Afro-Asian Journal of Scientific Research (AAJSR)
المجلة الأفروآسيوية للبحث العلمي (AAJSR)
E-ISSN: 2959-6505
Volume 2, Issue 3, July - September 2024, Page No: 132-148
Website: https://aajsr.com/index.php/aajsr/index
SJIFactor 2023: 3.908 0.51 :2023 (AIF) معامل التأثير العربي ISI 2023: 0.337

The Improvement of Back Propagation of Neural Networks

Nouri Ali [1] *, Iman Namroud [2], Mohsen Ibrahim [3]
[1] Higher Institute of Medical Sciences and Technologies, Bani Walid, Libya
[2] Collage of Electronics Engineering, Bani Walid, Libya
[3] Higher Institute of Engineering Technology, Bani Walid, Libya

تحسين الانتشار الخلفي للشبكات العصبية

نوري علي*[1]، إيمان نمرود[2]، محسن إبراهيم[3]
[1] المعهد العالي للتقنيات الطبية، بني وليد، ليبيا
[2] كلية التقنية الإلكترونية، بني وليد، ليبيا
[3] المعهد العالي للتقنيات الهندسية، بني وليد، ليبيا

*Corresponding author: nouri.alhashi@gmail.com

Received: May 29, 2024	Accepted: July 18, 2024	Published: July 30, 2024

Abstract:
This paper is concerned with the study of BP network and its evolvement as it is noticed that problems that exist in the Error back propagation algorithm are slow convergence and unlikely to approach minimum. The problems mentioned above led us to find improvements to accelerate this algorithm. In this research BP networks have been constructed, which includes accelerating procedures by using one of the gradient methods to get rid of zigzag phenomena and to approach global minimum. An adaptive learning rate of two methods has been suggested to reduce errors and accelerate training. it has been noticed that calculating appropriate initial weights (Suggested initial, Nguyen–Widrow) and using drive logarithmic function and an adaptive learning rate method have been suggested to reduce errors and accelerate training leading to adequate results and fast learning. A standard network has been applied to all suggestions to diagnose OSTEOARTHRITIN then, results of improving methods which we use have been displayed.

Keywords: Neural network, Image processing, Pattern recognition, back propagation algorithm.

الملخص
يهتم هذا البحث بدراسة شبكة BP وتطورها حيث يلاحظ أن المشاكل الموجودة في خوارزمية الخطأ الخلفي هي تقارب بطيء ومن غير المرجح أن يقترب من الحد الأدنى. وقد قادتنا المشاكل المذكورة أعلاه إلى إيجاد تحسينات لتسريع هذه الخوارزمية. في هذا البحث تم بناء شبكات BP والتي تتضمن إجراءات تسريع باستخدام إحدى طرق التدرج للتخلص من الظواهر المتعرجة والاقتراب من الحد الأدنى العالمي. وقد تم اقتراح معدل التعلم التكيفي لطريقتين لتقليل الأخطاء وتسريع التدريب. وقد لوحظ أن حساب الأوزان الأولية المناسبة (المبدئية المقترحة، Nguyen – Widrow) واستخدام الدالة اللوغاريتمية للقيادة وطريقة معدل التعلم التكيفي قد تم اقتراحها لتقليل الأخطاء وتسريع التدريب يؤدي إلى نتائج كافية وتعلم سريع. تم تطبيق شبكات قياسية على جميع الاقتراحات لتشخيص هشاشة العظام، ثم تم عرض نتائج تحسين الطرق التي نستخدمها.

الكلمات المفتاحية: الشبكات العصبية، معالجة صور، تمييز الانماط، خوارزمية الانتشار الخلفي.

1. Introduction

Neural networks are basically defined as mathematical models for processing information [2], or they are a mathematical method that simulates a biological neural network to obtain an artificial neural network, or they are a mathematical method because they use some mathematical methods to obtain weights (such as derivatives), and form models Neural networks are several connections of computational elements (nodes) that operate in parallel and are associated with typical weights (weights) equivalent to them during their use to demonstrate work or performance. These models are known by several names such as interconnection models, parallel distribution, and neural systems [34]. The function of artificial nerve systems is the parallel division of network computations, and most of the basic characteristics are of the neural network architecture. Some networks prepare immediate answers. Other networks require time to answer according to the characteristics and behavior they possess and are referred to as dynamic dynamics. There are various laws when weight changes, as neural networks show different speeds and efficiency in learning, and as a result, they indicate their ability to answer accurately (feeling the present input), that is, knowing what the output will be for this input. Neural networks consist of computational elements (nodes) connected by connecting lines attached to certain weights (a numerical value). These weights represent the information with which the network will begin solving the problem [35]. There are many types of neural networks, but they all contain three things: the cell (node), the processing unit, which I described as having connections between them (i.e., one cell is connected to another through weights), and it also contains the law of learning. rule), these three bodies together form the neural network model [35]. The memory of a digital calculator is estimated in bytes, and the memory of a neural network is estimated by correlations, as the speed of digital computers is clearly expressed in an instruction per second, and the speed of neural networks is measured by a correlation in every fraction of a second. [34] Learning is a phase in artificial neural networks when new data is produced in the network as a result of changing the weight of the network. The network is trained on the given application, that is, on the set of inputs to produce the desired output. Training is performed sequentially on the input vector when the weights of the network change according to specific laws. During training, the network weights gradually approach the ideal values and we obtain the desired output. Training in artificial neural networks is either supervised training or unsupervised training. Training requires the presence of a pair of input vector (Input) and expected output vector (Target Output), which together represent the training pair, where the output (Output) of the applied input vector (Target Output) is compared with the expected output vector (Target Output), and the difference between them represents the error (Error) returned during network in order to change the weights according to the algorithm in order to reduce the error. The vectors of the training set are applied sequentially, and the error is calculated and the weights are changed for each vector until the error of the input training set reaches a low level or level.

The second phase of artificial neural network information processing is the recall phase, where the given input is applied with the weights resulting from the first phase, that is, from the training phase of the neural network, and in one step we obtain the desired output. The call phase is feed forward only Error back propagation network neural is widely used due to its ease and ability to store information in the links representing weights that connect one cell to another. The law for learning the error back propagation network (EBP) is known as the error back propagation algorithm [36]. The goal of the training or (education) algorithm is to use the error in changing the weights to gradually reduce the error, and the algorithm updates the weights in batches during the iteration stages [36]. The network contains the activity change state (either inhibition or stimulation) of all cells in all layers and the weights that connect the cells of one layer to another based on certain equations. The process of training this network aims to reach a state of balance between the network's ability to give a good response to the input samples that are used in the training process and the network's ability to give a good response to input similar to, but not identical to, the input used in training [34]. Training the network stops when the desired output is obtained and the global minimum is reached.

1.1 The disease 'OSTEO ARTHRITIS'

Usually, osteoarthritis comes on slowly. Early in the disease, joints may ache after physical work or exercise. Osteoarthritis can occur in any joint. Most often it occurs at the hands, knees, hips, or spine.

Hands: One form of osteoarthritis that appears to have certain genetic traits—that is, to run in families is osteoarthritis of the fingers. It affects more women than males and is more common in women following menopause. When a person has osteoarthritis, their end joints develop tiny, bony bumps. fingers. Heberden's (HEB-err-denz) nodes is the term for them. Similar knobs on the middle joints of the fingers are known as Bouchard's (boo- SHARDZ) nodes. Fingers may swell and becoming gnarled; they may also hurt or feel numb and stiff. Osteoarthritis also frequently affects the base of the thumb joint

Knees: The body's main joints for bearing weight are the knees. They are therefore among the joints that osteoarthritis most frequently affects. It may be difficult to walk, climb, and get in and out of chairs and bathtubs due to their stiffness, swelling, and soreness. Osteoarthritis in the knees can cause impairment if left untreated. Pain and incapacity can be lessened with medication, exercise, weight loss, and walking assistance.

Hips: Hip osteoarthritis can be extremely painful, inflexible, and incapacitating. Individuals may experience pain in their knees, buttocks, inner thighs, groin, or hips. Walking aids like walkers or canes can ease the strain on the hip joint. Hip osteoarthritis can make it difficult to bend and move. This can make doing things like getting dressed and taking care of your feet difficult. Exercise, medicine, and walking aids can all help reduce pain and enhance mobility. If alternative measures fail to ease the extreme discomfort, the doctor might suggest a hip replacement.
Spine: Osteoarthritis of the spine can cause discomfort and stiffness in the lower back or neck. There may also be a consequence of arm or leg weakness or numbness. Some people find that sitting with back support pillows or sleeping on a firm mattress helps them feel better. Some find that using heat treatments or engaging in an exercise regimen that targets the abdomen and back muscles helps. [23].

1.2 The Central Nervous System (CNS)

Following McCulloch and Pitts' invention of simpler neurons, artificial neural networks were developed [29]. These neurons were shown as conceptual building blocks for circuits that might carry out computing tasks as well as models of biological neurons. The functioning of a biological neuron serves as the foundation for the fundamental model of the neuron. "Neurons are the basic signaling units of the nervous system" yet "each neuron is a discrete cell whose several processes arise from its cell body" . The structure of a neuron is divided into four primary parts. Two branches of the cell body, or soma, lead to presynaptic terminals: the axon and dendrites. The nucleus and protein synthesis are maintained by the cell body, which is the beating heart of the organism. Dendrites are the branches of a neuron that branch out in a structure resembling a tree and receive signals from neighboring neurons. Typically, a neuron contains a single axon that emerges from the axon hillock, a region of the cell body. Electric signals produced at the axon hillock are carried along the axon's whole length. We refer to these electrical signals as action potentials. A presynaptic terminal [21] may be reached by several branches that fork off from the axon's opposite end.

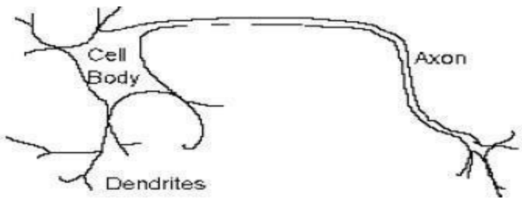

Figure 1: Nervous System**.**

1.3 The mathematical representation of neuron

There are three fundamental elements that are crucial for building a functioning model of a biological neuron. Initially, the neuron's synapses are represented as weights. The weight value indicates the strength of the link between an input and a neuron. Positive weight values indicate excitatory connections, whilst negative weight values show inhibitory connections. The actual activity occurring within the neuron cell is modeled by the next two components. An adder adds up all of the inputs that have been adjusted for each weight. [22] This process is known as "linear combination." Lastly, the amplitude of the neuron's output is managed by an activation function. Typically, an output must fall between 0 and 1, or between -1 and 1.

$$X = \sum_{i=1}^{n} x_i w_i \qquad Y = \begin{cases} +1, & \text{if } X \geq \theta \\ -1, & \text{if } X < \theta \end{cases}$$

The BP algorithm is one of the most popular methods in training MLPs. There are two relatively standard definitions of back propagation [24]. The first defines back propagation as a procedure for efficiently calculating the derivatives of some function of the outputs of any nonlinear differentiable system, with respect to all inputs and parameters of that system, through calculations proceeding backwards from outputs to inputs show figure 1. The second standard definition of back propagation is any technique for adapting the weights or parameters of a nonlinear system by using such derivatives or equivalent. Network size usually refers to the number of hidden layers and of neurons in each layer. The network size is a compromise between generalization and convergence. Convergence is the capacity of the network to learn the patterns on the training set and generalization is the capacity to respond correctly to new patterns. The best way is to implement the smallest network possible, so it is able to learn all patterns and, at the same time, provide good generalization [15]. Mathematically, this process is described in the figure 2. An object will be classified by neuron an object will be classified by neuron j into Class A if into Class A if

$$\sum w_{ij} x_i > \theta$$

where w_{ij} is the weight from neuron is the weight from neuron j to neuron to neuron i, x_i is the input from neuron the input from neuron i, and θ is the threshold on is the threshold on neuron j. If not, the object will be classified as Class B. If not, the object will be classified as Class B. • The weights on a perceptron model are adjusted by

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}$$

where $w_{ij}(t)$ is the weight from neuron) is the weight from neuron j to neuron to neuron i at time at time t (to the (to the t the iteration) and the iteration) and Δ wij is the weight adjustment. The weight change is computed by using the delta rule:

$$\Delta w_{ij} = \eta \delta_j x_i$$

where η is the learning rate (0<η<1) and δj is the error at neuron j

δj = Tj- Oj

where Tj is the target output value and Oj is the actual output of the network at neuron j.The process is repeated iteratively until convergence is achieved. Convergence is the process whereby the errors are minimized to an acceptable level. As in figure 2 below:

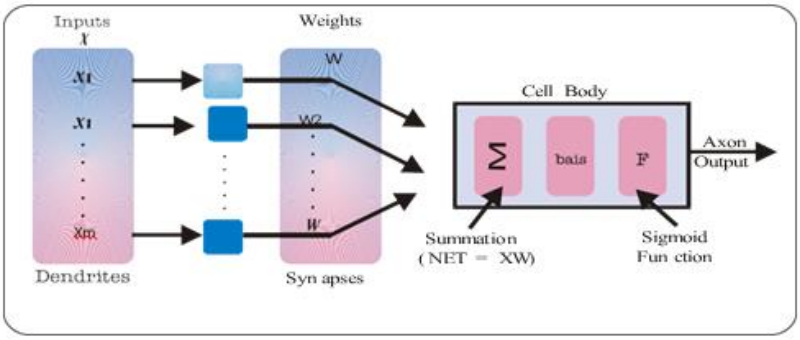

Figure 2: structure of neural network.

1.4 Activation function (sigmoid function)

As mentioned earlier, the activation function in a neural network act as a squashing function, making a neuron's output fall between predetermined values (often 0 and 1, or -1 and 1), often denotes three distinct types of activation functions. Initially, the Threshold Function is utilized, and its values range from 1 to 0, depending on whether the total input exceeds or equals the threshold value (v) or is less than or equal to that value [37].

$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases}$$

1.5 Neuron bias

In order for the network to function more quickly and get closer to the solution, the addition of a cell bias node is preferred in an error back propagation (EBP) network. With the exception of the fact that the input bias of the cell is always +1 rather than the other inputs for the other network cells, altering the weight of this cell's bias node is similar to changing the weights of the regular cells from which the network was created [21].

1.6 Learning in BP neural network

The feed-forward and feed-backward phases of the learning algorithm are executed [25]. The inputs are transmitted in the first phase. generating an output pattern as a reaction to the various processing element layers. The input pattern that was shown.

1.7 Forward Pass

The second stage is where Forward Pass begins to be used: network input values are used to calculate the values of the hidden layer, and the hidden layer values are then used to generate the output layer values. The collection is the product of this cell's entries with the associated weight (the product of a vector entry in the matrix of the weight of hidden income "), which is used to determine the values of the output cells in the hidden layer are then applied to each cell's output in that layer's output activation function. The output to the output layer, which is calculated for every cell in the hidden layer, is the input to the output layer. It operates in the same way as the preceding layer, applying the activation function (effectiveness) when the result is real network input [21].

1.8 Backward pass

Beginning with the third phase, the backward path computes the network's actual output and compares it with the planned output; the discrepancy between the two is referred to as error [26]. The regression in the value of E is represented by a vector made up of the partial derivatives of E for each weight in the network. The error E is thought of as a function of all the weight values in the network. The trend of the error value rapidly decreasing is represented by the negative regression [30]. A fast fix for the problem is to change the weight value. vij in terms of the direction of origin.

1.9 Image processing

Any type of signal processing where the input is an image, like photos, is known as image processing. The output of image processing can also be an image or a collection of parameters or features associated with the image. The majority of image processing methods handle the image as a two-dimensional signal and process it using conventional signal processing methods. While digital image processing is the most common type, optical and analog image processing are also feasible. The general methods discussed in this article are applicable to all of them [38].

1.10 Image normalization

The process of normalization modifies the range of values for pixel intensity. Applications include, for example, glare-related photos with low contrast. Another name for normalization is contrast stretching. It is known as dynamic range extension in broader data processing domains like digital signal processing. The term "normalization" refers to the process of bringing an image or other sort of signal into a range that is familiar or normal to the senses, which is typically the goal of dynamic range expansion in many applications. Achieving uniformity in the dynamic range for a set of signals, data, or visuals is frequently the driving force behind efforts to prevent mental weariness or distraction. For instance, a newspaper will aim to have a consistent grayscale range throughout all of the photographs in an issue. The process of normalization is linear. The procedure comprises deducting 50 from each pixel's intensity if the image's intensity range is 50 to 180 and the required range is 0 to 255. This makes the range 0 to 130. The range is then 0 to 255 after multiplying each pixel's intensity by 255/130. Image processing software that uses auto-normalization normally normalizes to the whole dynamic range of the number system that is provided in the image file format. Two photos of the same iris taken under different circumstances will have distinctive features at the same spatial position because the normalizing method will create iris regions with the same constant dimensions [22]. When addressing an image in neural networks, it is necessary to provide real numbers as input. These real numbers are then used as the activation function (sigmoid function) for calculating the real output per unit of output layer cells, and the extent to which this function falls between zero and one [0... 1]. The calibration data must be in a format that allows the image data to be distinguished within a range of private exchange activation, which is between 0 and 256 [30]. The calibration work is to be done in the range between [0... 1].

2. Proposed Method

It has been known the use of error BP in updating the weights demonstrated a slow progress because they use a fixed percentage of learning as well as slow convergence. In order to implement BP in neural network and diagnose Osteoarthritis, the following steps shod be flowed:

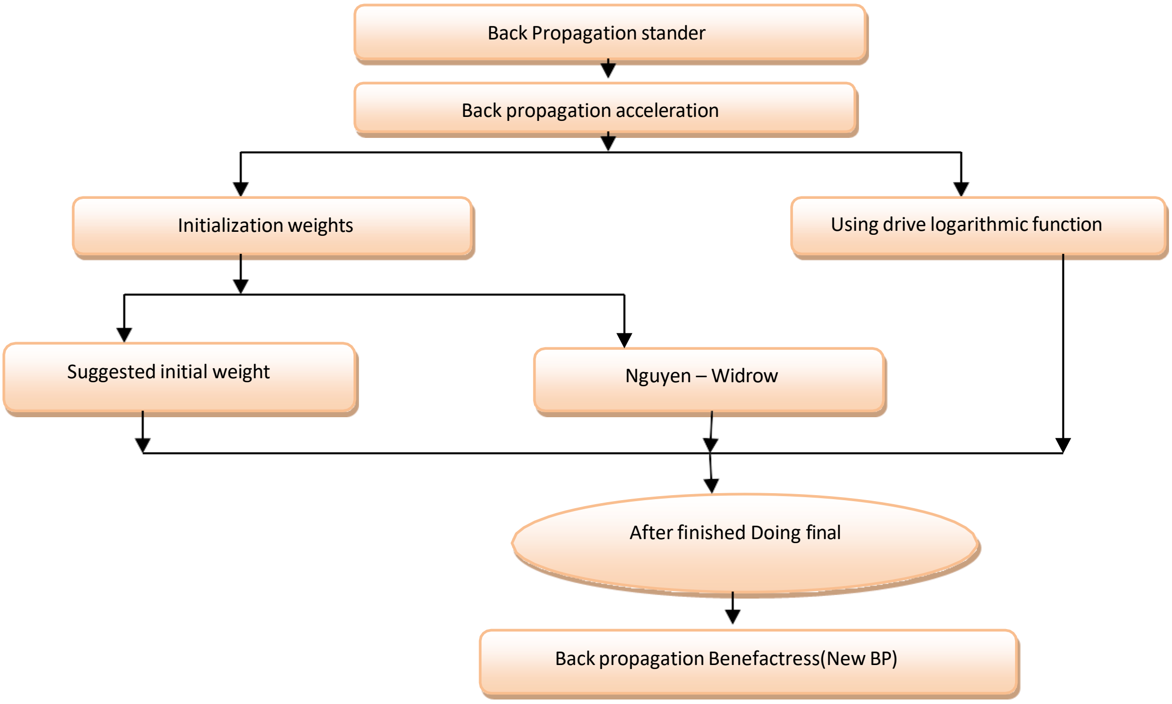

Figure 3: Proposed Method**.**

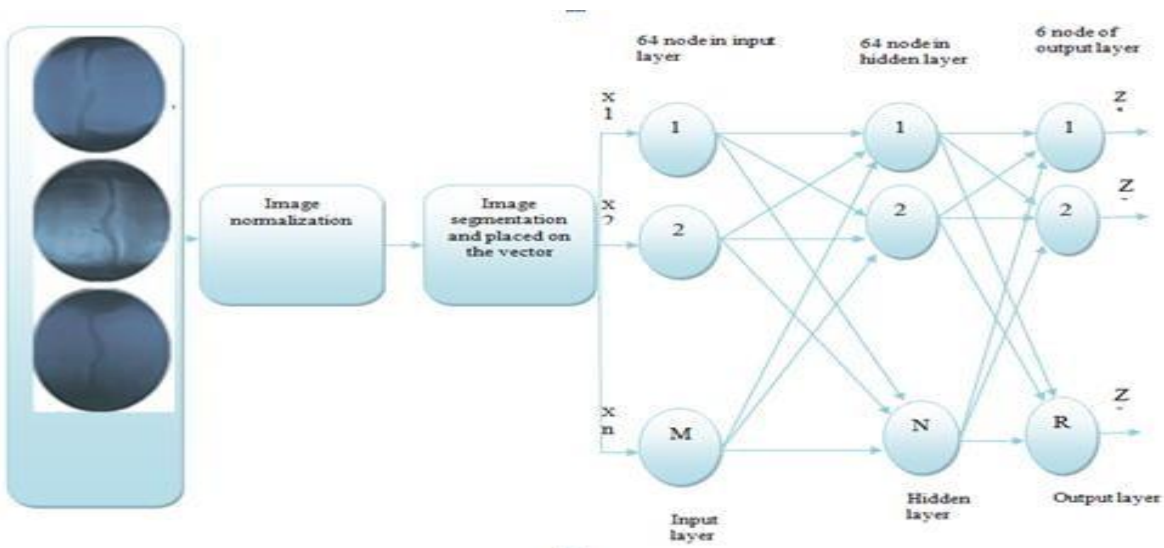

Figure 4: MLP hidden layer.

2.1 Back propagation acceleration

Error back propagation networks are known to have a sluggish convergence rate because they employ a set learning percentage, which causes the network to converge more slowly [15]. Consequently, the following categories of acceleration techniques

2.1.1 Initialization weights

The weights (w.rand) and bias in the area [-1,1] are chosen at random. If the configuration has very small weights, the hidden units and output units will approach zero, slowing down the educational process. As a result, it is necessary to select the right values for the network to be the primary input signal for the hidden units and output units in the area where the derived function has small activation values. [26] with the exception that we initialize each network's weights using the identical matrices [27].

2.1.2 using Suggested initial weight

There was another method to create the weights depend on the primary calculation of the rate of random weights for each column in the weight matrices and bias vectors. therefore, they depend on the weights of the primary network (w.typ 1). The method consists of the following steps:

1:- v_{ij} (old), w_{ij} (old): the values of a small random weights to the hidden layer and output layer, respectively, within the area [-1,1].

2:- Count rate of each column in the weight matrices M1i and m2j:

$$m1_i = \sum_{i=1}^{m} v_{ij}, \quad j = 1, \ldots, n$$

$$m2_i = \sum_{i=1}^{n} w_{ij}, \quad j = 1, \ldots, r$$

3:-To find a new weight vector , We divide the rate of the by the privies rate of weights matrices weight vector

$$v_{ij_{new}} = \frac{v_{ij_{old}}}{m1_i}$$

$$lw_{ij_{new}} = \frac{w_{ij_{old}}}{m2_i}$$

4:-Measure the rate values of bias weights of b1, b2, for the hidden and output vector respectively,we fined the new vector of the hidden bias of the new class

$$bias\_h_{new}$$

$$bias\_O_{new}$$

And the new bias for output layer As follows

$$bias\_h_{new} = \frac{bias_{old}}{b1}$$

$$bias\_O_{new} = \frac{bias_{old}}{b2}$$

2.1.3 using of Nguyen – Widrow

It is one of the methods of random configuration of weights were which used to obtain more rapid learning, to create the weights from hidden units to output units (as well as the bias of output) randomly and within the area [-0.5,0.5], while create the weights between input units and hidden layer units (as well as hidden bias) to improve the ability of the network (the hidden units to learn). [26]

This method consists (w.typ 2) of the following steps:

1:- Calculate the amount of following statement

$$\beta = 0.7(p)^{\frac{1}{n}} = 0.7\sqrt[n]{p}$$

Whereas:

n = number of units of input layer

p = number of units of hidden layer

For each hidden unit (j = 1,2, ..., P) we estimate the Wight, as: v_{ij} (old): the values of a small random weights of the hidden layer in the area [-1,1]. Then we measure the v_j (old). v_j (old) = the square root of the summation of square values of v_j(old). Then we re-create the weights as:

$$v_{ij}(\text{new}) = \frac{\beta \; v_{ij}}{\left\| v_j(\text{old}) \right\|}$$

We are now creating a hidden layer bias value:

bias $_{ij}$ = Random value within a small area [-β , β].

We proposed new techniques by modifying the way Nguyen-Windrow by adding a specific factor C1 and the new masseur new Wight and bias and calculating the new value of the bias in the second step explained as follows:

We re-create the weight of improved (w.typ 3) and as follows:

$$v_{ij}(\text{new}) = \frac{\beta \; v_{ij}}{c1 * \left\| v_j(\text{old}) \right\|}$$

We are now creating a hidden layer bias values improved:

bias $_{ij}$ = Be a random small value within a area [-β/c2 , β/c2].

C1 as fall within the area (0.1),

BP algorithm to create using the appropriate weights are as follows:

Begin

Do Initialization weights step

Do one gradient step

While (error >= Desired threshold)

Do one gradient step

End (while) End

2.1.4 using drive logarithmic function

activation function is proposed to accelerate Back propagation learning. Simulation using this activation function shows improvement in learning speed compared with other commonly used functions. This function may also be used in other multilayer feed forward training algorithms [31] explained in Figuren3.

$$0 = f(net) = \begin{cases} \ln(net+1) & net \geq 0 \\ -\ln(-net+1) & net < 0 \end{cases}$$

$$f'(net) = \frac{1}{1 + |net|}$$

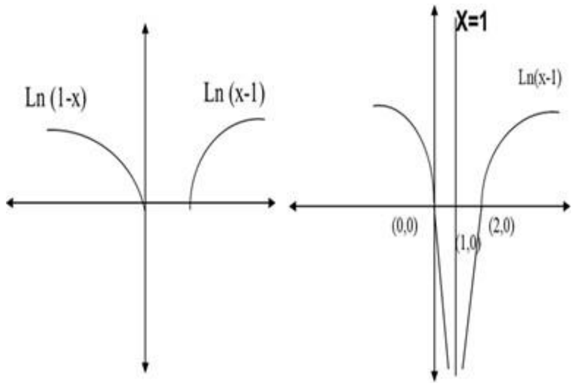

Figure 5: logarithmic function and drive**.**

3. Adapting learning rate

Learning ratio modification is another method that can demonstrate the error backpropagation network's effectiveness. The learning ratio can be continuously modified throughout the training process [29], as opposed to remaining constant throughout training [24]. A constant learning ratio results in a slow convergence of the network. The backward error propagation technique depends on knowing which way the square of the network's total error is decreasing faster in relation to the current weights when changing the weights. The network is better protected when appropriate learning rates are employed, as determined by the optimal step value in each learning stage. This is because the network is kept from deviating from the intended response and from surpassing the minimum error value that can arise in certain back-to-back error propagation algorithms. Using this method, it was possible for each weight to have its own rate of learning, and for these rates to vary over time as the learning process advances.

4. Supervised training

Training requires the presence of a pair of input vector (Input) and expected output vector (Target Output), which together represent the training pair, where the output (Output) of the applied input vector (Target Output) is compared with the expected output vector (Target Output), and the difference between them represents the error (Error) returned during the network is to change the weights according to the algorithm in order to reduce the error. The vectors of the training set are applied sequentially, and the error is calculated and the weights are changed for each vector until the error of the input training set reaches a low level or an acceptable level.

5. Image Recognition

BP improved when the conventional BP network was used. In a photograph, discrimination and the picture's requirements are represented by image calibration (normalization), image segmentation, edge detection, and fragmentation of the image [28].

6 Image Segmentation

The concept of fragmentation of the image is based on the drop in the gray levels of its constituent parts or on the similarity of these values. The method used in the project depends on the sharp changes in gray levels, the use of an edge detection algorithm, which is the way of specific edge (Sobel) [12]. Fragmentation is the process that divides the image file into its constituent parts or objects.

7 Edge detection

The border, often referred to as the edge between two sections, has reasonably defined gray features, allowing the transition between the two cuts to be made only on the basis of gray levels [12] or a quick shift in the point value for the surrounding area [32]. Solbel's candidate was utilized to ascertain the image's edges; it will take the form of a two-dimensional matrix with a size of (3 * 3) [12] as well as the following forms:

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Figure 6: Sobel filters in both directions of horizontal and vertical**.**

If the matrix A is the image, and the points a1 - a8 points are adjacent to the point of a (i, j) as shown in Figure 5.

a1	a2	a3	..	..a1,m
a4	a(i,j)	a5	..	..a2,m
a6	a7	a8	..	.
.	.	.	..	.
an,1	an,2	an,3	..	an,m

Figure 7: the matrix of the original image**.**

edge Gx, representing the value of the horizontal direction, and Gy, representing the value of the vertical edge direction is calculated, each is as follows

Gx = (a6+2a 7+a8) – (a1+2a2+a3)

Gy = (a3+2a5+a8) – (a1+2a4+a6)

Figure 8: shows the results.

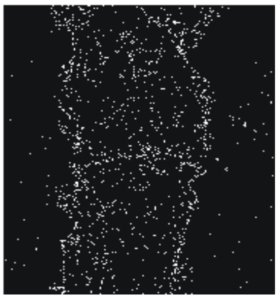

Figure 9: the application of candidate Sobel.

8. Experiment Results and Discussion

Discusses the results obtain from the experiments describe in the paper. This paper is very important as the outcomes decide whether the objectives of the study are achieved. In order to investigate the forecast result of BP learning with different number of input nodes and activation functions, empirical study were carried out on ANN model using several input datasets which have apply different types of data-pre-processing methods. Simulation with different datasets and comparison are adopted to evaluate the effectiveness of a modification.

8.1 Use of back propagation error network (standard and enhanced) in disease diagnosis

The backward error propagation network consists of three layers: an input layer, a hidden layer, and an output layer. The input layer consists of a group of cells, the number of which is according to the number of segments and is equal to 64 nodes (IN) for each image, so the input layer is a matrix with a number Rows equals the number of training models and equals 6 models (i.e. six columns), so one row contains 64 nodes and equals the number of columns of the matrix, the dimensions of the matrix (), and the number of cells of the hidden layer (hidden), so we choose 64 nodes (HID), so we have a vector with capacity , As for the output layer, the number of cells in it is equal to the number of models on which we train the network, and here it is equal to 6 nodes (OUT), which also represent a vector with capacity.

As for the goal (expected output), it is represented by a unit matrix with capacity that indicates the training models used in the network. For example, the first row refers to the first image, so the first element in that row is the number 1 and the rest of the numbers are 0. The second image represents the second row, so the first element is 0 and the first element is 0. The second is 1 and the rest are zeros, and so on for the rest of the rows, so each row refers to a model (image) of the network's training models, so the network begins to take the training pair consisting of an input vector and a target vector for a specific model (image) and trains the network on it, then calculates the real output of the network and compares it with the output. The expected and the difference between them is called the error. It is compared with a certain percentage, for example (er = 0.01). If error > 0.01, we begin the process of back-feeding the network, and we stop when error < 0.01 and the network training process ends.

8.2 The effect of initializing the initial weights on the number of steps of the standard BP

We noticed that the BP network (standard and improved) is sensitive to the initial weights because they affect the efficiency of the network. The weights and bias were initialized with small random values (rand). If these weights are not appropriate for the network, they will slow down the training process. For the purpose of accelerating the network's work, initialization was used. For initial weights, (w.ty 2) represents the (Nguyen – Widrow) method, (w.typ 3) represents the improved (Nguyen – Widrow) method, and (w.typ 1) represents the (weight matrix vector average) method. These methods have influenced on the number of training steps (iter.), since we use the same bias vectors and initial weight matrices in the rest of the networks used. As in Table 1 and Figure 10 for the standard BP network, and Table 2 and Figure 11) for the improved BP network using vectors.

Table 1 The effect of initializing the initial weights on the number of steps of the standard BP.

Weights	IN	HID	OUT	μ	Iter.
w.rand	64	64	6	0.08	26117
w.typ 1	64	64	6	0.08	17838
w.typ 2	64	64	6	0.08	10118
w.typ 3	64	64	6	0.08	8803

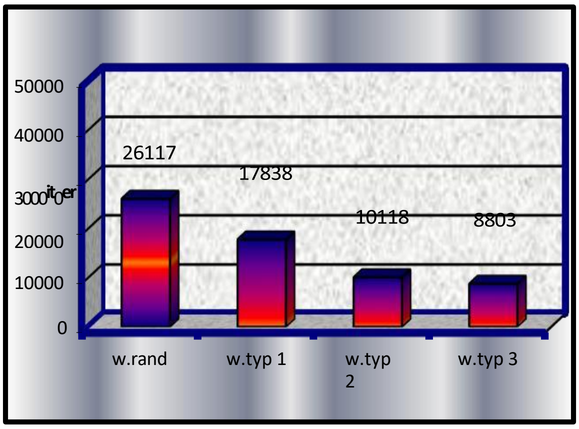

Figure 10: The relationship between initializing the initial weights and the number of steps to train the BP network.

8.3 Expected results after the implementation of road improvement in the selection of the appropriate weight in new B

We expect that the results are better than in Table 1 and that the network will be faster in the selection of Al weight, as he used to create the main weights (w.ty 2) is a method (Nguyen - Widrow) and (w.typ 3) is a method (Nguyen -- Widrow improve) and (w.typ 1) the way (the rate of weight vector arrays), we can expect that this will affect the number of steps and training (iter.), and will use the same bias and carriers in the rest of the matrices used in the initial weights of networks. The table 2 will make clear that our expectations will be in the implementation of the roads mentioned above and the previous table that follows is the results and our expectations is not the real results.

Table 2 The effect of initializing the initial weights on the number of steps of the improved BP network using vectors.

Weights	IN	ID	OUT	μ	Iter.
w.rand	64	64	6	0.08	23995
w.typ 1	64	64	6	0.08	14992
w.typ 2	64	64	6	0.08	10044
w.typ 3	64	64	6	0.08	8654

Figure 11: The relationship between initializing the initial weights and the number of steps to train the improved BP network using vectors**.**

8.4 The effect of an appropriate learning ratio parameter on the number of steps of the standard BP network

The BP network (standard and improved) relies on a fixed learning ratio (μ) during the training process. This learning ratio represents the length of the step and is limited between zero and one. If this ratio is inappropriate, it will lead to a slow training process and not reaching the desired convergence. It has been used Appropriate learning rates for the network based on regression vectors showed a noticeable improvement in reducing the number of training steps, as in Table 3 and Figure 12, where we notice that using the optimal step length (μ2) with weight (w.type 2) in learning is faster than The basic network with other types of different initial weights in terms of the number of training steps (iteration).

Table 3 Effect of an appropriate learning ratio on the standard BP network.

Weights	IN	HID	OUT	μ	μ1	μ2
w.rand	64	64	6	26117	5737	6037
w.typ 1	64	64	6	17838	4056	3190
w.typ 2	64	64	6	10118	2133	1599
w.typ 3	64	64	6	8803	2105	1532

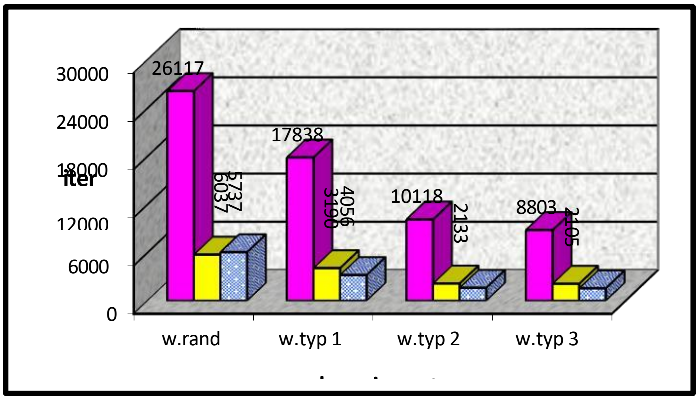

Figure 12: effect of an appropriate learning ratio parameter.

8.5 The effect of an appropriate learning ratio parameter on the number of steps of the improved BP network.

The improved BP network using the parallel tangent method is an acceleration network compared to the standard BP network. Another acceleration step was used for this network by making the learning ratio (μ) change with each iteration during the training process.

Table 4 Effect of an appropriate learning ratio on the improved BP network using matrices**.**

Weights	IN	HID	OUT	η	Iter.	
					μ 1	μ 2
w.rand	64	64	6	0.1	25883	25862
w.typ 1	64	64	6	0.1	18371	18364
w.typ 2	64	64	6	0.1	9272	9202
w.typ 3	64	64	6	0.1	8543	8527

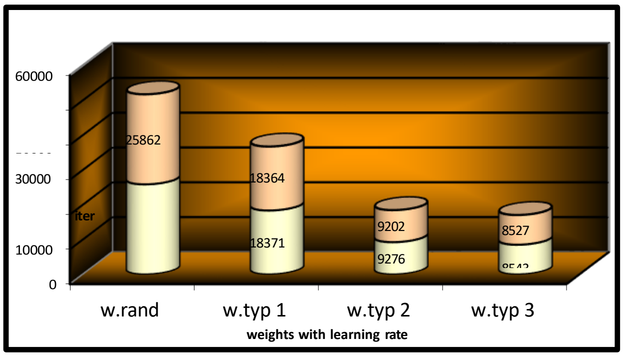

Figure 13: The relationship between an appropriate learning ratio and the number of steps for training the BP network (matrices).

Table 5 Effect of an appropriate learning ratio on the improved BP network using vectors**.**

Weights	IN	HID	OUT	η	Iter.	
					μ 1	μ 2
w.rand	64	64	6	0.1	3104	2879
w.typ 1	64	64	6	0.1	3275	2213
w.typ 2	64	64	6	0.1	2082	1542
w.typ 3	64	64	6	0.1	2057	1488

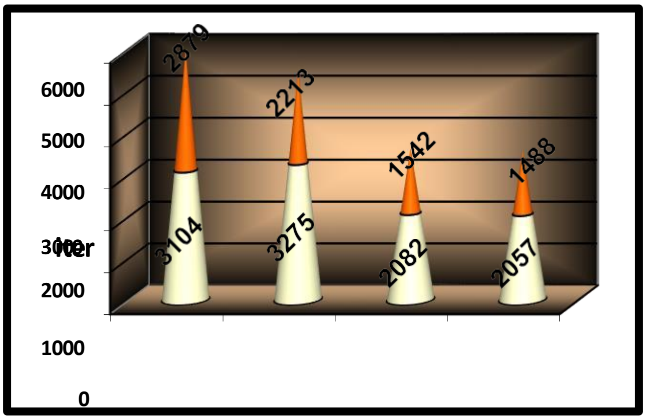

Figure 14: The relationship between an appropriate learning ratio and the number of steps for training the BP network (vectors)**.**

It was observed from Tables 4 and 5 and Figures 13,14, that the improved BP network using the weight (w.type 3) and the optimal step (μ2) is better than using other initial weights with (μ1) based on Number of training steps in network learning.

8.9 Comparison of the number of training steps for the standard and improved BP networks
Table 6, which includes the standard BP network and the improvements added to it. As shown in Figure 15, we note that the improved BP network using vectors with weight (w.type 3) and learning ratio (μ2) is better than the standard BP network with respect to the number of training steps.

Table 6 Comparison of the standard BP network with the improved BP network.

Network	Learning rate	Weight	Momentum	Iter
BP	μ = 0.08	w.rand	/	26117
	μ 2	w.typ 3	/	1532
	μ 2	w.typ 3	0.1	1387
BP.vec	μ = 0.08	w.rand	/	23995
	μ 2	w.typ 3	/	1488
	μ 2	w.typ 3	0.1	1341
BP.mat	μ = 0.08	w.rand	/	25862
	μ 2	w.typ 3	/	8527
	μ 2	w.typ 3	0.1	8535

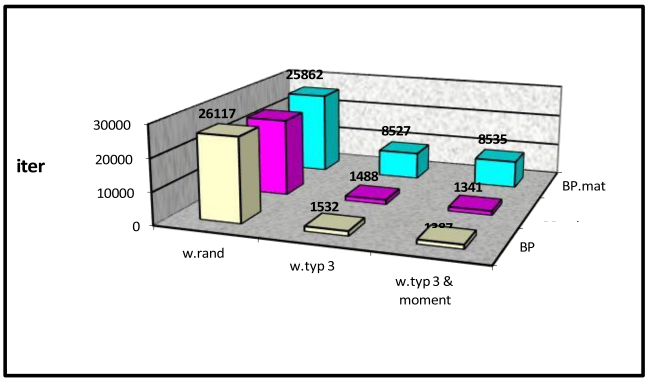

Figure 15: Additional improvements to the standard BP network**.**

From the above figure, it is clear that the improved BP network using vectors is better in terms of the number of training steps

8.10 The effect of the number of hidden layer cells on the number of training steps

As we mentioned previously, the input layer consists of a group of cells whose number is equal to 64 nodes (IN) for each image, and the number of cells in the output layer is equal to 6 nodes (OUT). As for the hidden layer, 50 nodes (HID) are chosen.

Table 7 shows the effect of the number of hidden layer cells on the number of training steps in the standard BP network**.**

weight	IN	HID	OUT	$\mu2$
w.typ 4	64	64	6	1532
	64	50	6	1184

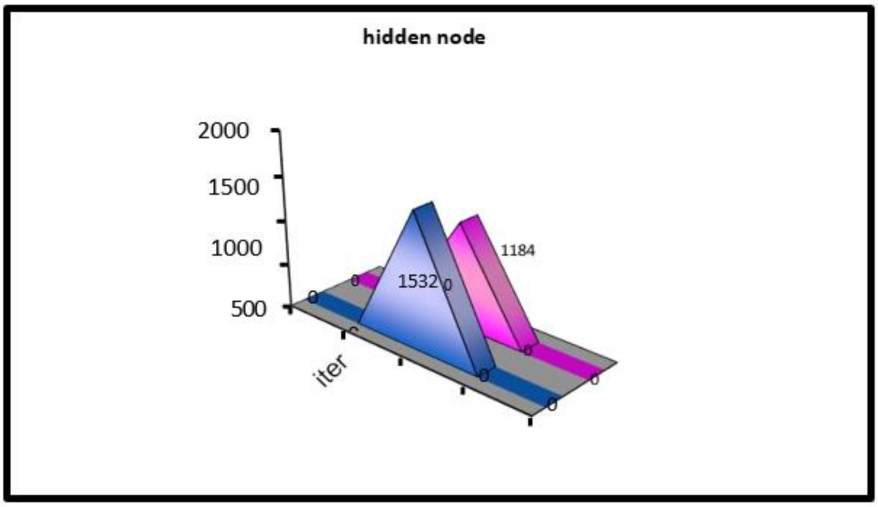

Figure 16: The relationship between the number of hidden layer cells and the number of training steps.

9.Conclusion

A neural network with standard and improved backpropagation of error was used in diagnosing osteoarthritis. An appropriate configuration of weights for the neural network was also used and compared with the results of using random weights, and we used learning ratios. Suiting the standard and improved network showed an acceleration in the network, and adding the motor torque parameter had an effect in reducing the number of training steps. It was also noted in Table 6 that the improved network with (optimal step size) $\mu2$, weight (w.typ 3) and momentum, is more efficient than the standard BP network and the other additions that were mentioned in Chapter Three in terms of the number of training steps while maintaining accuracy. Output. In Table 7, the number of hidden nodes of the network affects the efficiency of the network and the accuracy of training, as it was noted that the number 50 is more efficient than the number 64 with regard to the number of training steps, but sometimes the accuracy of the output at 64 is better.

References

[1] J. Samarabandu, and X. Liu ,"An Edge-Based Text Region Extraction Algorithm for in Door Mobile Robot Navigations". International Journal

[2] Abb0 K.K ,"Numerical Study In Interleaving PARTAN and Self-Scaling Variable Metric Updates by Different Switching Criteria", College of Computer Sciences and Mathematics, University of Mosul , (1998).

[3] AL-Assady, N.H. and AL-Bayati, A.Y." conjugate gradient method" , Technical Research report, NO.(1), school of computer studies, Leeds university , U.K. (1986)

[4] Bazarra.M. Sherali .H. and Shetty.C.M., “ Nonlinear Programming Theory And Application “, New York , John Wiley and Sons, Inc., (1993).
[5] Demuth H, Beale M.“ Neural Network Toolbox User’s Guide “, by The Math Works . Inc., (1992 – 1998).
[6] Dixon, L.C.W., “ Non-Linear Optimization “, English Universities Press, London, (1972) .
[7] Edgar, T.F., and Himmellau ,D.M.,“ Optimization Of Chemical Processes “, Mc Graw-Hill International Edition , (1989).
[8] Edward, J. Rzemoluck, “ Neural Network Data Analysis Using Simulnet “, Springer-Verlag New York, (1998).
[9] Fletcher, R.,“ An Overview Of Unconstrained Optimization “, numerical Analysis Report NA/149.Scotland, (1993).
[10] Fletcher R., “ Practical Methods Of Optimization “, John Wiley & Sons, Chichester , New York , (1987).
[11] Gill, P,E., and Murray ,W., “Practical Optimization “, Academic Press Inc., London , (1981).
[12] Gonzalez, Rafael C. and Woods, Richard E.,” Digital Image Processing “, Addison-Wesley Publishing Company Inc.,(1992).
[13] Hestenes , M.R.,“ Conjugate Direction Methods In Optimization “, Springer-Verlag Berlin , (1980).
[14] James L. Buchanan and Peter R. Turner, “ Numerical Methods And Analysis “, Mc Graw – Hill , (1992).
[15] Jose C. Principe ,Neil R. Euliano and Curt W.,” Neural And Adaptive Systems “, John Wiley &Sons , New York , (2000).
[16] Khaleel S.I., “Improvement Of The Backpropagation Neural Networks Algorithm & it’s Applications “, College of Computer Sciences and Mathematics, University of Mosul , (2000).
[17] Luger G.F.& Stubblefield W.A., “ Artificial Intelligence Structures And Strategies For Complex Problem Solving “, Addison Weley Longman , Inc., USA, (1998).
[18] Pao ,Yoh-Han “ Adaptive Pattern Recognition and Neural Networks “, Wesley Publishing Company ,Inc., New York , (1989).
[19] Wilde .J.D., “ Foundations Of Optimization “, by Prentice Hall , Inc., Englewood Cliffs.N.J. USA , (1967).
[20] Wasserman, P.D., “Neural Computing Theory And Practice “, Van Nostrand Reinhold, New York , (1989).
[21] Zurada, R. J. Marks, and C. J. Robinson, Eds. Piscataway, NJ: IEEE Press, 1994, pp. 352-363
[22] JP33. Joarder, A.H. and Al-Sabah, W.S. (2002). The dependence structure of conditional probabilities in a contingency table.
[23] International Journal of Mathematical Education in Science and Technology 33(3),475-480
[24] Arnold WD, Duque S, Iyer CC, Zaworski P, McGovern VL, Taylor SJ, von Herrmann KM, Kobayashi DT, Chen KS, Kolb SJ, Paushkin SV, Burghes AH. Normalization of Patient-Identified Plasma Biomarkers in SMNDelta7 Mice following Postnatal SMN Restoration. PloS one 2016;11(12):e0167077.
[25] Fogel l1,owners AJ ,Walsh MJ (1966) Artificial intelligence through simulated evolution ,wilky,new york.
[26] Argha Roy, Diptam Dutta, Kaustav Choudhury," Training Artificial Neural Network using Particle Swarm Optimization Algorithm ",
[27] Volume 3, Issue 3, March 2013
[28] Y Fukuoka, H Matsuki, H Minamitani, A Ishida "A modified back-propagation method to avoid false local minima",Neural networks, 1998•Elsevier
[29] Nicol N. Schraudolph, Jin Yu, Simon Günter Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, PMLR 2:436-443, 2007.
[30] B. Verma, , M. Blumenstein, and S. Kulkarni," A New Compression Technique Using an Artificial Neural Network " Open Access Published by De Gruyter February 1, 1999
[31] Warren S. McCulloch & Walter Pitts," A logical calculus of the ideas immanent in nervous activity " , Volume 5, pages 115–133, (1943)
[32] Scott E Umbaugh," Digital Image Processing and Analysis", Computer Vision and Image Analysis, 18 January 2023
[33] M H Hamza," Proceedings of the IASTED International Conference: Applied Informatics: International Symposium on Artificial Intelligence and Applications", IASTED International Conference on Applied Informatics 2002 - Innsbruck, Austria

[34] Junfeng Jing a b, Shenjuan Liu a, Gang Wang c, Weichuan Zhang d, Changming Sun ." Recent advances on image edge detection: A comprehensive review", Volume 503, 7 September 2022, Pages 259-271.

[35] Yu, Y. T. and Lau, M. F. 2006. A comparison of MC/DC, MUMCUT and several other coverage criteria for logical decisions. J. Syst. Softw. 79, 5 (May. 2006), 577-590. DOI= http://dx.doi.org/10.1016/j.jss.2005.05.030

[36] Eda Goz a, Mehmet Yuceer b, Erdal Karadurmus c," Total Organic Carbon Prediction with Artificial Intelligence Techniques" Volume 46, 2019, Pages 889-894.

[37] https://s.mriquestions.com/what-is-a-neural-network.html," Neural Network Definition and Components"

[38] Julia García Cabello," Mathematical Neural Networks", Department of Applied Mathematics, University of Granada, 18071

[39] Granada,Khuyen Le ," Activation functions", https://lekhuyen.medium.com/activation-functions-17f8e7d5fcf8.

[40] Sahil V. Khedaskar1 , Mohit A. Rokade1 , Bhargav R. Patil1 , Tatwadarshi P. N.2," A Survey of Image Processing and Identification Techniques", VIVA-Tech International Journal for Research and Innovation Volume 1, Issue 1 (2018).