# Combining Classifiers based on Minimization of a Bayes Error Rate

Hee-Joong Kang and Seong-Whan Lee
Center for Artificial Vision Research, Korea University
Anam-dong, Seongbuk-ku, Seoul 136-701, Korea
E-mail: {hjkang, swlee}@image.korea.ac.kr

## Abstract

*In order to raise a class discrimination power by combining multiple classifiers, the upper bound of a Bayes error rate bounded by the conditional entropy of a class variable and decision variables should be minimized. Wang and Wong proposed a tree dependence approximation scheme of a high order probability distribution composed of those variables, based on minimizing the upper bound. In addition to that, this paper presents an extended approximation scheme dealing with higher order dependency. Multiple classifiers recognizing unconstrained handwritten numerals were combined by the proposed approximation scheme based on the minimization of the Bayes error rate, and the high recognition rates were obtained by them.*

## 1. Introduction

Previous methods for combining multiple classifiers at an abstract level are voting method[7], Behavior Knowledge Space (BKS) method[3], the use of a Dempster Shafer formalism[7], and the use of a Bayesian formalism with an independence assumption[7] or a dependency based approximation[4]. Particularly, combination of multiple classifiers using the probability theory and the Bayesian formalism is formulated as follows. When an input $x$ is given to $K$ classifiers (e.g., $C_1, \cdots, C_K$) in parallel, a $K$-dimensional decision vector $c =< C_1(x) = M_1, \cdots, C_K(x) = M_K >$ is observed, where $L$ classes are denoted by $M = \{M_1, \cdots, M_L\}$. The main task of this combination is to choose a hypothesized class $m$ which maximizes a posterior probability $P^*$ which is $\max_m P(m \in M | C_1(x) = M_1, \cdots, C_K(x) = M_K)$. That requires an estimation of a $(K+1)$st-order probability distribution.

In estimating such a high order probability distribution, most studies have assumed that classifiers' decisions are conditionally independent of each other for the given class[7]. However, Huang and Suen [3] assumed no independence of classifiers in their proposed BKS method, and directly computed such a high order probability from the accumulated BKS table. More recently, Kang and Kim [4] proposed a dependency based approximation for the estimation, using the measure of closeness proposed by Lewis[5].

Unlike the tree dependence method by Chow and Liu [1], Wang and Wong [6] supported that the use of dependence tree approximation that minimized the upper bound of the Bayes misclassification rate was more suitable for class discrimination, particularly when the classes were not well separable. They defined CP (classes-patterns) mutual information and proposed an algorithm for finding optimal tree structure of patterns. However, in this paper, CD (classes-decisions) mutual information is introduced in order to raise a class discrimination power by combining multiple classifiers.

The CD mutual information measures the dependence between the class $m$ and the decision vector $c$. The Bayes error rate is related to the definition of mutual information as shown in [2]. Optimal product approximation is based on the minimization of the Bayes error rate. This product approximation is proposed to extend the tree dependence of Wang and Wong to higher order dependency, and thus it optimally approximates the high order probability distributions with a product of low order distributions from first- to $n$th-order dependency. Then, classifiers are combined by Bayesian rules using the product approximations. The experiments were executed with CENPARMI standardized data set of unconstrained handwritten digits.

## 2. Product Approximation by Minimizing the Bayes Error Rate

Let $C$ be a decision vector variable whose element $C_j$ represents the $j$th classifier's decision and its realization $c$ be an $K$-dimensional decision vector. Let $M$

be a class variable whose value $M_j$ represents a class label. To make an estimation of the $(K+1)$st-order probability distribution, an optimal product approximation by $n$th-order dependency should be identified in the approximation scheme. For the identification phase, a criterion is needed to measure how much an approximate distribution contributes to minimizing the upper bound of the Bayes error rate $P_e$. Such a criterion depends on the CD mutual information $M(M;C)$, as the following expressions:

$$P_e \leq \frac{1}{2}(H(M) - M(M;C)) \quad (1)$$

$$M(M;C) = \sum_M \sum_C P(m,c) \log \frac{P(m,c)}{P(m)P(c)} \quad (2)$$

The CD mutual information is defined as the quantitative measure about how much the occurrence of a particular decision vector $c$ tells us about the possibility of a class member $m$. It is obvious that minimizing the Bayes error rate $P_e$ leads to maximizing the CD mutual information $M(M;C)$.

Wang and Wong optimally approximated an $n$th-order variable distribution by a product of $(n-1)$ second-order component distributions. Because they focused on only the first-order dependency, however, their method was not appropriate for considering the higher order dependency. In this paper, the authors propose the product approximation scheme based on the minimization of the Bayes error rate for the optimal product approximation of the $(K+1)$st-order probability distribution with a product set of $n$th-order dependencies where $1 \leq n \leq (K-1)$. This product approximation scheme is regarded as a natural extension of the first-order dependence tree method by Wang and Wong. For notation convenience, the authors will denote $C_j(x) = M_j$ and $m \in M$ by $C_j$ and $m$ in the following probabilistic terms respectively.

When first-order dependency is considered, the approximate distribution of $C$ is defined in terms of second-order distributions as follows:

$$P_a(C_1, \cdots, C_K) = \prod_{j=1}^{K} P(C_{n_j} | C_{n_{i(j)}}),$$

such that $(0 \leq i(j) < j)$ holds, and the approximate distribution of both $C$ and $M$ is defined in terms of third-order distributions as follows:

$$P_a(C_1, \cdots, C_K, m) = \prod_{j=1}^{K} P(C_{n_j} | C_{n_{i(j)}}, m),$$

$$P_a(C_1, \cdots, C_K | m) = \frac{1}{P(m)} \prod_{j=1}^{K} P(C_{n_j} | C_{n_{i(j)}}, m),$$

such that $(0 \leq i(j) < j)$ holds and $C_{n_j}$ is conditioned on both $C_{n_{i(j)}}$ and $m$, and where $(n_1, \cdots, n_K)$ is an unknown permutation of integers $(1, \cdots, K)$. And $P(C_{n_j}|C_0, m)$ is equal to $P(C_{n_j}, m)$, by definition. By the dependence tree method of Wang and Wong, the authors can determine both the permutation $(n_1, \cdots, n_K)$ and their conditioned permutation $(n_{i(1)}, \cdots, n_{i(K)})$. As a branch weight in the dependence tree, $\Delta$ first-order mutual information is defined by the following expression:

$$\Delta M(C_j; C_{i(j)}) = M(C_j; C_{i(j)}, m) - M(C_j; C_{i(j)}) \quad (3)$$

Therefore, the optimal dependence tree is a maximum-weighted tree that has maximum total sum of $\Delta$ first-order mutual information, i.e. $\sum_{j=1}^{K} \Delta M(C_j; C_{i(j)})$.

When second-order dependency is considered, the approximate distribution of $C$ is defined in terms of third-order distributions as follows:

$$P_a(C_1, \cdots, C_K) = \prod_{j=1}^{K} P(C_{n_j} | C_{n_{i2(j)}}, C_{n_{i1(j)}}), \quad (4)$$

such that $(0 \leq i2(j), i1(j) < j)$ holds, and the approximate distribution of both $C$ and $M$ is defined in terms of fourth-order distributions as follows:

$$P_a(C_1, \cdots, C_K, m) = \prod_{j=1}^{K} P(C_{n_j} | C_{n_{i2(j)}}, C_{n_{i1(j)}}, m),$$

$$P_a(C_1, \cdots, C_K | m) = \frac{1}{P(m)} \prod_{j=1}^{K} P(C_{n_j} | C_{n_{i2(j)}}, C_{n_{i1(j)}}, m), \quad (5)$$

such that $(0 \leq i2(j), i1(j) < j)$ holds and $C_{n_j}$ is conditioned on both $C_{n_{i2(j)}}$ and $C_{n_{i1(j)}}$ as well as $m$, and where $(n_1, \cdots, n_K)$ is an unknown permutation of integers $(1, \cdots, K)$. And $P(C_{n_j}|C_0, C_0, m)$ is equal to $P(C_{n_j}, m)$, and $P(C_{n_j}|C_0, C_{n_{i1(j)}}, m)$ is equal to $P(C_{n_j}|C_{n_{i1(j)}}, m)$, by definition. For notation convenience, the authors will drop the subscript $n$ and denote, for example, $C_{n_j}$ by $C_j$ in subsequent discussions.

For an optimal second-order dependency approximation $P_a$ by minimizing the Bayes error rate in expression (1), the authors apply both the $(K+1)$st-order probability distribution $P$ (i.e. expression (5))and the $K$th-order probability distribution $P$ (i.e. expression (4))to the definition of CD mutual information (i.e. expression (2)), as in the following expressions:

$$\begin{aligned} M(M;C) &= \sum_C \sum_M P(c,m) \log \frac{P(c|m)}{P(c)} \\ &= \sum_{M,C} P(c,m) \log [\frac{1}{P(m)} \prod_{j=1}^{K} P(C_j | C_{i2(j)}, C_{i1(j)}, m)] \\ &\quad - \sum_C P(c) \log \prod_{j=1}^{K} P(C_j | C_{i2(j)}, C_{i1(j)}) \\ &= -\sum_M P(m) \log P(m) + \sum_{j=1}^{K} \sum_{M,C} P(c,m) \log P(C_j | C_{i2(j)}, C_{i1(j)}, m) \end{aligned}$$

$$= -\sum_{j=1}^{K}\sum_{C} P(c)\log P(C_j|C_{i2(j)}, C_{i1(j)})$$

$$= H(M) + \sum_{j=1}^{K}[M(C_j; C_{i2(j)}, C_{i1(j)}, m) - M(C_j; C_{i2(j)}, C_{i1(j)})] \quad (6)$$

$$\Delta M(C_j; C_{i2(j)}, C_{i1(j)}) = M(C_j; C_{i2(j)}, C_{i1(j)}, m) - M(C_j; C_{i2(j)}, C_{i1(j)}) \quad (7)$$

From the above derived expression (6), maximizing $M(M;C)$ is to maximize $\sum_{j=1}^{K}\Delta M(C_j; C_{i2(j)}, C_{i1(j)})$ which is the total sum of $\Delta$ second-order mutual information, since remaining term $H(M)$ is constant. Then, the next step is how to identify an optimal product set of second-order dependencies from all the permissible product sets. The process of identifying the optimal product set of second-order dependencies is algorithmically described as follows.

*Algorithm for second-order dependency*

*Input:*
The set of $s$ samples $S^1, S^2, \cdots, S^s$.
*Output:*
The optimal set of second-order dependencies as per the $\Delta$ first- and $\Delta$ second-order mutual information.
*Method:*
1. Estimate the second-, third-, and fourth-order marginals from the various samples.
2. Compute the weights $\Delta M(C_j; C_{i(j)})$ and $\Delta M(C_j; C_{i2(j)}, C_{i1(j)})$ for all pairs, triplets, and quadruplets of classifiers from the samples.
3. Compute the maximum weight of first- and second-order dependencies and its associated optimal product set.

```
maxTweight = 0;
for n = 1 to number of first-order dependencies of
ΔM(C_j;C_i(j)) do
    Tweight = 0;
    choose one of first-order dependencies as a constraint;
    Tweight = weight of the chosen first-order;
    while (number of untraversed classifiers > 0) do
      choose one of untraversed classifiers;
      find one of permissible second-order dependencies of
      ΔM(C_j;C_i2(j),C_i1(j)) associated with the chosen classifier;
      Tweight += weight of the found second-order dependency;
    end
    maxTweight = MAX(maxTweight, Tweight);
    store maxTweight and its associated first- and second-order
    dependencies;
end
obtain maximum maxTweight and its associated first-
and second-order dependencies;
```

4. Submit them as the desired optimal set.

*End of Algorithm*

## 3. Bayesian Decision Rules using Product Approximations

The $K$ classifiers are combined by the identified optimal product set, using the Bayesian formalism. For each hypothesized class $M_i$, a supported belief function $Bel(M_i)$ is defined by the following expression:

$$Bel(M_i) = P(M_i|C_1, \cdots, C_K) \quad (8)$$

To make the estimation of the probability distribution $P$ feasible, Bayesian theorem and the optimal product set of first-order dependencies were applied to the definition of the belief function. Thus, the authors have the following formula from the expression (8):

$$Bel(m) = P(m|C_1, \cdots, C_K) = \frac{P(C_1, \cdots, C_K, m)}{P(C_1, \cdots, C_K)}$$

$$= \frac{\prod_{j=1}^{K} P(C_{n_j}|C_{n_{i(j)}}, m)}{P(C_1, \cdots, C_K)} \approx \eta \prod_{j=1}^{K} P(C_{n_j}|C_{n_{i(j)}}, m),$$

with $\eta$ as a constant that ensures that $\sum_{i=1}^{L} Bel(M_i) = 1$. And, $(n_1, \cdots, n_K)$ is an unknown permutation of integers $(1, \cdots, K)$. Depending on the belief value $Bel(M_i)$, the authors choose a maximized posterior probability $P^*(m|C_1, \cdots, C_K)$, and then a combined decision is determined or not, according to the decision rule $D(c)$ given below:

$$D(c) = \begin{cases} M_i, & \text{if } Bel(M_i) = \max\limits_{M_j \in M} Bel(M_j) \\ L+1, & \text{otherwise.} \end{cases}$$

If the optimal product set of second-order dependencies is considered for the higher order dependency, then the combining formula is defined as follows:

$$Bel(m) = P(m|C_1, \cdots, C_K) = \frac{\prod_{j=1}^{K} P(C_{n_j}|C_{n_{i2(j)}}, C_{n_{i1(j)}}, m)}{P(C_1, \cdots, C_K)}$$

$$\approx \eta \prod_{j=1}^{K} P(C_{n_j}|C_{n_{i2(j)}}, C_{n_{i1(j)}}, m),$$

with $\eta$ as a constant that ensures that $\sum_{i=1}^{L} Bel(M_i) = 1$. And, $(n_1, \cdots, n_K)$ is an unknown permutation of integers $(1, \cdots, K)$. The authors can also apply the above decision rule $D(c)$.

## 4. Experimental Results

In this section, the experimental results of combining classifiers *NN1*, *NN2*, *NN3*, *NN4*, *R1*, *R2* will be shown. The *reject* results of a classifier were not considered in identifying the optimal product set. The performance of individual classifiers on the test set T was presented in Table 1, showing their rejection rates.

**Table 1. Performance of individual classifiers**

| Classifier | Recognition rate | Rejection rate |
|---|---|---|
| *NN1* | 96.00 | 0.00 |
| *NN2* | 94.15 | 0.00 |
| *NN3* | 84.45 | 12.25 |
| *NN4* | 78.75 | 0.00 |
| *R1* | 88.15 | 10.40 |
| *R2* | 90.95 | 8.15 |

All experiments were conducted on the test set T using the following combination methods: majority voting, the BKS method, a Conditional Independence Assumption Based (CIAB) Bayesian method, some $n$th-Order Dependency Based ($n$ODB) Bayesian methods

**Table 2. Combination of five classifiers**

| Comb. method | *NN1+NN2+ NN3+NN4+R1* | | *NN1+NN2+ NN3+NN4+R2* | | *NN1+NN2+ NN3+R1+R2* | |
|---|---|---|---|---|---|---|
| | Rec. | Rej. | Rec. | Rej. | Rec. | Rej. |
| Voting | 97.20 | 1.35 | 97.50 | 1.05 | 97.40 | 1.60 |
| BKS | 91.15 | 8.30 | 91.75 | 8.00 | 92.90 | 6.75 |
| CIAB | 96.55 | 0.00 | 97.10 | 0.00 | 97.35 | 0.00 |
| FODB | 96.55 | 0.00 | 97.10 | 0.00 | 97.00 | 0.00 |
| CFODB | 97.45 | 0.00 | 97.55 | 0.00 | 98.00 | 0.00 |
| SODB | 97.70 | 0.00 | 97.55 | 0.00 | 97.80 | 0.00 |
| ΔFODB | 97.65 | 0.00 | 97.70 | 0.00 | 98.25 | 0.00 |
| ΔSODB | 97.75 | 0.00 | 97.90 | 0.00 | 97.90 | 0.00 |

| Comb. method | *NN1+NN2+ NN4+R1+R2* | | *NN1+NN3+ NN4+R1+R2* | | *NN2+NN3+ NN4+R1+R2* | |
|---|---|---|---|---|---|---|
| | Rec. | Rej. | Rec. | Rej. | Rec. | Rej. |
| Voting | 97.35 | 1.50 | 96.55 | 2.25 | 96.55 | 2.40 |
| BKS | 92.20 | 7.40 | 92.00 | 7.35 | 90.90 | 8.35 |
| CIAB | 97.35 | 0.00 | 97.55 | 0.00 | 97.20 | 0.00 |
| FODB | 97.10 | 0.00 | 97.20 | 0.00 | 96.90 | 0.00 |
| CFODB | 97.85 | 0.00 | 97.45 | 0.00 | 97.30 | 0.00 |
| SODB | 97.70 | 0.00 | 97.40 | 0.00 | 97.70 | 0.00 |
| ΔFODB | 97.80 | 0.00 | 97.65 | 0.00 | 97.65 | 0.00 |
| ΔSODB | 98.05 | 0.00 | 97.60 | 0.00 | 97.90 | 0.00 |

**Table 3. Combination of six classifiers**

| Combination method | *NN1+NN2+NN3+NN4+R1+R2* | |
|---|---|---|
| | Recognition rate | Rejection rate |
| Voting | 97.60 | 1.10 |
| BKS | 89.55 | 10.25 |
| CIAB Bayesian | 97.60 | 0.00 |
| FODB Bayesian | 97.10 | 0.00 |
| CFODB Bayesian | 97.85 | 0.00 |
| SODB Bayesian | 97.95 | 0.00 |
| ΔFODB Bayesian | 98.00 | 0.00 |
| ΔSODB Bayesian | 98.05 | 0.00 |

in [4], and the proposed Δ $n$th-Order Dependency Based (Δ$n$ODB) Bayesian methods. The best recognition rate in each case of combining five classifiers, as shown in Table 2, was obtained by a Δ First-Order Dependency Based (ΔFODB) or a Δ Second-Order Dependency Based (ΔSODB) Bayesian method. The best recognition rate in Table 3 for combining six classifiers was obtained by the ΔSODB Bayesian method.

From all the experimental results, the Δ$n$ODB Bayesian methods showed better performance than the other methods. In most cases, the recognition rates obtained by the second-order dependency approximations were higher than those by the first-order. The lowered recognition rates of the BKS method were caused by the lack of a large enough and well representative training data set. The experimental results supported that the new combination method, which incorporated the product approximation based on the minimization of the Bayes error rate into the Bayesian formalism, contributed to raising the class discrimination power by combining multiple classifiers, although it required larger storage needs than the $n$ODB Bayesian methods for computing the Δ $n$th-order mutual information.

## 5. Conclusion

A new combination method based on the minimization of the Bayes error rate was presented in this paper. Minimizing the Bayes error rate leads to both the optimal product approximation of high order probability distributions and the best recognition rates. This extends the work of Wang and Wong. The CD mutual information of high order dependency was defined and an algorithm for the second-order dependency using the Δ second-order mutual information was proposed in this paper. The experimental results supported that combining classifiers based on the minimization of the Bayes error rate showed better performance than the other combination methods.

## Acknowledgements

This work was supported by Creative Research Initiatives of the Korean Ministry of Science and Technology. The authors would like to thank Prof. J. H. Kim and Prof. I.-S. Oh for their help and suggestions during this study.

## References

[1] C. K. Chow and C. N. Liu. Approximating Discrete Probability Distributions with Dependence Trees. *IEEE Trans. on Inf. Tech.*, 14(3):462–467, 1968.

[2] M. E. Hellman and J. Raviv. Probability of error, equivocation, and the Chernoff bound. *IEEE Trans. on Information Theory*, IT-16:368–372, 1970.

[3] Y. S. Huang and C. Y. Suen. A Method of Combining Multiple Experts for the Recognition of Unconstrained Handwritten Numerals. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(1):90–94, 1995.

[4] H.-J. Kang, K. Kim, and J. H. Kim. Optimal Approximation of Discrete Probability Distribution with $k$th-order Dependency and Its Applications to Combining Multiple Classifiers. *Pattern Recognition Letters*, 18(6):515–523, 1997.

[5] P. M. Lewis. Approximating Probability Distributions to Reduce Storage Requirement. *Information and Control*, 2:214–225, Sep. 1959.

[6] D. C. C. Wang and A. K. C. Wong. Classification of Discrete Data with Feature Space Transform. *IEEE Trans. on Automatic Control*, AC-24(3):434–437, 1979.

[7] L. Xu, A. Krzyzak, and C. Y. Suen. Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition. *IEEE Trans. on Systems, Man, and Cybernetics*, 22(3):418–435, 1992.