Iterative Methods for Searching Optimal Classifier Combination Function

Sergey Tulyakov, Chaohong Wu and Venu Govindaraju

***Abstract*— Traditional classifier combination algorithms use either non-trainable combination functions or functions trained with the goal of better separation of genuine and impostor class matching scores. Both of these approaches are suboptimal if the system is intended to perform identification of the input among few enrolled classes or templates. In this work we propose training combination functions with the goal of minimizing the misclassification rate. The main idea of proposed methods is to use a set of best or strong impostors, and attempt to construct a classifier combination function separating genuine and best impostor matching scores. We have to use iterative methods for such training, since the set of best impostors depends on currently used combination function. We present two iterative methods for constructing combination functions and perform experiments on handwritten word recognizers and biometric matchers.**

I. Introduction

Most combination methods in biometric applications assume the system working in verification mode. In such mode, given M matching scores from M biometric matchers we have to determine whether current verification attempt is genuine (test and enrolled templates belong to the same person) or impostor (different persons). The ratio of genuine and impostor score likelihoods can be used as an optimum combination method for this problem [1]. On the other hand, if we consider a problem of identifying the person among N enrolled persons given biometrics, identification problem, the optimum combination function for such mode of operation can be different from likelihood ratio [1]. The goal of this paper is to present possible algorithms leading to the construction of optimal combination function for biometric systems operating in identification mode.

The problem of constructing the optimal combination function for systems operating in identification mode has not received proper attention thus far. Usually, same combination approaches are considered for both verification and identification modes. For example, [2] explicitly reduce the problem of identifying an individual among N enrolled persons to N separate verification problems. [3], [4] try to predict the performance of identification system from the performance of equivalent verification system. As our work [1] implies, such approaches might produce suboptimal combination algorithms or incorrect performance predictions for identification systems. Whereas ROC or DET curves are suitable for evaluating performance in verification systems, identification systems require using other performance measures - the correct identification rate or Cumulative Match Curves (CMC). In this paper we are using the correct identification rate, or the number of successful identification trials, for comparing experimental results.

The authors are with the Center for Unified Biometrics and Sensors (CUBS), University at Buffalo, USA tulyakov@cubs.buffalo.edu

Formally, we consider a problem of combining the outputs of M matchers or classifiers in order to achieve better classification performance. We assume that all classifiers produce sets of matching scores s_i^j assigned to each of N classes, and our combination methods will be operating on these scores. There are two general approaches to classifier combination problem. In one approach, a combination function f of scores is chosen or trained, and the classification result C is determined by the corresponding combination rule:

$$C = \arg \max_{i=1,\ldots,N} f(s_i^1, \ldots, s_i^M) \quad (1)$$

Note that in our notation the upper index of the score corresponds to the classifier, which produced this score, and lower index corresponds to the class for which it was produced. The names of combination rules are usually directly derived from the names of used combination functions: the sum function $f(s^1, \ldots, s^M) = s^1 + \cdots + s^M$ corresponds to the sum rule, the product function $f(s^1, \ldots, s^M) = s^1 \ldots s^M$ corresponds to the product rule and so on. The combination functions are usually fixed and some justification is given why a particular combination function is used [5]. Such approach also usually requires some normalization of used scores.

Another approach considers combination problem as a pattern classification problem with N classes in MN-dimensional score space. For example, [6] considers combination of handwritten digit ($N = 10$) classifiers with neural networks operating in $M * 10$-dimensional score space. If the number of classes N increases, the training of pattern classifiers becomes an obstacle. One solution is to convert scores to ranks and perform classification in this new space; Behavior-Knowledge Spaces [7] is one of these methods. But even such conversion is not sufficient if the number of classes is big or variable.

In this work we consider the combination of handwritten word recognizers and the combination of biometric person matchers. In both applications (see section with the description of experiments) the number of classes N is in the order of thousands, and the pattern classification approach becomes infeasible. Thus, we want to use a combination rule (1) with some combination function f and we are faced with the problem of searching for the optimal combination function. In terms of combination framework we presented in [8], we restrict ourselves to the combinations of low complexity type. So far there is no agreement among researchers on what the optimal combination function is, and depending

978-1-4244-1597-7/07/$25.00 ©2007 IEEE

on the assumptions on classifiers and their scores different combination function can be considered as optimal [5].

The paper presents two new algorithms for iterative training of combination function. We employ the heuristic reasoning that such function should be aiming at separating the scores assigned to genuine classes and the scores assigned to a set of somehow determined best impostor classes. We investigate these algorithms in contrast to the algorithms trying to have best separation between sets of genuine and all impostor scores. An example of such algorithm, likelihood ratio combination rule, might not give the best performance in our application.

II. Traditional Approaches

With the development of biometric field the new application of matching algorithms became important - minimizing the cost of verifying the hypothesis of whether the input belongs to the prespecified class. In particular, for biometric verification system we need to determine whether the presented biometric input belongs to the claimed enrolled person. The verification problem is a two-class problem - the input does belong to the hypothesis class (genuine verification attempt) or does not (impostor). On the other hand, the traditional classification problem still takes place in biometrics as an identification problem: given biometric input determine the person among N enrolled persons. Note, that similar task division existed before in other pattern recognition tasks. As an example of verification system in a handwriting application, a bank check recognition system might hypothesize about the value of the check based on the legal field, and numeric string recognition module must confirm that courtesy value coincides with the legal amount[9]. In identification mode a handwriting recognition module is used to identify each word between N words in the lexicon.

Verification problems have to separate only two classes - genuine and impostor matching scores in the M-dimensional score space. The optimal solution can be achieved by considering the ratio of genuine and impostor score density functions (likelihood ratio) and thresholding it [10]:

$$f_{lr}(s^1,\dots,s^M) = \frac{p_{gen}(s^1,\dots,s^M)}{p_{imp}(s^1,\dots,s^M)} \tag{2}$$

Thus likelihood ratio seems to be the first candidate for the optimal combination function in corresponding identification systems. As we will show next, this is not necessarily true, and we have to look for optimal combination function elsewhere.

A. Likelihood Ratio Combination Rule

Let us investigate whether the likelihood ratio combination function will be optimal for identification systems. Suppose we performed a match of the input sample by all M matchers against all N classes and obtained MN matching scores $\{s_i^j\}_{i=1,\dots,N;j=1,\dots,M}$. Assuming equal prior class probabilities, the Bayes decision theory states that in order to minimize the misclassification rate the sample should be classified as one with highest value of likelihood function $p(\{s_i^j\}_{i=1,\dots,N;j=1,\dots,M}|\omega_i)$. Thus, for any two classes ω_1 and ω_2 we have to classify input as ω_1 rather than ω_2 if

$$\begin{aligned} p(\{s_i^j\}_{i=1,\dots,N;j=1,\dots,M}|\omega_1) > \\ p(\{s_i^j\}_{i=1,\dots,N;j=1,\dots,M}|\omega_2) \end{aligned} \tag{3}$$

Let us make an assumption that the scores assigned to each class are sampled independently from scores assigned to other classes; scores assigned to genuine class are sampled from M-dimensional genuine score density, and scores assigned to impostor classes are sampled from M-dimensional impostor score density:

$$\begin{aligned} &p(\{s_i^j\}_{i=1,\dots,N;j=1,\dots,M}|\omega_i) \\ &= p(\{s_1^1,\dots,s_1^M\},\dots,\{s_{\omega_i}^1,\dots,s_{\omega_i}^M\},\dots, \\ &\{s_N^1,\dots,s_N^M\}|\omega_i) = p_{imp}(s_1^1,\dots,s_1^M)\dots \\ &p_{gen}(s_{\omega_i}^1,\dots,s_{\omega_i}^M)\dots p_{imp}(s_N^1,\dots,s_N^M) \end{aligned} \tag{4}$$

After substituting 4 into 3 and canceling out common factors we obtain the following inequality for accepting class ω_1 rather than ω_2:

$$\begin{aligned} p_{gen}(s_{\omega_1}^1,\dots,s_{\omega_1}^M)p_{imp}(s_{\omega_2}^1,\dots,s_{\omega_2}^M) > \\ p_{imp}(s_{\omega_1}^1,\dots,s_{\omega_1}^M)p_{gen}(s_{\omega_2}^1,\dots,s_{\omega_2}^M) \end{aligned}$$

or

$$\frac{p_{gen}(s_{\omega_1}^1,\dots,s_{\omega_1}^M)}{p_{imp}(s_{\omega_1}^1,\dots,s_{\omega_1}^M)} > \frac{p_{gen}(s_{\omega_2}^1,\dots,s_{\omega_2}^M)}{p_{imp}(s_{\omega_2}^1,\dots,s_{\omega_2}^M)} \tag{5}$$

The terms in each part of the above inequality are exactly the values of the likelihood ratio function f_{lr} taken at the sets of scores assigned to classes ω_1 and ω_2. Thus, the class maximizing the MN-dimensional likelihood function of inequality 3 is the same as a class maximizing the M-dimensional likelihood ratio function of inequality 5. The likelihood ratio combination rule is indeed the optimal combination rule under used assumptions.

The results of experiments in section IV show that the performance of this rule is not necessarily optimal for identification problems. Moreover, such combination can have worse performance than a performance of single matcher used in combination. This failure is caused by the incorrectness in assuming that the matching scores in each identification trial are independent. We discussed the reasons for this more deeply in [1].

Note, that two types of dependence between matching scores exist. The first type of dependence is the dependence between scores produced by different matchers and assigned to the same class: $s_\omega^1,\dots,s_\omega^M$. Most combination algorithms account for such dependence; by considering reconstructed M-dimensional densities in p_{gen} and p_{imp} in likelihood ratio function (2) we take care of this dependence as well. The second type of dependence is the dependence between scores produced by the same recognizer and assigned to different classes: $s_{\omega_1}^j,\dots,s_{\omega_N}^j$. The existence of this dependence is precisely the reason why likelihood ratio function fails for combinations in identification systems, and why we need to construct separate combination algorithms for verification and identification systems.

B. Weighted Sum Combination Rule

One of the frequently used rules in classifier combination problems is the weighted sum rule with combination function $f(s^1,\ldots,s^M) = w_1 s^1 + \cdots + w_M s^M$. The weights w_j can be chosen heuristically with the idea that better performing matchers should have bigger weight, or they can be trained to optimize some criteria. In our case we train the weights so that the number of successful identification trials on the training set is maximized. Since we have two matchers in all configurations, we use brute-force method: we calculate the correct identification rate of combination function $f(s^1,s^2) = ws^1 + (1-w)s^2$ for different values of $w \in [0,1]$, and find w corresponding to highest rate.

The results of testing weighted sum combination rule are presented in section IV. This combination rule has better performance than the performances of combined matchers for all datasets and does not show failures similar to likelihood ratio combination rule. This is expected due to specific nature of weighted sum rule training - we seek to maximize the number of successful identification trials. In the worst case we might get one of the weights to be 0, and the performance of such rule will be equivalent to the performance of a single matcher.

III. Estimating Optimal Combination Function for Identification Systems

The failure of likelihood ratio combination function suggests that the densities of genuine and impostor matching scores are of little help for finding optimal combination function, and might be useful only if the scores in identification trials are independent. For dependent scores we have to consider the scores in each identification trial as a single training sample, and train the combination function on these samples.

This was precisely the technique we used to train the weighted sum rule for identification systems in section II-B. For each training identification trial we checked whether the genuine score pair produced bigger combined scores than all impostor score pairs. By counting the numbers of successful trials we were able to choose the proper weights.

Though the weighted sum rule provides a reasonable performance in our applications, its decision surfaces are linear and might not completely separate generally nonlinear score distributions. We might want our combination function to be more complex, trained with available training set and possibly approaching ideal optimal function when the size of the training set is increased. In this section we present two ideas on learning such combination functions. Since we do not know the exact analytical form of optimal combination function, the presented combination methods are rather heuristic.

A. Learning Best Impostor Distribution

The likelihood ratio combination function of section II-A separates the set of genuine score pairs from the set of all impostor score pairs. But we might think that for identification systems it is more important to separate genuine score pairs from the best impostor score pairs obtained in each identification trial. There is a problem, though, that we do not know which score pair is the best impostor in each identification trial. The best impostor score pair can be defined as one having biggest combined score, but the combination function is unknown.

To deal with this problem we implemented an iterative algorithm, where the combination function is first randomly initialized and then updated depending on found best impostor score pairs. The combination rule is based on the likelihood ratio function with the impostor density trained only on the set of found best impostor score pairs. The exact algorithm is presented below:

1) Make initialization of $f(s^1,s^2) = \frac{\hat{p}_{gen}(s^1,s^2)}{\hat{p}_{imp}(s^1,s^2)}$ by selecting random impostor score pairs from each training identification trial for training $\hat{p}_{imp}(s^1,s^2)$.
2) For each training identification trial find the impostor score pair with biggest value of combined score according to currently trained $f(s^1,s^2)$.
3) Update $f(s^1,s^2)$ by replacing impostor score pair of this training identification trail with found best impostor score pair.
4) Repeat steps 2-3 for all training identification trials.
5) Repeat steps 2-4 for predetermined number of training epochs.

The algorithm converges fast - after 2-3 training epochs, and found best impostor score pairs change little in the subsequent iterations. The trained combination function subsequently gets tested using a separate testing set. Table I (BestImp LR method) provides the results of the experiments.

The method seems to perform well, but weighted sum combination rule is still better for word recognizers and biometric li&C matchers. This method is not able to fully account for the dependence of scores in identification trials, and the learning of the optimal combination function will not be probably achieved with it.

B. Sum of Logistic Functions

Generally, the matching score reflects the confidence of the match, and we can assume that if the score is bigger, then the confidence of the match is higher. When the scores are combined, the higher score should result in higher combination score. Thus, the combination function $f(s^1,s^2)$ should be monotonically nondecreasing in both of its arguments. One type of monotonic functions, which are frequently used in many areas, are logistic functions:

$$l(s^1,s^2) = \frac{1}{1+e^{-(\alpha_1 s^1 + \alpha_2 s^2 + \alpha_3)}}$$

If $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$, then $l(s^1,s^2)$ is monotonically nondecreasing in both of its arguments. Our goal is to approximate the optimal combination function as a sum of such logistic functions. The sum of monotonically nondecreasing functions will also be monotonically nondecreasing.

Suppose we have one identification trial and $\mathbf{s_1} = (s_1^1, s_1^2)$ and $\mathbf{s_2} = (s_2^1, s_2^2)$ are two score pairs of this trial. Let $\mathbf{s_1}$ be a

genuine score pair, and $\mathbf{s_2}$ be an impostor score pair. Suppose also that we have some initial sum of logistic functions as our combination function. If both matchers gave a higher score to the genuine class and $s_1^1 > s_2^1$ and $s_1^2 > s_2^2$, then by our construction the combination score for genuine class will be higher than the combination score for impostor class. There is no need to do any modifications to our current combination function. If both matchers gave a lower score to the genuine class and $s_1^1 < s_2^1$ and $s_1^2 < s_2^2$, then we can not do anything - any monotonically nondecreasing function will give a lower combination score to the genuine class.

If one matcher gave a higher score to the genuine class and another matcher gave a higher score to the impostor class, we can adjust our combination function by adding corresponding logistic function to the current sum. For example, if $s_1^1 > s_2^1$ and $s_1^2 < s_2^2$ logistic function $l(s^1, s^2) = \frac{1}{1+e^{-(\alpha_1 s^1 + \alpha_3)}}$ will be increasing with respect to the first argument and constant with respect to the second argument. The input sample will be assigned genuine class since first matcher correctly identified it. We choose parameters α_1 and α_3 relative to the training sample:

$$l(s^1, s^2) = \frac{1}{1 + e^{-\frac{1}{h}\frac{1}{a-b}(s^1 - \frac{a+b}{2})}} \tag{6}$$

where $a = s_1^1$ and $b = s_2^1$, and h is the smoothing parameter. If a and b are close to each other, we get a steeper logistic function, which will allow us better separate genuine and impostor score pair. Similar logistic function is added to the current sum if second matcher is correct, and first is not: we replace s^1 by s^2 in equation (6), and $a = s_1^2$,$b = s_2^2$.

The overall training algorithm is similar to the training we did for best impostor likelihood ratio in the previous section:

1) Make initialization $f(s^1, s^2) = s^1 + s^2$, $n = 1$.
2) For each training identification trial and for each impostor score pair in this trial check if its combined score is higher than combined score of the genuine pair.
3) Update $f(s^1, s^2)$ by adding described above logistic function: $f(s^1, s^2) = \frac{1}{n+1}(nf(s^1, s^2) + l(s^1, s^2))$, $n = n + 1$.
4) Repeat steps 2-3 for all training identification trials.
5) Repeat steps 2-4 for predetermined number of training epochs.

The smoothing parameter h is chosen so that the performance of the algorithm is maximized on the training set. The convergence of this algorithm is even faster than the convergence of the best impostor likelihood ratio algorithm. Table I (Log Sum method) presents correct identification rate for this method.

The method outperforms weighted sum method for both biometric combinations, but not for the combination of word recognizers. This suggests that our heuristic was quite good, but still can be improved somehow. We can also see that the advantage of this method for second biometric combination outweighs its disadvantage for the combination of word recognizers, and thus we can consider it as the best combination rule so far.

IV. Experiments

We have performed three sets of experiments for this paper - one for combining two word recognizers and two for combining fingerprint and face biometric matchers. The reason of using the word recognizer combination is that this combination problem is similar to combining biometric matchers. Also, used word recognition datasets deliver a good example of failure of likelihood ratio method. Even though used biometric datasets do not show this, the failure of likelihood ratio combination method surely can happen on other biometric databases.

Two handwritten word recognizers are Character Model Recognizer (CMR)[11] and Word Model Recognizer (WMR)[12]. Both recognizers employ similar approaches to word recognition: they oversegment the word images, match the combinations of segments to characters and derive a final matching score for each lexicon word as a function of character matching scores.

Our data consists of three sets of 2654, 1723 and 1770 word images representing UK postal town and county names of approximately same quality (the data was provided as these three subsets and we did not regroup them). The word recognizers were run on these images and their match scores for the total of 1681 lexicon words were saved. Since our data was already separated into three subsets, we used this structure for producing training and testing sets. Each experiment was repeated three times, each time one subset is used as a training set, and two other sets are used as test sets. Final results are derived as averages of these three training/testing phases.

For combinations of biometric matchers we used the biometric matching score set BSSR1 distributed by NIST[13]. This set contains matching scores for a fingerprint matcher and two face matchers 'C' and 'G'. Fingerprint matching scores are given for left index 'li' finger matches and right index 'ri' finger matches. In this work we used both face matching scores and fingerprint 'li' scores and we do two types of combinations: 'li'&'C' and 'li'&'G'. We used bigger subsets of this data set with 6000 identification attempts to create a set of virtual persons and their matching scores. After discarding enrollees and identification trials with failed biometric enrollment we obtained two equal sets - 2991 identification trials with 2997 enrolled persons with each part used as training and testing sets in two phases.

The densities for genuine and impostor scores in two likelihood ratio methods were approximated by Parzen window method with gaussian kernels. The kernel widths are calculated using maximum likelihood leave-one-out cross-validation method on the training sets. Since iterative methods showed fast convergence we set the total number of training epochs to be 20 in all cases, and the combination function obtained at the end was evaluated on the test sets.

Table I shows the performance of all considered algorithms on all three combination tasks. The given numbers are the numbers of correct identification trials. Note, that for word recognizers, each sample is used two times for testing, so

Matchers	LR	Weighted Sum	BestImp LR	Log Sum
CMR&WMR	4293	5015	4922	5005.5
li&C	5817	5816	5803	5823
li&G	5737	5711	5742	5753

TABLE I
CORRECT IDENTIFICATION RATE FOR ALL CONSIDERED COMBINATION METHODS.

we divided the totals by 2.

The performance of likelihood ratio (LR column) on biometric matchers seems to be satisfactory which can be explained by the relatively low dependence between scores in identification trials (.32 correlation between genuine and best impostor scores for 'li' and approximately .14 correlations for 'C' and 'G'). The performance of likelihood ratio on word recognizers is not satisfactory (the correlations between genuine and best impostor scores are .79 and .44 for WMR and CMR correspondingly).

The performance of a weighted sum rule is relatively high for all tasks. Basically, it can serve as a landmark for all other algorithms. It is actually best performing method for word recognizers.

The two proposed methods, best impostor likelihood ratio ('BestImp LR') and the sum of logistic functions ('Log Sum'), have good performance on biometric matchers and better than likelihood ratio performance for handwritten recognizers. The sum of logistic functions have good performance on all tasks.

V. CONCLUSION

In this paper we tried to underscore the difficulty in finding the optimal combination function for combination rules. With independence assumption on the sets of scores produced by matchers during single identification trial this function coincides with likelihood ratio function. But if this assumption does not hold, the optimal combination function is not evident. The two presented iterative methods might hold a key for finding the optimal combination function in all cases.

The main idea of both methods is to modify the trained combination function depending on its performance in each identification trial. Thus we do not consider separate sets of genuine and impostor scores as in traditional likelihood ratio method, but each genuine score is considered simultaneously with the corresponding set of impostor scores obtained in the same identification trial. It is possible that some traditional pattern classification methods could be trained in the similar way resulting in the optimal combination function.

REFERENCES

[1] S. Tulyakov, V. Govindaraju, and C. Wu, "Optimal classifier combination rules for verification and identification systems," in *7th International Workshop on Multiple Classifier Systems*, Prague, Czech Republic, 2007.

[2] Y. Lee, K. Lee, H. Jee, Y. Gil, W. Choi, D. Ahn, and S. Pan, "Fusion for multimodal biometric identification," in *Audio- and Video-Based Biometric Person Authentication*, 2005, pp. 1071–1079.

[3] P. Grother and P. Phillips, "Models of large population recognition performance," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2, 2004, pp. II–68–II–75 Vol.2.

[4] R. Bolle, J. Connell, S. Pankanti, N. Ratha, and A. Senior, "The relation between the roc curve and the cmc," in *Automatic Identification Advanced Technologies, 2005. Fourth IEEE Workshop on*, 2005, pp. 15–20.

[5] J. Kittler, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, pp. 226–239, March 1998.

[6] D.-S. Lee, *Theory of Classifier Combination: The Neural Network Approach*. SUNY at Buffalo: Ph.D Thesis, 1995.

[7] Y. Huang and C. Suen, "A Method of Combining Multiple Experts for Recognition of Unconstrained Handwritten Numerals," *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 17, no. 1, pp. 90–94, 1995.

[8] S. Tulyakov and V. Govindaraju, "Classifier combination types for biometric applications," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), Workshop on Biometrics*, New York, USA, 2006.

[9] G.Kim and V.Govindaraju, "Bank check recognition using cross validation between legal and courtesy amounts," *Int'l J. Pattern Recognition and Artificial Intelligence*, vol. 11, no. 4, pp. 657–674, 1997.

[10] S. Theodoridis and K. K., *Pattern Recognition*. Academic Press, 1999.

[11] J. Favata, "Character model word recognition," in *Fifth International Workshop on Frontiers in Handwriting Recognition*, Essex, England, 1996, pp. 437–440.

[12] G. Kim and V. Govindaraju, "A lexicon driven approach to handwritten word recognition for real-time applications," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 4, pp. 366–379, 1997.

[13] "Nist biometric scores set. http://www.nist.gov/biometricscores/."