Classifier Combination – Statistical Approach

Sergey Tulyakov, Venu Govindaraju

Center for Unified Biometrics and Sensors,
University at Buffalo

Combination Field Overview

Approaches	Logic based	"Try them all"	Statistical
Description	-Assumptions on the meaning of combined data -The combination algorithm is a predetermined rule	Try few predetermined rules; choose one with best performance	The combination function is derived using training data and machine learning algorithms
Ease of use	Average	Easy	Difficult
Training data requirements	Low	Average	High
Optimality of combination	No	Somewhat	Yes

Example of logic based approach – Dempster-Shafer Theory

$\mathrm{P}(X)$ - the power set of X

$m : \mathrm{P}(X) \rightarrow [0,1]$ - basic belief assignment

Belief: $bel(A) = \sum_{B|B \subseteq A} m(B)$ Plausibility: $pl(A) = \sum_{B|B \cap A \neq \emptyset} m(B)$

Dempster's combination rule:

$$m_{1,2}(A) = (m_1 \oplus m_2)(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) m_2(C)$$

$$K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C)$$

Not optimal:

- basic belief assignments are heuristically chosen
- many other similar rules were proposed claiming superior performance
- assumes statistical independence of combined events

Example of "Try them all" approaches

Kittler et al., "On Combining Classifiers" , 1998:

- 6 rules are justified under different assumptions:
  - -Sum rule $S_i = f(s_i^1, \ldots, s_i^n) = s_i^1 + \ldots + s_i^n$
  - -Product rule $S_i = f(s_i^1, \ldots, s_i^n) = s_i^1 \times \ldots \times s_i^n$
  - -Max rule $S_i = f(s_i^1, \ldots, s_i^n) = \max(s_i^1, \ldots, s_i^n)$
  - -Min rule $S_i = f(s_i^1, \ldots, s_i^n) = \min(s_i^1, \ldots, s_i^n)$
  - -Median rule $S_i = f(s_i^1, \ldots, s_i^n) = median(s_i^1, \ldots, s_i^n)$
  - -Majority vote

$$S_i = f(s_i^1, \ldots, s_i^n) = \sum_j v_i^j, \quad v_i^j = \begin{cases} 1, \text{ if } s_i^j > s_i^k \ \forall k \\ 0, \text{otherwise} \end{cases}$$

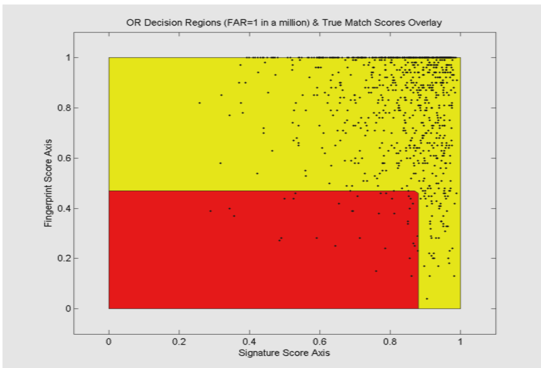
s_i^j - score assigned to class i by the classifier j

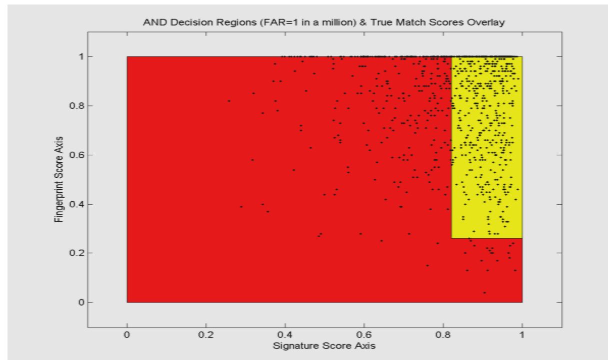
Somewhat optimal:

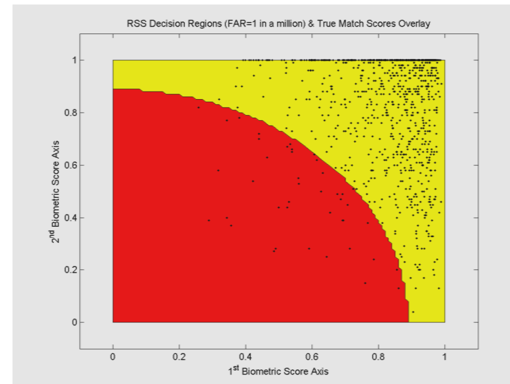
- choose best performing rule
- no confidence that chosen rule is close to optimal
- multiple published results show that different rules can be best in different problems

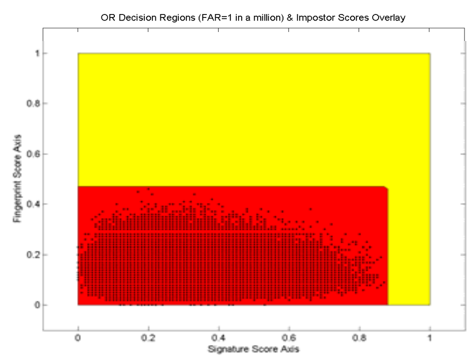
Example of "Try them all" approaches

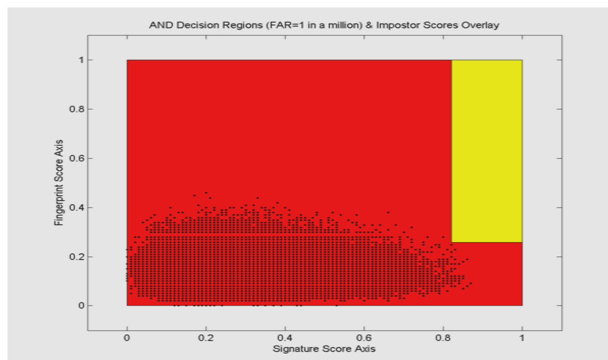
OR: 96.85% Accuracy | AND: 62.91% Accuracy | RMS: 96.11% Accuracy

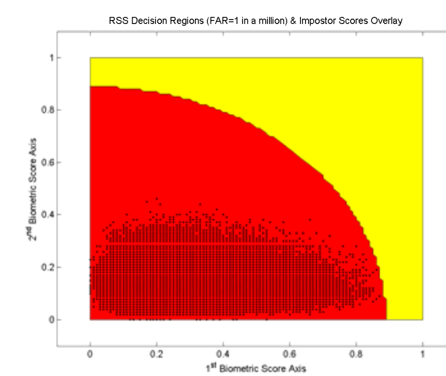








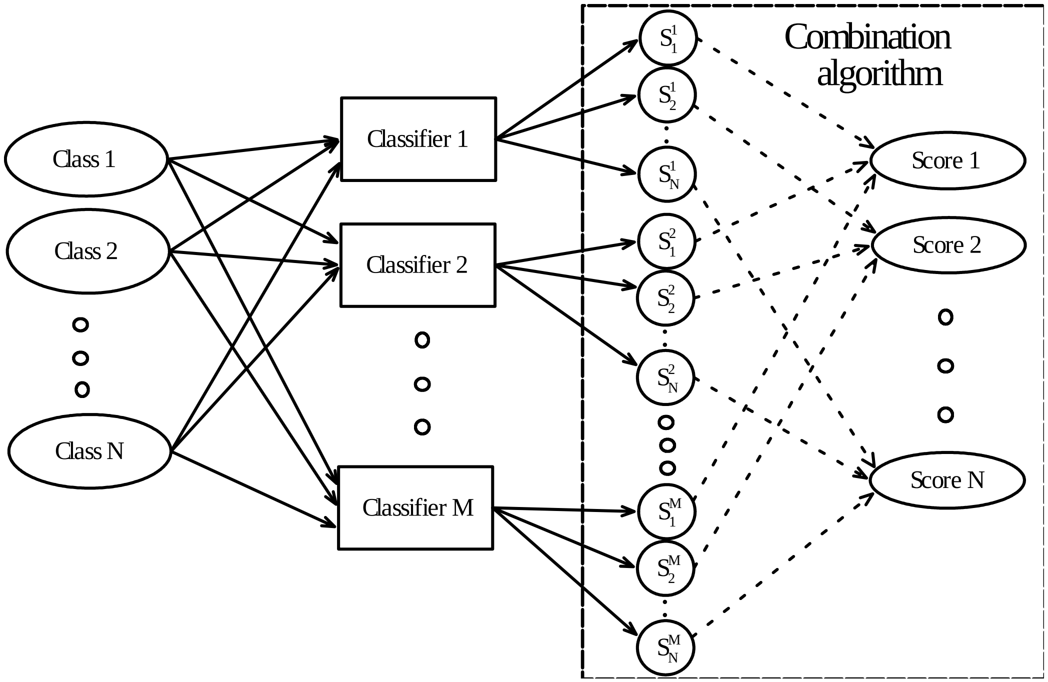




Match Zone *No-Match Zone*

Statistical approaches

- Combination problem - a problem of learning combination algorithm from training samples
- A set of learning algorithms is chosen with unknown parameters
- The best parameters are found with respect to the cost function and training data
- It is possible to give an estimate on the proximity of found solution to the optimal



Advantages:

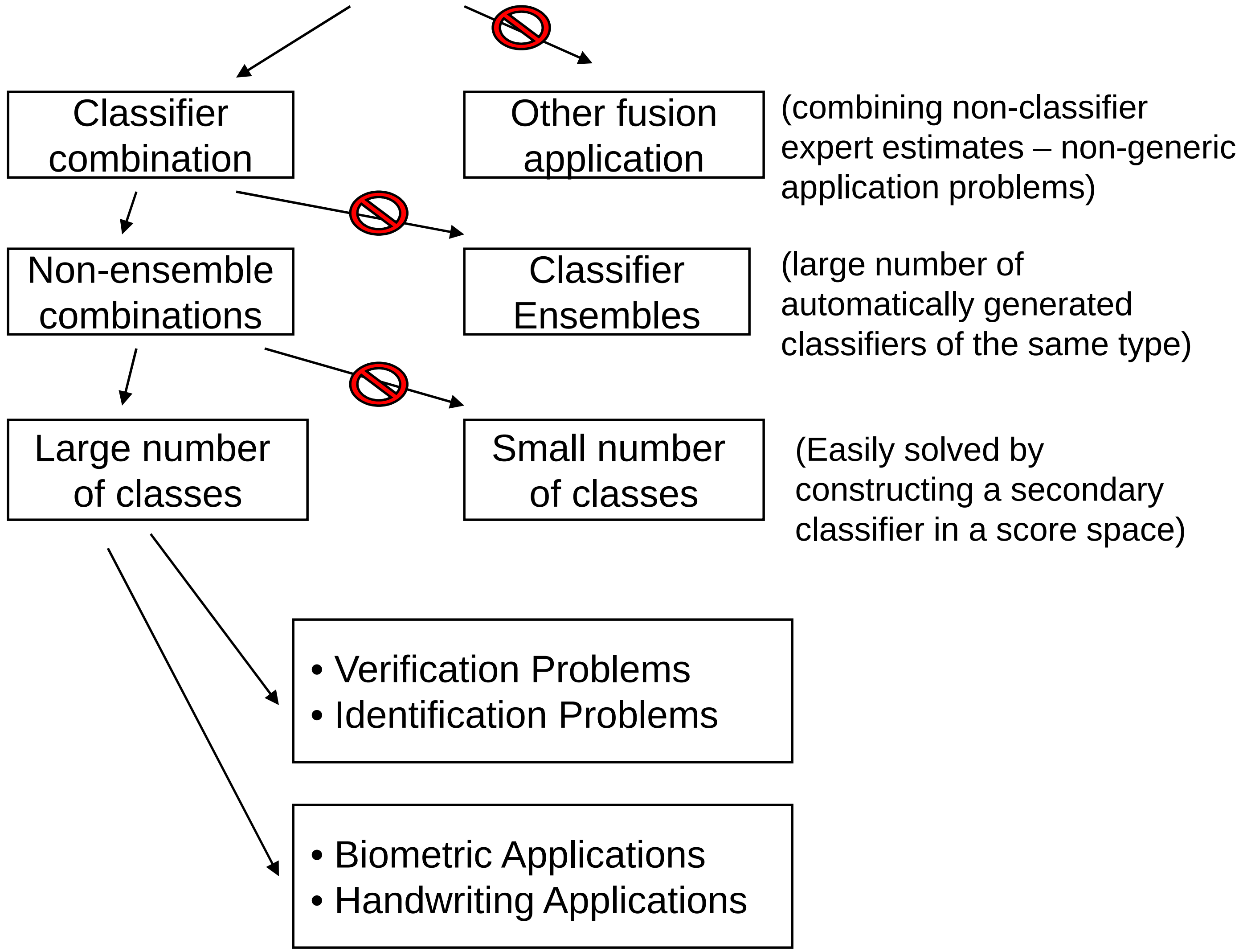
Universal approximation property guarantees the closeness to the optimal solution

But:

Need to properly choose cost function and avoid overfitting

cse@buffalo

Field of research

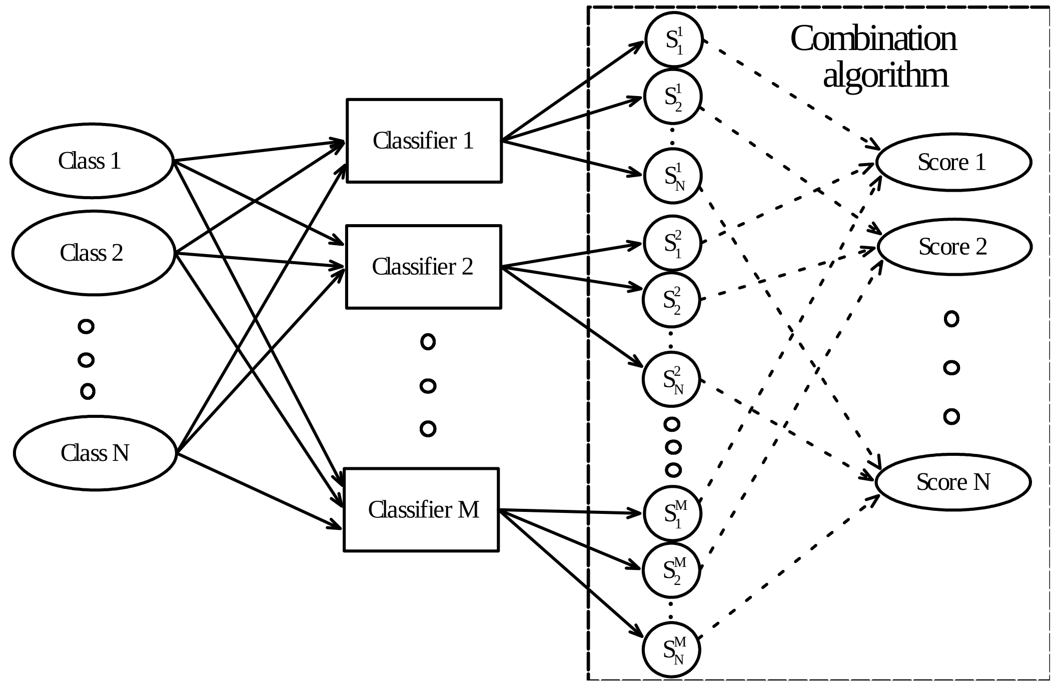


cse@buffalo

General statistical approach

• Most general combination algorithm is defined as map from (M classifiers)*(N classes) matching scores to N combined scores

$$S_i = f_i(\{s_j^k\}_{k,j})$$

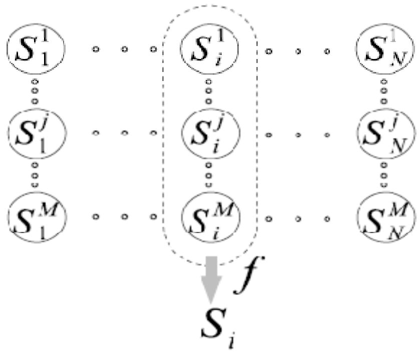


• Learning such combination algorithms is possible only if N and M are small

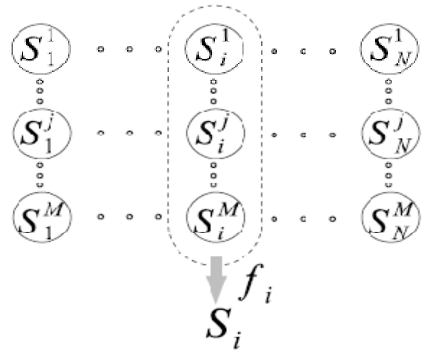
• Example (D. S. Lee): handwritten digit recognition (10 classes), 2 OCR algorithms, neural network with 10*2 inputs and 10 outputs

<u>Our problem</u>: number of classes N is much bigger (>1000), need different approach

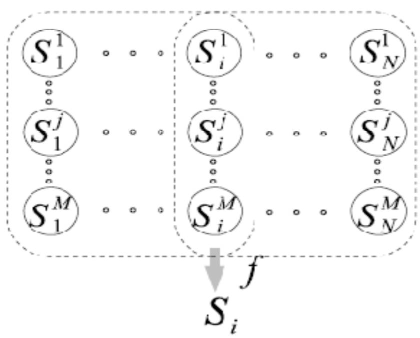
Research Result #1 – 4 Types of Combinations



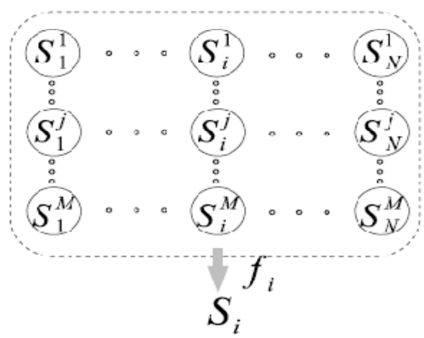
(a) Low



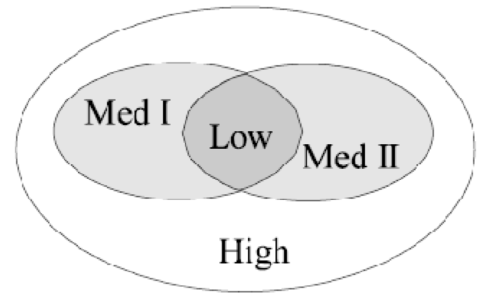
(b) Medium I



(c) Medium II



(d) High



M biometrics; N users

Low	$C_f(M)$
Medium I	$NC_f(M)$
Medium II	$C_f(NM)$
High	$NC_f(NM)$

$C_f(k)$ - complexity of the family of functions f accepting k dimensional input – VC (Vapnik-Chervonenkis) dimension [Tulyakov 06]

Research Result #1 – 4 Types of Combinations

Highlights:

- Theoretical proof that found combination types have different strengths:
  The optimal combination of lower complexity type might not achieve the same performance as sub-optimal combinations of higher complexity type
- The performance comparison of combinations of different type can be avoided
- Most existing combination techniques are of low complexity type and can be extended to higher complexity type
- Rank-based methods (e.g. Behavior-Knowledge Spaces) are usually of medium II complexity type, which explains their frequent superior performance (despite omitting raw score information)
- Need to search for efficient combinations of <u>higher complexity type</u> and operating on <u>raw matching scores</u>

cse@buffalo

Research Result #2 – Utilizing Score Set Statistics

$$\{s_1^j,\ldots,s_i^j,\ldots,s_N^j\}$$

(scores assigned to different classes by the same matcher during identification trial)

$$\{r_1^j,\ldots,r_i^j,\ldots,r_N^j\}$$

Rank-based methods convert scores to ranks; score information is lost

$$\{(s_1^j,r_1^j),\ldots,(s_i^j,r_i^j),\ldots,(s_N^j,r_N^j)\}$$

Composite score and rank-based methods are still not efficient

$$\{(s_1^j,t_1^j),\ldots,(s_i^j,t_i^j),\ldots,(s_N^j,t_N^j)\}$$

Our approach: instead of rank use more powerful score statistics

Examples of score statistics:

1. $t_i^j = r_i^j$ - Rank of score s_i^j among all scores $\{s_1^j,\ldots,s_i^j,\ldots,s_N^j\}$
2. $t_i^j = sbs_i^j$ - The best score besides s_i^j among all scores $\{s_1^j,\ldots,s_i^j,\ldots,s_N^j\}$
3. $t_i^j = \mu^j = \frac{1}{N}\sum_{i=1}^{N} s_i^j$ - sample mean $\quad t_i^j = \sigma^j = \left(\frac{1}{N-1}\sum_{i=1}^{N}(s_i^j-\mu^j)^2\right)^{1/2}$ - sample variance

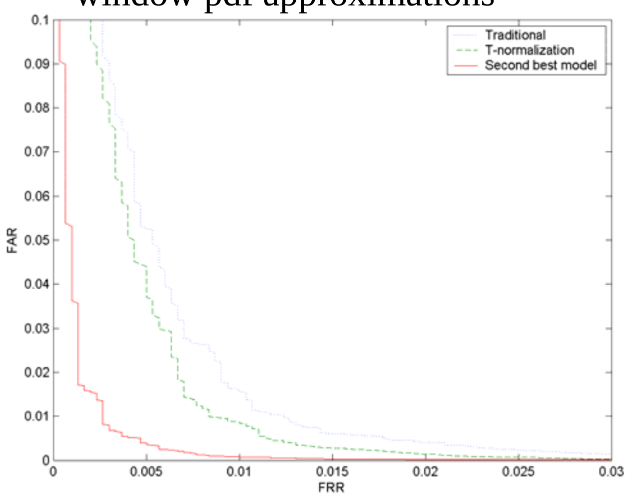
Research Result #2 – Utilizing Score Set Statistics

<u>Comparison of 3 methods:</u>

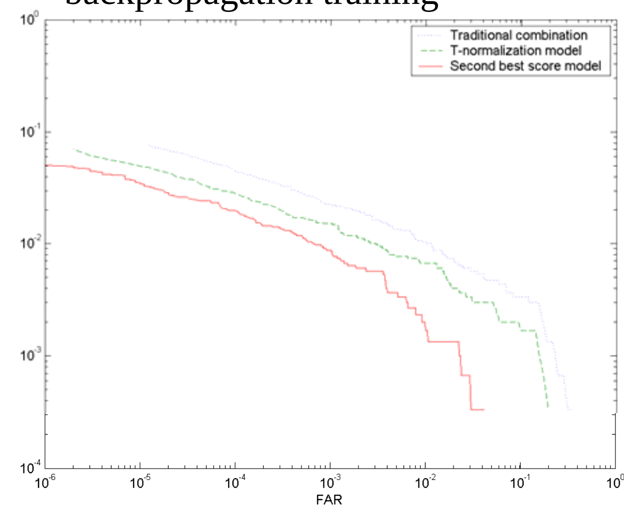
- *Traditional* - no score statistics; use only s_i^j
- *T-normalization*: use scores normalized with the help of μ^j and σ^j: $s_i^j \rightarrow \frac{s_i^j - \mu^j}{\sigma^j}$
- *Second best score model*: instead of s_i^j use pair (s_i^j, sbs_i^j)

Second best score model consistently provides better performance than T-normalization or traditional (no model) approaches (shown results on BSSR1 set, ‘li’&’C’)

Likelihood Ratio using Parzen window pdf approximations



Multilayer perceptron with backpropagation training



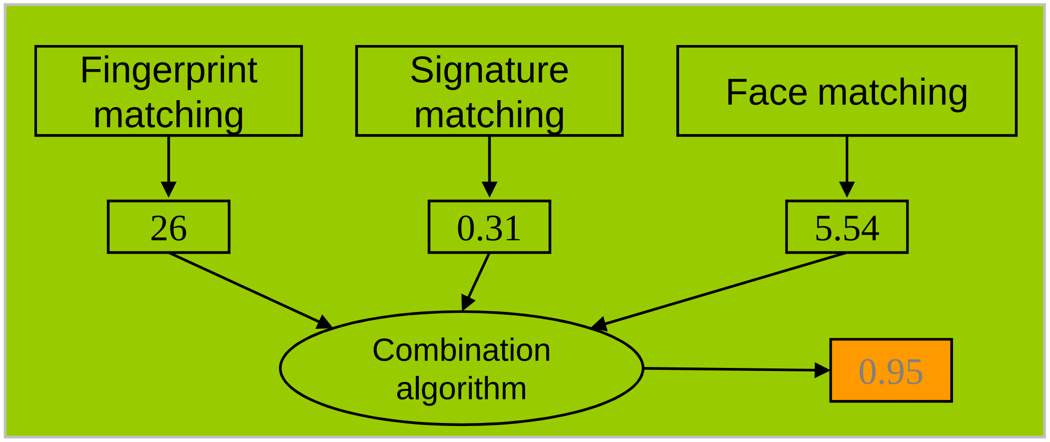
<u>Future research:</u>

- Use similar statistics for constructing medium I and high complexity combinations
- Other good statistics? Automatically finding useful statistics for a particular data?

Research Result #3 – Difference in Combinations for Verification and Identification Tasks

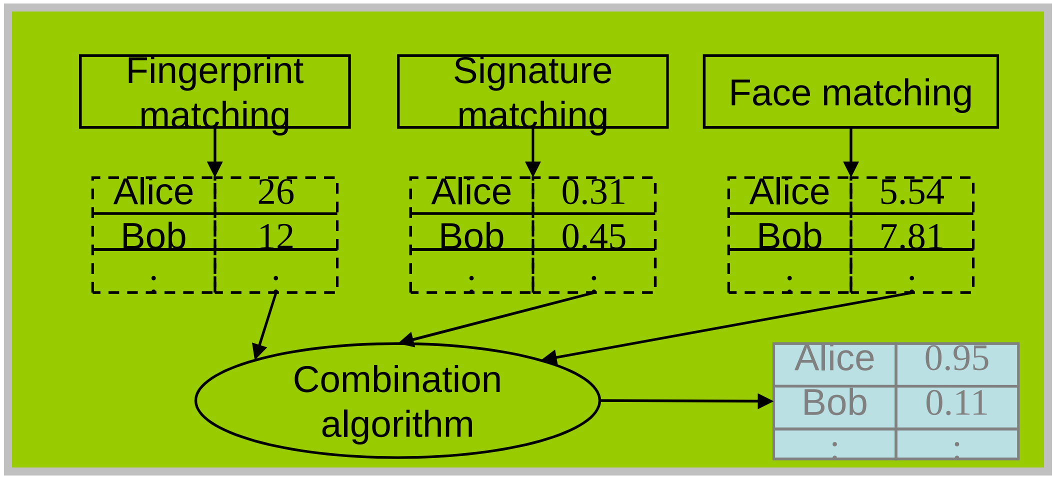
Verification Task:

- The combined score is tresholded to accept or reject verification hypothesis
- The optimization criteria: to minimize FRR for a given FAR
- Performance indicator: ROC curve



Identification Task:

- The class corresponding to the maximum of combined scores is chosen
- The optimization criteria: to maximize correct identification rate
- Performance indicator: correct identification rate, CMC curve



Research Result #3 – Difference in Combinations for Verification and Identification Tasks

Optimal combination algorithm:

Well-known

- Verification problem: *likelihood ratio* (ratio of genuine and impostor score densities)
- Identification problem:
  - likelihood ratio is optimal only under certain conditions (e.g. if scores assigned to different classes by the same matcher are statistically independent)
  - generally, such conditions do not hold and *likelihood ratio is not optimal*
  - likelihood ratio combination can have worse performance than a single matcher
  - it seems that it is not possible to analytically express optimal combination function for identification problem

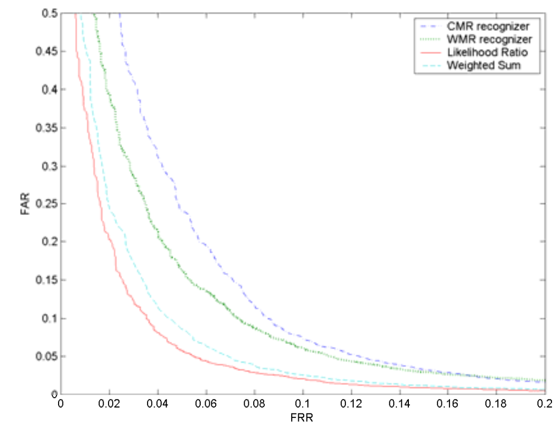
New results

(shown theoretically)

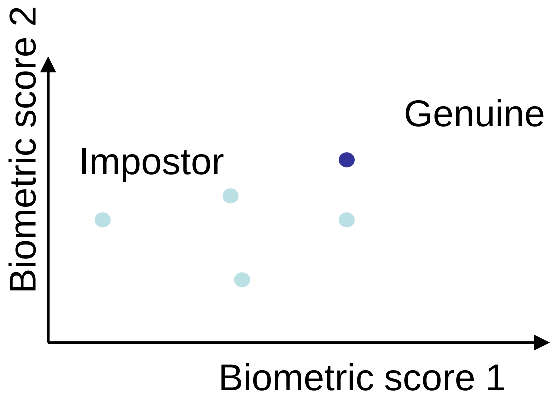
Example:

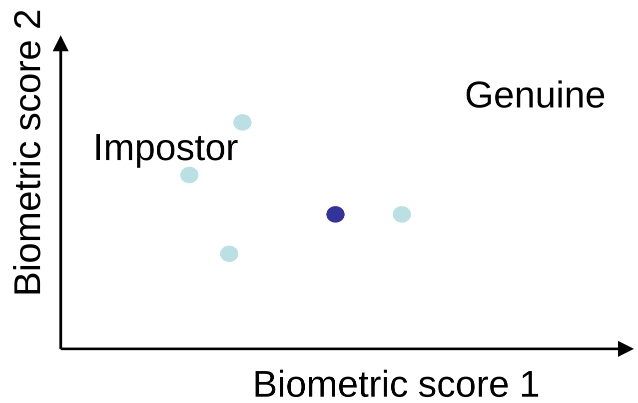
CMR is correct	54.8%
WMR is correct	77.2%
Both are correct	48.9%
Either is correct	83.0%
Likelihood Ratio	69.8%
Weighted Sum	81.6%

- Likelihood ratio combination has worse performance than a single matcher in identification mode;
- but it is superior to other methods in verification mode

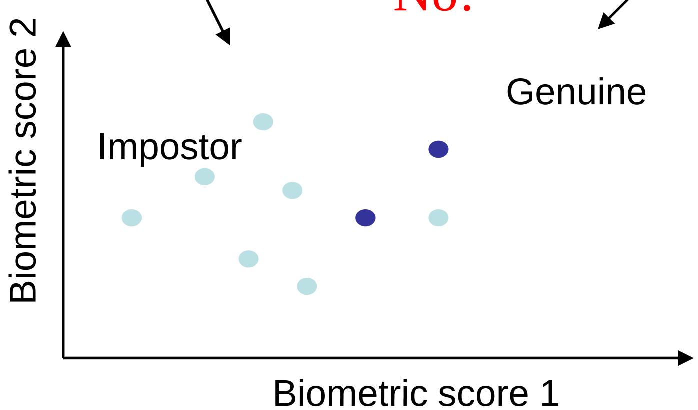


Research Result #4
– Iterative Methods for Finding Combination Algorithms in Identification Problems





No!



The training of the identification system combination should process scores from one identification trial as a <u>single training sample</u>.

Research Result #4 – Iterative Methods for Finding Combination Algorithms in Identification Problems

Ideas for proposed combination methods:

- Instead of using all impostor scores in identification trial use only single *best impostor score*
- *Best impostor score* can be determined using currently trained combination algorithm => iterative training

Considered approaches:

- Best impostor likelihood ratio
- Sum of logistic functions
- Neural networks utilizing best impostor scores

Some results:

	Likelihood Ratio	Weighted sum	Best Impostor Likelihood Ratio	Logistic Sum	Neural Network
CMR&WMR	4293	5015	4922	5005.5	5020.5
li & C	5817	5816	5803	5823	5826
li & G	5737	5711	5742	5753	5760

Future Research:

- Theoretically - still do not know if any of proposed algorithms is optimal
- Practically - need more experiments and possibly other algorithms

Research Result #5 – Utilizing Independence of Matchers

Observation:

- Multimodal biometric matchers produce statistically independent matching scores – can use this fact(?) for combination

Training set

$\{(s_1^g, s_2^g)_{k=1,\dots,K}\}$

$\{(s_1^i, s_2^i)_{l=1,\dots,L}\}$

⇒ Combination Algorithm ⇐

Independence Knowledge

$p_g(s_1^g, s_2^g) = p_g(s_1^g)\, p_g(s_2^g)$

$p_i(s_1^i, s_2^i) = p_i(s_1^i)\, p_i(s_2^i)$

- Instead of approximating 2-dimensional densities of scores $p_g(s_1^g, s_2^g), p_i(s_1^i, s_2^i)$ approximate 1-dimensional densities $p_g(s_1^g), p_g(s_2^g), p_i(s_1^i), p_i(s_2^i)$ and multiply them

Proved: Theorem: Product of approximations has the same order of error as individual 1-dimensional approximations. [Tulyakov 06]

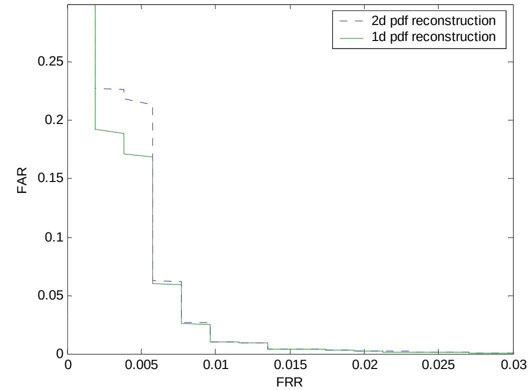
Research Result #5 – Utilizing Independence of Matchers

• Thus it is possible to improve the performance of combination algorithms by utilizing independence of matchers (better learning in low dimensions)
• Experimental results with likelihood ratio combination and Parzen window density approximations show that performance gains are rather small:

Num Train Samples	Not using	Using
30	0.205	0.216
100	0.079	0.062
300	0.051	0.020

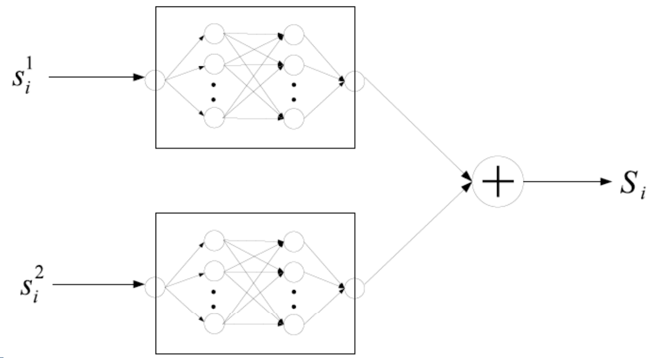
(Table shows averages of added error due to combination algorithm training over 100 runs)

•Simulated data
•Real data



Future Research:

Incorporate knowledge about classifier independence into other combination algorithms, e.g. neural network with special structure:



State of the Art?

- PAMI 1997 – Kittler et al., "On Combining Classifiers"
  - -attempt to justify different combination rules
  - - Our research: instead of combination rules use machine learning; it is only necessary to specify a complexity type of combination and optimization criteria
- PAMI 2005 – Snelick et al., "Large-Scale Evaluation of Multimodal Biometric Authentication Using State-of-the-Art Systems"
  - -"try them all" with adaptive normalization and user weighting
  - - not clear from paper if adaptive normalization results in medium II or high complexity combination type
  - - Our research: explicit use of matching score set statistics and differentiation between complexity types of combinations
- PAMI 2008 – Nandakumar et al., "Likelihood Ratio-Based Biometric Score Fusion"
  - -justify likelihood ratio combination method; use it with externally derived score quality measure
  - - Our research: the optimality of likelihood ratio method for verification problems is well-known, but it is not optimal for identification problems;
  - - score set statistics is a good alternative for externally derived quality measure

Conclusions

Theoretical results:

- 4 complexity types of classifier combinations defined by the amount of combined information and the number of trained combination functions
- Verification and identification problems require different optimal combination algorithms
- At least 4*2=8 optimal combination algorithms might exist for a single application
- Utilizing independence of combined matchers can be beneficial

Experimental results:

- Using second best score statistics delivers significant performance gains
- Iterative training can improve the performance of combination algorithms in identification problems
- Utilizing classifier independence delivers only small improvement

Considered methods provide an exhaustive framework for combination problems with a small number of classifiers and large number of classes