Tactile Gesture Recognition for Paralysed and Especially Abled Patients using Deep Learning and Neural Network

Siddhant Jain¹ Shubham Gera² Varun Goel³
1,2,3Maharaja Agrasen College, India

Abstract— An Artificial Neural Network (ANN) is a data handling worldview that is motivated by the way organic sensory systems, for example, the cerebrum, process data. The crucial element of this network is the novel structure of the data handling framework. It is made out of an expansive number of extremely interconnected processing components (neurons) working as one to tackle explicit issues. ANNs, similar to individuals, learn by precedent. An ANN is designed for a particular application, for example, pattern recognition or data characterization, through a learning procedure. Learning in natural frameworks includes changes in accordance with the conjunction associations that exist between the neurons. This is valid for ANNs also. This paper gives an overview of Artificial Neural Networks, working and preparing of ANN just as how an ANN can be utilized to the undertaking of hand gesture recognition explicitly for incapacitated and distinctively-able individuals.

Key words: Artificial Neural Network (ANN), Hand Gestures Image Database, Gesture Feature Extraction

I. INTRODUCTION

The study of the human brain is thousands of years old. With the arrival of contemporary physical science, it absolutely was solely natural to harness this thinking method. The stepping stone towards artificial neural networks came in 1943 once Warren McCulloch, a neurophysiologist, and a young mathematician, Walter Pitts, wrote a paper on how neurons might work. They sculptured a straightforward neural network with electrical circuits. Neural networks, having an outstanding ability to derive meaning from imprecise or complicated data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network will be thought of as an “expert” within the class of data it’s been given to analyse has been given to analyse. Other advantages include:

1) Adaptive Learning: A capability to be told the way to do tasks supported with the information given for initial expertise or training.

2) Self-Organization: An Artificial Neural Network will produce its own organization or illustration of the knowledge it receives throughout the learning process.

3) Real Time Operation: Artificial Neural Network computations could also be done out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability.

4) Fault Tolerance via Redundant Data Coding: Partial destruction of a network ends up in the corresponding degradation of performance. However, some network capabilities could also be maintained even with major network harm.

Neural networks take a different approach to problem solving that of conventional computers. Conventional computers use an algorithmic approach i.e. the computer follows a set of directions so as to resolve a haul. That restricts the problem determination and solving capability of standard computers to issues that we have tendency to already perceive and the knowledge to resolve. But computers would be most helpful if they do things that we have a tendency not to perceive and the knowledge to resolve. Neural networks process information in an exceedingly similar means as the human brain does. The network is composed of large number of highly interconnection processing elements (neurons) working in parallel to solve a specific problem. Neural networks learn by example. They cannot be programmed to perform a selected task. The examples should be elected fastidiously otherwise necessary time is wasted or worse the network could be functioning incorrectly.

The disadvantage is that as the network finds out the way to solve the matter by itself, its operation can be unpredictable. On the opposite hand, conventional computers use a cognitive approach to problem solving; the way the problem is solved must be know and stated in small unambiguous instructions. These directions are then translated to a high-level language program, then into machine code that the computer will perceive. These machines are fully predictable; if something goes wrong is thanks to a software system or hardware fault. Neural networks and traditional algorithmic computers aren’t in competition however complement one another. There are tasks are more suited to an algorithmic approach like arithmetic operations and task that are more suited to neural networks. Even more, an oversized large variety of tasks, need systems that use a mixture of the two approaches (normally a standard PC is employed to supervise the neural network) in order to perform at maximum efficiency.

A. Artificial Neural Network

Artificial Neural Networks are relatively crude electronic models based on the neural structure of the brain. The brain basically learn from experience. It is natural proof that some problems that are beyond the scope of current computers are indeed solvable by small energy efficient packages. This brain modeling also promises a less technical way to develop machine solutions. This new approach to computing also provides a more graceful degradation during system overload than its more traditional counterparts. These biologically inspired methods of computing are thought to be the next major advancement in the computing industry. Even the simple animal brains are capable of functions that are currently impossible for computers.

Whenever we talk about a neural network, we should more popularly say - Artificial Neural Network (ANN) ||. ANN are computers whose architecture is modeled after the brain. They typically consist of hundreds of simple processing units which are wired together in a complex communication network. Each unit or node is a simplified model of real neuron which sends off a new signal or fires if
it receives a sufficiently strong input signal from the other nodes to which it is connected.

Traditionally neural network used to refer as network or circuit of biological neurons, but modern usage of the term often refers to ANN. ANN is a mathematical model or procedure model, an information processing paradigm i.e. inspired by the approach of biological systema nervosum, such as the brain information system. ANN is made up of interconnecting artificial neurons which are programmed in order to mimic the properties of biological neurons. These neurons operate in union to resolve specific issues. ANN is organized for finding artificial intelligence issues while not making a model of real biological system. Artificial Neural Network is employed for speech recognition, image analysis, adaptive control etc. These applications are done through a learning method, like learning in biological system, which involves the adjustment between neurons through synaptic connection. Same happen in the ANN.

B. Working of ANN:
The other elements of the art of utilizing neural networks revolve around the myriad of the way these individual neurons are clustered along. This agglomeration happens within the human mind in such a way that data is processed in a very dynamic, interactive, and self-organizing way. Biologically, neural networks are constructed in a three-dimensional world from microscopic components. These neurons seem capable of nearly unrestricted interconnections. That is not true of any projected, or existing, man-made network. Integrated circuits, using current technology, are two-dimensional devices with a limited number of layers for interconnection. This physical reality restrains the kinds, and scope, of artificial neural networks which will be enforced in chemical element.

Currently, neural networks are easy agglomeration of the primitive artificial neurons. This agglomeration happens by making layers that are then connected to one another. How these layers connect is the different part of the "art" of engineering networks to resolve world issues.

C. Training an Artificial Neural Network
Once a network has been structured for a selected application, that network is prepared to be trained. To start this method the initial weights are chosen haphazardly. Then, the training, or learning, begins. There are generally two approaches to learning - supervised and unsupervised learning. Supervised learning involves a mechanism of providing the network with the specified output either by manually "grading" the network's performance or by providing the specified outputs with the inputs. Unsupervised learning is wherever the network has make sense out of the inputs while not using outside facilitate. The vast bulk of networks utilize supervised training. Unsupervised learning is employed to perform some initial characterization on inputs. However, within the full blown sense of being actually self-learning, it's still simply a shining promise that's not totally understood, does not completely work, and thus is relegated to the lab.

![A simple neural network](image1)

Fig. 1: A Simple Neural Network Diagram.

Basically, all artificial neural networks have a similar structure or topology as shown in Figure-1. In that structure a number of the neurons interface to the real world to receive its inputs. Other neurons offer the real world with the network's outputs. This output could be the character that the network thinks that it has scanned or the image it thinks is being viewed. All the remainder of the neurons area unit hidden from read.

But a neural network is not just a bunch of neurons. Some early researchers tried to connect neurons in a very random manner, without much success. Now, it's glorious that even the brains of snails are structured devices. One of the best ways in which to style a structure is to form layers of parts. It is the grouping of those neurons into layers, the connections between these layers, and the summation and transfer functions that comprises a functioning neural network. The general terms wont to describe these characteristics are common to all or any networks.

Although there are helpful networks that contain just one layer, or even one element, most applications require networks that contain at least the three normal types of layers - input, hidden, and output. The layer of input neurons receives the information either from input files or directly from electronic sensors in time period applications. The output layer sends info on to the surface world, to a secondary computer process, or to other devices such as a mechanical control system. Between these two layers is several hidden layers. These internal layers contain several of the neurons in varied interconnected structures. The inputs and outputs of every of those hidden neurons merely move to different neurons.
1) **Supervised Training.**

In supervised coaching, both the inputs and therefore the outputs are provided. The network then processes the inputs and compares its ensuing outputs against the specified outputs. Errors are then propagated back through the system, causing the system to adjust the weights which control the network. This method happens over and over because the weights are frequently tweaked. The set of information that permits the learning is termed the "training set. Many layered networks with multiple nodes are capable of memorizing data. To monitor the network to see if the system is solely memorizing its information in some non-vital means, supervised learning has to hold back a certain set of data in order for it to be used to test the system after it has completely undergone its training.

2) **Unsupervised, or Adaptive Training.**

The other style of training is termed unsupervised training or learning. In unsupervised training, the network is provided with inputs but not with desired outputs. The system itself should then decide what options it’ll use to cluster the input file. This is often referred to as self-organization or adaptation. At this time, unsupervised learning isn't well understood. This adaptation to the atmosphere is that the promise which might modify fantasy styles of robots to repeatedly learn on their own as they encounter new things and new environments. Life is stuffed with things where actual coaching sets don't exist. Some of these things involve group action where new combat techniques and new weapons may well be encountered. Because of this sudden facet to life and therefore the human need to be ready, there continues to be research into, and hope for, this field. Yet, at this time, the immense bulk of neural network work is in systems with supervised learning. Supervised learning is achieving results.

II. **LITERATURE REVIEW**

The paper by N. Dawar and N. Kehtarnavaz [1], talks about deep learning based gesture recognition. The hybrid system consists of depth camera, which is used by convolution neural network and a wearable inertial sensor which is used along with memory elements in the convolution network. It does segmentation of all the actions and then matches it to the recognizable actions. The developed system is used for two applications: for smart TV gestures and one for home healthcare system. P. Wang, W. Li, P. Ogunbona, J. Wan and S. Escalera [2], talks about the recent advances in the area of RGB-D based gesture recognition. In particular, Convolution Neural Networks (CNN) and Recurrent Neural Networks (RNN) architectures have been adopted for RGB-D based motion detection. The paper surveys a few methods for human motion detection and also talks about the drawbacks and positives of the existing techniques. The method for encoding spatial temporal structural inherently. A research paper on Tactile Gestures by Voyles and P. Khosla [3], states that Gesture based programming is used to ease the burden to train robots. Programming real world, complex problems is based on non how well we are able to do verbal communication. This paper uses an application of distributed perception for inferring an user’s intention by observing the tactile gestures. These gestures consist of sparse, inexact, physical “nudges”; applied to the robot’s end effector for the purpose of modifying its trajectory in free space. A set of independent agents each with its own local, fuzzified, heuristic model of a particular trajectory parameter observes data from a wrist force/torque sensor to evaluate the gestures. The agents then independently determine the confidence of their respective findings and distributed arbitration resolves the interpretation through voting. H. Liu, J. Graco, X. Song, J. Bimbo, L. Seneviratne and K. Althoefer [4], proposes a novel algorithm for identifying the shape of object which in contact with a robotic finger through the tactile pressure sensing, algorithm is capable of differentiating the contact shapes between a set of low-resolution pressure map. In this research paper a novel feature extraction technique is formed which converts apressure map into a 512-feature vector. The extracted feature of the pressure map is indifferent to scale, positioning and partial occlusion, and is independent of the sensor’s resolution or image size. Tests using four different contact shapes achieved an average classification precision of 91.07%. A limitation of this research paper’s developed algorithm is that the feature extraction algorithm is not used to the object rotation, which is to be worked upon. Another paper by K. Bernardin, K. Ogawara, K. Ikeuchi and R. Dillmann [5], uses both hand shape and contact point information obtained from a data glove and tactile sensors to recognize and understand continuous human grasp sequences. The sensor fusion, enables classification and task segmentation are made by a Hidden Markov Model recognizer that differentiates 14 grasp types; as displayed in Kamakura’s taxonomy. An accuracy of up to 92.2% for a single human system, and 90.9% for a multiple user system can be achieved. Both finger joint angles and information on contact surfaces, obtained by the data glove and tactile sensors, are used efficiently. Using Hidden Markov Models, the system combines the two sensor inputs, detects the grasping phases in a user demonstration and displays the grasps with a single, statistically perfect approach. This research paper’s future work will include further improving the tactile input on key points of the hand, using in addition visual features, like rough object size or shape, and analyzing grasps transitions, which happens when the grasp type is changed without releasing the object. [6] In this research paper, machine learning techniques (namely, support vector machines and extreme learning machines) supports a pattern-recognition framework that can fully use the tensor morphology of the tactile signal. Furthermore in this research paper, a practical strategy is discussed to address the intricacies of the training procedure. Experimental results show the efficiency of the proposed approach. The idea behind this approach is twofold. First, tactile interactions are translated by stimuli that can only be fully characterized by exploiting tensor morphology. Second, inductive learning proves to be an effective tool to model complex, non-linear mechanisms. An appealing characteristic of this research paper’s proposed framework is the ability to surround the class of regularized kernel methods. This research paper indeed exploited this feature to propose a practical criterion to support model selection, which may represents an important task in the presence of noisy training data. B. Toghiani-Rizi, C. Lind, M. Svensson and M. Windmark [7], state that, an automated bartender system was developed for making orders in a bar using hand gestures. The gesture...
Tactile Gesture Recognition for Paralysed and Especially Abled Patients using Deep Learning and Neural Network

III. METHODOLOGY

This section will introduce the proposed tactile gesture based hand gesture recognition system to recognize the chosen set of five hand gestures, termed as: “Best of Luck”, “Fist”, “Palm”, and “Double U”. The proposed algorithm consists of three basic stages: pre-processing, feature extraction, and classification. As known the main problems of 2D object recognition are the size changing, translation of position, and rotation by angle from the principle axes. In this algorithm two methods are proposed; in the first method the hand contour is extracted as a geometric feature and the method treats the problems of size changing and translation (in some cases). The second method treats the problem of rotation in addition to previous problems using the hand complex moments feature. The extracted features are entered to the last stage (neural networks using the supervised back-propagation learning algorithm), and this stage is responsible for recognizing and deciding to which class the hand gesture belongs. It is known that to have a robust classification it is necessary to choose properly the features as well as the correct way to present the features to the classifier. This work explores mainly this topic finding proper features and proper way to represent them that matches the classifier characteristics. The proposed hand gesture recognition algorithm consists of two methods: neural network with hand contour and neural network with hand complex moments. Either of these methods is formed by four serial stages: hand image capture, image pre-processing, feature extraction, and classification, as seen in Fig. 3. The role of the pre-processing module is to section the pattern of interest (hand) from the background, removing noise and any other operation which will contribute to defining a compact representation of the pattern.

Fig. 3: Overview of gesture recognition system

In the training phase, the feature extraction module finds the appropriate features for representing the input patterns and also the classifier is trained using the back-propagation algorithm. In the testing phase, the trained classifier will assign the input pattern (test hand gesture image) to one of the pattern classes under consideration based on the measured features. The next sections present a detailed explanation of the proposed methods.

A. Hand Gestures Image Database

The starting point of this project was the creation of a database with all the hand gesture pictures that might be used for training and testing. The construction of such a database is clearly dependent on the application. Each gesture represents a gesture command mode. These commands are widely used in various application programs. Therefore, these gestures are allowed to be flexible and natural so that they can successfully be applied to a wide range of people and situations. The gestures images are real images with different sizes acquired using digital camera and taken from one subject.

The database consists of 500 images for training set, with hundred samples for each gesture with scaling effect, translation, and rotation.

B. Neural Network with Hand Contour

The recognition process consists of two phases: training (learning) and classification (testing). Steps included in each phase are as described in the following lines.

1) Training Phase
   1) Capturing Gesture Image from training set.
   2) Segmentation to isolate hand object from the background.
   3) Noise reduction to remove any noise from the segmented image.
   4) Edge detection to search out the hand gesture image boundaries.
   5) Contour detection as a geometric feature.
   6) Calculating the peak offset and dimension offset as a general feature.
   7) Saving feature image as a training pattern.

2) Testing Phase
   1) Capturing Gesture Image from testing set.
2) Segmentation to isolate hand object from the background.
3) Noise reduction to remove any noise from the segmented image.
4) Edge detection to find the gesture image boundaries.
5) Contour detection as a geometric feature.
6) Calculating height offset and width offset as a general feature.

C. Pre-Processing

The primary goal of the pre-processing stage is to make sure a uniform input to the classification network. This stage includes hand segmentation to isolate the foreground (hand gesture) from the background, removing any noises caused by segmentation process using special filters. This stage also includes edge detection to search out the ultimate form of the hand.

1) Hand segmentation

The hand image is segmented from the background. The segmentation process, however, should be fast, reliable, and consistent, and able to produce the best-quality output possible giving the constraints, which must produce an image suitable to recognize the gesture of the hand. In this work, a Thresholding algorithm is used for segmentation of the gesture image as it is fast and requires a very low time comparing with the other methods which is a very important feature especially for real-life applications.

2) Noise reduction

Once the hand gesture image has been segmental, a special filter is applied. The goal of applying a filter is to eliminate all the single white pixels on a black background and all the single black pixels on a white foreground. To accomplish this goal, a median filter is applied to the segmented image.

3) Edge detection

For recognizing static gestures in our system, the model parameters are derived from description of the shape, and the boundary of the hand is extracted for further processing. Therefore, several different edge detector operators were tried.

In addition to segmentation and noise reduction processes as in previous method, the pre-processing involves another operation that includes image trimming, scaling, and coordinate normalization.

a) Image Trimming

The image of hand gesture may include additional empty lines and columns that haven’t any information (space lines); these empty lines should be eliminated by tracing from outside margins towards inside and stopping at the first occurrence of the pixel at each side of the four edges.

b) Image Scaling

The dimensions of the hand gesture images are varying due to capturing process therefore the image size is adjusted to fixed size (250 × 250) in order to facilitate the calculation of the next stage.

D. Gesture Feature Extraction

The objective of the feature extraction stage is to capture and discriminate the most relevant characteristics of the hand gesture image for recognition. The selection of good features can strongly affect the classification performance and reduce the computational time. These selected features, consequently, result in an easier classification task. The features used must be suitable for the application and the applied classifier. In a proposed method two types of features are extracted, namely, the hand contour as a geometric feature and height width off sets of the hand image as assistant general features.

1) Geometric Feature (Hand Contour)

One of the most important geometric features for static hand gesture recognition is hand contour. Once the contour map is produced, feature image scaling takes place. Feature image scaling is a very simple method which reduces the feature image size by selecting several rows and columns. For all the images in the training set, a scaling is performed. The result is feature image with 32, the number of the rows, and 32, the number of the columns. After the feature image scaling, a feature gesture image must be prepared before entering the last stage, i.e., classifier stage. The preparing step includes shifting the hand gesture section to the origin point (0, 0) to solve translation problem.

IV. EXPERIMENT AND RESULT

Hand gesture recognition is a decision concerning the gesture of the user which cannot be made unless the gesture or recognition algorithm passes through several operations on unknown hand gesture. These operations are required before deciding to which class the gesture belongs. Hand gesture recognition algorithm includes two methods, extracting the hand contour as a geometric feature and treating the problem of rotation in addition to previous problems using the hand complex moments feature. Each of these methods consists of two phases: training phase and testing phase. The training phase works on the gesture used to train neural networks while the testing phase works on the hand gestures used to decide to which class they belong, which will not be used in the training phase. Both phases consist of two main processes: pre-processing and feature extraction. In this section, the results obtained from applying the proposed hand gesture recognition algorithm to both methods and the effects of each of the processes are presented. For each hand gesture of the five selected gesture classes, several samples are taken under different light conditions (natural and artificial light conditions).

Fig. 4. Central Coordinates Normalization
A. Hand Gesture Segmentation
All the techniques that are used in this paper are based on hand shape. The acquiring color gesture image is segmented to isolate the foreground hand from the background.

B. Noise Reduction
The segmented hand image may contain some noises that will affect the result values produced by the feature extraction stage. Hence, the values of the computed features will be different for the same image if it contains noise. So using the median filter will reduce the noise as much as possible.

1) Feature Extraction
The hand gesture image that has passed through image pre-processing stage is fed to the feature extraction stage to compute the feature information about the image. As soon as contour feature is computed the image size is adjusted so that each hand gesture image size becomes $32 \times 32$. Image resizing makes the system faster; this operation will reduce the negative effects of the size change. The general features (height offset and width offset) will be computed implicitly.

a) Training Phase
In this phase, the composite feature vectors computed earlier and stored in a feature images database are fed to the next stage of our system as inputs. These feature vectors are used to train the neural networks. The learning process for the five multilayer neural networks is accomplished using the following parameters such as the Input Layer, the Hidden Layer, the Output Layer, Number of images for each training set, and Learning Rate. The neural networks are trained through successive iteration, after each iteration the square error over the validation set is computed.

b) Testing Phase
After training neural networks, performance is estimated by applying the testing set to the network inputs and computing the classification errors. The test gesture feature image will be entered into the first neural network. In this phase, if the network succeeds to recognize the gesture, the test operation is stopped. If this network does not recognize the gesture features, the second network will be activated and so on. If all networks fail to identify the features, “gesture not recognized” message will appear to announce the failure in recognition. In this phase, 15 hand gesture images are used to test the system with different light conditions and with scaling and translation effects. The system now is able to recognize and classify any unknown gestures if they are in the original database. Each gesture has a table of recognition results and with neural network outputs for one gesture image the performance of the proposed system is evaluated based on its ability to correctly recognized gestures to their corresponding input gestures, the metric that is used to accomplish this job is called the recognition rate. The recognition rate is defined as the ratio of the number of correctly recognized gestures to the total number of input gestures as shown in the equation:

\[
\text{Recognition Rate} = \frac{\text{No. of correctly recognized gestures}}{\text{No. of total recognized gestures}} \times 100
\]

A summary of all recognition results and the recognition rates for each of the five gestures is presented by Table 1, and the recognition rates with each class are shown in Fig. 5.

<table>
<thead>
<tr>
<th>Gesture Meaning</th>
<th>Number of Test Gestures</th>
<th>Successful Recognition</th>
<th>Recognition Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best of Luck</td>
<td>50</td>
<td>48</td>
<td>96</td>
</tr>
<tr>
<td>Fist</td>
<td>50</td>
<td>49</td>
<td>98</td>
</tr>
<tr>
<td>Palm</td>
<td>50</td>
<td>45</td>
<td>90</td>
</tr>
<tr>
<td>Peace</td>
<td>50</td>
<td>46</td>
<td>92</td>
</tr>
<tr>
<td>Double U</td>
<td>50</td>
<td>46</td>
<td>92</td>
</tr>
<tr>
<td>Total</td>
<td>250</td>
<td>234</td>
<td>93.6</td>
</tr>
</tbody>
</table>

Table 1. Summary of recognition results and recognition rates

The pre-processing stage in this method includes image trimming followed by normalization process. The effect of these operations will be presented in the next sections.

![Fig. 5: Percentages of Correct Recognition](image-url)
2) **Image Trimming Effect**

The hand gesture filtered image may contain unused space surrounding the hand gesture so the image trimming process is used to extract the hand gesture from its background.

3) **Coordinate Normalization**

After scaling each image size to fixed size 250x250, the coordinates for the hand image are normalized between [-1,1]. Each hand gesture in the training set will have a feature vector of ten values and these values represent the complex moments starting with zero order up to nine order.

**REFERENCES**


