IDENTIFICATION OF HAND SIGNALS USING ARTIFICIAL NEURAL NETWORKS

Dr P.J.Arul Leena Rose
*Department of Computer Application, FSH,SRMIST, Chennai-603203

ABSTRACT

In this paper a simple and fast algorithm will be developed using orientation histograms to work on a workstation. It will identify static hand signals, namely, American Sign Language (ASL). Previously data gloves or markers have used for input in the system. A pattern recognition system will be using a transform that converts an image into a feature vector, which will then be compared with the feature vectors of a training set of signals. Hand signal recognition procedures with sign language have been used. In sign language, each signal has an assigned meaning. Computer recognition of hand signals may provide a more natural-computer interface, allowing people to point, or rotate a CAD model by rotating their hands. Hand signals can be classified in two categories: static and dynamic. A static signal is a particular hand configuration and poses, represented by a single image. A dynamic signal is a moving signal, represented by a sequence of images. The final system will be implemented with a Perceptron network.

Key words: *ASL, Neural Networks, MATLAB, Perceptron.*

1.INTRODUCTION

American Sign Language is the language of choice for most deaf people in the United States. It is part of the "deaf culture" and includes its own system of puns, inside jokes, etc. However, ASL is one of the many sign languages of the world. As a Tamil speaker would have trouble understanding someone speaking Punjabi, a speaker of ASL would have trouble understanding the Sign Language of Sweden. ASL also has its own grammar that is different from English. ASL consists of approximately 6000 signals of common words with finger spelling used to communicate obscure words or proper nouns[1]. Finger spelling uses one hand and 26 signals to communicate the 26 letters of the alphabet. ASL uses facial expressions to distinguish between statements, questions and directives. The eyebrows are raised for a question, held normal for a statement, and furrowed for a directive.

2. OBJECT RECOGNITION

2.1 Large Object Tracking

Image moments[3], which are fast to compute, provide a very coarse summary of global averages of orientation and position. If the hand is on a uniform background, this method can distinguish hand positions and simple pointing signals.

2.2 Shape recognition

Most applications, such as recognizing particular static hand signal, require a richer description of the shape of the input object than image moments provide. If the hand signals fell in a predetermined set, and the camera views a close-up of the hand, we may use an example-based approach, combined with a simple method top analyze hand signals called orientation histograms .

3. IMAGE DATABASE

The image database can have different formats. Images can be either hand drawn, digitized photographs or a 3D dimensional hand. Photographs were used, as they are the most realistic approach. Images came from two main sources. Various ASL databases on the Internet and photographs took with a digital camera. This meant that they have different sizes, different resolutions and some times almost completely different angles of shooting[2]. Images belonging to the last case were very few but they were discarded, as there was no chance of classifying them correctly. The database itself was constantly changing throughout the completion, as it was it that would decide the robustness of the algorithm.

4. OPERATION

The program can be 'divided' in 6 steps. Lets examine them one by one.

Step1: The first thing for the program to do is to read the image database. A for loop is used to read an entire folder of images and store them in MATLAB's memory. The user from menus selects the folder. A menu will pop-up asking whether the user want to run the algorithm on test or train sets. Then a second menu will pop-up for the user to choose which ASL sign he wants to use.

Step2: Resize all the images that were read in Step1 to 150x140 pixels. This size seems to be the optimal one for offering enough detail while keeping the processing time low.

Step3: Next thing to do is to find the edges. As mentioned before 2 filters were used. For the x direction x = [0 -1 1] For the y direction y== $\begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}$,, which is the same as x but transposed and multiplied by –1

Step 4: Dividing the two resulting matrices (images) dx and dy element by element and then taking the atan (1 tan−□). This will give the gradient orientation.

Step 5: Then the MATLAB function im2col is called to rearrange the image blocks into columns. This is not a necessary step but it has to be done if we want to display the orientation histogram.

Step 6: Converting the column matrix with the radian values to degrees. This way we can scan the vector for values ranging from o 0 to o 90. This is because for real elements of X, a tan (X) is in the range [π/2, π/2]. This can also be seen from the orientation histograms where values come up only on the first and last quarter. Determining the number of the histogram bins was another issue that was solved by experimenting with various values

5.FORM OF RESULT:

For '0' the classification error is very small. For the 5th image the amount of noise (28 gaussian) is very high but it still classifies correctly. It will not tolerate blurring above a pixel radius of 2.0 though. On the other hand translation doesn't seem to cause any problems.

6.CONCLUSION

The concept is to use this technique in conjunction with Neural Networks. In other approaches of pattern recognition that orientation histograms have been used different ways of comparing and classifying were employed. Euclidean distance is a straightforward approach to it. It is efficient as long as the data sets are small and not further improvement is expected. Another advantage of using neural networks is that we can draw conclusions from the network output. This Research paper helps to transform the society in a way to help for the deaf and dump people to identify the signals through sign languages.

7. REFERENCES

[1]Christopher M. Bishop, "Neural networks for Pattern Recognition" Oxford, 1995.

[2]Duane Hanselman, Bruce Littlefield, "Mastering MATLAB, A comprehensive tutorial and reference", Prentice Hall

[3]Maria Petrou, Panagiota Bosdogianni, "Image Processing, The Fundamentals", Wiley