A Method for Locating BK Anatomical Landmarks for a Laser Scanning Imager

Daniel Jiménez, M.S. (jimenezd@uthscsa.edu)
Thomas Darm, C.P.O.
Bill Rogers, M.S.
Nicolas Walsh, M.D.

The University of Texas Health Science Center at San Antonio
Department of Rehabilitation Medicine
7703 Floyd Curl Drive
San Antonio, TX 78284
USA

Kurzfassung

Die CAD Bearbeitung eines Schaftes für eine Unterschenkelprothese nimmt ihren Ursprung in der digitalen Entsphrechung der äusseren Form des Stumpfes. Üblicherweise werden anatomische Fixpunkte auf dem Stumpf (bei Scannern mit Markerkennung) oder auf der bereits digitalisierten Form markiert, die die Zuordnung optimaler Druckbe- und -entlastungsbereiche vereinfacht. Diese Methode ist ziemlich anfällig für systematische Fehler, die dem Anwender unterlaufen. Um diesen Subjektivitätsfaktor zu eliminieren, wird im vorliegenden Beitra einer texanishen Forschungsgruppe eine Idee und deren Umsetzung vorgestellt, mit Hilfe dere die Software automatisch die Markierungspunkte auf der digitalisierten Stumpfoberfläche erkennt. Probates Mittel für eine derartige Aufgabe ist die Anwendug eines Musterkennugsalgorithmus, in diesem Fall auf der Basis "Neuronaler Netzwerke", der die Friformfläche sozusagen "abscannt" und die gesuchten Puntke detektiert. Diser Automatismus reduziert Fehler und hilft, durch Übernahme eines Arbeitsganges die Bearbeitungsdauer zu verkürzen. Exemplarisch wurden die "Neuronalen Netzwerke" auf das "Aufspüren" des Mittelpunktes des Patellarsehnenbandes, des Fibulaköpfchens und des distalen Tibiaendes "abgerichtet". Unter "Neuronalen Netzwerken" versteht man die Computersimulation biologischer Denkschemen mit dem Ziel, Muster, wie beispielweise wiederkehrende Formen in einem ausgedehnten 3D-Relief, zu erkennen und zu klassifizieren. Die "Neuronalen Netzwerke" müssen für ihre Aufgabe zuvor trainiert werden. Im vorliegenden Fall geschah dies anhand von manuell lokalisierten Fixpunten auf einer Reihe von zu digitalisierenden Stüpfen. Sobald der Algorithmus gelernt hat, die gewünschten Orte zu finden, kann er in ein CAD/CAM Programm integriert und auf beliebige Unterschenkelstüpfe angewandt werden. In praxi wird jedes "Neuronale Netzwerk" nur auf jeweils einen zu erkennenden Punkt hin trainiert.

Ergebnisse aus experimentellen Studien zeigen, dass die Methode mindestens so genau ist, wie ein erfahrener Orthopädietechniker. Die Musterkennungsalgorithmmen wurder in die Software integriert, die den San Antonio Laser Imager steuert. Innerhalb von Sekunden nachdem das Bein des Patienten gescannt worder ist, werden die vorhergesagten Fixpunkte auf der digitalisierten Oberfläche markiert und können so unmittelbar als Anker für die Modellierungsregionen dienen.

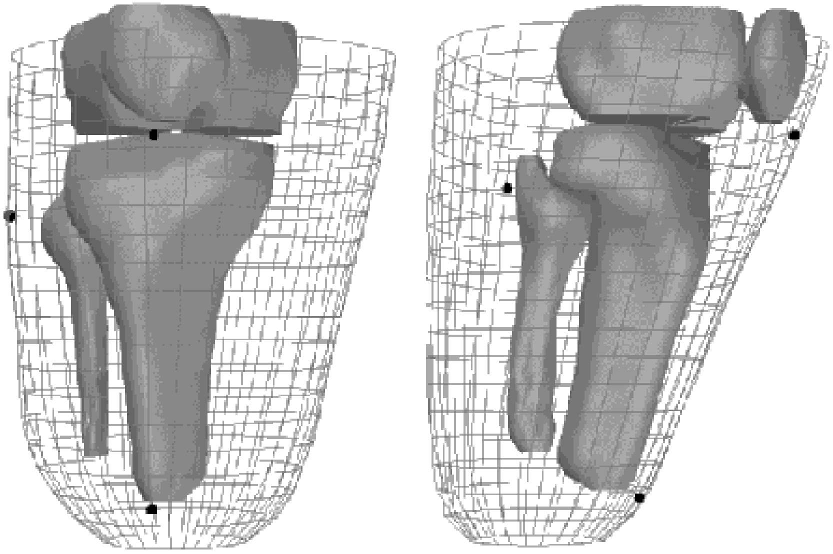
Abstract

Computer aided design of a prosthesis for a below-the-knee (trans-tibial) amputee begins with a digitized representation of the shape of the residual limb. Certain anatomical landmarks must be located *on this shape to identify optimal areas for load and pressure relief. Normally, the technician or prosthetist operating the digitizer places the marks on the shape before or after digitization. This method is somewhat prone to human error and bias. The approach presented here uses a pattern recognition algorithm in the digitizer's control software to locate automatically these landmarks at the time of digitization. Automating this task reduces error and allows the techician to work more efficiently. Neural networks are used to find the midpoint of the patellar tendon, the head of the fibula, and the distal end of the tibia. Neural networks are computational devices, simulated in a computer program, that are capable of recognizing and classifying patterns such as features in a digitized shape. The neural networks are trained to place markers on a set of shapes for which the landmarks have been located manually. Once they have learned to recognize the landmarks in the training set, the neural networks can then be used to find them for arbitrary shapes. Each neural network is trained on a different anatomical landmark. Experimental results show that the method is at least as accurate as a trained prosthetist. The method has been incorporated into software controlling a laser-scanning device (The San Antonio Laser Imager) that digitizes the surface of the skin. Within seconds after the patient's residual limb is scanned, the predicted landmarks are placed onto the digitized shape, which can then be modified in a CAD program for socket design.*

1. Introduction

The computer aided design (CAD) and manufacture (CAM) of a BK prosthetic limb socket begins with the digitization of the surface of the amputee's residual limb. This can be done using a mechanical digitizer on the inside of a plaster cast of the residual limb or by using a laser imager that scans the residual limb with a laser as it rotates around the patient's leg. A prosthetic socket CAD program modifies the resulting shape to that of a biomechanically correct socket. The original shape is modified so that the resulting socket will apply pressure in pressure tolerant areas and relieve pressure sensitive areas. A computer controlled milling machine then carves the modified shape in plaster to make a pattern for socket fabrication. Alternately, a rapid-prototype system such as stereo lithography can be used to directly fabricate the socket.

During the modification process, certain anatomical areas receive particular attention. Three of these areas are the midpoint of the patellar tendon (MPTN), the distal end of the crest of the tibia (DECT) and the head of the fibula (HDFB). These landmarks are represented on the digitized shape by the point on the skin closest to the corresponding anatomical feature. Knowing the locations of these areas is essential in modifying the shape; the MPTN bears most of the load in a patellar tendon bearing (PTB) socket, and the DECT and HDFB are sensitive areas the prosthetist must protect by providing relief. Figure 1 shows a lateral and anterior views of a digitized shape of a BK amputee along with the bones; in this case, the bone shapes are from data in a CT scan of the patient. This technique is not usually used for routine residual limb imaging because of the cost, unnecessary exposure to radiation, and other factors. In other types of imaging, the positions of the bones must be deduced just from the surface of the skin.



Landmarks are shown with black markers. From top to bottom, MPTN, HDFB, and DECT.
Figure 1: Bone Structure of a BK Amputee, with Landmarks

Locating the landmarks is usually done by placement of nonreflective markers (recognized during digitization) from palpation of the skin or by visual inspection in the CAD program. There are several factors affecting the accuracy of these techniques:

- Human error: sometimes the technician puts the marker for the landmark in the wrong place. Sometimes, a patient will move his or her leg during the digitization process, causing errors to be introduced.
- Different technicians may have slight differences of opinion about precisely where a landmark should be placed. This may not be a problem as long as the technician is consistent, but if a prosthetist has to deal with more than one technician a standard method is needed.
- Landmarks on the residual limbs of larger people are difficult to locate due to excess adipose tissue obscuring bony prominences such as the head of the fibula.
- There are several different surgical procedures for a BK amputation, each leaving a different appearance of the surface of the skin[4,5]. Also, traumatic amputations often leave little room for decision in this area; the location of the DECT can vary widely from shape to shape.

A method for locating these landmarks with a higher degree of accuracy and consistency would be of benefit to prosthetics design. Previous research has used surface curvature anaylsis[1] to locate the landmarks. Using this technique, landmarks are located by measuring areas of greatest curvature on the surface of the skin which, presumably, correspond to bony prominences. The authors reported achieved a root mean squared error of 8.4mm, 7.7mm, and 8.5mm between their algorithm and a human technician locating the MPTN, HDFB, and DECT, respectively. However, the research was limited to eight shapes and it is unclear whether the technician or algorithm was more accurate.

Our approach to locating these landmarks is to train neural networks[6] to locate each landmark on about one hundred shapes. The networks are tested by locating landmarks on other shapes. We achieved good accuracy on shapes held out for testing, and have confirmed the locations of the landmarks by having a two prosthetists locate the landmarks for each shape. We have also incorporated this method into software that controls The San Antonio Imager, a non-contact laser imager designed to image the residual limbs of amputees. This way, the technician taking a scan of a patient's leg can simply click a button and have the markers automatically appear on the shape on the screen. The nonreflective markers can still be used, but are not needed except to occasionally confirm the accuracy of the neural networks' predictions.

2. The San Antonio Laser Imager

2.1. Imaging Technology

The San Antonio Laser Imager uses a structured lighting system where a plane of laser light illuminates the residual limb of the amputee. The patient sits in a chair and places the limb into a large cannister where the imaging takes place. Often the limb is clothed in a compression sock to firm the flesh so that the image will not reflect distortion due to gravity. Where the plane of light interects the limb, a line is produced and obliquely viewed by two video cameras. A custom video board extracts the location of the line from the video images in real time. The imager rotates around the residual limb viewing the limb from 64 angular locations, taking approximately five seconds to do a complete scan. Small black tape markers are placed on the residual limb at the regions of anatomical interest such as MPTN, DECT, and HDFB; the imager software finds these markers by searching for "holes" in the data. A name for each marker must be entered since the control software can't determine which one is which.

2.2. Control Hardware

The system is controlled through a custom video board installed in an IBM-PC compatible Pentium system running Windows NT or Windows 95. The video board controls the two cameras in the imager, and is accessed through programmed input/output via a driver written for the video board.

2.3. Control Software

The user interface to the imager is a point-and-click type Windows application program. When a scan is taken, the resulting data is assembled into a three dimensional shape. The user clicks different buttons on the program to run the imager and save the resulting image to a standard AAOP1 format shape file that can be read into CAD programs such as Seattle ShapeMaker and Sockets. The locations of the black tape markers show up as holes in the data that are recognized as landmark locations; these markers are placed by a technician who has palpated the leg. The markers are approximately one square centimeter, so a small amount of the surface is obscured when the scan is taken. This area replaced in the resulting shape by interpolation.

2.4. File Format

The AAOP1 file format is used for the output of the imager. The shape is written to a file as a sequence of approximately 50 to 100 data slices. Each *slice* of data is composed of 64 points of equal angular distance from one another (i.e., $\Delta\theta = 2\pi / 64$). The same angular values are used for each slice, so from top to bottom corresponding points form a *profile*. The distances between slices are not uniform, so a list of z values, one for each slice, is written to the data file. The landmark locations are also included in the file with a short descriptive string typed by the technician.

3. Neural Networks

Artificial neural networks (shorted to simply "neural networks" in this paper) are an idealized version of what happens in a biological neural network. They are used for pattern recognition, classification, and other statistical tasks. They are often well-suited to situations where data is incomplete or incorrect, or there is not a good conventional way to approach the problem. A technical discussion of neural networks is beyond the scope of this paper, but here we present some information about them.

A neural network has several *input units* and *output units*, represented by real numbers (often in the range 0..1 or -1..1). It may also contain *hidden units*, connected to the input units, output units, or each other. Each connection between units has a real-valued *weight* representing the weakness or strength of the connection. The network is "stimulated" by placing values in the input units. These values are propogated through the network's connections, multiplied by the weights. In most neural network architectures, an *activation function* is applied to a value before it is applied to a connection; this simulates behavior observed in real neurons.

A neural network can be used for prediction by presenting it with an input vector (via the input units), propogating the values through the network, and examining the output units. If the network is *trained* properly, its output units will estimate the desired function of the input units.

One type of training, called *supervised learning*, involved presenting sample input and output pairs to a neural network and getting the network to produce the correct output for a given input. The network will, hopefully, generalize and produce the correct output for inputs it has not been presented with before.

Neural networks have been used for many biomedical applications. Some examples are:

- Classifying mammograms as to whether they contain malignancies[7].
- Coronary artery disease diagnosis[8].
- Pap smear classification[9]. This application is now a common commercial pap-smear test, well beyond the research stage.

Neural networks have also been suggested for automated alignment of BK prostheses[10], and used in for controlling prosthetic limbs[11].

There are many different methods for training neural networks. The two used for this project are backpropogation and ADALINE. Both minimize the output error by adjusting the weights. Backpropogation uses hidden units and a sigmoidal (S-shaped) activation function; ADALINE uses no hidden units and identity activation function.

4. Methods

4.1. Shape representation

For training and testing the neural networks, 120 different residual limb shapes collected from the San Antonio Laser Imager over a period of several years were used. Each shape included the locations of the anatomical landmarks placed there at the time of digitization.

4.2. Approach to the problem

We approached the problem of finding a landmark as finding the pair (θ, z) of angle and z-value for the landmark. Finding the radius is not needed since there can only be one radius for each (θ, z) pair (note: this is not true for a general shape in cylindrical coordinates, but is true for the roughly cylindrical geometry of the BK residual limb); the mapping to the surface of the skin is straightforward.

4.3. Algorithm

The algorithm to find a landmark given the shape information is:

- Preprocess the shape into a grid of normalized radius values.
- Rotate the grid for many values of θ, each time presenting the grid to a set of trained one-output neural networks. Choose the predicted θ-value, θ', as that with the best response from the ensemble.
- For many values of z, present a portion of the grid close to (θ', z) to another set of neural networks. Choose the predicted z-value, z', as that with the best response from the networks.

4.4. Preprocessing

The preprocessing algorithm places radius values for each slice/profile pair into corresponding positions in a two-dimensional array of reals. The array is resized from (64 x number of slices) to (96 x 96) and interpolated radius values chosen so that the spacing between slices is uniform. The values in the array are then smoothed by averaging with nearest neighbors to reduce any noise from the digitization process. The values are then normalized so that the mean is 0, then scaled so that all the values are between -1 and 1. The neural networks for predicting θ (θ-networks) use a (16 x 16) grid whose values are chosen from equidistant points in the larger array. The neural networks for predicting z (z-networks) use a (16 x 64) grid of values from the neighborhood of (θ', z) (for some z) for testing, or (θ, z) for training. Larger and smaller grid sizes were tested, but the best results in terms of accuracy and network training times were achived with the (16 x 16) grid size for θ-networks and (16 x 64) for z-networks. Figure 2 shows greyscale images of the normalized (96 x 96) grid, a (16 x 16) grid suitable for presentation to the θ-networks, and a (16 x 64) grid suitable for presentation to the z-networks.

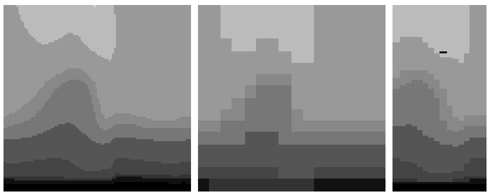


Figure 2: Greyscale normalized (96 x 96) array, θ-network input grid, z-network input grid

4.5. Training

Training was done using neural network software developed in-house. One hundred different residual limb shapes were used for training the neural network. The shapes were divided into two sets of 50 by choosing a random permutation of files and dividing it in two. Two more sets of 50 were chosen using another random permutation. The first two were used to train and cross-validate two θ-networks and two

z-networks. The second two were similarly used to produce four more θ- and z-networks.

Experimentation showed that ADALINE [6] neurons were a good choice for the θ-networks. Backpropagation nets with eight hidden units proved to work well for the z-networks.

As the preprocessed grid was rotated clockwise (as seen from above the shape), the outputs of the θ-networks were trained to change from 1 to 0 as the correct value for θ crossed the middle profile of the grid. For the z-networks, the outputs were trained to change from 1 to 0 as the correct z-value became greater than the z-value for the current (16 x 64) input grid. We validated each network by using it in an ensemble of size one and applying it using the algorithm described above. For the θ- networks, we used the absolute value of the angular distance between θ and θ' as a measure of error; training was stopped when the error on the validation set reached a minimum. We trained the z-networks similarly, using the absolute value of the distance between z and z' as a measure of error. The ADALINEs usually reached a mean squared error (MSE) of 0.05 by the time training was stopped. The backpropogation networks usually reached an MSE of from 0.001 to 0.003 before training was stopped. The θ-nets typically reached an error of within one profile, or 0.1 radians for the HDFB and DECT. The nets for the MPTN had somewhat less error. The z-values reached an error of within 1.5 slices for each marker.

4.6. Ensemble technique

The sets of neural networks use for predicting marker locations are an example of the ensemble neural networks technique. Ensemble networks combine the outputs of several neural networks[2]. The output of the ensemble is a weighted average of the outputs of each network, with the ensemble weights determined as a function of the relative error of each network determined in training[2]; the resulting network often outperforms the constituent networks. There is a growing body of research into ensemble methods, for example, improvements in performance can result from training the individual networks to be decorrelated with each other[3]. We used a variation on ensemble techniques in neural networks literature to improve the accuracy of the predictions made by the networks.

5. Results

We had a trained prosthetist examine the twenty residual limb shapes we held out for testing. We hid the markers that had been placed by palpation at the time of the scan so that he was blinded to all previous markers. We asked him to place the markers for DECT, HDFB, and MPTN where he thought they should go. He had access to our Sockets CAD program with a false-colour mode showing the surface curvature at each point to help find the bony prominences. We then ran our algorithm on the same shapes and compared the algorithm's error with the prosthetist's error. Table 1 shows the results of this experiment as median and mean of the errors in Euclidean distance between the palpated and predicted landmarks for both the algorithm and the human.

Landmark	Algorithm Median, Mean, and Standard Deviation of Error	Human Median, Mean, and Standard Deviation of Error
MPTN	8.5, 7.7, 6.1	5.9, 9.4, 7.3
HDFB	7.4, 9.1, 6.2	8.6, 9.4, 5.2
DECT	6.3, 8.9, 7.3	9.0, 9.2, 3.8

Table 1: Median, Mean, and Standard Deviation of Errors for Algorithm, Human, in millimeters

Figure 3 shows the anterior of a typical shape after the prosthetist and the algorithm have made their predictions. The palpated landmark is represented with an X, the prosthetists prediction with a square, and the algorithms prediction with a diamond.

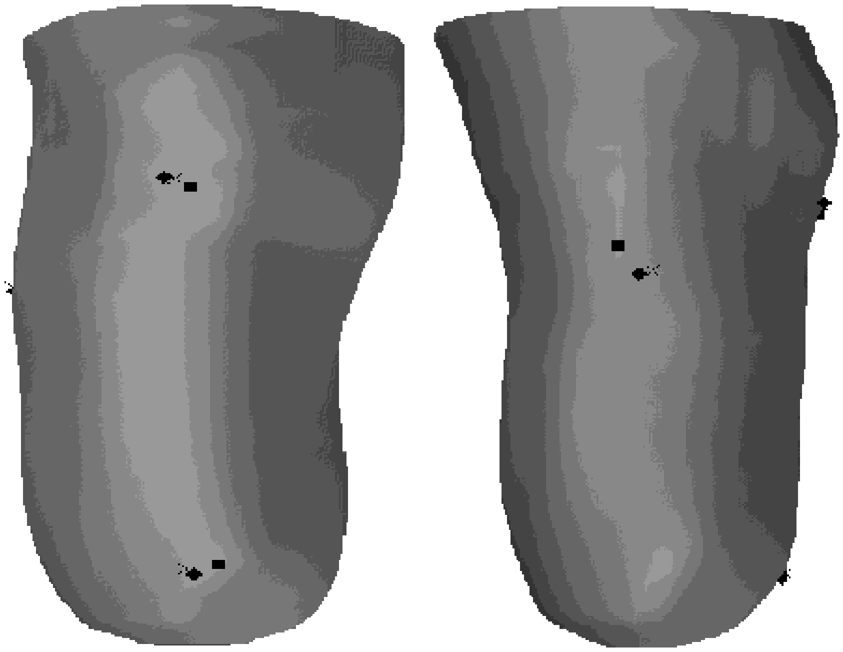


Figure 3: Typical Shape with Predictions, Anterior and Lateral Views
Diamond - computer predicted, Box - human predicted, Cross - palpated

6. Using the method with the imager

We have incorporated the neural network method into the controlling software for the San Antonio Laser Imager and begun using the new software to image patients. A few different ways of getting the programs to work together were tried. We decided to have the neural network prediction software work as a separate executable program that can be run from the imager software as a sort of "plug-in." When the "neural-locator" button on the imager software is clicked (after a patient has been scanned), the shape data is written to a temporary file and then the neural network locator program is called. The program examines the data in this file, then prints a list of predicted markers with their locations in cylindrical coordinates. This list is read in by the imager software and the markers placed into the shape. The whole process, from the time the amputee places his limb into the imager to the time the final shape file with the predicted markers is written to disk, takes about 20 to 30 seconds.

7. Conclusions and future directions

From the results, we believe that our method is at least as accurate as a trained professional sitting at a CAD program finding the markers. It remains to be seen whether our technique is more accurate; ideally, we would like to use shapes where the bone positions are determined by a CT scan, rather than palpated or guessed. To our knowledge, a large enough corpus of such shapes doesn’t exist yet. Much of the development of the neural network method took place in our CAD program, Sockets. We are putting the newly enhanced imager into practice by using it to image patients. Preliminary results show that the neural network technique works well in practice.

8. Acknowledgments

Over a period of years, Virgil Faulkner has been gathering residual limb shape data in his lab; we obtained many of the shapes for this project from him. Virgil Faulkner was also the prosthetist who placed the markers on the testing shapes. Thanks to him for his contributions. Thanks also to the Otto Bock Foundation for sponsoring part of the expense of bringing this paper to the International Symposium on CAD/CAM-Systems in Pedorthics, Prosthetics and Orthotics.

9. Bibliography

[1] Chan, R.B., Rovick, J.S., and Childress, D.S., ‘‘Surface Curvature Analysis for Enhanced Computer-Aided-Design of Prosthetic Sockets,’’ Proceedings of the 15th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 1993.
[2] Krogh, A. and Vedelsby, J., ‘‘Neural Network Ensembles, Cross Validation, and Active Learning’’, *Neural Information Processing Systems*, volume 7. MIT Press, 1995.
[3] Rosen, B.E., ‘‘Ensemble Learning using Decorrelated Neural Networks’’ *Connection Science*, 8, 3-4, pp. 373-384, 1996.
[4] Porter, J.M., ‘‘Chronic Lower-Extremity Ischemia Part II,’’ *Current Problems in Surgery*, volume 28, number 2, Mosby-Year Book, 1991.
[5] Schwartz. S.I., Shires, G.T., and Spencer, F.C., *Principles of Surgery, Fifth Edition*, pp 2027-2029., McGraw-Hill, 1989.
[6] Fausett, L. *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*, pp. 80-86, Prentice-Hall, 1994.
[7] P. Downes. 1994. "Neural Network Recognition of Multiple Mammographic Lesions." *World Congress on Neural Networks*, San Diego, CA, USA, Vol. I, pp. 133-137 (5-9 June).
[8] Hiroshi Fujita, Tetsuro Katafuchi, Toshiisa Uehara, Tsunehiko Nishimura. 1992. "Neural Network Approach for the Computer-Aided Diagnosis of Coronary Artery Diseases in Nuclear Medicine." *Proceeding of the 1992 International Joint Conference on Neural Networks (IJCNN’92)*, Baltimore, MD, USA, Vol. III, pp. 215-220 (7-11 June 1992).
[9] M. Kharazi, S. Keyhani-Rofagha, R. O’Toole. 1993. "Evaluation of the PAPNET automated cytological findings with biopsy." *American Journal of Clinical Pathology*, Vol. 99, pp. 337-338.
[10] A.I. Chahande, V.W. Faulkner, S.R. Billakanti, N.E. Walsh. 1994. "Neural Network Models for Customized Alignment of Endoskeleton BK Prosthesis." *International Conference on Neural Networks (ICNN’94)*, IEEE, Orlando, FL, USA (26 June - 2 July 1994).
[11] R.K. Elsley. 1990. "Adaptive Control of Prosthetic Limbs Using Neural Networks." *International Conference on Neural Networks (ICNN’90)*, San Diego CA, USA, Vol. II, pp. 771-776.