Recognition of Sign Language Subwords Based on Boosted Hidden Markov Models

Liang-Guo Zhang[1,3], Xilin Chen[1,2], Chunli Wang[1], Yiqiang Chen[1], Wen Gao[1,2,3]
[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China
[2]Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China
[3]Graduate School of the Chinese Academy of Sciences, Beijing 100080, China
{lgzhang, xlchen, clwang, yqchen, wgao}@jdl.ac.cn

ABSTRACT
Sign language recognition (SLR) plays an important role in human-computer interaction (HCI), especially for the convenient communication between deaf and hearing society. How to enhance the traditional hidden Markov models (HMM) based SLR is an important issue in the SLR community. And how to refine the boundaries of the classifiers to effectively characterize the property of spread-out of the training samples is another significant issue. In this paper, a new classification framework applying adaptive boosting (AdaBoost) strategy to continuous HMM (CHMM) training procedure at the subwords classification level for SLR is presented. The ensemble of multiple composite CHMMs for each subword trained in boosting iterations tends to concentrate more on the hard-to-classify samples so as to generate more complex decision boundary than that of the single HMM classifier. Experimental results on the vocabulary of frequently used Chinese sign language (CSL) subwords show that the proposed boosted CHMM outperforms the conventional CHMM for SLR.

Categories and Subject Descriptors
H.1.2 [**Models and Principles**] User/Machine Systems – *Human information processing*; I.5.2 [**Pattern Recognition**]: Design Methodology – *Classifier design and evaluation*

General Terms
Algorithms, Design, Experimentation, Human Factors, Languages

Keywords
Human-computer interaction, sign language recognition, HMM, AdaBoost

1. INTRODUCTION
Sign language, as a kind of most structured human gesture, is regarded as one of the most natural means of exchanging information for deaf people. The goal of SLR is to provide an efficient and accurate mechanism to transcribe sign language into text or speech so that the communication between deaf and hearing society can be more convenient. SLR, as one of the important research areas of HCI, has attracted more and more interest in HCI society for its significant academic value as well as broad application prospect.

Up to now, SLR can be classified into two classes according to the devices used to capture gestures, i.e. datagloves based SLR [1~4] and vision based SLR [5~9]. Kim *et al*. [1] applied fuzzy min-max neural network and fuzzy logic approach to recognize 31 manual alphabets and 131 Korean signs based on datagloves and reported an accuracy of 96.7% for manual alphabets and 94.3% for sign words. Liang and Ouhyoung [2] employed HMM as recognition method and used the datagloves as input device to recognize continuous Taiwan SLR with the average recognition rate of 80.4% for 250 signs. Gao *et al*. [3] used datagloves as input devices and HMM as recognition method to recognize 5177 isolated signs with 94.8% accuracy and recognize 200 sentences with 91.4% word accuracy. Starner *et al*. [4] used single camera to extract two-dimensional features as the input of HMM to realize continuous American SLR. The word accuracy of 92% or 98% was achieved when the camera was mounted on the desk or in a user's cap in recognizing the sentences based on a 40-word lexicon. Vogler *et al*. [5] used computer vision methods to extract the three-dimensional feature parameters of a signer's arm motions and applied HMM to recognize continuous ASL sentences with a best accuracy of 95.83% on a vocabulary of 53 signs. In addition, Vogler *et al*. [6] used phonemes instead of whole signs as the basic units and modeled them with parallel HMM (PaHMM). Grobel and Assan [7] used HMM to recognize the isolated signs obtaining 91.3% accuracy on a 262-sign vocabulary with the aid of colored gloves. HMM was also employed by Hienz and Bauer [8] to recognize continuous German sign language and an accuracy of 91.7% can be achieved in recognition of sign language sentences with 97 signs. Zhang *et al*. [9] developed a system to recognize 439 frequently used CSL signs from a frontal view with an accuracy of 92.5% achieved. Tied-mixture density HMM was used to speed up the recognition without the significant loss of accuracy.

From the review above, we know that most current researches of SLR mainly take HMM as the recognition method. How to improve the performance of conventional HMM is still an important issue. Some variants of HMM, e.g., PaHMM, are studied as attempts to this improvement. Furthermore, as we know, the pattern recognition errors usually occur when there exist some hard-to-classify samples in the sample space. Especially, in SLR, we must deal with the recognition of single handed signs and (or) similar sign pairs, which is usually considered as a difficult problem. Therefore, how to refine the boundaries of the classifiers to effectively characterize the property of spread-out of the

ICMI'05, October 4–6, 2005, Trento, Italy.

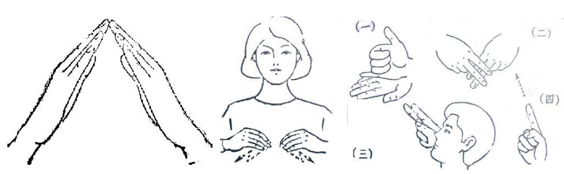
training samples, especially focusing more on hard samples, is also significant in the research of SLR.

In this paper, AdaBoost is incorporated into CHMM based classifier training procedure to generate the ensemble of multiple CHMM classifiers so as to improve the performance of the traditional single CHMM based SLR. The work is motivated by combining the benefits of a well-tuned probabilistic model for sequential signal processing, i.e., HMM, with the state-of-the-art margin-maximizing technique from machine learning, i.e., boosting.

Boosting has caused great excitement in the field of machine learning for several years. Although it has been studied in the areas such as speech recognition [11]~[13] and lip reading [14][15], very little research has examined the application of techniques to the problem of SLR. Only recently, Ong [10] constructed boosted classifier tree to detect and classify hands shapes. Moreover, like in speech recognition, there have been some work on the integration of boosting and HMM [13]~[15].

Inspired by the related works mentioned above, we integrate the AdaBoost and CHMM in the following way. We convert multi-class SLR problem into many binary classification problems since the output of the CHMM is probabilistic measure, not a Boolean decision. As for boosted CHMM training, at each boosting round, the base classifiers of all classes are trained and verified in the parallel and interrelated style. Through a series of sequential rounds of base classifier training and weight adjusting, boosted CHMM classifiers ensemble for each class is trained.

Subword [16], similar to *phoneme* in speech, is introduced to facilitate the future realization of large vocabulary SLR. A subword is defined as the smallest contrastive unit which has its own meaning and distinguishes one sign from another. In CSL, an example of such a subword is the “room” (Fig.1 (a)), which can form the gesture “classroom” with another subword “education” (Fig.1 (b)). While the Chinese idiom “zuo-jing-guan-tian” (Fig.1 (c)) consists of four different subwords. Thus, boosting is applied to HMM training at the subword classification level. For future research, we can extend the boosted CHMM classification approach to isolated or continuous SLR by the means of linking the subword models.



(a) (b) (c)

Figure 1. (a) Subword ”room”, (b) Subword “education”, (c) Chinese idiom “zuo-jing-guan-tian” [16]

The rest of this paper is organized as follows. In section 2, we first briefly introduce some background of HMM and AdaBoost. Then, we discuss the proposed boosted CHMM in detail. In section 3, we describe the recognition system based on boosted CHMM. In section 4, we evaluate the performance of the proposed method. Section 5 concludes the paper.

2. BOOSTED CHMM

In this section, some basic ideas of HMM and boosting learning algorithm, especially AdaBoost, are briefly described. More detailed information of HMM and boosting can refer to [17] and [18], respectively.

2.1 HMM

HMM [17] has been proven to be one of the most successful statistical modeling methods in the area of speech recognition, and employed by more and more SLR researchers in recent years.

Formally, an HMM λ consisting of N states $s_1, s_2, ..., s_N$ can be specified by its parameters as $\lambda = (\pi, A, B)$. Here, π stands for the vector of the initial probabilities π_i of the system starting in state s_i. The parameter A represents the matrix of state-transition probabilities a_{ij} that a transition from state s_i to state s_j is taken at regularly spaced discrete time intervals. The parameter B has two representing forms, i.e., discrete form and continuous form. In the former way, B denotes observation probability distribution of state s_i as $b_i(v)$, v is any discrete observation symbol and discrete HMM is determined. In the latter way, B represents observation probability density function of state s_i as $b_i(X)$, X is any continuous observation vector. Usually, $b_i(X)$ can be described as multivariate Gaussian mixture density function as follows

$$b_i(X) = \sum_{m=1}^{M} c_{im} G(X, \mu_{im}, \Sigma_{im}), \tag{1}$$

where

$$G(X, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^D |\Sigma|}} \exp\left[-\tfrac{1}{2}(X-\mu)' \Sigma^{-1} (X-\mu)\right] \tag{2}$$

and c_{im} is the mixing coefficient, which satisfies $\sum_{m=1}^{M} c_{im} = 1$. In this case, continuous HMM (CHMM) is drawn.

Let $X = [X^{(1)}, X^{(2)}, \ldots, X^{(K)}]$ be the set of K training samples for one class (e.g., corresponding to one sign in SLR) and $X^{(k)} = [X_1^{(k)} X_2^{(k)} \cdots X_{T_k}^{(k)}]$ be the k^{th} training sample data with T_k frames. Given $X^{(k)}$ and λ, we define the following forward and backward variables:

$$\alpha_t^{(k)}(i) = P(X_1^{(k)} X_2^{(k)} \cdots X_t^{(k)}, q_t = s_i \mid \lambda) \tag{3}$$

$$\beta_t^{(k)}(i) = P(X_{t+1}^{(k)} X_{t+2}^{(k)} \cdots X_{T_k}^{(k)} \mid q_t = s_i, \lambda). \tag{4}$$

Based on the above two variables, we define $\xi_t^{(k)}(i,j)$, the *a posteriori* probability of transition from state s_i at time t to s_j at time $t+1$, and $\gamma_t^{(k)}(i,m)$, the condition density of $X_t^{(k)}$ coming from m^{th} component of the state s_i, as follows:

$$\xi_t^{(k)}(i,j) = P(q_t = s_i, q_{t+1} = s_j \mid X^{(k)}, \lambda) = \frac{\alpha_t^{(k)}(i) a_{ij} b_j(X_{t+1}^{(k)}) \beta_{t+1}^{(k)}(j)}{P(X^{(k)} \mid \lambda)} \tag{5}$$

$$\gamma_t^{(k)}(i,m) = \frac{\alpha_t^{(k)}(i)\beta_t^{(k)}(i)}{\sum_{i=1}^{N}\alpha_t^{(k)}(i)\beta_t^{(k)}(i)} \cdot \frac{c_{im}G(X_t^{(k)},\mu_m,\Sigma_m)}{\sum_{m=1}^{M}c_{im}G(X_t^{(k)},\mu_m,\Sigma_m)} \tag{6}$$

Using the standard *Baum-Welch algorithm* [17], the parameters of the CHMM can be obtained in the following.

$$\overline{a}_{ij} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k-1}\xi_t^{(k)}(i,j)}{\sum_{k=1}^{K}\sum_{t=1}^{T_k-1}\sum_{j}\xi_t^{(k)}(i,j)} \tag{7}$$

$$\overline{c}_{im} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\gamma_t^{(k)}(i,m)}{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{m=1}^{M}\gamma_t^{(k)}(i,m)} \tag{8}$$

$$\overline{\mu}_{im} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\gamma_t^{(k)}(i,m)X_t^{(k)}}{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\gamma_t^{(k)}(i,m)} \tag{9}$$

$$\overline{\Sigma}_{im} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\gamma_t^{(k)}(i,m)(X_t^{(k)}-u_{im})(X_t^{(k)}-u_{im})'}{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\gamma_t^{(k)}(i,m)} \tag{10}$$

In general, the recognition result of an HMM based *W*-class classification problem is determined as follows:

$$w^* = \arg\max_{1\le w\le W} P(X_{test} \mid \lambda^w), \tag{11}$$

where $P(X_{test} \mid \lambda^w)$ is the decoded probability of the test sample X_{test} under the model λ^w of the w^{th} class.

2.2 Boosting and AdaBoost

Boosting [18][19] is a family of ensemble techniques to improve the performance of classifiers by sequentially training and then combining a collection of base (or called weak) classifiers through an iterative process, which makes the succeeding new base classifier(s) put more emphasis on the training samples that are hardest to be classified by previous iterations. It has been theoretically proved that boosting converges to optimal classifiers ensemble on the training set and has low generalization error.

The most successful and promising boosting algorithm widely used for binary classification problem is adaptive boosting, i.e., AdaBoost [18]. AdaBoost can generate a stronger classifier with good generalization property by linearly combining multiple base classifiers (learners) trained from a sequence of boosting iterations. That is, it maintains a distribution of weights over the training samples, and emphasizes those hard-to-discriminate samples by iteratively learning a new classifier using a base learner and then reweighting the training samples. Thus, the whole process of adaptive boosting consists of a series of rounds of base classifier training and weight adjusting. The basic algorithm of AdaBoost for binary classification problem is outlined in Figure 2.

If the training error ε_t of λ_t is written as $\varepsilon_t = 0.5 - \gamma_t$ $(\gamma_t > 0)$, the error is bounded as follows [19].

$$\varepsilon \le \prod_t [2\sqrt{\varepsilon_t(1-\varepsilon_t)}] = \prod_t \sqrt{1-4\gamma_t^2} \le \exp\left(-2\sum_t \gamma_t^2\right) \tag{12}$$

Thus, if any weak learner has an error less than 0.5, the overall training error decreases exponentially.

To deal with multi-class classification problem, there exist some extensions of the binary-class AdaBoost, e.g., AdaBoost.M1 and AdaBoost.M2 [19].

Input: $(x_1, y_1), \ldots, (x_K, y_K)$,

where $x_k \in X, y_k \in Y = \{-1,+1\}$.

Initialize: $D_1(k) = 1/K$.

Repeat: for boosting round $t = 1, \ldots, T$:

- *Train weak learner* λ_t *using distribution* D_t;
- *Get weak hypothesis* $h_t : X \to \{-1,+1\}$ *with estimated error* $\varepsilon_t = \Pr_{k\sim D_t}[h_t(x_k) \ne y_k]$;
- *Loop terminates if* $\varepsilon_t \ge \frac{1}{2}$ *and set* $T = t-1$;
- *Set* $\alpha_t = \frac{1}{2}\ln((1-\varepsilon_t)/\varepsilon_t)$;
- *Update the distribution*:

$$D_{t+1}(k) = \frac{D_t(k)\exp(-\alpha_t y_k h_t(x_k))}{Z_t}$$

where Z_t *is a normalization factor to make* $D_{t+1}(k)$ *a distribution.*

Output: the final hypothesis

$$H(x) = sign\left(\sum_{t=1}^{T}\alpha_t h_t(x)\right)$$

Figure 2. AdaBoost algorithm [18]

2.3 Boosted CHMM

In this section, we discuss the parameters estimation of each boosted base classifier, the construction of boosted CHMM ensemble and boosted CHMM based recognition in detail.

2.3.1 Parameters Estimation

As described in Sec. 2.2, AdaBoost maintains a distribution of weights over the training samples. Thus, when training CHMM in the framework of boosting, the forward-backward reestimation equations in Sec. 2.1 can be adjusted to incorporate the set of weights $\{D_k\}$ in each training round as follows.

$$\overline{a}_{ij} = \frac{\sum_{k=1}^{K}D_k\sum_{t=1}^{T_k-1}\xi_t^{(k)}(i,j)}{\sum_{k=1}^{K}D_k\sum_{t=1}^{T_k-1}\sum_{j}\xi_t^{(k)}(i,j)} \tag{13}$$

$$\bar{c}_{im} = \frac{\sum_{k=1}^{K} D_k \sum_{t=1}^{T_k} \gamma_t^{(k)}(i,m)}{\sum_{k=1}^{K} D_k \sum_{t=1}^{T_k} \sum_{m=1}^{M} \gamma_t^{(k)}(i,m)} \qquad (14)$$

$$\bar{\mu}_{im} = \frac{\sum_{k=1}^{K} D_k \sum_{t=1}^{T_k} \gamma_t^{(k)}(i,m) X_t^{(k)}}{\sum_{k=1}^{K} D_k \sum_{t=1}^{T_k} \gamma_t^{(k)}(i,m)} \qquad (15)$$

$$\bar{\Sigma}_{im} = \frac{\sum_{k=1}^{K} D_k \sum_{t=1}^{T_k} \gamma_t^{(k)}(i,m)(X_t^{(k)} - u_{im})(X_t^{(k)} - u_{im})'}{\sum_{k=1}^{K} D_k \sum_{t=1}^{T_k} \gamma_t^{(k)}(i,m)} \qquad (16)$$

where $\xi_t^{(k)}(i,j)$ and $\gamma_t^{(k)}(i,m)$ are defined as in Eq. (5) and Eq. (6), respectively.

The reestimation equations $\{\pi_i\}$ are the same to the conventional single CHMM. As proved in [20], the imposing of certain weights on training samples, as in reestimation equations above, still maintains the convergence property of the maximum likelihood estimation as in the conventional HMM parameters reestimation. In fact, equations (13)~(16) may be considered as the output of a variant of the standard Baum-Welch algorithm.

2.3.2 Ensemble Construction

To realize boosted CHMMs ensemble training, the problem of the evaluation of the classification error of any base learner should be first resolved. And to deal with boosted CHMM recognition, how to make the final decision is another problem.

Assume that in boosted CHMM training for a *W*-class classification problem, class w $(1 \le w \le W)$ has an ensemble of T base CHMMs $\{\lambda_1^w, \lambda_2^w, \ldots, \lambda_T^w\}$ and K training samples $X^w = \{X_1^w, X_2^w, \ldots, X_K^w\}$. As in Eq. (11), the recognition result of CHMM classifier is determined by comparing all probabilities of the test sample under all CHMMs in the recognition lexicon, but not in a definite Boolean form like other types of classifiers. Thus, in our boosted CHMM framework, we convert *W*-class problem into multiple binary classification problems using the *one-against-all-others* strategy. That is, the correct class label of one sample under the classifiers ensembles of all classes is determined as follows:

$$\lambda^* = \arg\max_{\lambda} P(X_k^w \mid \lambda_t^w) . \qquad (17)$$

It can be inferred from Eq. (17) that the correct prediction result is obtained only when the observation probability of correct model measures greater than that of any other models; otherwise, an error occurs. Thus, the binary hypothesis of any sample X_k^w under the base CHMM λ_t^w is determined as

$$h_t^w(X_k^w) = \begin{cases} 1 & if \quad P(X_k^w \mid \lambda_t^w) > P(X_k^w \mid \lambda_{t'}^{w'}) \\ -1 & otherwise \end{cases} , \qquad (18)$$

where $w \ne w'$.

Therefore, the training error of any base CHMM λ_t^w over all the training samples of class w at boosting round t is estimated as follows:

$$\varepsilon_t^w = \Pr_{X_k^w \sim D_t^w}[h_t^w(X_k^w) = -1] = \sum_{k:h_t^w(X_k^w)=-1} D(X_k^w) . \qquad (19)$$

From Eq. (18) and Eq. (19), we can see that at any boosting round, the training error of one class is correlated with other. As a result, with the progress of the boosted training, the training errors of all trained base CHMMs may change. Thus, we should update the changed errors of all CHMMs.

To treat the base classifier of each class more equally, at every boosting round, we train each base CHMM with the corresponding weights over the training samples for each class (sign) in a parallel fashion. The procedure of boosted CHMM training for a *W*-class classification problem is outlined in Fig. 3.

Initialize: $D_1^w(X_k^w) = 1/K$.

Repeat: for boosting round $t = 1, \ldots, T$:

- *Train* W *base classifiers* $\{\lambda_t^w\}$ *with distribution* $\{D_t^w\}$ *using Eq.* (13)~(16) *for* W *classes in parallel*;
- *Compute base hypothesis* $h_t^w : X^w \to \{-1,+1\}$ *using Eq.* (18) *and the corresponding error* ε_t^w *using Eq.* (19) *for each class* w . *Compute errors* $\{\varepsilon_{t'}^{w'}\}$ *of all other trained base classifiers*.
- *Loop terminates if any* $\varepsilon_t^w \ge \frac{1}{2}$ *and set* $T = t-1$;
- *Calculate the weight* $\alpha_t^w = \frac{1}{2}\ln((1-\varepsilon_t^w)/\varepsilon_t^w)$ *of* λ_t^w *for each class* w . *If the error of any other trained classifier* $\lambda_{t'}^{w'}$ *changes, the corresponding* $\alpha_{t'}^{w'}$ *is also recalculated*;
- *Update the distribution of the weight over samples*:

$$D_{t+1}^w(X_k^w) = \frac{D_t^w(X_k^w)\exp(-\alpha_t h_t(X_k^w))}{Z_t} ,$$

where Z_t *is a normalization factor to make* $D_{t+1}^w(X_k^w)$ *a distribution*.

Figure 3. Boosted CHMM based training procedure, taking *W*-class classification problem as an example

2.3.3 Boosted CHMM Based Recognition

After boosted CHMM training, for each class w, the ensemble Ω^w comprises T base CHMMs $\{\lambda_1^w, \lambda_2^w, \ldots, \lambda_T^w\}$ and T corresponding classifier weights $\{\alpha_1^w, \alpha_2^w, \ldots, \alpha_T^w\}$. Thus, for recognition of any test sample X, the normalized log-probability of observed X under the ensemble Ω^w is computed as follows:

$$\overline{P}(X \mid \Omega^{w}) = \frac{1}{T}\sum_{t=1}^{T}\log[\alpha_t^{w} P(X \mid \lambda_t^{w})]. \tag{20}$$

Therefore, similar to Eq. (11), the final decision of the boosted CHMM based recognition result is made as follows:

$$w^* = \arg\max_{1 \le w \le W} \overline{P}(X \mid \Omega^{(w)}). \tag{21}$$

3. SUBWORDS RECOGNITION SYSTEM BASED ON BOOSTED CHMM

The structure of the CSL subwords recognition system based on the proposed boosted CHMM is shown in Figure 4. The overall recognition system works as follows. First, the collected sign data are fed into the data preprocessing and feature extraction module indicated by the upper dashed frame. Then, the feature vectors of training and test samples are input into the boosted CHMM based recognizer training and sign recognition module, which is indicated by the bottom dashed frame in Figure 4.

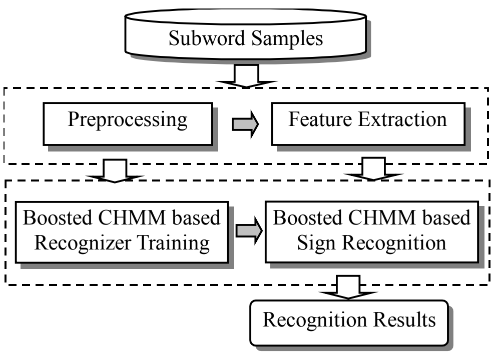


Figure 4. System overview

Two Cybergloves and three Pohelmus 3SAPCE-position trackers are used as input devices. Two trackers are positioned on the wrist of each hand and another is fixed at signer's back (as the reference tracker). The Cybergloves collect the variation information of hand shape with the 18-dimensional data each hand, and the position trackers capture the variation information of orientation, position, and movement trajectory.

In order to extract the features invariant to signer's position, the tracker at signer's back is chosen as the origin of the reference Cartesian coordinate system, and the position and orientation at each hand with respect to the reference system are calculated to be invariant features. Through this transformation, the feature data are composed of a relative three-dimensional position vector and a three-dimensional orientation vector for each hand, which don't change with the signer position and orientation. In the case of two hands, a 48-dimensional vector is formed, including the hand shape, position and orientation vector. As each component in the vector has different dynamic range, its value is normalized between 0 and 1.

4. EXPERIMENTS AND DISCUSSIONS

All experiments were performed on the vocabulary of 102 frequently used CSL single-handed subword signs, where every two or more signs are some similar to each other. Each sign is collected 30 times by one signer, i.e., totally 3060 samples are obtained with 2550 samples for training and remained 510 samples for test. For the convenience of comparison, each (base) classifier is a 3-state CHMM with 5 Gaussian mixture components.

The first experiment is to test the recognition performance of the boosted CHMM comparing with that of the classic CHMM. Figure 5 shows the average test results of the boosted CHMM at different number of boosting iterations compared with that of the single CHMM. From Figure 5, we can see that the boosted CHMM begins to converge only after 5 iterations since there are not too many training samples employed. The average recognition rates of 89.8% for the single CHMM and the best accuracy of 92.7% for the boosted CHMM are obtained. Thus, an accuracy improvement of 2.9% is achieved.

The improved performance above is due to the following reason. For each subword sign, through boosted CHMMs training, an ensemble of composite classifiers is obtained, which definitely generate more complex classification boundary for the current sign versus all others, especially the ensemble focuses more on the hard-to-classify samples. That is, the boosted CHMM ensemble can more effectively characterize the property of the spread-out of samples than the single CHMM. Therefore, the boosted CHMM obtains better accuracy than that of the single CHMM.

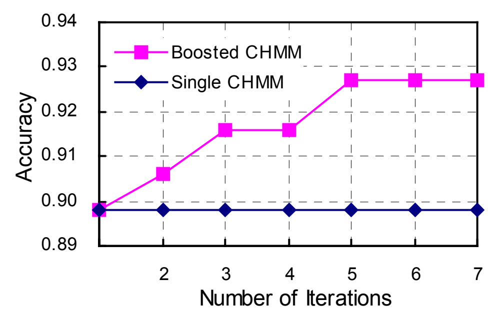


Figure 5. The recognition accuracy comparison of two models

Besides, we evaluate the classification performance by using cumulative match score (CMS) [21], which describes "*is the correct answer in the top n matches*?" and the CMS curve is shown in Figure 6. It can be observed that the boosted CHMM effectively improves the accuracy relative to the single CHMM. Here, the boosting iteration number is 5.

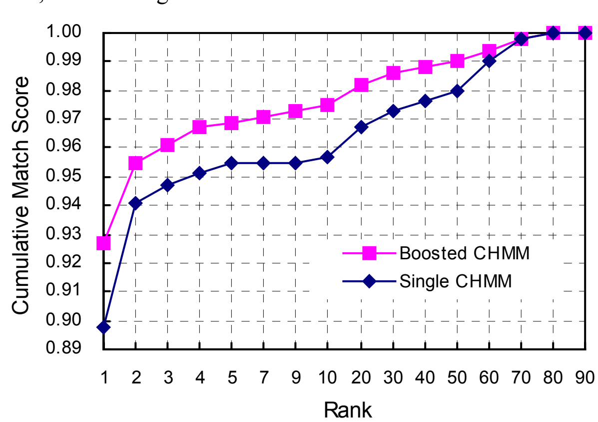


Figure 6. CMS comparison of two models

5. CONCLUSIONS

In this paper, a boosted CHMM classification framework, which incorporates AdaBoost strategy into the conventional CHMM training procedure, is presented and applied to the recognition of sign language subwords. The ensemble of multiple boosted composite CHMMs for each subword tends to focus more on the hard-to-classify samples so as to generate more complex decision boundary than that of the single HMM classifier. Experimental results on the vocabulary of 102 frequently used CSL subwords show that boosted CHMM improves the recognition accuracy of about 3%, which demonstrates that the performance of the proposed boosted CHMM classifiers ensemble is better than that of the traditional single CHMM classifier for SLR. Future work will focus on the isolated and continuous SLR based on the proposed boosted CHMM method.

6. ACKNOWLEDGMENTS

This work was supported by Natural Science Foundation of China (Grant No. 60303018). The authors are grateful to Liang Dong for his valuable suggestions.

7. REFERENCES

[1] J.S. Kim, W. Jang, and Z. Bien. A dynamic gesture recognition system for the Korean sign language (KSL). *IEEE Trans. Systems, Man, and Cybernetics*, vol. 26, no. 2, 1996, 354-359.

[2] R.H. Liang, and M. Ouhyoung. A real-time continuous gesture recognition system for sign language. In *Proc. Third Int'l Conf. Automatic Face and Gesture Recognition (FGR)*, 1998, Nara, 558-565.

[3] W. Gao, J.Y. Ma, J.Q. Wu, and C.L. Wang. Sign language recognition based on HMM/ANN/DP. *Int'l J. Pattern Recognition and Artificial Intelligence*, vol. 14, no. 5, 2000, 587-602.

[4] T. Starner, J. Weaver, and A. Pentland. Real-time American sign language recognition using desk and wearable computer based video. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, 1998, 1371-1375.

[5] C. Vogler and D. Metaxas. Adapting hidden Markov models for ASL recognition by using three-dimensional computer vision methods. In *Proc. IEEE Int'l Conf. Systems, Man and Cybernetics (SMC)*, 1997, 156-161.

[6] C. Vogler and D. Metaxas. A framework for recognizing the simultaneous aspects of American sign language. *Computer Vision and Image Understanding **81***, 2001, 358-384.

[7] K. Grobel and M. Assan. Isolated sign language recognition using hidden Markov models. In *Proc. IEEE Int'l Conf. SMC*, 1996, 162-167.

[8] B.Bauer and H. Hienz. Relevant features for video-based continuous sign language recognition, In *Proc. FGR*, 2000, 440-445.

[9] L.-G. Zhang, Y. Chen, G. Fang, X. Chen, W. Gao. A vision-based sign language recognition system using tied-mixture density HMM. In *ACM Sixth Int'l Conf. Multimodal Interfaces (ICMI)*, 2004, 198-204.

[10] E.-J. Ong and R. Bowden. A boosted classifier tree for hand shape detection. In *Proc. FGR*, 2004.

[11] H. Schwenk. Using boosting to improve a hybrid HMM / neural network speech recognizer. In *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 1999.

[12] G. Zweig, M. Padmanadhan. Boosting Gaussian mixtures in an LVCSR system. In *Proc. ICASSP*, 2000.

[13] C. Meyer. Utterance-level boosting of HMM speech recognizers. In *Proc. ICASSP*, 2002.

[14] S.W. Foo and L. Dong. A boosted multi-HMM classifier for recognition of visual speech elements. In *Proc. ICASSP*, 2003.

[15] P. Yin, I. Essa, J.M. Rehg. Asymmetrically boosted HMM for speech reading. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[16] C. Wang, W. Gao, and S. Shan. An approach based on phonemes to large vocabulary Chinese sign language recognition. In *Proc. FGR*, 2002, 411-416.

[17] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, vol. 77, no. 2, 1989, 257-285.

[18] R.E. Schapire. The boosting approach to machine learning: an overview. In *MSRI Workshop on Nonlinear Estimation and Classification*, 2002.

[19] Y. Freund and R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Science*, vol. 55, no. 1, 1997, 119–139.

[20] L.M. Arslan, J.H.L. Hansen. Selective training for hidden Markov models with application to speech recognition. *IEEE Trans. Speech and Audio Processing*, vol. 7, no. 1, 1999, 46-54.

[21] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss. The FERET evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, 2000, 1090–1104.