Abstract—This paper documents the design and testing of a software system which automates the task of extracting the two-dimensional cartesian coordinates of human neutrophils from subsequent video frames captured by a Scion Frame Grabber. A single output Radial Basis Function (RBF) neural network is utilized for the recognition of cells as well as determining the cell’s exact position. The semi-automated training set construction and learning techniques are presented. The resulting algorithm is an accurate and computationally efficient blood cell locator suitable for real-time systems.

I. INTRODUCTION

Previous research involving Artificial Neural Networks (ANN) for the processing of blood cell images have concentrated primarily on algorithms which classify the cells in to one of the three mammalian classes: red, white, or platelet. Some of these classification efforts have been made based on size and RGB color components [1], contrasted HSL hue [2], wavelet transform features [3], nuclei shape [4], or a combination of shape and color features [5]. One of the most common applications for blood cell classification systems is to count white blood cells in order to detect the pathological effects of diseases such as Leukemia, which increases a patient’s white blood cell count. Shape features also help ANNs to distinguish between healthy or cancerous cells. The system proposed here utilizes an ANN classifier not for blood cell classification into one of the three major classes or any of the numerous subclasses, but for the classification of an image as a blood cell which is spatially centered at a known xy coordinate. Location is required to observe the interaction of flowing blood cells with each other or their surroundings. The RBF neural network is presented in section II. Section III describes the semi-automated training set construction and the learning process. Finally, the results are presented in section IV.

II. THE ARTIFICIAL NEURAL NETWORK

A. Activation

In artificial neural models used for computation each neuron computes a scalar value which represents its level of activation on a given input. An activation of zero means that a neuron will not increase the outputs of the neural network because it will not affect the value of the linear summation performed in the output layer. The scalar activation is a function of the input and synaptic weight vectors. The ANN in this system utilizes a Gaussian basis activation function. Where \( \text{width} \) is the width of the Gaussian curve and synaptic weight vector \( \mathbf{w} \) is the center for calculating the Euclidean distance to the input vector \( \mathbf{x} \), equation 1 determines the radial basis activation \( \Phi \).

\[
\Phi(\mathbf{x}, \mathbf{w}) = \exp\left((-1) \sum_{i=1}^{n} \frac{(x_i - w_i)^2}{\text{width}^2}\right)
\]

Negating the input to the exponential function produces the desired activation behavior: as the Euclidean distance approaches infinity, the activation approaches zero.

B. Neural Network Topology and Usage

The ANN is a feedforward backpropagation neural network with 121 (one for each of the \( 11 \times 11 \) grayscale pixels) inputs, a single (selected through trial and error) hidden neuron in the single hidden layer, and a single bounded linear output. An image based ANN which does not enjoy the reduced dimensionality of extracted features should, in general, not be executed on every pixel of an image in a brute force pass. The first step of optimization was to reduce each area of interest within the image to a \( 75\% \) lower resolution. This was accomplished by averaging adjacent pixels of each \( 22 \times 22 \) pixel area down to the 121 element input vector. The majority of CPU time is saved by threshold background subtraction and the application of a low
cost filter which prevents ANN operations from being performed on images which are obviously not blood cells.

Fig. 1. Note that the “phase-bright” section that arises due to the phase contrast optical imaging technique used to accentuate the contrast at the cell edges has also been subtracted and ignored by the filter.

Only when the binary image satisfies the filter criteria is the grayscale image propagated forward through the ANN. The average pixel position of the binary image must be sufficiently close to the center of the image (a center of mass may be used for grayscale or color images). There must also be a sufficient number of white pixels in the area being examined.

III. LEARNING

A. Semi-Automated Training Set Construction

The cell locator software system was constructed in the C++ programming language utilizing OpenGL and Microsoft Windows for the GUI. In the mode for training set construction the user can step through several frames of the video marking each blood cell in the frame by selecting it with the mouse. Each cell marked will add a record to the database storing frame number, x coordinate and y coordinate. Dozens of frames may need to be viewed in order to mark a sufficiently large variety of cell images. Cells which are distorted, such as quickly moving blurred cells, should also be marked for training.

The desired ANN output required for calculating the mean squared error (MSE) and adjusting the synaptic weights is nearly identical to the hidden layer activation function. Figure 3 displays the desired output at each of the pixel coordinates surrounding a marked blood cell.

B. Training

Once a few dozen cells have been marked the application can be run in the training mode. It will execute until the ANN has been trained on each of the marked cells stored in the database, then it will save the synaptic weights to the hard drive and exit. The RBF neural network converges quickly after seeing only a few frames of video. Figure 4 is a graph of the MSE after a single pass through the training set. While the total MSE seen after ten passes through the training frames (Figure 5) is lower than that seen in the first pass, the ANN does not successfully locate all of the cells after ten passes through the training set. The ANN cannot generalize as well and becomes overfitted to the training set after seeing the training data more than once. Therefore it is preferred to prepare a larger training set over presenting a smaller training set to the ANN multiple times.

Fig. 2. Three blood cells marked with the mouse for training the ANN

Fig. 4. The MSE over the first pass through the training set.

Fig. 5. A lower total MSE is seen during a tenth pass through the training set.

For these experiments, the training consisted of a single pass through four cell-marked video frames.
IV. RESULTS

The trained system is capable of recognizing or counting blood cells with 100% accuracy. However, the goal of this project was not to count blood cells, but to accurately locate their xy coordinates. It is clear in Figure 6 that the system is consistently a few pixels away from the center of each cell. Further optimization will be required to locate the exact centers of the flowing cells. Promising candidate areas of improvement are in a modified ANN topology or calculating the desired ANN output with a function other than the exponential function.

This novel neutrophil locator may function as an important component in future software applications for the analysis of biofluid mechanics. It could also be used to locate blood cells for classification.

REFERENCES
