A Static Hand Gesture Recognition System Using a Composite Neural Network

Mu-Chun Su, Woung-Fei Jean, and Hsiao-Te Chang
Department of Electrical Engineering, Tamkang University, Taiwan, R.O.C.
muchun@cyber.ee.tku.edu.tw

Abstract

A system for the recognition of static hand gestures is developed. Applications of hand gesture recognition range from teleoperated control, to hand diagnostic and rehabilitation or to speaking aids for the deaf. We use two EMI-Gloves connected to an IBM compatible PC via HyperRectangular Composite Neural Networks (HRCNNs) to implement a gesture recognition system. Using the supervised decision-directed learning (SDDL) algorithm, the HRCNNs can quickly learn the complex mapping of measurements of ten fingers' flex angles to corresponding categories. In addition, the values of the synaptic weights of the trained HRCNNs were utilized to extract a set of crisp IF-THEN classification rules. In order to increase tolerance on variations of measurements corrupted by noise or some other factors we propose a special scheme to fuzzify these crisp rules. The system is evaluated for the classification of 51 static hand gestures from 4 "speakers". The recognition accuracy for the testing set were 93.9%.

1. Introduction

Hand gestures involve relative flexure of the user's fingers and consist of information that is often too abstract to be interpreted by a machine. Applications of hand gesture recognition widely range from teleoperated control to medicine or to entertainment [1], [2]. For instance, transform of human hand motion for telemanipulation is especially important in hazardous environments [3]. To motivate a patient to take hand rehabilitation exercise an appealing idea is to customize the hand rehabilitation procedure and then package the exercise into a video game format [4]. Another important application of hand gesture recognition is to improve the quality of life of the deaf or non-vocal persons through a hand-gesture to speech system. Due to congenital malfunction, disease, head injuries, or virus infections, deaf or non-vocal individuals are unable to communicate with hearing people through speech. Deaf or non-vocal persons use sign language or hand gestures to express themselves. However, most hearing people do not have the special sign language expertise. This is a major barrier between these two groups in daily communication. How to overcome this barrier to help the former persons to integrate into society is a very challenging research area.

Recently, several approaches have been proposed to implement hand-gesture to speech systems [5], [6], [7]. Basically, these speaking aids consist of three main components : (1) sensing gloves which convert hand gestures into computer readable data, (2) a recognition unit which maps hand-gestures to corresponding semantic interpretation, and (3) a speech synthesizer which outputs speech signals for hearing persons. The three most popular models of sensing gloves are VPL Data Glove [8], Virtex Cyber Glove [9], and Mattel Power Glove. They all have sensors that measure some or all of the finger joint angles. Each has its own advantages and disadvantages. As for the recognition unit, it is the heart of hand-gesture to speech systems. The simplest approach to recognizing hand gestures is to use a minimum distance classifier [10]. Designing a minimum distance classifier is straightforward. We simply store the sample values of each hand gesture and let each set of samples be represented as a prototype vector. When an unknown hand gesture is to be classified, we assign the unknown gesture to the class of its closest prototype. Although it requires little construction time, the price paid for this simplicity is that the resulting recognition accuracy is not very high. The reason is that measurement data is usually corrupted by disturbance (e.g. sensor noise, glove slipping

0-7803-3645-3/96 $5.00©1996 IEEE

on the hand etc.). Another approach is to use a Bayes classifier which is optimum in the sense that, on average, its use yields the lowest probability of committing classification errors [10]. Unfortunately, the statistical properties of the hand gestures often are unknown or involve considerable amounts of data preprocessing to estimate statistical parameters of each hand gesture. After the parameters have been estimated, the structure of the Bayes classifier is fixed, and its eventual performance will depend on how well the actual pattern population satisfy the underlying statistical assumptions made in the derivation of the estimation method being used.

The use of neural networks for the recognition of gestures has been examined by several researchers. Neural networks do not make any assumption regarding the underlying probability density functions or other probabilistic information about the pattern classes under consideration. They yield the required decision function directly via training. A two-layer backpropagation network with sufficient hidden nodes has been proven to be a universal approximator [11], [12]. Wibur has trained a neural network to recognize approximately 203 hand gestures derived from the American Sign Language (ASL) [13]. Fels and Hinton also trained backpropagation networks to recognize 66 root words, each with up to six different ending. The total size of the vocabulary is 203 words which were from a single "speaker". Although these results illustrated the potential of backpropagation networks for the recognition of hand gestures, we have difficulty in deciding the number of nodes in the hidden layer of a backpropagation network. There are no known rules for specifying the number of hidden nodes, so this number usually is based either on prior experience or simply chosen arbitrarily and then refined by testing. Besides, the backpropagation algorithm may converge slowly or be trapped at a local minimum. In addition, if we want to increase the number of hand gestures in the original vocabulary, we have to reconstruct the backpropagation network. This may require substantial training time.

We propose to train a HyperRectangular Composite Neural Networks (HRCNNs) for the recognition of hand gestures. There are two apparent advantages for this approach : (1) The training time is short because the SDDL algorithm is not a gradient-based method. (2) The values of synaptic weights of a trained HRCNN can be utilized to extract a set of IF-THEN rules. Based on these extracted rules, we can implement a conventional expert system or a fuzzy system. If we want to increase the number of hand gestures, all we need to do is to refine part of the corresponding rules. This greatly decreases reconstruction time. This paper is organized into five sections. Section II discusses the characteristics of HRCNNs. A description of the supervised decision-directed learning (SDDL) algorithm is given in section III. We use a data base consisting of 51 static hand gestures utilized by deaf persons in Taiwan from 4 "speakers" to evaluate the proposed method. The experimental results are shown in section IV. Finally, several concluding remarks are given in section V.

2. HyperRectangular Composite Neural Networks

A symbolic representation of a two-layer HRCNN is illustrated in Fig. 1. The mathematical description of a two-layer HRCNN is given as follow :

$$Out(\underline{x}) = f(\sum_{j=1}^{J} Out_j(\underline{x}) - \eta), \qquad (1)$$

$$Out_j(\underline{x}) = f(net_j(\underline{x})), \qquad (2)$$

$$net_j(\underline{x}) = \sum_{i=1}^{n} f((M_{ji} - x_i)(x_i - m_{ji})) - n, \quad (3)$$

and

$$f(x) = \begin{cases} 1 & if\, x \geq 0 \\ 0 & if\, x < 0 \end{cases} \qquad (4)$$

where M_{ji} and $m_{ji} \in R$ are adjustable synaptic weights of the jth hidden node, $\underline{x} = (x_1, \cdots, x_n)^T$ is an input pattern, n is the dimensionality of input variables, η is a small positive real number, and $Out(\underline{x}) : R^n \rightarrow \{0, 1\}$ is the output function of a two-layer HRCNN with J hidden nodes. The supervised decision-directed learning (SDDL) algorithm was developed to generate a two-layer HRCNN in a sequential manner by adding hidden nodes as needed [14]. As long as there are no identical data over different classes, we can obtain 100% recognition rate for training data. From our previous works [14]-[19], we have shown that the values of the synaptic weights of a trained HRCNN can be interpreted as a set of crisp IF-THEN rules. The IF-THEN classification rules extracted from a trained HRCNN with J hidden nodes can be represented as :

0-7803-3645-3/96 $5.00©1996 IEEE

$$
\begin{array}{l}
\text{IF } (\underline{x} \in [m_{11}, M_{11}] \times \cdots \times [m_{1n}, M_{1n}]) \\
\qquad \text{THEN } Out(\underline{x}) = 1; \\
\qquad \qquad \vdots \\
\text{IF } (\underline{x} \in [m_{J1}, M_{J1}] \times \cdots \times [m_{Jn}, M_{Jn}]) \\
\qquad \text{THEN } Out(\underline{x}) = 1.
\end{array}
\tag{5}
$$

where the rule antecedents define a set of n-dimensional hyperrectangles. Note that the representativeness of each rule can be measured by calculating the number of patterns contained by the corresponding hyperrectangle. The larger the number is the more representative the rule is.

There are many factors which will corrupt the measurements of the ten fingers' joint angles. First, sensor noise is usually unavoidable. Second, EMI Gloves tend to slip slightly when the user is using them to form hand gestures, therefore, some measurements will vary slightly. Third, as a different user uses the gloves measures are prone to change, causing the spread of each gesture class with respect to its prototype to become large. Actually, even if the same user forms the same hand gesture, repeatability is not guaranteed. Therefore some measurements can not be explained by those extracted crisp classification rules. So, some measured data to be classified is not contained in any one of the hyperrectangles defined by M_{ji} and m_{ji}. In order to generalize the extracted crisp classification rules to explain such patterns we incorporate fuzzy sets introduced by Zadeh [20] into the recognition system implemented by HRCNNs. We propose a special scheme to fuzzify crisp rules. A membership function $m_j(\underline{x})$ is used to measure the degree to which a pattern $\underline{x}$ is close to the hyperrectangle HR_j defined by $[m_{j1}, M_{j1}] \times \cdots \times [m_{jn}, M_{jn}]$. As $m_j(\underline{x})$ approaches 1, the pattern $\underline{x}$ is more close to HR_j, with the value 1 representing that the hyperrectangle HR_j contains the pattern $\underline{x}$. The membership function $m_j(\underline{x})$, as is shown in Fig. 2(a) and (b), is defined as follow :

$$m_j(\underline{x}) = \exp[-s_j^2(per_j(\underline{x}) - per_j)^2] \tag{6}$$

where

$$per_j(\underline{x}) = \sum_{i=1}^{n} \max(M_{ji} - m_{ji},\, M_{ji} - x_i,\, x_i - m_{ji}) \tag{7}$$

$$per_j = \sum_{i=1}^{n} (M_{ji} - m_{ji}) \tag{8}$$

and s_j is a sensitivity parameter which regulates how fast the membership value decrease as the distance between $\underline{x}$ and HR_j increases. In order to reflect the representative of each rule, it is reasonable to set the value of s_j to be in proportional to the difference between 1 and the fraction of patterns that fall within the hyperrectangle HR_j, which means

$$s_j = c(1 - p_j) \tag{9}$$

where p_j is ratio of the number of patterns contained by the jth hyperrectangle to the total number of patterns and c is a real-valued constant. To numerically combine the J fuzzy rules in order to compute the final membership value, two kinds of defuzzifiers are proposed.

(1) Maximum method

$$m(\underline{x}) = \max_{j=1,\cdots,J} m_j(\underline{x}) \tag{10}$$

It is the simplest and straightforward method.

(2) Maximum-likelihood method

$$m(\underline{x}) = \sum_{j=1}^{J} p_j\, m_j(\underline{x}) \tag{11}$$

This method is motivated by the popular probabilistic methods of maximum-likelihood and maximum-*a posteriori* parameter estimation [21].

At last a pattern $\underline{x}$ is assigned to class k if

$$m^{(k)}(\underline{x}) > m^{(l)}(\underline{x}), \qquad l = 1, 2, \cdots, M, \;\; l \neq k \tag{12}$$

where M represents the total number of classes and $m^{(l)}(\underline{x})$ represents the grade to which the pattern $\underline{x}$ belongs to the class l.

3. The Supervised Decision-Directed Learning (SDDL) Algorithm

The supervised decision-directed learning (SDDL) algorithm generates a two-layer feedforward network in a sequential manner by adding hidden nodes as needed. As long as there are no identical data over different classes, we can obtain 100% recognition rate for training data. First of all, training patterns are divided into two classes : (1) a "positive class" from which we want to extract the "concept" and (2) a "negative class" which provides the counterexamples with respect to the "concept". A "seed" pattern is used as the base of the "initial concept" (a hyperrectangle with arbitrarily small size). The seed pattern is arbitrarily chosen from the positive class. Then we try to generalize (expand) the initial concept (hyperrectangle) to include next positive pattern . After this , we have to check whether there is any negative pattern falling inside the present hyperrectangle in order to prevent the occurrence of "overgeneralization". The following step is to fetch next positive pattern and to generalize the initial concept to include the new positive pattern. This process involves growing the original hyperrectangle to make it larger to include the new positive pattern. After the process of generalization, again we use negative patterns to prevent overgeneralization. Fig. 3 illustrates the process of generalization and shows the way to prevent overgeneralization. This process is repeated for all the remaining positive patterns. If there is any unrecognized positive pattern, another hidden node is self-generated and the process of learning is repeated again and again until all positive patterns are recognized. The flowchart of the SDDL algorithm is given in Fig. 4. A more detailed description of the training procedure is given in [14], [15]. Given all description of the training algorithm, we now give pseudo-code description of two important procedures:

```
• Procedure of generalization
  begin ($\underline{x}$ is a positive pattern)
    for $i$ from 1 to dimensions-of-input
      begin
        if $x_i \ge M_{ji}(t)$
        then $M_{ji}(t+1) = x_i + \varepsilon$;
        else if $x_i \le m_{ji}(t)$
        then $m_{ji}(t+1) = x_i - \varepsilon$;
      end;
    end;
  end.
• Procedure of prevention-of-overgeneralization
  begin ($\underline{x}$ is a counterexample)
    for $i$ from 1 to dimensions-of-input
      begin
        if $x_i < M_{ji}(t)$
        then $M_{ji}(t+2) = x_i - \delta$ ($\delta$ should be chosen to ensure $x_i - \delta \ge M_{ji}(t)$);
        else if $x_i > m_{ji}(t)$
        then $m_{ji}(t+2) = x_i + \delta$ ($\delta$ should be chosen to ensure $x_i + \delta \le m_{ji}(t)$);
      end;
    end;
  end.
```

There are similarities between the fuzzy min-max networks classifier [22] and the hyperrectangular composite neural network. Both approaches utilize hyperrectangles as rule representation elements. However, there are many major differences between these two classes of neural networks [14]. First, the hyperrectangular composite neural network was originally developed to generate crisp IF-THEN rules. After crisp rules have been generated, a reasonable membership function is utilized to fuzzyify these crisp rules. In contrast, the fuzzy min-max neural networks focus on finding fuzzy rules. Second, the fuzzy min-max neural network classifier finds a set of hyperrectangles under a special expansion criterion so that the sizes of hyperrectangles are limited. On the contrary, hyperrectangular composite neural networks find hyperrectangles whose sizes are as large as possible. Third, the contraction procedures are different in these two classes of neural networks.

4. Experimental results

In our experiments, the ten fingers' joint angles were sensed by a pair of prototype EMI Gloves developed by the "Intelligent Information Processing" laboratory in Tamkang university. As compared with the Data Glove and Cyber Glove, the EMI Glove is a much cheaper alternative. In order to keep costs low, the EMI Glove used low-priced electromechanical strain gauges based on slide resistors. A single sensor measured all the joints in the finger at once. The flexure information of each finger is measured directly by a change of resistance. This change in resistance induces a change of voltage which is then amplified and digitized by an A/D converter. The digitized voltage is sampled by a IBM 486 compatible PC. We did not further transform the raw measurements into exact joint angle data since we hoped not to waste computation resources in order to develop a real-time gesture recognition system. Despite its low precision and limited sensing capability, the fact that the prototype EMI Glove was low-priced and easily designed has made us be able to conduct the experiments.

Since there was no sensor placed on the prototype EMI Glove to determine the hand orientation, our experiments focused on static gesture recognition which relies only on the information about the angles of the fingers. Presently the data base consists of 51 gestures formed by 4 users. Each gesture was repeated ten times by each user. All these gestures are currently used by the deaf in Taiwan. Note that the gestures used by the deaf will be different if they are from different countries. Even if they are in the same country, variations of gestures still may exist. The total number of data in our data base is 2040. We split the data base into two sets : (1) a training set consisting of 1530 data and (2) a testing set consisting of 510 data. Some examples of the gestures used in our data base are given in Table I. After training, the average accuracies for the training set is 100% and for testing set is 93.9%. The experimental results are very encouraging.

5. Concluding Remarks

A real-time static hand gesture recognition system is presented in this paper. By training HyperRectangular Composite Neural Networks incorporated with a special fuzzifying scheme, the complex mapping of hand gestures to corresponding semantic interpretation is leaned from a data base consisting of 51 gestures formed by 4 persons. The recognition accuracy for the testing set is 93.9%. We are currently investigating the method of recognizing dynamic gestures which relies on taking hand movement into account.

Acknowledgment

This work is supported in part by the National Science Council under the Grant No. NSC 85-2213-E-032-009.

References

[1] R. Skalwsky, The Science of Virtual Reality and Virtual Environments, Addison-Wesley, 1993.

[2] G. Burdea and P. Coiffet, Virtual Reality Technology, Joh Wiley & son, Inc., 1993.

[3] T. H. Speeter, "Transforming human hand motion for telemanipulation," Presence, vol. 1, no. 1, pp. 63-70, 1992.

[4] G. Burdea, D. Langrana, D. silver, R. Stone and D. Dipaolo, "Diagonstic / rehabilitation system using dextrous force feedback," Proceedings of Interface to Real and Virtual Worlds Conference, Montpellier, France, pp. 255-265, March, 1992.

[5] J. Kramer, L. Leifer, "The talking glove : a speaking aid for nonvocal deaf and deaf-bilid individuals," Proc. of RESNA 12th Annual Conference, New Orloans, Louisiana, pp. 471-472, 1989.

[6] J. Kvamer and L. Leifer, "The talking glove : a speaking aid for nonvocal deaf and deaf-blind individuals," Proc. of the RESNA 12th Annual Conf., New Orleans, Louisiana, pp. 471-472, 1989.

[7] Fels, S. Sidney, and Geoffrey E. Hinton, "Glove-talk : a neural network interface between a data-glove and a speech synthesizer," IEEE Trans. on Neural Networks, vol. 4, no. 1, pp. 2-8, Jan., 1993.

[8] VPL Research Inc., DataGlove Model : User's Manual, Redwood City, CA, 1987.

[9] Virtex Co., Company brochure, Standford, CA, October, 1992.

[10] J. T. Tou and R. C. Gonzalez, Pattern Recognition Principles, Addison-Wesley Pub. Comp., 1977.

[11] G. Cybenko, "Approximation by superpositions of a sigmoid function," Mathematics of Control, Signals, and Systems, no. 2, pp. 303-314, 1989.

[12] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," Neural Networks, no. 2, pp. 359-366, 1989.

[13] R. B. Wilbur, American Sign Language and Sign System. Baltimore, MD : University Park Press, 1979.

[14] M. C. Su, A Neural Network Approach to Knowledge Acquisition, Ph.D. Dissertation, University of Maryland, August, 1993.

[15] M. C. Su, "Use of neural networks as medical diagnosis expert systems," in Computers in Biology and Medicine, vol. 24, no. 6, 1994.

[16] C. H. Hsieh, M. C. Su, and C. T. Tseng, "A mandarin digits recognition system based on a novel class of hyperrectangular composite neural networks," IASTED International conference on Modeling, Simulaiton & Identification, pp. 241-244, Japan, 1994.

[17] M. C. Su, "A neuro-fuzzy approach to medical diagnosis expert systems," The 1st Medical Engineering Week of the Workd, Taiwan, pp. 138-143, 1994.

[18] M. C. Su, C. J. Kao, K. M. Liu, and C. M. Wu, "Neural Network-based Fuzzy System," 1994 International Computer Symposium, pp. 1246-1250, Taiwan, 1994.

[19] C. T. Hsieh, M. C. Su, and S. C. Chien, "Use of a self-learning neuro-fuzzy system for syllabic labeling of continuous speech," Fuzz-IEEE/EFES'95, pp. 1727-1734, Japan, 1995.

[20] L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, pp. 338-353, 1965.

[21] B. Kosko, Neural Networks and Fuzzy Systems : A Dynamical Systems Approach to Machine Intelligence, Prentice-Hall, Inc., New Jersey, 1992.

[22] P. K. Simpson, "Fuzzy min-max neural networks-part 1 : Classification," IEEE Trans. on Neural Networks, vol. 3, pp. 776-786, Sep. 1992.

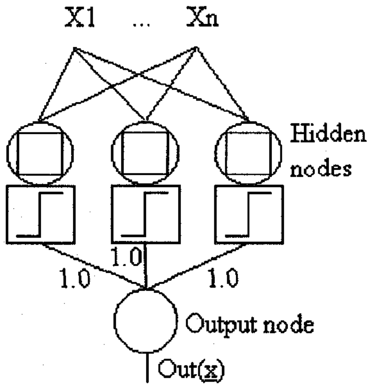
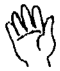
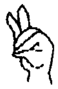
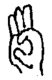
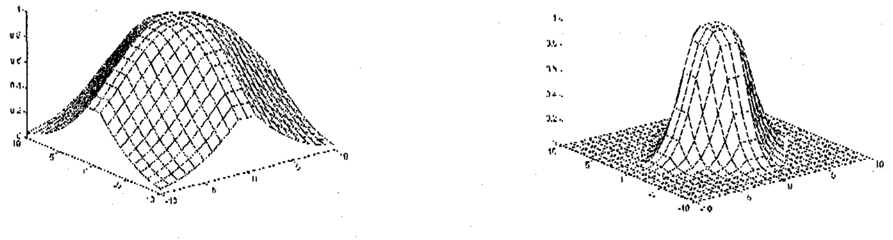


Figure 1. Symbolic representation of a two-layer HRCNN.

Table I. Examples of hand gestures used in Taiwan.

No.	Chinese	English	Hand Gesture
1	五	five	
2	百	hundred	
3	廁所	bathroom	



(a) (b)

Figure 2. Membership function, $m_j(\underline{x})$, with (a) s_j=1.0, (b) s_j=10.0.

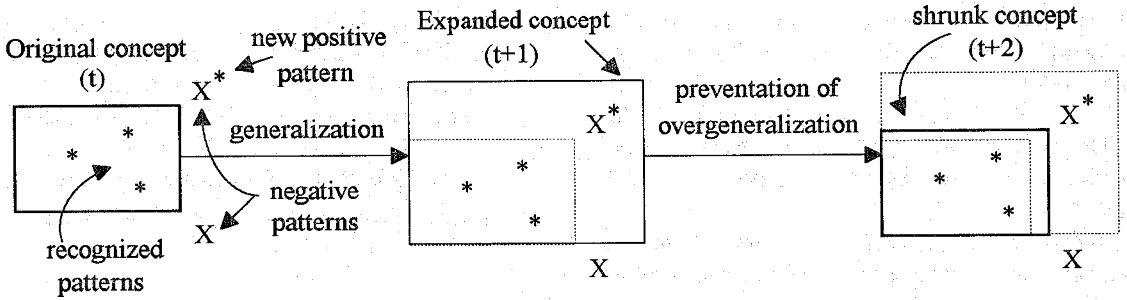


Figure 3. The process of generalization and prevention of overgeneralization.

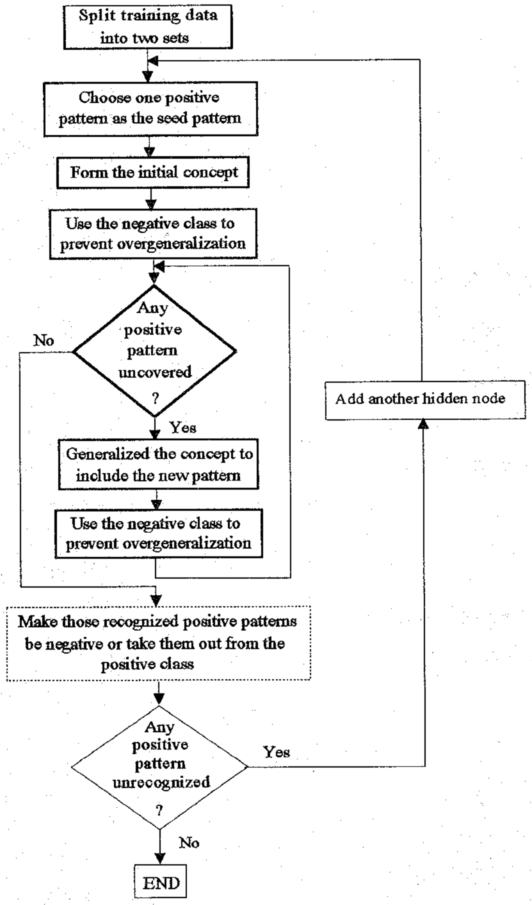

Split training data into two sets
Choose one positive pattern as the seed pattern
Form the initial concept
Use the negative class to prevent overgeneralization
Any positive pattern uncovered ?
No
Yes
Generalized the concept to include the new pattern
Use the negative class to prevent overgeneralization
Make those recognized positive patterns be negative or take them out from the positive class
Any positive pattern unrecognized ?
Yes
No
END
Add another hidden node

Figure 4. Flowchart of the SDDL algorithm.