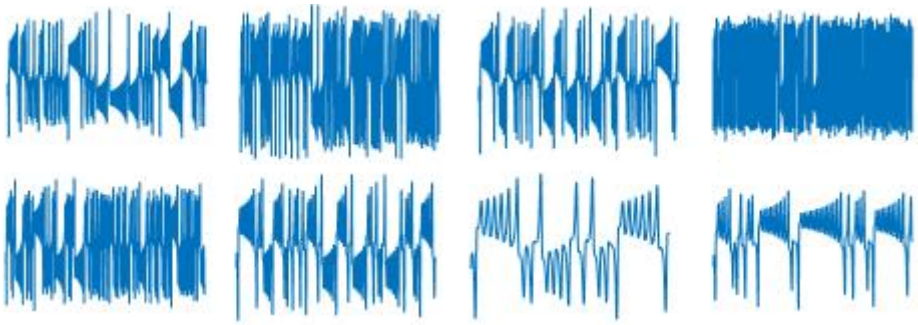

Figure 14

Sample images obtained from y coordinates of time series over Lorenz chaotic system

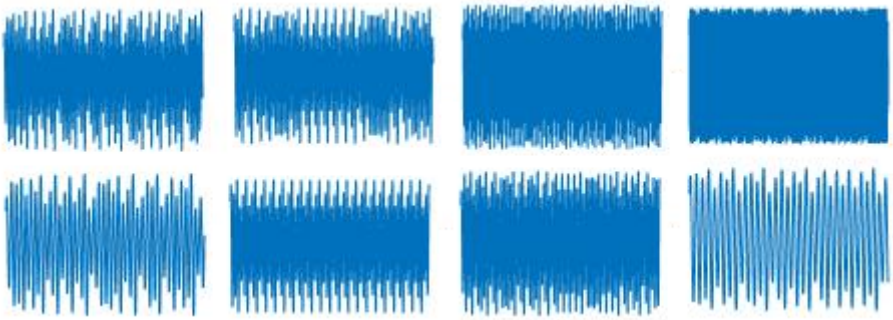

Figure 15

Sample images obtained from the y coordinates of the time series over the Rossler chaotic system.

Figure 16

Sample images obtained from z coordinates of time series over Lorenz chaotic system.

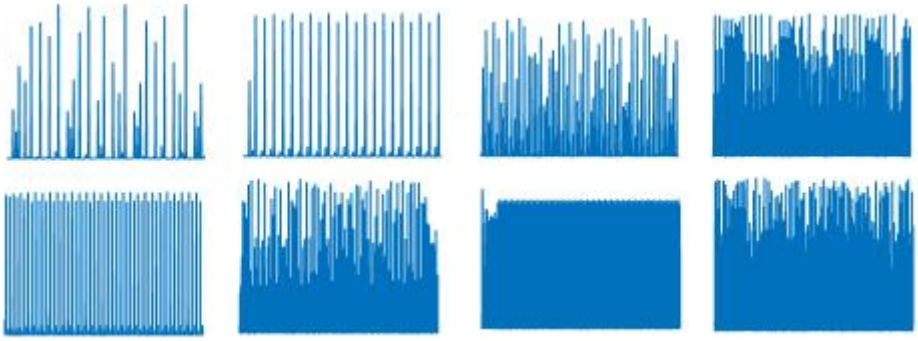

Figure 17

Sample images obtained from the z coordinates of the time series over the Rossler chaotic system.

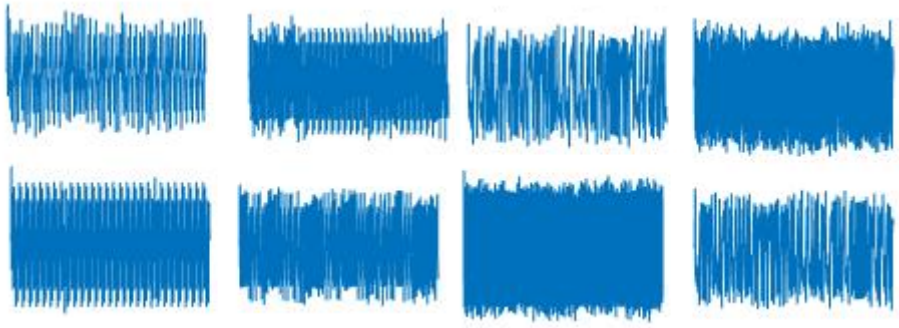

Figure 18

Sample images obtained from x coordinates of time series via Chen Chaotic System.

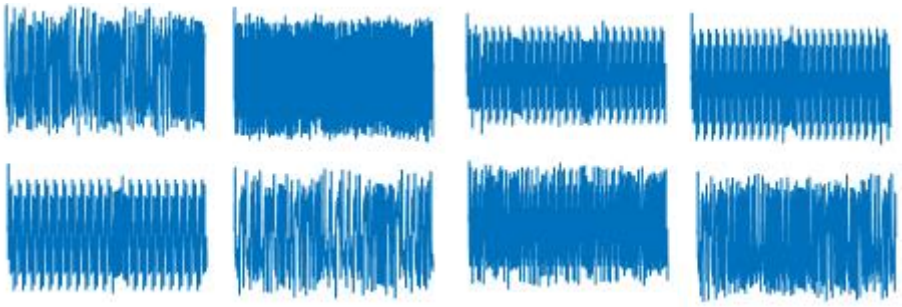

Figure 19

Sample images obtained from y coordinates of time series over Chen chaotic system.

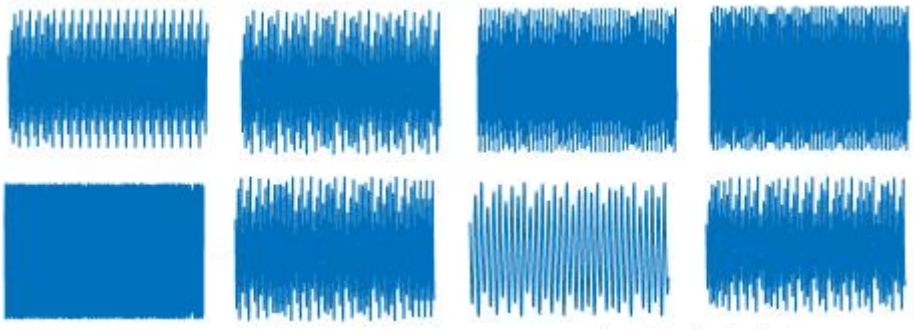

Figure 20

Sample images obtained from z coordinates of time series over Chen chaotic system