Utilizing Independence of Multimodal Biometric Matchers

Sergey Tulyakov and Venu Govindaraju

Center for Unified Biometrics and Sensors (CUBS)
SUNY at Buffalo, USA

Abstract. The problem of combining biometric matchers for person verification can be viewed as a pattern classification problem, and any trainable pattern classification algorithm can be used for score combination. But biometric matchers of different modalities possess a property of the statistical independence of their output scores. In this work we investigate if utilizing this independence knowledge results in the improvement of the combination algorithm. We show both theoretically and experimentally that utilizing independence provides better approximation of score density functions, and results in combination improvement.

1 Introduction

The biometric verification problem can be approached as a classification problem with 2 classes: claimed identity is the true identity of the matched person (genuine event) and claimed identity is different from the true identity of the person (impostor event). During matching attempt usually a single matching score is available, and some thresholding is used to decide whether matching is a genuine or an impostor event.

If M biometric matchers are used, then a set of M matching scores is available to make a decision about match validity. This set of scores can be readily visualized as a point in M-dimensional score space. Consequently, the combination task is reduced to a 2-class classification problem with points in M-dimensional score space. Thus any generic pattern classification algorithm can be used to make decisions on whether the match is genuine or impostor. Neural networks, decision trees, SVMs were all successfully used for the purpose of combining matching scores.

If we use biometric matchers of different modalities (e.g. fingerprint and face recognizers) then we possess an important information about independence of matching scores. If generic pattern classification algorithms are used subsequently on these scores, the independence information is simply discarded. Is it possible to use the knowledge about score independence in combination and what benefits would be gained?

In this paper we will explore the utilization of the classifier independence information in the combination process. We assume that classifiers output a set of scores reflecting the confidences of input belonging to the corresponding class.

2 Previous Work

The assumption of classifiers independence is quite restrictive for pattern recognition field since the combined classifiers usually operate on the same input. Even when using

completely different features for different classifiers the scores can be dependent. For example, features can be similar and thus dependent, or image quality characteristic can influence the scores of the combined classifiers. Much of the effort in the classifier combination field has been devoted to dependent classifiers and most of the algorithms do not make any assumptions about classifier independence. Though independence assumption was used to justify some combination methods[1], such methods were mostly used to combine dependent classifiers.

One recent application where independence assumption holds is the combination of biometric matchers of different modalities. In the case of multimodal biometrics the inputs to different sensors are indeed independent (for example, there is no connection of fingerprint features to face features). The growth of biometric applications resulted in some works, e.g. [2], where independence assumption is used properly to combine multimodal biometric data.

We approach classifier combination problem from the perspective of machine learning. Biometric scores usually correspond to some distance measure between matched templates. In order to utilize the independence knowledge the scores should be somehow normalized before combination to correspond to some statistical variables, e.g. posterior class probability. Such normalization should be considered as a part of the combination algorithm, and the training of the normalization algorithm as a part of the training of the combination itself. Thus combination rule assuming classifier independence (such as product rule in [1]) requires training similar to any classification algorithm used as a combinator. The question is whether the use of independence assumption in combination rule gave us any advantage over using generic pattern classifier in a score space.

Our knowledge about classifier independence can be mathematically expressed in the following definition:

Definition 1. *Let index $j, 1 \leq j \leq M$ represent the index of classifier, and $i, 1 \leq i \leq N$ represent the index of class. Classifiers C_{j_1} and C_{j_2} are independent if for any class i the output scores $s_i^{j_1}$ and $s_i^{j_2}$ assigned by these classifiers to the class i are independent random variables. Specifically, the joint density of the classifiers' scores is the product of the densities of the scores of individual classifiers:*

$$p(s_i^{j_1}, s_i^{j_2}) = p(s_i^{j_1}) * p(s_i^{j_2})$$

Above formula represents an additional knowledge about classifiers, which can be used together with our training set.

Our goal is to investigate how combination methods can effectively use the independence information, and what performance gains can be achieved. In particular we investigate the performance of Bayesian classification rule using approximated score densities. If we did not have any knowledge about classifier independence, we would have performed the approximation of M-dimensional score densities by, say, M-dimensional kernels. The independence knowledge allows us to reconstruct 1-dimensional score densities of each classifier, and set the approximated M-dimensional density as a product of 1-dimensional ones. So, the question is how much benefit do we gain by considering the product of reconstructed 1-dimensional densities instead of direct reconstruction of M-dimensional score density.

In [4] we presented the results of utilizing independence information on assumed gaussian distributions of classifiers' scores. This paper repeats main results of those experiments in Section 4. The new developments presented in this paper are the theoretical analysis of the benefits of utilizing independence information with regards to Bayesian combination of classifiers (Section 3), and experiments with output scores of real biometric matchers (Section 5).

3 Combining Independent Classifiers with Density Functions

As we noted above, we are solving a combination problem with M independent 2-class classifiers. Each classifier j outputs a single score x_j representing the classifier's confidence of input being in class 1 rather than in class 2. Let us denote the density function of scores produced by the j-th classifier for elements of class i as $p_{ij}(x_j)$, the joint density of scores of all classifiers for elements of class i as $p_i(\mathbf{x})$, and the prior probability of class i as P_i. Let us denote the cost associated with misclassifying elements of class i as λ_i. Bayesian cost minimization rule results in the decision surface

$$f(\lambda_1, \lambda_2, \mathbf{x}) = \lambda_2 P_2 p_2(\mathbf{x}) - \lambda_1 P_1 p_1(\mathbf{x}) = 0 \tag{1}$$

In order to use this rule we have to learn M-dimensional score densities $p_1(\mathbf{x})$, $p_2(\mathbf{x})$ from the training data. In case of independent classifiers $p_i(\mathbf{x}) = \prod_j p_{ij}(x_j)$ and decision surfaces are described by the equation

$$\lambda_2 P_2 \prod_{j=1}^{M} p_{2j}(x_j) - \lambda_1 P_1 \prod_{j=1}^{M} p_{1j}(x_j) = 0 \tag{2}$$

To use the equation 2 for combining classifiers we need to learn $2M$ 1-dimensional probability density functions $p_{ij}(x_j)$ from the training samples. So, the question is whether we get any performance improvements when we use equation 2 for combination instead of equation 1. Below we will provide a theoretical justification for utilizing equation 2 instead of 1 and following sections will present some experimental results comparing both methods.

3.1 Asymptotic Properties of Density Reconstruction

Let us denote true one-dimensional densities as f_1 and f_2 and their approximations by Parzen kernel method as $\hat{f}_1$ and $\hat{f}_2$. Let us denote the approximation error functions as $\epsilon_1 = \hat{f}_1 - f_1$ and $\epsilon_2 = \hat{f}_2 - f_2$. Also let f_{12}, $\hat{f}_{12}$ and ϵ_{12} denote true two-dimensional density, its approximation and approximation error: $\epsilon_{12} = \hat{f}_{12} - f_{12}$.

We will use the mean integrated squared error in current investigation:

$$MISE(\hat{f}) = E\left(\int_{-\infty}^{\infty} (\hat{f} - f)^2(x) dx \right)$$

where expectation is taken over all possible training sets resulting in approximation $\hat{f}$. It is noted in [3] that for d-dimensional density approximations by kernel methods

$$MISE(\hat{f}) \sim n^{-\frac{2p}{2p+d}}$$

where n is the number of training samples used to obtain $\hat{f}$, p is the number of derivatives of f used in kernel approximations (f should be p times differentiable), and window size of the kernel is chosen optimally to minimize $MISE(\hat{f})$.

Thus approximating density f_{12} by two-dimensional kernel method results in asymptotic MISE estimate

$$MISE(\hat{f}_{12}) \sim n^{-\frac{2p}{2p+2}}$$

But for independent classifiers the true two-dimensional density f_{12} is the product of one-dimensional densities of each score: $f_{12} = f_1 * f_2$ and our algorithm presented in the previous sections approximated f_{12} as a product of approximations of one-dimensional approximations: $\hat{f}_1 * \hat{f}_2$. MISE of this approximations can be estimated as

$$\begin{aligned} MISE(\hat{f}_1 * \hat{f}_2) = E\bigg(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\big(\hat{f}_1(x) * \hat{f}_2(y) - f_1(x) * f_2(y)\big)^2 dxdy\bigg) = \\ E\bigg(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\big((f_1(x)+\epsilon_1(x)) * (f_2(y)+\epsilon_2(y)) - f_1(x) * f_2(y)\big)^2 dxdy\bigg) = \\ E\bigg(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\big(f_1(x)\epsilon_2(y) + f_2(y)\epsilon_1(x) + \epsilon_1(x)\epsilon_2(y)\big)^2 dxdy\bigg) \end{aligned} \quad (3)$$

By expanding power 2 under integral we get 6 terms and evaluate each one separately below. We additionally assume that $\int_{-\infty}^{\infty} f_i^2(x)dx$ is finite, which is satisfied if, for example, f_i are bounded (f_i are true score density functions). Also, note that $MISE(\hat{f}_i) = E\bigg(\int_{-\infty}^{\infty}(\hat{f}_i - f_i)^2(x)dx\bigg) = E\bigg(\int_{-\infty}^{\infty}(\epsilon_i)^2(x)dx\bigg) \sim n^{-\frac{2p}{2p+1}}$.

$$E\bigg(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_1^2(x)\epsilon_2^2(y)dxdy\bigg) = \int_{-\infty}^{\infty} f_1^2(x)dx * E\bigg(\int_{-\infty}^{\infty}\epsilon_2^2(y)dy\bigg) \sim n^{-\frac{2p}{2p+1}} \quad (4)$$

$$E\bigg(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_2^2(y)\epsilon_1^2(x)dxdy\bigg) = \int_{-\infty}^{\infty} f_2^2(y)dy * E\bigg(\int_{-\infty}^{\infty}\epsilon_1^2(x)dx\bigg) \sim n^{-\frac{2p}{2p+1}} \quad (5)$$

$$\begin{aligned} E\bigg(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_1(x)\epsilon_1(x)f_2(y)\epsilon_2(y)dxdy\bigg) = \\ E\bigg(\int_{-\infty}^{\infty} f_1(x)\epsilon_1(x)dx\bigg) * E\bigg(\int_{-\infty}^{\infty} f_2(y)\epsilon_2(y)dy\bigg) \\ \leq \sqrt{\int_{-\infty}^{\infty} f_1^2(x)dx}\sqrt{E\bigg(\int_{-\infty}^{\infty}\epsilon_1^2(x)dx\bigg)} \\ \times \sqrt{\int_{-\infty}^{\infty} f_2^2(y)dy}\sqrt{E\bigg(\int_{-\infty}^{\infty}\epsilon_2^2(y)dy\bigg)} \\ \sim \sqrt{n^{-\frac{2p}{2p+1}}}\sqrt{n^{-\frac{2p}{2p+1}}} = n^{-\frac{2p}{2p+1}} \end{aligned} \quad (6)$$

$$
\begin{aligned}
&E\left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_1(x)\epsilon_1(x)\epsilon_2^2(y)dxdy\right) = \\
&E\left(\int_{-\infty}^{\infty} f_1(x)\epsilon_1(x)dx\right) * E\left(\int_{-\infty}^{\infty} \epsilon_2^2(y)dy\right) \leq \\
&\sqrt{\int_{-\infty}^{\infty} f_1^2(x)dx}\sqrt{E\left(\int_{-\infty}^{\infty} \epsilon_1^2(x)dx\right)E\left(\int_{-\infty}^{\infty} \epsilon_2^2(y)dy\right)} \\
&\sim \sqrt{n^{-\frac{2p}{2p+1}}}n^{-\frac{2p}{2p+1}} = o\big(n^{-\frac{2p}{2p+1}}\big)
\end{aligned}
\tag{7}
$$

Similarly,

$$
E\left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \epsilon_1^2(x)f_1(x)\epsilon_2(y)dxdy\right) = o\big(n^{-\frac{2p}{2p+1}}\big) \tag{8}
$$

$$
\begin{aligned}
&E\left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \epsilon_1^2(x)\epsilon_2^2(y)dxdy\right) = \\
&E\left(\int_{-\infty}^{\infty} \epsilon_1^2(x)dx\right)E\left(\int_{-\infty}^{\infty} \epsilon_2^2(y)dy\right) = o\big(n^{-\frac{2p}{2p+1}}\big)
\end{aligned}
\tag{9}
$$

Thus we proved the following theorem:

Theorem 1 *If score densities of two independent classifiers f_1 and f_2 are p times differentiable and bounded, then the mean integrated squared error of their product approximation obtained by means of product of their separate approximations $MISE(\hat{f}_1 * \hat{f}_2) \sim n^{-\frac{2p}{2p+1}}$, whereas mean integrated squared error of their product approximation obtained by direct approximation of two-dimensional density $f_{12}(x,y) = f_1(x) * f_2(y)$ $MISE(\hat{f}_{12}) \sim n^{-\frac{2p}{2p+2}}$.*

Since asymptotically $n^{-\frac{2p}{2p+1}} < n^{-\frac{2p}{2p+2}}$, the theorem states that under specified conditions it is more beneficial to approximate one-dimensional densities for independent classifiers and use a product of approximations, instead of approximating two or more dimensional joint density by multi-dimensional kernels. This theorem partly explains our experimental results of the next section, where we show that 1d pdf method (density product) of classifier combination is superior to multi-dimensional Parzen kernel method of classifier combination. This theorem applies only to independent classifiers, where knowledge of independence is supplied separately from the training samples.

4 Experiment with Artificial Score Densities

In this section we summarize the experimental results previously presented in [4]. The experiments are performed for two normally distributed classes with means at (0,0) and (1,1) and different variance values (same for both classes). We used a relative combination added error, which is defined as a combination added error divided by the Bayesian error, as a performance measure. For example, table entry of 0.1 indicates that the combination added error is 10 times smaller than the Bayesian error. The combination added

error is defined as an added error of the classification algorithm used during combination [4].

The product of densities method is denoted here as '1d pdf'. The kernel density estimation method with normal kernel densities [5] is used for estimating one-dimensional score densities. We chose the least-square cross-validation method for finding a smoothing parameter. We employ kernel density estimation Matlab toolbox [6] for implementation of this method. For comparison we used generic classifiers provided in PRTools[7] toolbox. '2d pdf' is a method of direct approximation of 2-dimensional score densities by 2-dimensional Parzen kernels. SVM is a support vector machine with second order polynomial kernels, and NN is back-propagation trained feed-forward neural net classifier with one hidden layer of 3 nodes. For each setting we average results of 100 simulation runs and take it as the average added error. These average added errors are reported in the tables.

In the first experiment (Figure 1(a)) we tried to see what added errors different methods of classifier combination have relative to the properties of score distributions. Thus we varied the variances of the normal distributions (σ) which varied the minimum Bayesian error of classifiers. All classifiers in this experiment were trained on 300 training samples. In the second experiment (Figure 1(b)) we wanted to see the dependency of combination added error on the size of the training data. We fixed the variance to be 0.5 and performed training/error evaluating simulations for 30, 100 and 300 training samples.

σ	1d pdf	2d pdf	SVM	NN
0.2	1.0933	1.2554	0.2019	3.1569
0.3	0.1399	0.1743	0.0513	0.1415
0.4	0.0642	0.0794	0.0294	0.0648
0.5	0.0200	0.0515	0.0213	0.0967

(a)

Training size	1d pdf	2d pdf	SVM	NN
30	0.2158	0.2053	0.1203	0.1971
100	0.0621	0.0788	0.0486	0.0548
300	0.0200	0.0515	0.0213	0.0967

(b)

Fig. 1. The dependence of combination added error on the variance of score distributions (a) and the dependence of combination added error on the training data size (b).

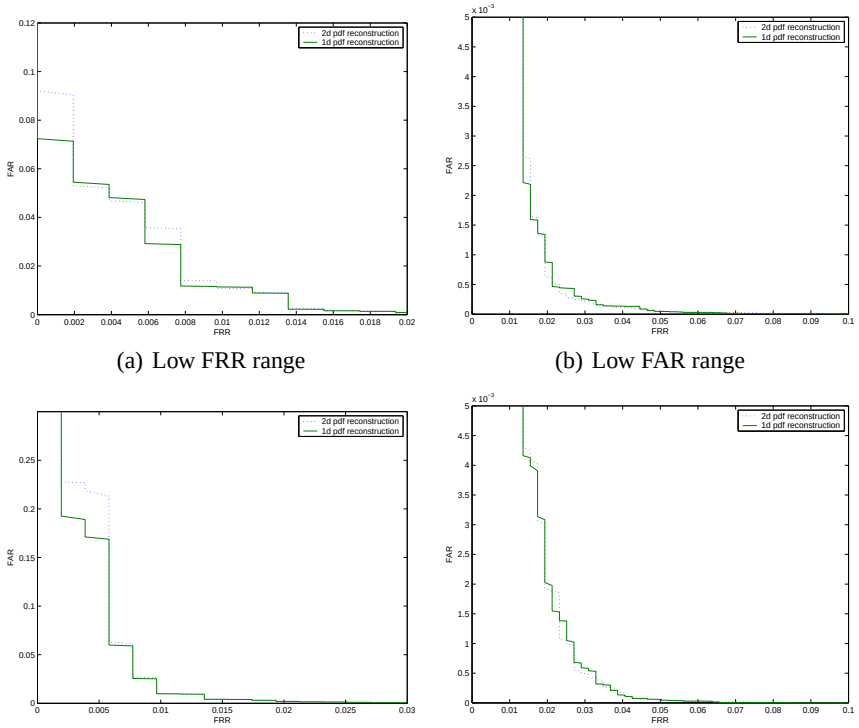
As expected, the added error diminishes with increased training data size. It seems that the 1d pdf method improves faster than other methods with increased training data size. This correlates with the asymptotic properties of density approximations of Section 3.1.

These experiments provide valuable observations on the impact of utilizing the knowledge of the score independence of two classifiers. The reported numbers are averages over 100 simulations of generating training data, training classifiers and combining them. Caution should be exercised when applying any conclusions to real life problems. The variation of performances of different combination methods over these simulations is quite large. There are many simulations where 'worse in average method' performed better than all other methods for a particular training set. Thus, in practice it is likely

that the method, we find best in terms of average error, is outperformed by some other method on a particular training set.

5 Experiment with Biometric Matching Scores

We performed experiments comparing performances of density approximation based combination algorithms (as in example 1) on biometric matching scores from BSSR1 set [8]. The results of these experiments are presented in Figure 2.



(a) Low FRR range (b) Low FAR range

(c) Low FRR range (d) Low FAR range

Fig. 2. ROC curves for BSSR1 fingerprint and face score combinations utilizing ('1d pdf reconstruction') and not utilizing ('2d pdf reconstruction') score independence assumption: (a), (b) BSSR1 fingerprint (li set) and face (C set); (c), (d) BSSR1 fingerprint (li set) and face (G set) .

In the graphs (a) and (b) we combine scores from the left index fingerprint matching (set li) and face (set C) matching. In graphs (c) and (d) we combine the same set of fingerprint scores and different set of face scores (set G). In both cases we have 517 pairs of genuine matching scores and 517*516 pairs of impostor matching scores. The experiments are conducted using leave-one-out procedure. For each user all scores for

this user (one identification attempt - 1 genuine and 516 impostor scores) are left out for testing and all other scores are used for training the combination algorithm (estimating densities of genuine and impostor matching scores). The scores of 'left out' user are then evaluated on the ratio of impostor and genuine densities providing test combination scores. All test combination scores (separately genuine and impostor) for all users are used to create the ROC curves. We use two graphs for each ROC curve in order to show more detail. The apparent 'jaggedness' of graphs is caused by individual genuine test samples - there are only 517 of them and most are in the region of low FAR and high FRR.

Graphs show we can not assert the superiority of any one combination method. Although the experiment with artificial densities shows that reconstructing one-dimensional densities and multiplying them instead of reconstructing two-dimensional densities results in better performing combination method on average, on this particular training set the performance of two methods is roughly the same. The asymptotic bound of Section 3 suggests that combining three or more independent classifiers might make utilizing independence information more valuable, but provided data set had only match scores for two independent classifiers.

6 Conclusion

The method for combining independent classifiers by multiplying one-dimensional densities shows slightly better performance than a comparable classification with approximated two-dimensional densities. Thus using the independence information can be beneficial for density based classifiers. The experimental results are justified by the asymptotic estimate of the density approximation error.

The knowledge about independence of the combined classifiers can also be incorporated into other generic classification methods used for combination, such as neural networks or SVMs. We expect that their performance can be similarly improved on multimodal biometric problems.

References

1. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. Pattern Analysis and Machine Intelligence, IEEE Transactions on **20** (1998) 226–239
2. Jain, A., Hong, L., Kulkarni, Y.: A multimodal biometric system using fingerprint, face and speech. In: AVBPA. (1999)
3. Hardle, W.: Smoothing Techniques with Implementation in S. Springer-Verlag (1990)
4. Tulyakov, S., Govindaraju, V.: Using independence assumption to improve multimodal biometric fusion. In: 6th International Workshop on Multiple Classifiers Systems (MCS2005), Monterey, USA, Springer (2005)
5. Silverman, B.W.: Density estimation for statistics and data analysis. Chapman and Hall, London (1986)
6. Beardah, C.C., Baxter, M.: The archaeological use of kernel density estimates. Internet Archaeology (1996)
7. Duin, R., Juszczak, P., Paclik, P., Pekalska, E., Ridder, D.d., Tax, D.: Prtools4, a matlab toolbox for pattern recognition (2004)
8. NIST: Biometric scores set. http://www.nist.gov/biometricscores/ (2004)