Locating Anatomical Landmarks for Prosthetics Design using Ensemble Neural Networks

Daniel Jiménez, Thomas Darm, Bill Rogers, Nicolas Walsh

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

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. A method of locating the midpoint of the patellar tendon, the distal end of the tibia and the head of the fibula is presented. The method involves training ensembles of neural networks on shapes for which the markers have been located manually; the neural networks are then used to find the landmarks for arbitrary shapes. Experimental results show that the method is at least as accurate as a trained prosthetist.

1. Introduction

The first step in the computer aided design of a below-the-knee (BK) prosthetic limb socket is 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 provide pressure in pressure tolerant areas and relief in pressure sensitive areas. A computer controlled milling machine then mills out the modified shape to make a pattern for socket fabrication.

During the modification process, three areas on the surface of the skin receive particular attention: 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 shape by the point on the skin closest to the corresponding anatomical feature. A bar is shaped across the MPTN is to receive the greatest part of the load. The HDFB and DECT are augmented to provide relief, as these areas are sensitive to pressure. 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 made from data in a CT scan of the patient. Unfortunately, most patients cannot be imaged this way because of the cost, unnecessary exposure to radiation, and other factors; the positions of the bones must be deduced just from the surface of the skin.

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.

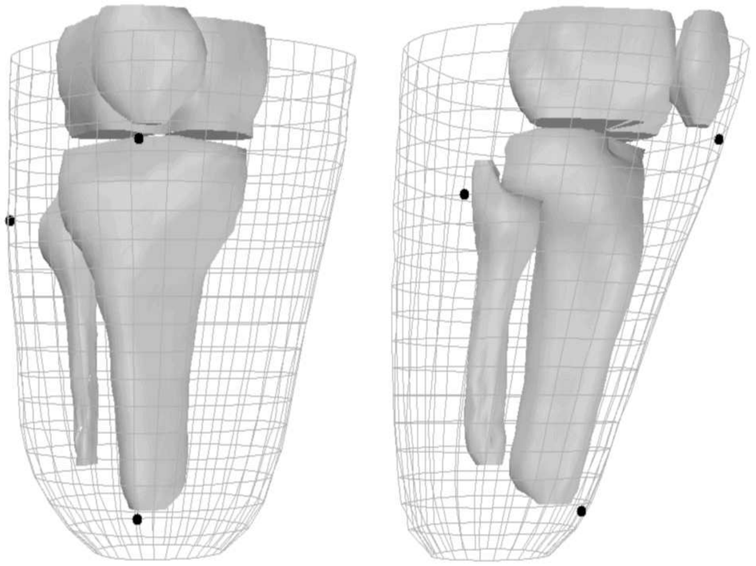
- Different technicians may have slight differences of opinion about precisely where a landmark should be placed.

- 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.

Figure 1. Bone Structure of a BK Amputee, with Landmarks

Landmarks are shown with black markers. From top to bottom, MPTN, HDFB, and DECT.



A method for locating these landmarks with a higher degree of accuracy and consistency would be a 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 ensembles of neural networks for each landmark using about one hundred shapes. We achieved a high degree of accuracy on shapes held out for testing, and have confirmed the locations of the landmarks by having a prosthetist locate the landmarks for each shape.

2. Methods

2.1. Shape representation

For this project, 120 different residual limb shapes were obtained from a laser imager. The laser imager uses a structured lighting system where a plane of laser light illuminates the residual limb of the amputee. Where the plane of light interects the limb, a line is produced. Two video cameras obliquely view this line. 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. The scan time is approximately five seconds. Small black tape markers are placed on the residual limb at the regions of anatomical interest such as MPTN, DECT, and HDFB. The locations of these markers show up as holes in the data that are recognized as landmark locations. The resulting data is assembled into a three dimensional shape. This shape is written to a file as a sequence of approximately 40 to 50 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.

2.2. 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 [1]; the mapping to the surface of the skin is straightforward.

[1] This is not true for a general shape in cylindrical coordinates, but is true for the roughly cylindrical geometry of the BK residual limb

2.3. Algorithm

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

1. Preprocess the shape into a grid of normalized radius values.
2. Rotate the grid for many values of θ, each time presenting the grid to an ensemble of trained one-output neural networks. Choose the predicted θ-value, $\hat{\theta}$, as that with the best response from the ensemble.
3. For many values of z, present a portion of the grid close to $(\hat{\theta}, z)$ to another ensemble. Choose the predicted z-value, $\hat{z}$, as that with the best response from the ensemble.

2.4. 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 $(64 \times$ number of slices$)$ to $(96, 96)$ and interpolated radius values chosen so that the spacing between slices is uniform. The values in the array are then smoothed (averaged 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 16×16 grid whose values are chosen from equidistant points in the larger array. The neural networks for predicting z (z-networks) use a 16×64 grid of values from the neighborhood of $(\hat{\theta}, z)$ (for some z) for testing, or (θ, z) for training. Figure 2 shows greyscale images of the normalized 96×96 grid, a 16×16 grid suitable for presentation to the θ-networks, and a 16×64 grid suitable for presentation to the z-networks.

2.5. Training

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 16×64 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 $\hat{\theta}$ 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 $\hat{z}$ as a measure of error. The ADALINEs usually reached a mean squared error (MSE) of 0.05 by the time training was stopped. The backprop 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.

2.6. Ensemble technique

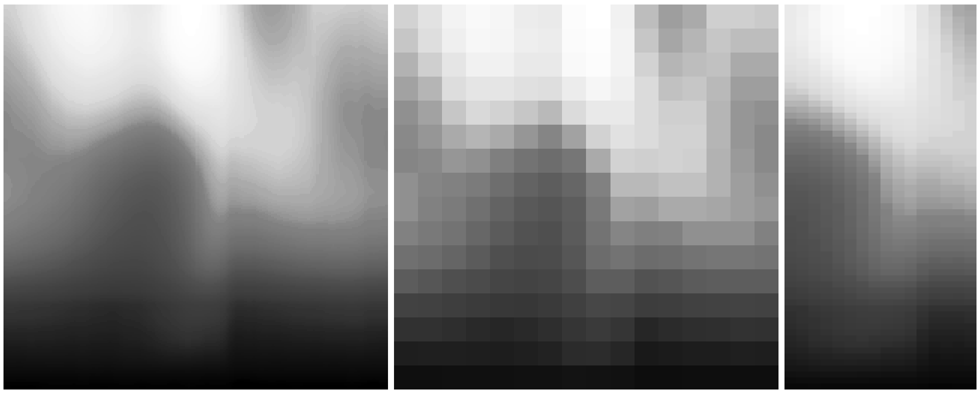
Ensemble networks combine the outputs of several neural networks[2]. The output of the ensemble is a weighted average of the outputs of each network, with the ensemble weights determined as a function of the relative error of each network determined in training[2]; the resulting network often outperforms 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]. Our technique differs in that the ensemble weights are determined dynamically, i.e., upon each propagation through the network, as opposed to statically, as part of the training algorithm. This allows the ADALINE networks to benefit from ensemble techniques and, we believe, also improves the accuracy of backpropagation ensembles. Both our ADALINEs and backprop net give an output in the range [0..1]. A threshold function can be applied to, e.g., outputs of 0.6 and 0.9, but the output of 0.6 has a higher probability of being wrong. Our ensemble technique attempts to capture this idea. The *uncertainty* $u(y)$ of the output y of a network is defined as:

$$u(y) = \begin{cases} y & \text{if } y < 1/2 \\ 1 - y & \text{otherwise} \end{cases}$$

Clearly, the uncertainty rises for outputs $y \geq 0.5$ as y falls, and for outputs $y < 0.5$ as y rises. If the outputs of the individual networks (without thresholding) are $y_1, y_2, \ldots, y_n$, then the ensemble weight w_i for output unit y_i given by:

$$w_i = \frac{\exp(-\gamma u(y_i)^2)}{\sum_{j=1}^{n} \exp(-\gamma u(y_j)^2)}$$

Figure 2. Greyscale normalized 96×96 array, θ-network input grid, z-network input grid



where γ is a gain term determined empirically. The weights are multiplied by their respective outputs to find the output of the ensemble. A rigorous analysis of this technique is beyond the scope of this paper, but the point is that it allows ADALINE outputs, which are simply dot products of weights and inputs, to escape the linearity inherent in ensemble combinations (otherwise, any weighted average of ADALINE outputs is simply the output of a third "virtual" ADALINE with no more computational power than its component neurons). The result seems to work better than static weights for backprop networks, too, but we have only tried it on the problem at hand and, with ADALINE neurons, the two-input XOR problem, which it can solve (unlike the standard ADALINE).

3. 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 and asked him to place the markers for DECT, HDFB, and MPTN where he thought they should go. He had access to a sophisticated CAD program with a false-color 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. 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.

4. 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. We also believe our ensemble technique, which began as an *ad hoc* method of improving performance, deserves more in depth study.

5. 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.

6. 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," to appear in *Connections Science*, 1996.

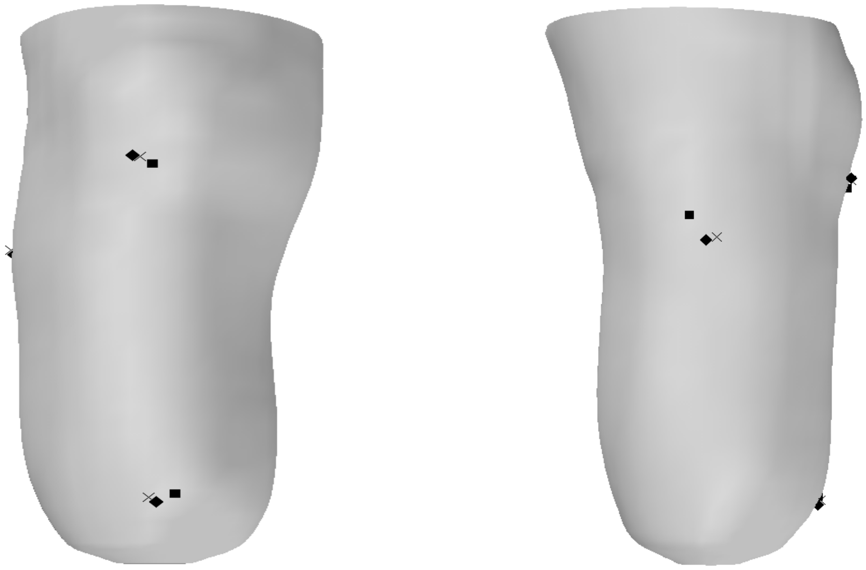
[4] Porter, J.M., "Chronic Lower-Extremity Ischemia Part

Table 1. Median, Mean, and Standard Dev. of Errors for Algorithm, Human, in millimeters

Landmark	Algorithm Median, μ, and σ of Error	Human Median, μ, and σ 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

Figure 3. Typical Shape with Predictions, Anterior and Lateral Views

$\diamond$ - computer predicted, $\square$ - human predicted, $\times$ - palpated



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.