applied sciences

MDPI

Article

An Agent-Ensemble for Thresholded Multi-Target Classification

Nathan H. Parrish *, Ashley J. Llorens and Alex E. Driskell

The Johns Hopkins University Applied Physics Laboratory, 11100 Johns Hopkins Road, Laurel, MD 20723, USA; ashley.llorens@jhuapl.edu (A.J.L.); Alexander.Driskell@jhuapl.edu (A.E.D.)
* Correspondence: nathan.parrish@jhuapl.edu

Received: 7 January 2020; Accepted: 22 January 2020; Published: 18 February 2020

Abstract: We propose an ensemble approach for multi-target binary classification, where the target class breaks down into a disparate set of pre-defined target-types. The system goal is to maximize the probability of alerting on targets from any type while excluding background clutter. The agent-classifiers that make up the ensemble are binary classifiers trained to classify between one of the target-types vs. clutter. The agent ensemble approach offers several benefits for multi-target classification including straightforward in-situ tuning of the ensemble to drift in the target population and the ability to give an indication to a human operator of which target-type causes an alert. We propose a combination strategy that sums weighted likelihood ratios of the individual agent-classifiers, where the likelihood ratio is between the target-type for the agent vs. clutter. We show that this combination strategy is optimal under a conditionally non-discriminative assumption. We compare this combiner to the common strategy of selecting the maximum of the normalized agent-scores as the combiner score. We show experimentally that the proposed combiner gives excellent performance on the multi-target binary classification problems of pin-less verification of human faces and vehicle classification using acoustic signatures.

Keywords: score normalization; multi-agent systems; ensemble classification; pin-less verification; ground vehicle classification

1. Introduction

Multi-target binary classification problems occur in many sensor-based observation systems that must detect and alert on exemplars from a broad and disparate set of targets to the exclusion of background clutter. In many cases, the purpose of such a system is to alert or assist a human operator in identifying malicious or dangerous exemplars masked within a crowded clutter environment. In order to be useful to the operator, such systems must be able to correctly classify targets while maintaining a low probability of false alarm (PFA). Examples include chemical agent detection systems that must alert on a wide number of different harmful chemical plumes vs. benign ones [1], pin-less biometric verification systems that must correctly identify a group of persons of interest while excluding the general public [2] and malware and intrusion detection systems that must alert on different families of attacks to the exclusion of benign network traffic [3].

We address the multi-target binary classification problem by employing an ensemble of classification agents, each trained to classify between one of the pre-defined target-types and clutter. Perhaps the most straightforward approach to this problem is to train a single binary classifier that combines all target-types as a single target class. Our combination of agents approach provides several benefits over using a single classifier. First, it offers the potential to reduce the necessary complexity for any individual agent as compared to a single, multi-target classifier. Second, it simplifies the process of making in-situ adjustments to the relative importance of individual target-types (including adding or

deleting targets), an attribute that is often desirable in detection and classification applications. Third, it provides alignment between the algorithmic structure and a human operator's understanding of the target class breakdown, enhancing the ability of an operator to interact with the classifier by tuning it to reflect changes or drift in the target population. Finally, it enables the classifier to provide an indication of the most likely target-types, given an alert.

The ensemble of agents classifier approach requires a combination algorithm that, when presented with a test sample, fuses the outputs from the agents into a single target vs. clutter decision. The goal of the combiner is to maximize the probability of alerting on test samples from any of the target-types while maintaining a low PFA. Several previous works have used the maximum agent score as the combiner output [3–5]. These works have shown that normalization of the agent scores prior to choosing the maximum significantly impacts performance. An approach such as this was titled the *any-combiner* in [5], as it classifies a test example as target if the output score of any of the agents exceeds a threshold. However, the motivation and implementation of the any-combiner was driven by developer intuition and justified via empirical results, with little theoretical justification.

The main contribution of this paper is to provide a theoretically motivated combiner framework for the ensemble of agents approach that is equivalent to Neyman-Pearson optimal combination under a conditionally non-discriminative assumption on the agent scores. We provide a comparison of the proposed approach to other agent-classifier combination strategies including meta-classification and the any-combiner. Furthermore, we compare previously proposed agent-normalization strategies for the any-combiner to a new weighted likelihood ratio normalization that is motivated by the derivation of the proposed combiner. Finally, we perform agent-combination experiments using simulated data, face recognition data, and ground vehicle recognition data from acoustic recordings.

2. Problem Description

We consider the binary classification problem where the positive class consists of L different target-types and the negative class consists of background clutter. We assume that we have either trained or are provided with L different classification agents, each of which is trained to distinguish between one of the L different target-types and clutter. Specifically, we assume that training for agent j uses target-type j exemplars as the positive class, clutter exemplars as the negative class and does not use target exemplars from any target class i, $i \neq j$.

When presented with a test feature vector, $\mathbf{v}$, each agent j, $j = 1, ...L$ provides as its output a real-valued score, $x_j = g_j(\mathbf{v})$. The agent classification functions, $g_j(\mathbf{v})$, should be designed such that one can infer a degree of confidence that $\mathbf{v}$ belongs to target-type j vs. clutter from the output score. For example, the agents could be binary support vector machines or likelihood-based generative classifiers. We define the vector $\mathbf{x} = [x_1, x_2, ..., x_L]^{\mathrm{T}}$, where superscript T indicates matrix transpose, as the concatenated agent-scores for test sample $\mathbf{v}$.

We further assume that we are given a training dataset of score/label pairs $\{(\mathbf{x}_i, y_i)\}$, where $\mathbf{x}_i \in \mathbb{R}^L$ is the vector of agent score outputs and $y_i \in \{-1, 1, 2, ..., L\}$ is the corresponding class label, with -1 indicating clutter and $1, 2, ..., L$ indicating that the i^{th} sample is representative of the respective target-type. The purpose of this training data is to train a combiner function that takes the output of the L agents and produces a single score that we can threshold to classify the test sample as either target or clutter. We note that these training scores should be unbiased, i.e., computed via cross-validation or on a holdout set of data that is independent of the data used to train the agent classifiers.

A combiner for the agent-classifiers produces a label output $\hat{y}(\mathbf{x})$. Since the primary goal is to identify a test sample as target vs. clutter, the fusion combiner returns either $\hat{y}(\mathbf{x}) = 1$ for target or $\hat{y}(\mathbf{x}) = -1$ for clutter. In order to focus on the low-PFA region that is of interest in alert-driven automation systems, the learning objective for the combiner maximizes a partial area under the curve (PAUC) measure [5,6] that is limited to a low-PFA region of the ROC curve.

3. Related Work

There is a wide body of literature on classifier combiners. Kuncheva [7] breaks the literature down into two broad categories: classifier selection and classifier fusion. Of these two categories, classifier selection ensembles are the most closely related to the proposed ensemble of classifier agents. Classifier selectors are often based on the mixture of experts originally proposed by Jacobs et al. [8]. The expert classifiers are trained to perform well in a region of the input space, and thus a selection combiner chooses from one of the experts to classify each test sample based on the location of the test feature vector. Mixtures of experts have found wide use in a variety of applications [9–12]. The motivation for our agent-classifiers is similar to that for classifier experts; in each case, the constituent classifiers are specialized with respect to a particular subproblem. However, whereas experts are trained for a specific region of the input space, agents are tuned for a particular target-type. This difference makes a standard classifier selector inappropriate for agent combination.

Meta-classification, also known as classifier stacking, is another method of combining classifiers whereby the set of score/label pairs $\{(\mathbf{x}_i, y_i)\}$ is treated as a new set of classifier training data to which standard classifiers are applied [13–15]. We relate our proposed agent-combination method to meta-classification in Section 4.3.

Several authors have proposed ensembles of classifiers that are similar in spirit to the agents that we propose. Malisiewicz et al. proposed a mixture of exemplar-support vector machines (SVMs) for object classification in computer vision systems [4]. An exemplar-SVM is a linear SVM trained to distinguish one target training sample from clutter. This is an extreme version of what we refer to here as a classifier agent, where each target training sample serves as its own target-type. Kantchelaian et al. proposed an ensemble of agents approach for computer malware classification [3]. They specifically evaluated the advantages this approach has over traditional binary classifiers including responsiveness to changes in the target class and the semantic information it can provide to the operator. Additionally, in previous work, we proposed a multi-agent combination for target classification and a combiner algorithm that we called the *any-combiner* given as:

$$\hat{y}(\mathbf{x}) = \text{sign}\left(\max_{j\in 1,\ldots,L}\left(f(x_j)\right) - \alpha\right), \tag{1}$$

where $f(\cdot)$ is a normalizing function (we describe several examples in detail in Section 4.4). The any-combiner combination strategy, classifying a test sample as target if any of the constituent agents classifies it as target, was used as well in both [3,4]. However, none of these previous studies evaluated when or if this approach is optimal, or what the optimal normalization for the individual agents may be. We extend the results of these previous works by proposing a new combination strategy that is similar to the any-combiner and by providing theory on when this approach is optimal for combining the output of the agents.

A critical element of the any-combiner (1) and the new combiner that we propose in this paper is agent normalization. Biometrics researchers have studied classifier normalization extensively [16–19]. The problem studied in these previous works differs from that studied here in that they assumed that at test time a user would claim an identity, and the system would only evaluate the classifier associated with that user. Score normalization allowed the use of a single threshold for any claimed identity. The Z-norm [17], F-norm [16], equal error rate (EER) norm, and model-specific log-likelihood ratio (MS-LLR) norm [18] have all been investigated in such systems. In contrast, we assume that a test sample could belong to any one of the pre-defined target-types or clutter, requiring evaluation and combination of multiple agent scores. We discuss the effect of the previously proposed normalizations within the context of an agent-combiner in Section 4.4 and compare them in the experiments section.

Finally, we note that both a *max*-combiner and a *sum*-combiner were previously proposed in the classifier combination literature [20]. Although the any-combiner given in (1) and our new proposed combiner given in Section 4 contain a max and sum respectively, they are different than those previously

proposed. Kittler derives the previously proposed sum and product rules from an independence assumption on the underlying classifiers [20]. However, in Section 4.1 we show that our proposed combiner is not based on an independence assumption but rather a *conditionally non-discriminative* assumption. In fact, we believe that the independence assumption is inappropriate for combing agent-classifiers as the outputs are likely correlated, particularly if several of the targets are similar in feature space.

4. Proposed Agent Combiner

A block diagram of our proposed agent combiner is given in Figure 1. Each classification agent evaluates the test sample, $\mathbf{v}$, and provides a confidence score output indicating whether or not the test sample belongs to its target-type vs. clutter. The combiner normalizes the confidence scores using the weighted likelihood ratio function,

$$f_{wlr}(x_j) = p(Y = j|Y \geq 1)\frac{p(x_j|Y = j)}{p(x_j|Y = -1)}, \tag{2}$$

sums the normalized scores, and then classifies the sample as target if the resulting sum exceeds threshold α. The fusion algorithm is therefore given as

$$\hat{y}(\mathbf{x}) = \text{sign}\left(\sum_{j=1}^{L} f_{wlr}(x_j) - \alpha\right) \tag{3}$$

with α set to satisfy the PFA constraint.

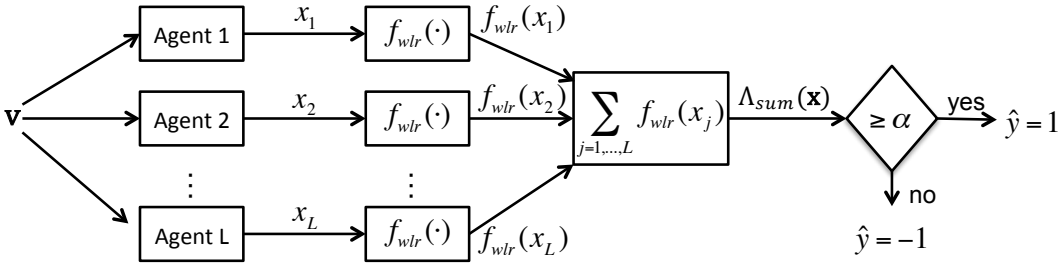

Figure 1. Block diagram of the proposed agent combiner.

4.1. Relationship to the Neyman-Pearson Combiner

We motivate the combination approach given in (3) by showing that under certain assumptions it is equivalent to the Neyman-Pearson optimal agent combiner. The Neyman-Pearson combiner for the agent score vector $\mathbf{x}$ is

$$\hat{y}(\mathbf{x}) = \text{sign}(\Lambda_{\text{NP}}(\mathbf{x}) - \alpha^*), \tag{4}$$

where the Neyman-Pearson fusion score, $\Lambda_{\text{NP}}(\mathbf{x})$, is given by the likelihood ratio:

$$\Lambda_{\text{NP}}(\mathbf{x}) = \frac{p(\mathbf{x}|Y \geq 1)}{p(\mathbf{x}|Y = -1)}, \tag{5}$$

and α^* is set in order to satisfy

$$P(\Lambda_{\text{NP}}(\mathbf{x}) - \alpha^* \geq 0|Y = -1) = \text{PFA}^*, \tag{6}$$

where PFA* is the allowable PFA for the system.

In Theorem 1 below, we show assumptions under which (3) and (4) are equivalent. First we give the following definitions for non-discriminative and conditionally non-discriminative random variables, which provide the basis for the assumptions of the theorem.

Definition 1. *A random variable Z is* non-discriminative *for target-type j if the conditional pdf of Z given j is the same as that for Z given clutter, $p(z|Y = j) = p(z|Y = -1)$.*

Definition 2. *A random variable Z_1 is* conditionally non-discriminative *for target-type j given random variable Z_2 if $p(z_1|z_2, Y = j) = p(z_1|z_2, Y = -1)$.*

Theorem 1. *If the agents not trained for target-type j are jointly conditionally non-discriminative for target-type j conditioned on the output of agent j for all $j \in 1, ..., L$, then (3) is equivalent to the optimal Neyman-Pearson combiner given in (4).*

Proof. The proof is a straightforward application of the total-probability theorem to the Neyman-Pearson Fusion score given in (5):

$$\begin{aligned}\Lambda_{\mathrm{NP}}(\mathbf{x}) &= \frac{p(\mathbf{x}|Y \geq 1)}{p(\mathbf{x}|Y = -1)} \\ &= \frac{\sum_{j=1}^{L} p(\mathbf{x}|Y = j)p(Y = j|Y \geq 1)}{p(\mathbf{x}|Y = -1)} \\ &= \sum_{j=1}^{L} \frac{p(\mathbf{x}_{\bar{j}}|x_j, Y = j)p(x_j|Y = j)p(Y = j|Y \geq 1)}{p(\mathbf{x}_{\bar{j}}|x_j, Y = -1)p(x_j|Y = -1)},\end{aligned}$$

where $\mathbf{x}_{\bar{j}}$ is the vector of agent-outputs with the j^{th} agent removed. If the conditionally non-discriminative assumption of the theorem is met, then $p(\mathbf{x}_{\bar{j}}|x_j, Y = j) = p(\mathbf{x}_{\bar{j}}|x_j, Y = -1)$ and thus

$$\Lambda_{NP}(\mathbf{x}) = \sum_{j=1}^{L} p(Y = j|Y \geq 1) \frac{p(x_j|Y = j)}{p(x_j|Y = -1)}, \tag{7}$$

which, when substituted into (4) results in the proposed agent combiner given in (3). □

There are several benefits to implementing (3) as opposed to (4) given that the assumptions of Theorem 1 are met. Implementation of (3) requires estimation of only single variable probability density functions (pdf), as opposed to (4) which requires that we estimate joint pdfs for target and clutter. The standard methods of simplifying or regularizing joint density estimation problems, such as assuming independence or diagonal loading of the covariance matrix of Gaussian distributions, are not appropriate for this problem as some of the underlying agents are likely to be correlated given that they are all functions of the same underlying feature vector $\mathbf{v}$. This is particularly true for agents trained on targets that are similar in feature space. Furthermore, the fact that (3) requires only one-dimensional pdfs makes it much easier to use non-parametric density estimation methods.

Theorem 1 requires the conditionally non-discriminative condition to hold for the agents in a multi-target classification ensemble. The following lemmas give two conditions on the underlying agent-classifiers under which the assumption holds.

Lemma 1. *Classifier agents $x_j, j \neq k$ are jointly conditionally non-discriminative for target-type k given $x_k = g_k(\mathbf{v})$ if $g_k(\mathbf{v})$ is the likelihood ratio for target-type k vs. clutter.*

Proof. If $g_k(\mathbf{v})$ is the likelihood ratio for target-type k vs clutter, then by definition

$$\frac{p(\mathbf{v}|Y = k)}{p(\mathbf{v}|Y = -1)} = \frac{p(x_k|Y = k)}{p(x_k|Y = -1)}. \tag{8}$$

Furthermore, we can write

$$\frac{p(\mathbf{v}|Y=k)}{p(\mathbf{v}|Y=-1)} \tag{9}$$

$$= \frac{p(\mathbf{v},\mathbf{x}|Y=k)}{p(\mathbf{v},\mathbf{x}|Y=-1)} \tag{10}$$

$$= \frac{p(\mathbf{v},\mathbf{x}_{\bar{k}}|x_k,Y=k)}{p(\mathbf{v},\mathbf{x}_{\bar{k}}|x_k,Y=-1)} \frac{p(x_k|Y=k)}{p(x_k|Y=-1)}, \tag{11}$$

where $\mathbf{x}_{\bar{k}}$ is the vector of agent-outputs with the k^{th} element removed. We go from (9) to (10) using the fact that the x's are functionally dependent on $\mathbf{v}$. Since (11) is equal to the right hand side of (8), the first fraction in (11) must equal one, indicating that $\mathbf{x}_{\bar{k}}$ and $\mathbf{v}$ are conditionally non-discriminative for target-type k given x_k. □

Lemma 2. *Assume that we have linear classifier agents, $g_j(\mathbf{v}) = \mathbf{h}_j^{\mathrm{T}}\mathbf{v}$, and define $\mathbf{h}_{j\perp k}$ as the component of $\mathbf{h}_j$ that is perpendicular to $\mathbf{h}_k$. If the $n_{j,k} = \mathbf{h}_{j\perp k}^{\mathrm{T}}\mathbf{v}$ are jointly non-discriminative for target-type k and are independent of x_k, then $\mathbf{x}_{\bar{k}}$ is conditionally non-discriminative for target-type k given agent k.*

Proof. Under the assumption of linear classifiers, we can write $x_j = c_{j,k}x_k + n_{j,k}$ where $c_{j,k} = \frac{\mathbf{h}_j^{\mathrm{T}}\mathbf{h}_k}{\|\mathbf{h}_j\|\|\mathbf{h}_k\|}$ and $n_{j,k}$ is defined as in the Lemma. Therefore, we can write

$$\frac{p(\mathbf{x}_{\bar{k}}|x_k,Y=k)}{p(\mathbf{x}_{\bar{k}}|x_k,Y=-1)} \tag{12}$$

$$= \frac{p(n_{1,k},n_{2,k},...,n_{L,k}|x_k,Y=k)}{p(n_{1,k},n_{2,k},...,n_{L,k}|x_k,Y=-1)} \tag{13}$$

$$= \frac{p(n_{1,k},n_{2,k},...,n_{L,k}|Y=k)}{p(n_{1,k},n_{2,k},...,n_{L,k}|Y=-1)} \tag{14}$$

$$= 1. \tag{15}$$

In the above, we go from (13) to (14) using the independence assumption of the Lemma, and we go from (14) to (15) using the non-discriminative assumption. □

Lemmas 1 and 2 show that the essential quality for conditionally non-discriminative agents is that any discriminative power that they may offer between a target-type and clutter is *redundant* to that provided by the agent that is trained for that target-type. In the case of Lemma 1 any function of $\mathbf{v}$ is redundant as x_k is a sufficient statistic for discrimination between target k and clutter. In the second case, the information given by knowledge of $x_j, j \neq k$ that is not already available through knowledge of x_k provides no additional discriminative power for target k. Although our agents may not meet these assumptions in practice, experiments show that the proposed combiner strategy works well for combining classifier agents.

4.2. Adapting to Target Population Drift

A challenge in multi-target classification systems is that the distribution among target-types is often not known a-priori when the system is initially trained. Furthermore, it may change over time or change based on system location, a situation referred to previously in the literature as *population drift* [21] or *adversarial drift* when the targets are a malicious class [3]. Mis-specification of the priors among target-types can greatly impact the performance of the system [21]. A benefit of the proposed approach is that it easily handles a changing distribution among the target-types by adjustment of prior, $p(Y = j|Y \geq 1)$, in the weighted likelihood ratio (2) (in this scenario, we do not assume the more general *concept drift* [22] where the label of the one of the targets changes to clutter, in which case the underlying agents would need to be retrained in order to exclude the former target-type). Furthermore, novel targets can be added in-situ without completely retraining the system

by adding a new agent trained for that target-type to the ensemble and adjusting the parameters of the normalization function appropriately.

4.3. Relationship to Meta-Classifiers

Meta-classifiers use the combiner training data to train a classifier for the score vector $\mathbf{x}$. Meta-classifiers can be either generative or discriminative. Generative meta-classifiers model the conditional distributions $p(\mathbf{x}|y)$. Therefore, the proposed agent-combiner can be viewed as a generative meta-classifier derived using the conditionally non-discriminative assumptions of Theorem 1.

If the conditionally non-discriminative assumption holds, then using a meta-classifier as opposed to the proposed approach offers no benefit as modeling the joint distribution of the training data can provide no additional discriminative power for the combiner. Even if this assumption does not hold, then we must be able to learn the model accurately from the training data. Thus, in practice, the simpler model that assumes agents are conditionally non-discriminative may perform better given limited training data, a phenomenon that we explore experimentally in Section 5.

Another benefit of the proposed combiner over meta-classifiers is the ability to easily adapt to changes in the target profile as described in the previous section. In order to adjust to any change in the target profile, a discriminative meta-classifier, such as a support vector machine, will need to re-train with a set of training data that is weighted or sampled to reflect the change [23]. This is also the case if an agent is added to or removed from the ensemble.

4.4. Relationship to the Any-Combiner Rule and a Comparison of Normalization Functions

If we replace the sum in (3) with a maximum, then we arrive at the any-combiner agent combination function that was originally proposed in [5]:

$$\hat{y}(\mathbf{x}) = \text{sign}\left(\max_{j=1,\ldots,L} f(\mathbf{x}_j) - \alpha\right). \tag{16}$$

The any-combiner given in (16) is more flexible than the proposed combiner in that it can use a number of different normalization functions. Several other score-normalization functions have been proposed in the biometrics literature for user verification. In user verification, a user claims an identity, and thus only one of the L agents is used at test time. Therefore, the complete ensemble is not evaluated at test time; however score-normalization is still useful for biometric verification so that each agent can be compared to a common threshold or so that the outputs of multiple biometric systems can be combined for different users in a consistent manner [18]. As these score normalization functions are still useful for our problem, we describe them here and evaluate their effectiveness in the experiments section.

Poh and Kittler [18] and Poh and Tistarelli [24] review score normalization functions and compare their use for biometric verification. Define $\hat{\mu}_j^{(-1)}$ and $\hat{\mu}_j^{(j)}$ as estimates of the mean value of x_j given clutter and target-type j, respectively, and define $\hat{\sigma}_j^{(-1)}$ and $\hat{\sigma}_j(j)$ as the estimated standard deviations. The normalization functions that [18,24] analyze are the Z-norm

$$f_Z(x_j) = \frac{x_j - \hat{\mu}_j^{(-1)}}{\hat{\sigma}_j^{(-1)}}, \tag{17}$$

F-norm

$$f_F(x_j) = \frac{x_j - \hat{\mu}_j^{(-1)}}{\hat{\mu}_j^{(j)} - \hat{\mu}_j^{(-1)}}, \tag{18}$$

MS-LLR norm (model-specific log-likelihood ratio)

$$f_{MSLLR}(x_j) = \log \frac{\hat{p}(x_j|Y=j)}{\hat{p}(x_j|Y=-1)}, \tag{19}$$

and EER-norm (equal error rate)

$$f_{EER}(x_j) = x_j - \Delta_j, \tag{20}$$

where Δ_j is an estimate of the score at which the probability of false positive equals the probability of false negative for agent j.

The Z-norm requires only clutter data for training, and we note that the Z-norm is a Gaussian parametric form of equal-PFA normalization proposed in [5]:

$$f_{EQPFA}(x_j) = 1 - \hat{\mathrm{F}}(x_j|Y=-1), \tag{21}$$

where the latter term in (21) is an estimate of the conditional cumulative distribution function of agent j's output given clutter. Both of these normalization functions are designed so that agents commit the same number of false alarms at any operating point. Thus, they can be motivated by a *fairness* perspective if one views the system-level PFA rate as a resource that is shared among the agents. However, these normalization functions can result in lower system-level probabilities of detection as they do not consider the difference in probability of detection among the different agents.

The F-norm, EER-norm and MS-LLR norm all use the target and clutter data to train the normalization functions. The F-norm, however, only uses the mean of the score distributions, and can thus perform poorly if the higher-order statistics of the score-outputs differ significantly. The EER norm considers second order statistics, but is designed only for a threshold of $\alpha = 0$. If this threshold is far from the desired operating point, then the EER norm will perform poorly. The MS-LLR norm is very similar to the proposed weighted likelihood ratio normalization; however, by not including the a-priori probabilities of the various target-types, this normalization will result in poor performance when the target sub-type distributions or risks are asymmetric.

An additional normalization function that has been used previously in the literature for SVM agents is Platt probabilistic normalization [4,5,25]. Platt probabilistic normalization uses a sigmoidal function to convert the unnormalized output of SVM classifiers into an estimate of the target-class posterior:

$$\begin{aligned} f_{Platt}(x_j) &= \hat{P}(Y=j|x_j) \\ &= \frac{1}{1+\exp(Ax_j+B)}, \end{aligned} \tag{22}$$

where A and B are parameters that are learned via a regularized maximum likelihood estimation using the class j training data vs. the clutter data. It is possible to use (22) directly in (16). We can also use Platt normalization in the proposed combiner (3) by relating (22) to the weighted likelihood normalization proposed in (2). We first rewrite the target posterior estimate as:

$$\hat{P}(Y=j|x_j) = \frac{\hat{p}(x_j|Y=j)\hat{P}(Y=j)}{\hat{p}(x_j)} \tag{23}$$

and

$$\hat{P}(Y=-1|x_j) = 1 - \hat{P}(Y=j|x_j) \tag{24}$$

$$= \frac{\hat{p}(x_j|Y=-1)\hat{P}(Y=-1)}{\hat{p}(x_j)}. \tag{25}$$

To eliminate the influence of the class prior probabilities in (23) and (25), we define n_j as the number of sub-type j training data and n_{-1} as the number of clutter training data that we use to learn (22). We can then relate (22) to (2) via

$$f_{wlr}(x_j) = P(Y = j|Y \geq 1)\frac{f_{Platt}(x_j)n_{-1}}{(1 - f_{Platt}(x_j))n_j}. \quad (26)$$

In practice, we have found that using (26) to estimate (2) for SVM agents results in better performance than estimating (2) via Gaussian distributions. This is likely due to the better fit of the sigmoidal function for the output of the SVM than a Gaussian parametric model as demonstrated in [25].

5. Experiments

In this section, we perform three experiments to demonstrate the effectiveness of our proposed agent combination strategy. We first perform an experiment with simulated data, then perform a pin-less verification experiment on face images using the Yale faces database, and finally perform an experiment where we classify tracked ground vehicles from wheeled based on their acoustic signature. In the two real-data experiments, we train the agent-classifiers using nonlinear Gaussian RBF SVMs implemented via LIBSVM [26].

In all experiments, the metric that we optimize is the partial area under the curve (PAUC) [5,6]. The PAUC differs from the AUC in that it computes the performance only in a specified region of the ROC curve. We believe that this metric is more appropriate than the AUC for sensor-based alert systems as we can tune the PAUC to focus on performance only in the low-PFA region where these systems operate. We use PAUC limited to the 0%-10% PFA range for all of our results. Additionally, in all experiments we perform statistical significance testing using the paired t-test with 95% confidence. Results that are the best, or statistically significantly tied for the best, are boldface in the results tables.

5.1. Simulated Data

We evaluate the effectiveness of the proposed combiner and the any-combiner with the different normalization strategies given in Section 4.4 vis-a-vis the conditionally non-discriminative condition defined in Section 4.1. We do this by simulating scores for agents that are distributed according to a multi-variate Gaussian distribution based on the label. We compare the following agent-combiners: 1. (*Prpsd. Gaussian WLR*) the proposed combiner given in (3) with a Gaussian model for the agent score distributions, 2. (*Joint Gaussian*) a jointly Gaussian generative meta-classifier that models the joint distribution of the agent scores conditioned on the label as $p(\mathbf{x}|Y = j) = \mathcal{N}(\mathbf{x}; \bar{}_j, \Sigma_j)$, 3. (*Ind. Gaussian*) an independent Gaussian generative meta-classifier that models the joint distribution of the agent scores conditioned on the label as $p(\mathbf{x}|Y = j) = \prod_{i=1}^{L} \mathcal{N}(x_i; \mu_{i,j}, \sigma_{i,j})$, 4. (*AC Gaussian WLR*) the any-combiner with weighted-likelihood normalization, 5. (*AC Z-norm*) the any-combiner with Z-normalization, 6. (*AC F-norm*) the any-combiner with F-normalization, and 7. (*AC EER-norm*) the any-combiner with EER-normalization where the EER rate is calculated assuming Gaussian densities as in [18]. Both the joint Gaussian and independent Gaussian combiners form a test statistic using the target vs. clutter likelihood ratio, that is,

$$\Lambda = \frac{\sum_{j=1}^{L} p(\mathbf{x}|Y = j)p(Y = j|Y \geq 1)}{p(\mathbf{x}|Y = -1)}.$$

We first evaluate the combiners when the conditionally non-discriminative assumption holds. We do this by simulating a ten-agent problem where the agent-outputs are independent and non-discriminative for targets for which they are not trained. We simulate a complex problem where the degree of difficulty for different targets varies significantly, a situation that has been shown to occur in practice [27]. Define $\mu_j^{(i)}$ and $\sigma_j^{(i)}$ as the mean and standard deviation of the j^{th} agent's output when $Y = i$. We first draw agent-score distribution parameters for clutter data, drawing $\mu_j^{(-1)}$ from a uniform $[-3 \ -1]$ distribution and $\sigma_j^{(-1)}$ from a uniform [0.5 1.25] distribution for each agent. Next,

we make each agent j non-discriminative for targets other than j by setting $\mu_j^{(i)}$ and $\sigma_j^{(i)}$, $\forall i \neq j, i \geq 1$ equal to the clutter parameters for agent j: $\mu_j^{(-1)}$ and $\sigma_j^{(-1)}$. Finally, we draw $\mu_j^{(j)}$ from a uniform [1 3] distribution and $\sigma_j^{(j)}$ from a uniform [0.5 1.25] distribution for each agent.

Figure 2 plots ROC curves in the low-PFA region for the ten agents on their specific target-type for one draw of the agent-score parameters. We generate ten-thousand test exemplars and a varying number of combiner training samples in order to evaluate the effectiveness of the combiners when they must estimate their parameters using either few or many combiner training samples. We set class-priors to 0.5 for clutter and 0.05 for each target-type. Due to the uniformity among the target-type priors the weighting in the proposed weighted-likelihood ratio normalization is irrelevant and thus the results compare the performance of the normalizations without considering the weighting. We repeat the experiment fifty times, generating new agent-output distributions and data for each iteration.

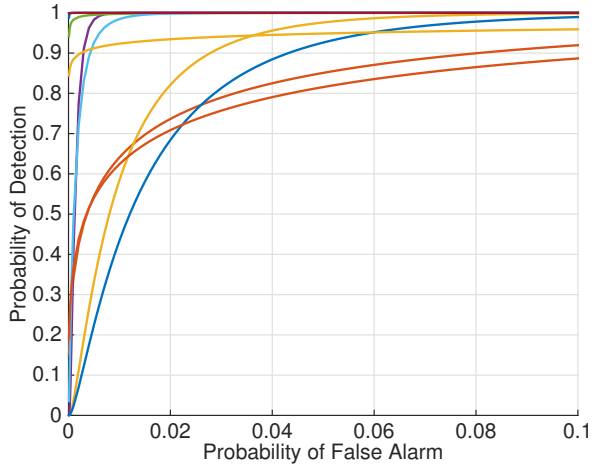

Figure 2. Agent ROC curves for the first simulation in Section 5.1. Each line shows a ROC curve for one of the 10 random agents. The ROC curves show that the simulation mimics a scenario where some targets are much harder to distinguish from clutter than others.

Table 1 gives the results averaged over the fifty experiments for each number of combiner training samples. The rightmost column gives an upper bound on the performance of each combiner when provided with the true parameters, and the joint Gaussian, independent Gaussian, and proposed method all perform the best when provided with the true parameters. However, when the combiners must estimate the parameters from data, the joint and independent Gaussian combiners perform worse in all cases than the proposed method. The proposed method is the best or statistically significantly tied for the best with 95% confidence at all numbers of training samples. At low numbers of training samples, the any-combiner with Z-normalization is statistically significantly tied for the best with the proposed method. The results show that, given limited training data, the performance of the proposed combiner greatly exceeds that of the joint combiner as well as the combiner based on an independence assumption. We can also see that the weighted likelihood normalization outperforms the normalization methods given in Section 4.4. However, we note that the performance of the different normalization methods will depend on a variety of factors including the number of combiner training samples and how well the agent distributions match the assumptions used to derive the normalization methods. Regularization techniques for estimating the model parameters can also improve the performance when there are few training samples [18].

Table 1. Average partial area under the curve (PAUC) for the first simulation with conditionally non-discriminative agents. Columns give the results for differing number of combiner training samples, and the column titled 'model' gives the results when the combiners are provided with the model parameters. Boldface identifies results that are the best or statistically significantly tied for the best with 95% confidence.

	100 Samples	**1000 Samples**	**2000 Samples**	**10,000 Samples**	**Model Params**
Prpsd. Gaussian WLR	**0.813**	**0.855**	**0.856**	**0.855**	**0.856**
Joint Gaussian	0.208	0.799	0.831	0.851	**0.856**
Ind. Gaussian	0.600	0.844	0.850	0.854	**0.856**
AC Gaussian WLR	0.812	0.854	0.855	0.854	0.855
AC Z-norm	**0.809**	0.829	0.832	0.831	0.832
AC F-norm	0.716	0.753	0.751	0.756	0.757
AC EER-norm	0.780	0.791	0.792	0.792	0.821

We now compare performance when the agents are not conditionally non-discriminative. We simulate a problem with four target-types and four corresponding agents. We again use Gaussian conditional densities, $p(\mathbf{x}|Y=j) = \mathcal{N}(\mathbf{x}; \bar{}_j, \Sigma)$. We set the parameters of the conditional distributions to make agents one and two offer discriminative power on each other's targets. In order to do this, we set:

$$\begin{aligned}
\mu_{-1} &= [-1,\ -1,\ -1,\ -1]^{\mathrm{T}}, \\
\mu_1 &= [1,\ 1,\ -1,\ -1]^{\mathrm{T}}, \\
\mu_2 &= [1,\ 1,\ -1,\ -1]^{\mathrm{T}}, \\
\mu_3 &= [-1,\ -1,\ 1,\ -1]^{\mathrm{T}}, \\
\mu_4 &= [-1,\ -1,\ -1,\ 1]^{\mathrm{T}},
\end{aligned}$$

and

$$\Sigma = \begin{bmatrix} 1 & \alpha & 0 & 0 \\ \alpha & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Using these definitions we can determine the following conditional distributions:

$$\begin{aligned}
p(x_2|x_1, Y=1) &= \mathcal{N}\left(\alpha x_1 + (1-\alpha), \sqrt{1-\alpha^2}\right), \\
p(x_2|x_1, Y=-1) &= \mathcal{N}\left(\alpha x_1 - (1-\alpha), \sqrt{1-\alpha^2}\right)
\end{aligned}$$

and

$$\begin{aligned}
p(x_1|x_2, Y=2) &= \mathcal{N}\left(\alpha x_2 + (1-\alpha), \sqrt{1-\alpha^2}\right), \\
p(x_1|x_2, Y=-1) &= \mathcal{N}\left(\alpha x_2 - (1-\alpha), \sqrt{1-\alpha^2}\right).
\end{aligned}$$

We can control how closely the conditionally non-discriminative assumption holds by varying α between 0 and 1. When $\alpha = 0$ agents one and two are independent and discriminative for one another's target-types; thus the conditionally non-discriminative assumption is poor. At $\alpha = 1$ the conditionally non-discriminative condition holds.

Tables 2 and 3 give the results, averaged over fifty random draws of the data for $\alpha \in \{0.95, 0.8, 0.6, 0.4, 0.1\}$ when the combiners are provided with 100 and 1000 training samples, respectively. Table 2 shows that with fewer training samples, the proposed combiner is the best or

statistically significantly tied for the best at all α values other than $\alpha = 0.1$, where the combiner based on the independence assumption performs better. Table 3 shows that when the combiners have more training samples, the proposed combiner is the best at only the larger values of $\alpha = 0.95$ and $\alpha = 0.8$. These results show that, in the case of limited data samples, the proposed combiner exhibit good performance even when the conditionally non-discriminative assumption does not hold.

Table 2. Average PAUC for simulation two with 100 combiner training samples. Boldface identifies results that are the best or statistically significantly tied for the best with 95% confidence.

	$\alpha = 0.95$	$\alpha = 0.8$	$\alpha = 0.6$	$\alpha = 0.4$	$\alpha = 0.1$
Prpsd. Gaussian WLR	**0.464**	**0.473**	**0.476**	**0.500**	0.508
Joint Gaussian	0.371	0.383	0.402	0.432	0.483
Ind. Gaussian	0.418	0.440	0.463	**0.497**	**0.538**
AC Gaussian WLR	0.457	0.463	0.461	0.480	0.487
AC Z-norm	**0.461**	0.466	**0.474**	0.490	0.503
AC F-norm	0.452	0.460	0.471	0.483	0.490
AC EER-norm	**0.464**	0.468	**0.478**	0.491	0.510

Table 3. Average PAUC for simulation two with 1000 combiner training samples. Boldface identifies results that are the best or statistically significantly tied for the best with 95% confidence.

	$\alpha = 0.95$	$\alpha = 0.8$	$\alpha = 0.6$	$\alpha = 0.4$	$\alpha = 0.1$
Prpsd. Gaussian WLR	**0.479**	**0.492**	0.504	0.521	0.543
Joint Gaussian	0.469	0.489	**0.506**	**0.535**	0.584
Ind. Gaussian	0.444	0.467	0.496	0.534	**0.589**
AC Gaussian WLR	0.471	0.482	0.490	0.504	0.521
AC Z-norm	0.471	0.483	0.491	0.505	0.522
AC F-norm	0.469	0.481	0.490	0.503	0.522
AC EER-norm	0.465	0.474	0.481	0.492	0.510

5.2. Pin-Less Verification with Yale Faces

We perform a pin-less biometric verification experiment using the cropped version of the Extended Yale Faces dataset [28]. The dataset contains frontal images of thirty-nine people, with sixty-five images of each person taken under different lighting conditions. In order to perform the pin-less verification experiment, we make persons one through five the targets, referred to as clients within the biometrics literature, and the remaining users the clutter, referred to as impostors.

The original images are 192 x 168 pixels. We down sample the images by a factor of two (The Yale Faces data is often down sampled in order to improve computational efficiency [29,30]), vectorize the resulting pixels, and perform principal components analysis retaining components that contain 95% of the data variance. This results in a sixty-four dimensional feature vector for each image.

We randomly select n images from each person in the client group for training and use the rest for testing. We randomly select twenty persons from the impostor group and use all of their images for training while using all images from the remaining fourteen impostors for testing. We vary the number of training images for each client over the set $n \in \{5, 10, 20, 30, 40\}$. We repeat the experiment twenty-five times for each n, randomly selecting different training and test images for each iteration.

We train five SVM agents, with the training data for the i^{th} agent consisting of the n images for client i as target and all images from the twenty training impostors as clutter. We choose the SVM parameters via cross-validation on the training data, and then get unbiased training data for training the combiners by cross-validation.

We compare the combiner strategies from the previous section along with the addition of the proposed combiner with weighted likelihood normalization estimated via the Platt probabilistic normalization as described in Section 4.4 (*Prpsd. Platt WLR*) and a meta-SVM classifier using the

Gaussian RBF kernel (*Meta SVM*). We choose parameters for the meta-SVM via cross-validation on the combiner training data from the same pool of parameters used to train the agent classifiers.

Table 4 gives the results. If we look at the top three rows, we can see that at a high number of training samples the joint and independent Gaussian combiners result in better performance than the proposed combiner with a Gaussian model for weighted likelihood ratio. However, the proposed combiner with the weighted likelihood ratio estimated via Platt probabilistic normalization is statistically tied for the best for all numbers of training samples, outperforming both the generative joint Gaussian model and the discriminative meta-SVM. The any-combiner with F-normalization performs very well also, being statistically tied for the best at all numbers of training samples other than five. At five training samples, the any-combiner with Z-normalization is tied with the proposed combiner with Platt probabilistic normalization.

We also use this experiment to show that the normalized agent scores give additional information indicating which target-type causes an alert. To do this, we calculate the percentage of true-positive alerts for which the maximum normalized agent-score is from the agent trained for the target-type that causes the alert. Table 5 gives the average accuracy for the different normalization methods when we set the threshold, α, in order to get a false alarm rate of five percent. The results show that the combiners that achieved the highest probability of detection according to Table 4 also achieve high accuracy in terms of estimating the correct target-type.

Table 4. Average PAUC for the Yale Faces pin-less verification experiment. Column titles give the number of training examples from each client person. Boldface identifies results that are the best or statistically significantly tied for the best with 95% confidence.

	5	**10**	**20**	**30**	**40**
Prpsd. Gaussian WLR	0.453	0.545	0.688	0.735	0.754
Prpsd. Platt WLR	**0.503**	**0.597**	**0.761**	**0.821**	**0.846**
Joint Gaussian	0.358	0.534	0.698	0.775	0.798
Ind. Gaussian	0.435	0.548	0.691	0.745	0.756
AC Gaussian WLR	0.447	0.544	0.688	0.735	0.754
AC Z-norm	**0.507**	0.552	0.662	0.699	0.711
AC F-norm	0.452	**0.610**	**0.751**	**0.821**	**0.841**
AC EER-norm	0.237	0.338	0.572	0.732	0.729
Meta-SVM	0.452	**0.568**	0.680	0.771	0.820

Table 5. Average percent accuracy when using the maximum normalized agent-output to estimate the client that causes an alert when the threshold is set to give five percent probability of false alarm (PFA). Column titles give the number of training examples from each client person.

	5	**10**	**20**	**30**	**40**
Prpsd. Gaussian WLR	95	97	98	98	99
Prpsd. Platt WLR	97	98	99	99	99
AC Gaussian WLR	95	97	98	98	99
AC Z-norm	97	98	99	98	98
AC F-norm	95	97	98	99	99
AC EER-norm	82	92	96	99	99

5.3. Classification of Ground Vehicles Using Acoustic Signatures

The acoustic-seismic classification identification dataset (ACIDS) contains acoustic time-series data collected from nine different types of ground vehicles as they pass by a fixed location. An array of three microphones recorded the sound emitted from each passing vehicle. The recordings were made in four different locations including desert, arctic, and mid-Atlantic environments, with vehicle speeds varying from five to forty km/hour, and closest point of arrival (CPA) distances from twenty-five to one hundred meters. The data consists of 274 labeled recordings of CPA events, and Table 6 gives a

breakdown of the number of events for each type of vehicle. The Army Research Lab collected the data and made it available [31]. We use the acoustic data from microphone one only. The sampling frequency is 1025.641 Hz and the data is bandpassed between twenty-five and four hundred Hz.

Table 6. Number of closest point of arrival (CPA) events for each type of vehicle in the ACIDS dataset.

Vehicle	Number of Events	Number of Scans
1	62	4960
2	37	2960
3	9	720
4	27	2160
5	39	3120
6	37	2960
7	7	560
8	35	2800
9	21	1680

We pre-process the data and extract features in a manner that matches closely that of a previous study using the ACIDS dataset by Wu and Mendel [32]. We first estimate the CPA for each time-series. We do this by filtering the magnitude of the acoustic response with a two-second moving-average filter and estimating CPA as the point where the output achieves its maximum value. We then limit the data to a forty-second window centered on the CPA. We convert the windowed time-series into a spectrogram by taking the short-time Fourier transform with a one-second window, frequency resolution of one Hz, and fifty percent overlap. This gives a spectrogram that contains eighty time-scans and 376 frequency bins. We pass the spectrogram through a spectrum normalizer to remove the broadband energy trend, so that the end result is a spectrogram that characterizes the narrowband content of the vehicle's acoustic signature. The normalized spectrogram for the first CPA event in the ACIDS dataset is shown in Figure 3.

We featurize the spectrograms using the magnitude of the spectrum at the second through twelfth harmonics of the fundamental frequency as was done in [32]. To get these features, we first estimate the fundamental frequency of the predominant harmonic set in each spectrogram that, as described in [33], typically relates to the engine cylinder firing rate. We estimate the predominant fundamental using an algorithm similar to the harmonic relationship detection algorithm described in [33]. The algorithm detects the presence of harmonic sets with fundamental frequencies between six and twenty Hz on a scan-by-scan basis by looking for a repeated pattern of narrowband energy at the correct harmonics within the spectrum. We smooth the resulting estimate using a median filter, and take the magnitude of the spectrum at the second through twelfth harmonics, normalized to have a maximum of one, as the feature vector for the scan. This results in eighty features per event. These features are suitable for this application due to their invariance to changes in the vehicle's speed [33].

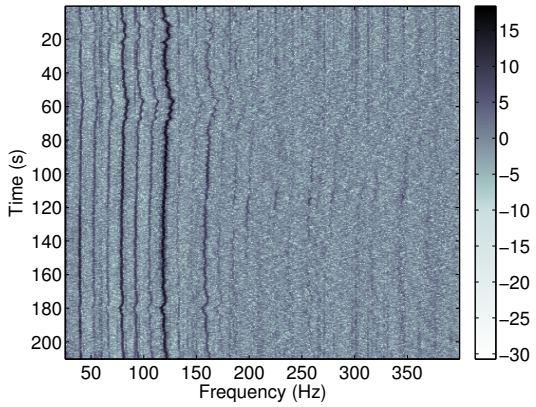

Figure 3. Normalized magnitude spectrogram, in dB, for the first vehicular event in the ACIDS dataset.

Vehicles 1, 2, 8, and 9 form a group of heavy tracked vehicles and we define these four vehicle types as our targets and the others as clutter. We perform ten-fold cross-validation. In each fold, we set aside ten percent of the events as test and the rest as training. We use the scans in the training data to train an ensemble of Gaussian RBF SVM classifier-agents, one for each target vs. clutter. We choose the SVM parameters for each classifier agent by ten fold cross-validation on the training data and also get unbiased data for combiner training by ten-fold cross-validation.

We compare the proposed combiner with weighted-likelihood ratio normalization, where we model the likelihood using Platt probabilistic normalization, Neyman-Pearson combination using a joint Gaussian model, Neyman-Pearson combination using an independent Gaussian model, the any-combiner using Z, F, and EER normalization and a meta-SVM combiner using a non-linear Gaussian RBF SVM trained to classify the output of the combiners. We train the meta-SVM combiner using the same parameters and procedure with which we train the classifier agents. We repeat this experiment twenty-five times, randomizing the cross-validation indices each time, to get our final set of results.

The SVM agents classify scans. In order to classify the events, we combine the scan-by-scan scores along the eighty scans for each event. Define $c_i, i = 1, ..., 80$ as the combiner output along the eighty scans for an event. Assuming that the scans are independent, we combine the scores for an event to get a likelihood ratio of target vs. clutter according to:

$$\Lambda = \sum_{i=1}^{80} \log \left(\frac{p(c_i|y=1)}{p(c_i|y=-1)} \right). \tag{27}$$

The two Neyman-Pearson combiners as well as the proposed combiner give score outputs that are already in the form of (27), so we sum the log of the combiner outputs on scans to get the event score. In order to convert the Z-norm, F-norm, and EER-norm combiner outputs to a probability, we make the assumption that the output of the combiner is conditionally Gaussian. For the Meta-SVM, we train a Platt-scaling function for the Meta-SVM output and convert it to a likelihood ratio as in Section 4.4.

Table 7 gives the PAUC, averaged over the twenty-five experiments, in the 0%–10% PFA operating region when classifying scans and events. The result shows that the proposed combiner with Platt weighted-likelihood normalization results in the highest PAUC in this region for both scans and events, and that the result is statistically significant at 95 % confidence.

Table 7. Average PAUC over the 0%–10% PFA operating region for the various combiner methods on scan-by-scan features and events from the ACIDS dataset. Boldface identifies results that are the best or statistically significantly tied for the best with 95% confidence.

	Scans	**Events**
Prpsd. Gaussian WLR	0.563	0.794
Prpsd. Platt WLR	**0.575**	**0.847**
Joint Gaussian	0.554	0.786
Ind. Gaussian	0.546	0.764
AC Gaussian WLR	0.562	0.797
AC Z-norm	0.556	0.784
AC F-norm	0.467	0.753
AC EER-norm	0.385	0.539
Meta-SVM	0.558	0.814

6. Conclusions

We have proposed a combination strategy for an ensemble of agent-classifiers that is optimal under a conditionally non-discriminative assumption on the agent-scores. We showed that the proposed combiner naturally handles target-class population drift without the need for re-training the underlying classifier agents. We compared this approach to several other combiner strategies

including meta-classification and the any-combiner approach with different agent-normalization functions. We have shown empirically that the proposed combiner achieves excellent performance when the conditionally non-discriminative assumption holds. The proposed approach showed excellent performance in a pin-less verification experiment using face data and in a vehicle classification experiment using acoustic signatures.

Author Contributions: Conceptualization, N.H.P.; Methodology, N.H.P. and A.J.L.; Software, N.H.P. and A.E.D.; Investigation, N.H.P., A.J.L. and A.E.D.; Writing–original draft preparation, N.H.P.; Writing–review and editing, N.H.P. and A.J.L.; Funding acquisition, A.J.L. All authors have read and agree to the published version of the manuscript.

Funding: This research was funded by the Office of Naval Research.

Acknowledgments: The authors would like to thank the US Army Research Laboratory for providing the ACIDS dataset.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pearman, W.F.; Fountain, A.W. Classification of chemical and biological warfare agent simulants by surface-enhanced Raman spectroscopy and multivariate statistical techniques. *Appl. Spectrosc.* **2006**, *60*, 356–365. [CrossRef] [PubMed]
2. Wayman, J.; Jain, A.; Maltoni, D.; Maio, D. An introduction to biometric authentication systems. *Biom. Syst.* **2005**, 1–20.
3. Kantchelian, A.; Afroz, S.; Huang, L.; Islam, A.C.; Miller, B.; Tschantz, M.C.; Greenstadt, R.; Joseph, A.D.; Tygar, J. Approaches to adversarial drift. In Proceedings of the ACM Workshop Artificial Intelligence and Security, Berlin, Germany, 4 November 2013; pp. 99–110.
4. Malisiewicz, T.; Gupta, A.; Efros, A.A. Ensemble of exemplar-svms for object detection and beyond. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 89–96.
5. Parrish, N.; Llorens, A.J. The any-combiner for multi-agent target classification. In Proceedings of the 16th International Conference on Information Fusion, Istanbul, Turkey, 9–12 July 2013; pp. 166–173.
6. McClish, D.K. Analyzing a portion of the ROC curve. *Med. Decis. Mak.* **1989**, *9*, 190–195. [CrossRef]
7. Kuncheva, L.I. *Combining Pattern Classifiers: Methods and Algorithms*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2004.
8. Jacobs, R.A.; Jordan, M.I.; Nowlan, S.J.; Hinton, G.E. Adaptive mixtures of local experts. *Neural Comput.* **1991**, *3*, 79–87. [CrossRef] [PubMed]
9. Collobert, R.; Bengio, S.; Bengio, Y. A parallel mixture of SVMs for very large scale problems. *Neural Comput.* **2002**, *14*, 1105–1114. [CrossRef] [PubMed]
10. Enzweiler, M.; Gavrila, D.M. A multilevel mixture-of-experts framework for pedestrian classification. *IEEE Trans. Image Proc.* **2011**, *20*, 2967–2979. [CrossRef] [PubMed]
11. Ebrahimpour, R.; Kabir, E.; Yousefi, M.R. Improving mixture of experts for view-independent face recognition using teacher-directed learning. *Mach. Vis. Appl.* **2011**, *22*, 421–432. [CrossRef]
12. Yuksel, S.E.; Wilson, J.N.; Gader, P.D. Twenty years of mixture of experts. *IEEE Trans. Neural Netw. Learn. Syst.* **2012**, *23*, 1177–1193. [CrossRef]
13. Sakkis, G.; Androutsopoulos, I.; Paliouras, G.; Karkaletsis, V.; Spyropoulos, C.D.; Stamatopoulos, P. Stacking classifiers for anti-spam filtering of e-mail. *arXiv* **2001**, arXiv:cs/0106040.
14. Wang, S.Q.; Yang, J.; Chou, K.C. Using stacked generalization to predict membrane protein types based on pseudo-amino acid composition. *J. Theor. Biol.* **2006**, *242*, 941–946. [CrossRef]
15. Domingos, P. A few useful things to know about machine learning. *Commun. ACM* **2012**, *55*, 78–87. [CrossRef]
16. Poh, N.; Bengio, S. *An Investigation of F-Ratio Client-Dependent Normalisation on Biometric Authentication Tasks*; Technical Report, Research Report 04-46; IDIAP: Martigny, Switzerland, 2004.

17. Fierrez-Aguilar, J.; Ortega-Garcia, J.; Gonzalez-Rodriguez, J. Target dependent score normalization techniques and their application to signature verification. *IEEE Trans. Syst. Man Cybern. C Appl. Rev.* **2005**, *35*, 418–425. [CrossRef]
18. Poh, N.; Kittler, J. Incorporating variation of model-specific score distribution in speaker verification systems. *IEEE Trans. Audio Speech Lang. Process.* **2008**, *16*, 594–606. [CrossRef]
19. Poh, N.; Ross, A.; Lee, W.; Kittler, J. A user-specific and selective multimodal biometric fusion strategy by ranking subjects. *Pattern Recognit.* **2013**, *46*, 3341–3357. [CrossRef]
20. Kittler, J.; Hatef, M.; Duin, R.P.; Matas, J. On combining classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.* **1998**, *20*, 226–239. [CrossRef]
21. Kelly, M.G.; Hand, D.J.; Adams, N.M. The impact of changing populations on classifier performance. In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, 22–27 August 1999; pp. 367–371.
22. Kolter, J.Z.; Maloof, M.A. Dynamic weighted majority: An ensemble method for drifting concepts. *J. Mach. Learn. Res.* **2007**, *8*, 2755–2790.
23. Klinkenberg, R.; Joachims, T. Detecting Concept Drift with Support Vector Machines. In Proceedings of the Seventeenth International Conference on Machine Learning, Stanford, CA, USA, 29 June–2 July 2000; pp. 487–494.
24. Poh, N.; Tistarelli, M. Customizing biometric authentication systems via discriminative score calibration. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2681–2686.
25. Platt, J.C. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classif.* **1999**, *10*, 61–74.
26. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27. Available online: http://www.csie.ntu.edu.tw/~cjlin/libsvm (accessed on 10 February 2020). [CrossRef]
27. Doddington, G.; Liggett, W.; Martin, A.; Przybocki, M.; Reynolds, D. *Sheep, Goats, Lambs and Wolves: A Statistical Analysis of Speaker Performance in the NIST 1998 Speaker Recognition Evaluation*; Technical Report, DTIC Document; National Institutes of Science and Technology: Gaithersburg, MD, USA, 1998.
28. Georghiades, A.; Belhumeur, P.; Kriegman, D. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 643–660. [CrossRef]
29. Wright, J.; Yang, A.Y.; Ganesh, A.; Sastry, S.S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 210–227. [CrossRef]
30. Naseem, I.; Togneri, R.; Bennamoun, M. Linear regression for face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 2106–2112. [CrossRef] [PubMed]
31. Pham, T.; Srour, N. *ACIDS*; Technical Report; US Army Research Laboratory: Adelphi, MD, USA 1998.
32. Wu, H.; Mendel, J.M. Classification of battlefield ground vehicles using acoustic features and fuzzy logic rule-based classifiers. *IEEE Trans. Fuzzy Syst.* **2007**, *15*, 56–72. [CrossRef]
33. Robertson, J.A.; Mossing, J.C.; Weber, B.A. Artificial neural networks for acoustic target recognition. In *Proceedings of the SPIE Symposium. OE/Aerospace Sensing and Dual Use Photonics*; International Society for Optics and Photonics: Bellingham, WA, USA 1995; pp. 939–950.