On-line anomaly detection and resilience in classifier ensembles

Hesam Sagha *, Hamidreza Bayati, José del R. Millán, Ricardo Chavarriaga

Defitech Chair in Non-Invasive Brain–Machine Interface, Center for Neuroprosthetics and Institute of Bioengineering, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

ABSTRACT

Detection of anomalies is a broad field of study, which is applied in different areas such as data monitoring, navigation, and pattern recognition. In this paper we propose two measures to detect anomalous behaviors in an ensemble of classifiers by monitoring their decisions; one based on Mahalanobis distance and another based on information theory. These approaches are useful when an ensemble of classifiers is used and a decision is made by ordinary classifier fusion methods, while each classifier is devoted to monitor part of the environment. Upon detection of anomalous classifiers we propose a strategy that attempts to minimize adverse effects of faulty classifiers by excluding them from the ensemble. We applied this method to an artificial dataset and sensor-based human activity datasets, with different sensor configurations and two types of noise (additive and rotational on inertial sensors). We compared our method with two other well-known approaches, generalized likelihood ratio (GLR) and One-Class Support Vector Machine (OCSVM), which detect anomalies at data/feature level.

We found that our method is comparable with GLR and OCSVM. The advantages of our method compared to them is that it avoids monitoring raw data or features and only takes into account the decisions that are made by their classifiers, therefore it is independent of sensor modality and nature of anomaly. On the other hand, we found that OCSVM is very sensitive to the chosen parameters and furthermore in different types of anomalies it may react differently. In this paper we discuss the application domains which benefit from our method.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The field of activity recognition has gained an increasing level of interest driven by applications in health monitoring and assistance, manufacturing and entertainment. In addition, given the advances in portable sensing technologies and wireless communication many of these systems rely on the fusion of the information from (heterogeneous) sensor networks. In this framework, the detection of anomalous (e.g. faulty or misbehaving) nodes constitute an important aspect for the design of robust, resilient systems. Besides activity recognition, anomaly detection (AD) is an important issue in the fields of control systems, navigation, and time series analysis. Generally, an anomalous pattern is one that is not desired or not expected. Therefore it often decreases the system performance or generates abnormal behavior in the data stream. It can be due to sensor failure, signal degradation, environmental fluctuations, etc. In most fields, such as health monitoring, a desirable characteristic is to process the data stream online and get a real-time anomaly detection to take an appropriate counteraction.

As mentioned above, there is a tendency toward the use of large number of sensors and with different sensor modalities to have more information about the observed environment. In a pattern recognition system, there are different levels to fuse sensor information; data, feature, or classifier. Generally, the goal of data fusion is to achieve more reliable data. Feature fusion concatenates different features from all the sensors before classification (Fu et al., 2008). Classifier fusion is applicable when an ensemble of classifiers is used and each classifier is assigned to different subset of sensors, and finally, the decision is made by a combination of the classifier decisions (Ruta and Gabrys, 2000). This architecture allows for a decentralized classification system where each classifier decides about a separate data stream. Therefore, if a particular stream is faulty or misbehaving we can remove the corresponding channel easily from the fusion in order to continue the classification, keeping the system performance as high as possible. A good example of such systems is human activity recognition (HAR) with on-body sensors mounted on specific limbs. A good practice is to devote a distinct classifier to each of the sensors (Roggen et al., 2011). In these systems, one inevitable anomaly is the rotation or sliding of the sensors. If these anomalies can be detected, it is easier to take an appropriate counter-action for the misbehaving sensor, without the need of

* Corresponding author. Tel.: +41 788757348.
  E-mail address: hesam.sagha@epfl.ch (H. Sagha).

reconfiguring the whole system (Sagha et al., 2011a; Chavarriaga et al., 2011). This desirable characteristic is the core feature of *Opportunistic activity recognition systems* where elements in the network may appear, disappear or change during life time (Roggen et al., 2009).

Given a classifier ensemble, the anomaly detection process can be applied at different levels; raw signal or feature level, or at classifier/fusion level. Hereafter, we call both raw and feature level as low level and the detection process as *Low Level Anomaly Detection* (LoLAD), while for the later case we name the process as *Fusion Level Anomaly Detection* (FuLAD). The former case is the most commonly applied (Chandola et al., 2009), however, it is often not applicable as different sensor modalities may be available – requiring to design specific AD for each modality – as well as the energy and computational cost in the case of wireless sensor networks. Therefore in this paper we introduce an anomaly detection mechanism at the level of classifier fusion, based on the consistency of the classifier decisions. Applying the method on two human activity datasets, we show that, upon detection of anomalous classifiers, an adaptation strategy such as removing them from the fusion chain, leads to a graceful performance degradation.

The structure of the paper is as follows; the next section summarizes related work on anomaly detection and resilience, then we describe the proposed method in Section 3. The description of the experiments and the results are presented in Sections 4 and 5, respectively. Finally, we discuss about the use of the method and its pros and cons in Section 6, followed by a conclusion.

2. Related work

We can handle anomalies in two manners: through detection and isolation, or by anomaly resilience and adaptation. In the former case, the goal is to detect whether there is an anomaly in the data or not (detection) and which part of the system is affected (isolation). While for the latter one, the system is designed to be tolerant against anomalies, or to be able to take suitable counteraction whenever an anomaly is detected. For example, by changing the network structure in wireless sensor networks (WSN) or adapting parameters to the new data trend.

2.1. Anomaly detection and isolation

Numerous studies have been undertaken to detect anomalies at the data level. Chandola et al. (2009) survey methods to detect anomalous patterns in a pool of patterns. These methods (such as computing distance to the nearest neighbor or to a cluster center, estimating statistical models on the data, discriminating normal and anomalous patterns using artificial neural networks, or support vector machines) are used in fault detection in mechanical units (Jakubek and Strasser, 2002), structural damage detection (Brotherton and Johnson, 2001), sensor networks (Ide et al., 2007), etc.

Time series change detection (Basseville and Nikiforov, 1993) has been applied to fraud detection (Bolton et al., 2002), computer intrusion detection (Schonlau et al., 2001) and concept drift (Lane and Brodley, 1998). One of the well-known approaches is CUmulative SUM (CUSUM) (Page, 1954) which is particularly used when the parameters of the changed signal are known. CUSUM computes the cumulative sum of the log-likelihood ratio of the observations and once this value exceeds a threshold (which can be adaptive) it is considered a change. It is widely used in change and drift detection in time series (Wu and Chen, 2006), exerting neuro-fuzzy models (Xie et al., 2007), auto-regressive models (El Falou et al., 2000), and Kalman filters (Severo and Gama, 2010) to generate residuals whose changes should be detected. Li et al. proposed a subspace approach to identify optimal residual models in a multivariate continuous-time system (Li et al., 2003).

When the parameters of the changed signals are unknown, generalized likelihood ratio (GLR) (Lorden, 1971) and adaptive CUSUM are proposed. GLR maximizes the likelihood ratio over possible values of the parameter of the changed signal, assuming the normal data follows a Gaussian distribution. It is widely used for change detection in brain imaging (Bosc et al., 2003), diffusion tensor imaging for monitoring neuro-degenerative diseases (Boisgontier et al., 2009), for detecting land mines using multi-spectral images (Anderson, 2008), and target detection and parameter estimation of MIMO radar (Xu and Li, 2007). In the same class, Adaptive CUSUM is able to detect changes by suggesting a distribution for the unknown change model based on the distribution of the known model of unaltered data (Alippi and Roveri, 2006a,b, 2008). One-Class Support Vector Machine (OCSVM) is another common approach for anomaly detection (Das et al., 2011). Its rationale is to compute a hyperplane around train data and samples are considered anomalous if they fall outside that hyperplane.

Other approaches have also been proposed in control systems to detect abnormal sensors. One is to extract the model of the system and detect faults by monitoring residual error signal (Hwang et al., 2010). Another more complex way is to create the input–output model of the system by regression methods and detect potential faults when there is a change in the estimated parameters (Ding, 2008; Smyth, 1994).

There are many studies and methods toward detection of fault and changes in the sensor networks. In this area, anomaly detection can be done by having redundancies in the network. Either in the form of physical redundancy such as adding extra sensors, thus increasing the cost of the system deployment; or from analytical redundancies (Betta and Pietrosanto, 1998). For instance, Andrieu et al. (2004) discuss particle methods to model validation, change/fault detection, and isolation. In this scope, model validation is a process to ensure reliable operations. A survey on the approaches to fault detection and isolation in unknown environments has been done by Gage and Murphy (2010). Chen et al. proposed a probabilistic approach that recognizes faulty sensors based on the difference in the measurements of a sensor and its neighbors (Chen et al., 2006). Other approaches are based on the similarity between two time series (Yao et al., 2010), OCSVM (Rajasegarar et al., 2007) and clustering (Rajasegarar et al., 2006) to detect outliers in a sensor network, and kernel-based methods (Camps-Valls et al., 2008) to detect changes in remote imaging systems.

2.2. Anomaly resilience and adaptation

In sensor networks there are different strategies to introduce robustness against changes and drifts in the sensors. Luo et al. (2006) discuss how to optimize the parameters of the model of a sensor under both noisy measurement and sensor fault. In turn, Demirbas (2004) proposes scalable design of local self-healing for large-scale sensor network applications, while Koushanfar et al. (2002) propose to use an heterogeneous back-up scheme to substitute faulty sensors. Other different designs, principles and service managements have been proposed to provide self-healing services and diagnosing the true cause of performance degradation (Krishnamachari and Iyengar, 2003; Ruiz et al., 2004; Sheth et al., 2005).

Alternatively, the system can be adapted to dynamic changes. Snoussi and Richard (2007) model the system dynamics, including abrupt changes in behavior, and select only a few active nodes based on a trade-off between error propagation, communications constraints and complementarity of distributed data. Wei et al. (2009) used discrete-state variables to model sensor states and

continuous variables to track the change of the system parameters. Alternatively, fault detection in WSN has been addressed by finding linear calibration parameters using Kalman and particle filters, and state models (Balzano, 2007). Furthermore, Wang et al. (2005) proposed a fault-tolerant fusion based on error-correction codes in order to increase the classification performance where each node sends class labels to the fusion process. Alternatively, Chavarriaga et al. (2011) used mutual information between classifier decisions to assess the level of noise in an activity recognition scenario.

2.2.1. AD in activity recognition systems

Advances in sensing, portable computing devices, and wireless communication has lead to an increase in the number and variety of sensing enabled devices. Activity recognition systems can fuse information from networks of on-body and ambient sensors for better performance. In such systems there are several approaches to handle anomalies at the data, feature or classifier level. Some of the mentioned approaches in Section 2.1 could be applicable at the feature level. Another strategy is to identify possible anomalies and propose a method to compensate for them during the network's life-time. i.e., for specific well-defined anomalies, ad hoc strategies can be devised, but this is application-dependent and may not be easily generalized for other domains. For example, in accelerometer-based HAR, sensor rotations are hardly avoidable. One way to cope is to detect how much rotation is added to the signals, based on a reference, and then rotate the sensor readings back before classification (Kunze and Lukowicz, 2008; Steinhoff and Schiele, 2010). An alternative way is to use features or classifiers that are robust against a particular anomaly. For example, Förster et al. (2009) extract discriminative features based on genetic programming, which are robust against sensor sliding in body area sensor networks. Another approach is to use an adaptive classifier robust to specific changes in the feature distribution using an Expectation–Maximization approach (Chavarriaga et al., 2013a). These methods are limited due to the fact that there is a need to pre-determine the possible anomalies. Moreover, it may not be feasible to characterize all of them in real world applications and it may be impractical to have a rectifier for all of them.

Depending on the limitations of a sensor network, such as the bandwidth, the power consumption, the modality and the cost of each node, in practice it may not be feasible to design an efficient LoLAD for a pattern recognition system as the ones described in this Section. This process can increase the requirements of computational power and energy consumption of each node, depending on the sampling rate and/or bandwidth. Also, as mentioned previously, designing an AD for each modality may not be feasible. In order to come up with these limitations there is a need to reduce the communication of the nodes and make the AD independent of the modality. Aiming at overcoming these limitations, we propose a method for AD at the fusion level, particularly suitable for classifier ensembles.

Classifier ensembles can provide great flexibility to recognition systems. In particular, when an anomalous classifier is detected, the ensemble can be reconfigured so as to reduce performance degradation. The simplest way would be to remove it from the ensemble by turning the node off or forcing it into stand-by mode. Other approaches can attempt to adapt or retrain the anomalous classifiers with the new data. In this case, since the ground truth labels are not provided online, we can use the best classifier in the ensemble as a trainer (Calatroni et al., 2011). Another way, is to update the fusion parameters based on the output of the new classifiers (Sannen et al., 2010). An alternative approach could be to regenerate the decisions of anomalous classifiers based on the decisions of the other classifiers and then use the same fusion structure as before (Sagha et al., 2010). In this work we focus on the detection of the anomalous classifiers and use the first strategy

(i.e., removal) to assess the increased robustness of the entire system.

3. Method

We propose a method to automatically detect anomalies in a classifier ensemble at the fusion level, as shown in Fig. 1. The classifier ensemble architecture is more convenient in the sensor networks deployed for pattern recognition in the sense that different modalities could be covered with different classifiers and also the ensemble could be easily reconfigured to attain different criteria such as cost and accuracy (Chavarriaga et al., 2011).

The rationale behind the method is to find a model of classifier coherence from the training set. Then, at run time, looking at a sequence of classifiers' outputs, we determine how much each classifier deviates from this model. Whenever this deviation exceeds a defined threshold the classifier is considered as anomalous. Finally, to make the system resilient, we remove the anomalous classifiers and reconfigure the ensemble.

To compute the deviation, we present two measures. One is based on the Mahalanobis distance, Section 3.1, and the other is based on an information theoretic approach, Section 3.2. Later, we will discuss the pros and cons of each approach. The first was originally proposed by Sagha et al. (2011b) and here we provide further characterization of it. Moreover, in order to compare the performance of the fusion level and feature level methods, in Sections 3.3 and 3.4 we describe two commonly used anomaly detection methods: GLR and OCSVM.

3.1. Distance based detection (DB)

We assume that a given classifier s in the ensemble and the fused output yield posterior probability vectors, $o_s = [p_1, p_2, \ldots, p_C]$ and $f = [p_1, p_2, \ldots, p_C]$ respectively, where each element denotes the probability of the sample belonging to a specific class $c \in [1, \ldots, C]$. Fig. 1 shows the process of detection. By calculating the distance between each classifier output and the final fusion output we can deduce how similar they are. At runtime, if the average distance over the last w decisions exceeds a pre-defined threshold, the corresponding classifier will be marked as *anomalous*, while other classifiers are considered *healthy*.

The computed distance should take into account the information obtained from the training set. We propose the Mahalanobis distance, $\mathcal{D}_{sc}$, which stores information about the classifier coherence and correlation in a covariance matrix. The average distance over past w decisions, $\tilde{\mathcal{D}}_{sc}$, is defined as

$$\tilde{\mathcal{D}}_{sc'}(t') = \frac{1}{w} \sum_{t=t'-w}^{t'} (o_{s_t} - f_t)^T \Phi_{sc'}^{-1} (o_{s_t} - f_t), \qquad (1)$$

where o_{s_t} is the output of the classifier $s \in [1, \ldots, N]$ and f_t is the fusion output at time t. $c' \in [1, \ldots, C]$ is the recognized class after
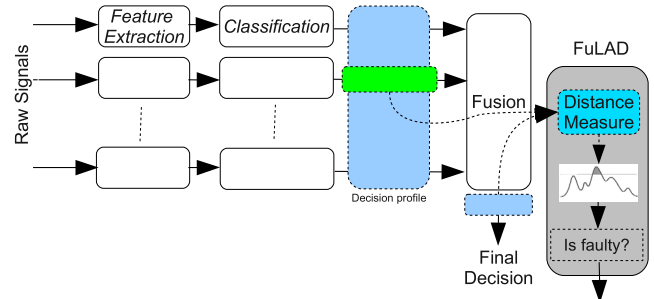

Fig. 1. Classifier fusion architecture. FuLAD (DB) process is shown in gray.

classifier fusion. $\Phi_{sc'}$ represents the covariance between the classifier s and the fusion output, for class c'. The covariance matrix is estimated based on the training data set,

$$\Phi_{sc} = E((o_s^c - f^c)^T (o_s^c - f^c)), \tag{2}$$

where o_s^c and f^c are the output of the classifier s and the output of the fusion for the specific class $c \in [1, \ldots, C]$, respectively, and $E(\cdot)$ is the mathematical expectation. When the distance between a classifier and fused output is larger than the corresponding threshold for the chosen class, we label the classifier as anomalous. Thresholds, Θ_{sc}, are set individually for each classifier and class such that

$$\Theta_{sc} = k\,max(\mathcal{D}_{sc}^{tr}), \tag{3}$$

where $max(\mathcal{D}_{sc}^{tr})$ is the maximum distance for class c and classifier s computed on the training set. The constant $k > 0$ is the same for all the classes and classifiers, and denotes the sensitivity of the detection. Larger values of k result in less sensitive detection. In this work, we use the *max* function which is simple yet reasonable, although more complex threshold estimations could also be applied (e.g. by assessing the distance distribution).

Once one classifier is labeled as anomalous, we exclude it from the fusion process. As a result the final fusion decision may change. Therefore, to detect all the anomalous parts we perform the detection process iteratively until the distances of the remaining classifiers are below the threshold or a predefined minimum number of healthy classifiers, nH_{min}, is reached.

Moreover, we add a fixed epsilon value (0.01) to the diagonal elements of the covariance matrix to avoid singularities, and an upper bound is used to avoid large Mahalanobis distance values, which affect the anomaly recognition. The bound is set for all classes and classifiers using training data and is obtained as a scaled factor of the maximum value of $\mathcal{D}_{sc}^{tr}$. So

$$\mathcal{D}_{sc} = \begin{cases} \mathcal{D}_{sc} & \mathcal{D}_{sc} < \Gamma max(\mathcal{D}_{sc}^{tr}) \\ \Gamma max(\mathcal{D}_{sc}^{tr}) & \mathcal{D}_{sc} \geqslant \Gamma max(\mathcal{D}_{sc}^{tr}) \end{cases} \tag{4}$$

The final algorithm is as follows:

begin
$\Phi_{\mathbf{sc}} \equiv$ Covariance matrix for classifier s and class c
$\Theta_{sc} \equiv$ Distance threshold for classifier s and class c
nHealthy $\equiv$ Number of healthy detected classifiers (init. = N)
$nH_{min} \equiv$ minimum number of classifiers for fusion
$\tilde{\mathcal{D}}_{sc} \equiv$ Average distance over specified window
$f(\mathcal{S}) \equiv$ Fusion of classifiers in set $\mathcal{S}$
$\mathcal{H} \equiv$ set of healthy sensors (initially $\mathcal{H} = [s_1..s_n]$)
faulty $\equiv$ set of anomalous classifiers (init. $= \phi$)
while nHealthy > nH_{min}
  $f = fusion(\mathcal{H})$
  $c' = argmax_c(f)$
  Compute distance $\tilde{\mathcal{D}}_{sc'} \; \forall s \in \mathcal{H}$
  if $max(\tilde{\mathcal{D}}_{sc'}) < \Theta_{sc}$
    exit
  else
    faulty = [faulty argmax$_s(\tilde{\mathcal{D}}_{sc'})$]
    nHealthy = nHealthy $- 1$
  end
end

3.1.1. Computational cost

This procedure iterates at most $N - nH_{min}$ times for computing distances, where N is the number of classifiers and nH_{min} is the minimum number of classifiers for fusion. At each iteration *nHealthy* distances between each classifier and the fusion output are computed. The cost of calculating each distance is two vector

subtractions of size C (number of classes) and a multiplication of three matrices of sizes $1 \times C$, $C \times C$, and $C \times 1$. Finally the order of computation is $O(N^2 C^3)$.

3.2. Information Theoretic approach (IT)

Information theory is a wide spread domain which has been used for signal processing, cryptography, feature selection, among others (Shannon, 2001). Two key concepts in information theory are the entropy, which denotes the amount of information of a random variable, and the mutual information, which quantifies the shared information between two random variables. The entropy of a random variable X is obtained by

$$H(X) = -\sum_i p(x_i) log_2 p(x_i), \tag{5}$$

where x_i are possible values of X. Mutual information (MI) between random variables X and Y is defined by

$$MI(X;Y) = \sum_{x,y} p(x,y) log_2 \frac{p(x,y)}{p(x)p(y)}, \tag{6}$$

where $p(x,y)$ is the joint probability.

In the case of an anomalous classifier, we expect its mutual information with the rest of the classifiers to change (Chavarriaga et al., 2011). In consequence, MI changes between the classifiers could be used to detect anomalies in the ensemble.

To identify which part of the system is anomalous, some care should be taken when using these values; In order to have an on-line detection, the computation of MI on the test set, $\mathbf{MI_{test}}$, is done in a sliding window of classifier decisions. Inside the window, the distribution of ground truth labels may vary compared to the training set (e.g. always feeding data from one class would result in entropies and MI values equal to zero). This variation causes a common change in the estimation of MI elements. We call it *common information change (CIC)* which is measured in bits and computed as the minimum difference value between pairwise elements of $\mathbf{MI_{test}}$ and the one on training set, $\mathbf{MI_{train}}$. To remove this effect, we remove this difference from the computed MI. The algorithm is as follows:

begin
faulty $\equiv$ set of faulty detected classifiers (init. $= \phi$)
nHealty $\equiv$ Number of healthy detected classifiers, (init. $= N$)
dif = MI$_{\mathbf{train}}$ $-$ MI$_{\mathbf{test}}$
CIC = $min(\mathbf{dif})$
sub = dif $-$ CIC
sub$_{\mathbf{ii}}$ = 0; % removing entropy
while *true*
  $s_i = \sum_{j \notin faulty} \mathbf{sub_{ij}}$
  if $max(s) >$ nHealhty $\times$ k
    faulty = [faulty argmax(s)]
    nHealthy = nHealthy $- 1$
  else
    break
  end
  if nHealhty = nH_{min}
    break
  end
end

Again, k denotes the sensitivity, *dif* is the difference between MI of training and testing and its minimum value corresponds to the *CIC*. Afterwards, we set the diagonal values of the consequent matrix to zero to remove the entropy. Finally, the loop starts detecting anomalous classifiers until no value exceeds the threshold.

Opposite to the previous measure, this approach does not require the classifiers/fusion to provide their outputs as a probability vector. The decision of the classifiers can be discrete or nominal value.

3.2.1. Computational cost

The calculation requires the estimation of MI in a window of length w, $O(wC^2N^2)$, then we have a matrix subtraction $O(N^2)$ and the iteration continues at most until we reach to the nH_{min}. So finally the computation order is $O(wC^2N^2 + N^3)$.

3.3. Generalized Likelihood Ratio (GLR)

As a comparison we used two standard approaches that detects anomalies at the feature level. The first approach is based on Generalized Likelihood Ratio for vector based problems where the initial distribution of data θ_0 is known but it is unknown after change,

$$\Theta(t) = \begin{cases} = \theta_0, & \text{when } t < t_0, \\ \neq \theta_0, & \text{when } t \geqslant t_0, \end{cases} \qquad (7)$$

where t_0 is the time when change happens. The GLR solution leads to the following equation (Basseville and Nikiforov, 1993):

$$g_t = \max_{t-w \leqslant j \leqslant t} \frac{t-j+1}{2} (\bar{Y}_j^t - \theta_0)^T \Sigma^{-1} (\bar{Y}_j^t - \theta_0), \qquad (8)$$

where $\bar{Y}_j^t$ is the mean of the data vector from time j to time t. We mark the data as anomalous when g_t is greater than a threshold. Here, we set the threshold as the maximum value of g_t on the training set.

3.4. One-Class Support Vector Machine (OCSVM)

One-Class Support Vector Machine computes a hyperplane around the training data, every test data point outside this hyperplane is considered as anomalous. Two parameters must be tuned for this method; v, which is the upper bound of the fraction of the training data that are considered to be outside the hyperlane and bandwidth, σ, which affects the smoothness of decision boundary. To set these parameters, different optimization approaches are proposed (Zhuang and Dai, 2006; Lukashevich et al., 2009). Here, we experimentally set the best parameters so as to have a good detection accuracy on the rotational noise, Section 4.2.4. We have used LIBSVM implementation (Chang and Lin, 2011).

4. Experiments

The performance of the proposed methods has been evaluated on one artificially generated dataset and two real datasets of human activities. The activity datasets contain several sensor modalities, and different ensemble configurations were tested based on the sensor modalities or location.

4.1. Synthetic dataset

The synthetic dataset consists of different classifier ensembles where we systematically test different sizes, accuracies and levels of noise. The number of classifiers and classes ranges from 2 to 16 and 3 to 15 with the step of 2 and 3, respectively. The number of samples per class was set to 60 for both the training and the testing sets and they are randomly distributed over time. In each simulation, the accuracy of the healthy classifiers was set to 80%, while the performance of anomalous classifiers ranges from 10% to 70%. The number of anomalous classifiers varies from 0% to 100% of the available classifiers for each experiment. These values pro-

Table 1
Sensor configurations. The number of classifiers for each configuration is written in parenthesis for car manufacturing dataset and opportunity dataset, respectively.

Mode	Acc	Acc + Gyro	Acc + Gyro + Magnetic
L	$Config_{l1}(7,5)$	$Config_{l2}(7,5)$	$Config_{l3}(7,5)$
S	–	$Config_{s2}(14,10)$	$Config_{s3}(21,15)$

vide a general assessment on the methods and the effect of noise for different ensemble configurations and number of classes.

In each experiment, we change one of the following characteristics: number of classes, number of classifiers, percentage of anomalous classifiers, or the level of anomaly. Presented results are the average across 5 repetitions per each experiment.

4.2. Human activity datasets

4.2.1. Datasets

The two real datasets contain data from body mounted inertial sensors, recorded while human subjects perform different activities. In order to emulate changes in the sensor network we artificially added different levels of noise to the sensor readings and assessed the method ability to detect noisy channels, as well as the recognition performance.

The first dataset corresponds to a *car manufacturing scenario* (Stiefmeier et al., 2008). It contains data from 8 subjects performing 10 recording sessions each (except one subject who recorded only 8 sessions). The sensors are accelerometers, rate gyros, and magnetic sensors mounted on different parts of the body (i.e., hands, upper and lower arms, and chest). There are 20 activity classes to be recognized, such as Open/Close hood, Open/Check/Close trunk.[1] We use leave-one-subject-out cross validation for evaluating the performance, and the classes are segmented and randomly distributed.

The second dataset, named *Opportunity dataset*, comprises sensory data of different modalities in a breakfast scenario. The database is fully described by Chavarriaga et al. (2013b) and is publicly available as a benchmark for HAR methods.[2] The dataset was recorded in an instrumented kitchen with two doors and a dining table at the center. The different sensor modalities are camera, microphone, localization,[3] reed switch, and inertial sensors. Inertial and localization sensors are all mounted on a jacket worn by the subject during the recording. For the current simulations we use a subset of this dataset including 4 subjects and motion jacket inertial sensors. Seventeen activities are detected, such as Open/Close door, Open/Close drawer. For this dataset we also evaluate the methods with a leave-one-subject-out cross validation and the data are previously segmented and randomly distributed.

4.2.2. Configurations

We tested different sensor configurations using one, two or three modalities for both datasets (i.e., accelerometer, gyro and magnetic sensors). The configurations are listed in Table 1. For one group of them, denoted as $config_l$ for location based, each classifier uses data from a set of sensors placed at the same location. For the other group, denoted as $config_s$ for sensor type based, we used one classifier per sensor per location (e.g. one classifier for the tri-axial accelerometer located on the right upper arm). For each case the number of classifiers are also provided in the table.

---

[1] For the complete list of the labels please refer to Stiefmeier et al. (2008).
[2] <http://archive.ics.uci.edu/ml/datasets/OPPORTUNITY+Activity+Recognition>.
[3] Ubisense: <http://www.ubisense.net/>.

4.2.3. Classification

For classification we used Quadratic Discriminant Analysis (QDA), unless otherwise stated. As features we used the mean and variance of the segmented signals. For each class (i.e., activity), the covariance matrix has been estimated, which is constrained to be diagonal due to the lack of sufficient data to estimate the full covariance matrix. The decisions of individual classifiers are fused using a naïve Bayesian approach. It uses the normalized confusion matrix of each classifier to represent the total reliability of this classifier for each class. The fused output can be computed as follows:

$$C_{out} = arg\,max_a\left(\prod_s P(C = c_i|O_s = o_s)\right), \quad o_s \in c_1, \ldots, c_m, \tag{9}$$

where O_s is the class label from the classifier s. Computing the probabilities is straightforward using Bayes rule:

Fig. 2. The effect of noise on the overall classification accuracy after fusion. HAR datasets – $Config_{l2}$. Left: rotational noise, the level is in degree. Right: additive noise, the level is in SNR.

$$P(C = c_i|O_s = o_s) \propto P(C)P(O_s = o_s|C = c_i). \tag{10}$$

We suppose that the priors $P(C)$ of all the classes are the same, and $P(O_s = o_s|C = c_i)$ is the confusion matrix estimated on the training data.

In addition, to show the generality of the approach we also perform simulations using other classification and fusion methods (Linear driscriminant analysis and Dempster–Shafer fusion).

4.2.4. Simulated anomalies

We simulate anomalies by introducing rotational and additive noise to the testing signals. Each type of noise was tested separately in the three datasets for different configurations (i.e., number of faulty classifiers, level of noise). In each simulation, noise is added to a fixed number of randomly chosen sensors for the entire test set. We report results on 10 repetitions of each configuration.

In the case of rotational noise, for each simulation the level noise is randomly chosen between 0° and 60° (with intervals of 10°). For additive noise the levels correspond to Signal to Noise Ratios of 0 to 10 dB with a step of 2 dB. As a reference, Fig. 2 shows how the classification performance in the HAR datasets degrades with the increase of the noise level and the amount of faulty classifiers.

5. Results

5.1. Synthetic dataset

Fig. 3 illustrates the effect of the number of faulty classifiers and their accuracies for the synthetic dataset. It shows the average computed distances (over the number of classes and classifiers) for the healthy and faulty classifiers (*Mean Healthy* and *Mean Faulty*, respectively) for the two proposed measures; Mahalanobis distance and mutual information. The results suggest that the IT measure performs better than DB when the number of noisy sensors and noise itself are higher. That is because a large number of faulty sensors will affect the fusion output, thus affecting the detection using the DB approach. In contrast, the IT approach only
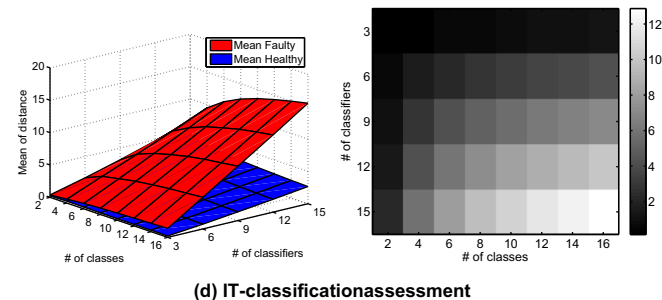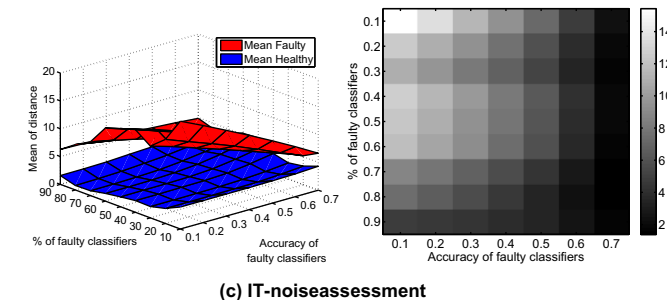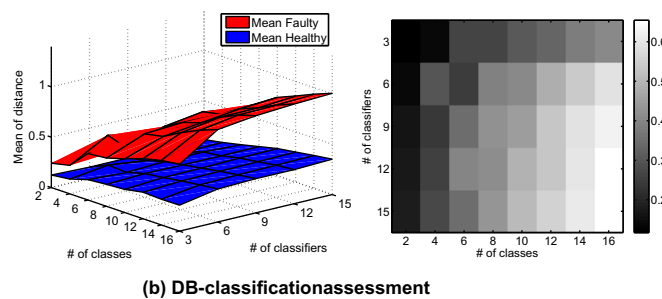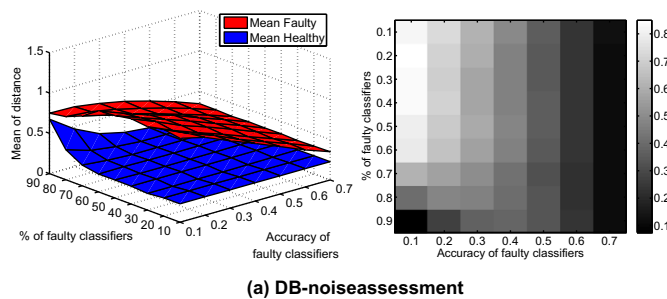

Fig. 3. Synthetic dataset. Left: the effect of (a) and (c) noise and ratio of noisy sensors, (b) and (d) number of classes and classifiers on the distance. Right: difference between faulty and healthy. Each plot shows the average distance computed for faulty and healthy classifiers (red and blue traces respectively). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

considers the relation between classifiers without considering the fused output.

In contrast, as it is shown in Fig. 3(b) and (d) when there are few classifiers or classes, DB is able to detect the anomalies better. In general, the number of classes has more effect on DB, while the number of classifiers has more effect on the IT-based measure.

5.2. Human activity datasets

Besides comparison with the LoLAD approach (i.e., GLR and OCSVM methods), we also compare the proposed methods with a reference performance corresponding to a 'perfect removal' approach. In this case we exclude from the ensemble those classifiers whose decisions are affected by the noise. This yields an upper bound on the performance that can be achieved based on the removal of anomalous classifiers. Obviously, such a removal cannot

be performed at run-time since there is no information about noise. For all simulations we set $k = 1$, window length $= 50$, and in the case of DB measure, $\Gamma = 5$.

The classification accuracies for the car manufacturing dataset are shown in Fig. 4 with different sensor configurations for rotational and additive noise, respectively. Similarly, Fig. 5 presents the results on the *Opportunity* dataset. Given the characteristics of the latter dataset, i.e., larger number and variety of realistic activities, the classification accuracy is lower (e.g. below 70% with QDA) than in the car manufacturing dataset.

Obviously, when no action is taken the performance decreases dramatically in both datasets (c.f. Fig. 2; green traces in Figs. 4 and 5). In general, GLR performs satisfactorily in all situations, as well as the IT-based method. Although for the latter, the performance decreases dramatically when there are less than three healthy classifiers in the Location-based configurations ($config_{l*}$). This is certainly due to the low number of available classifiers.
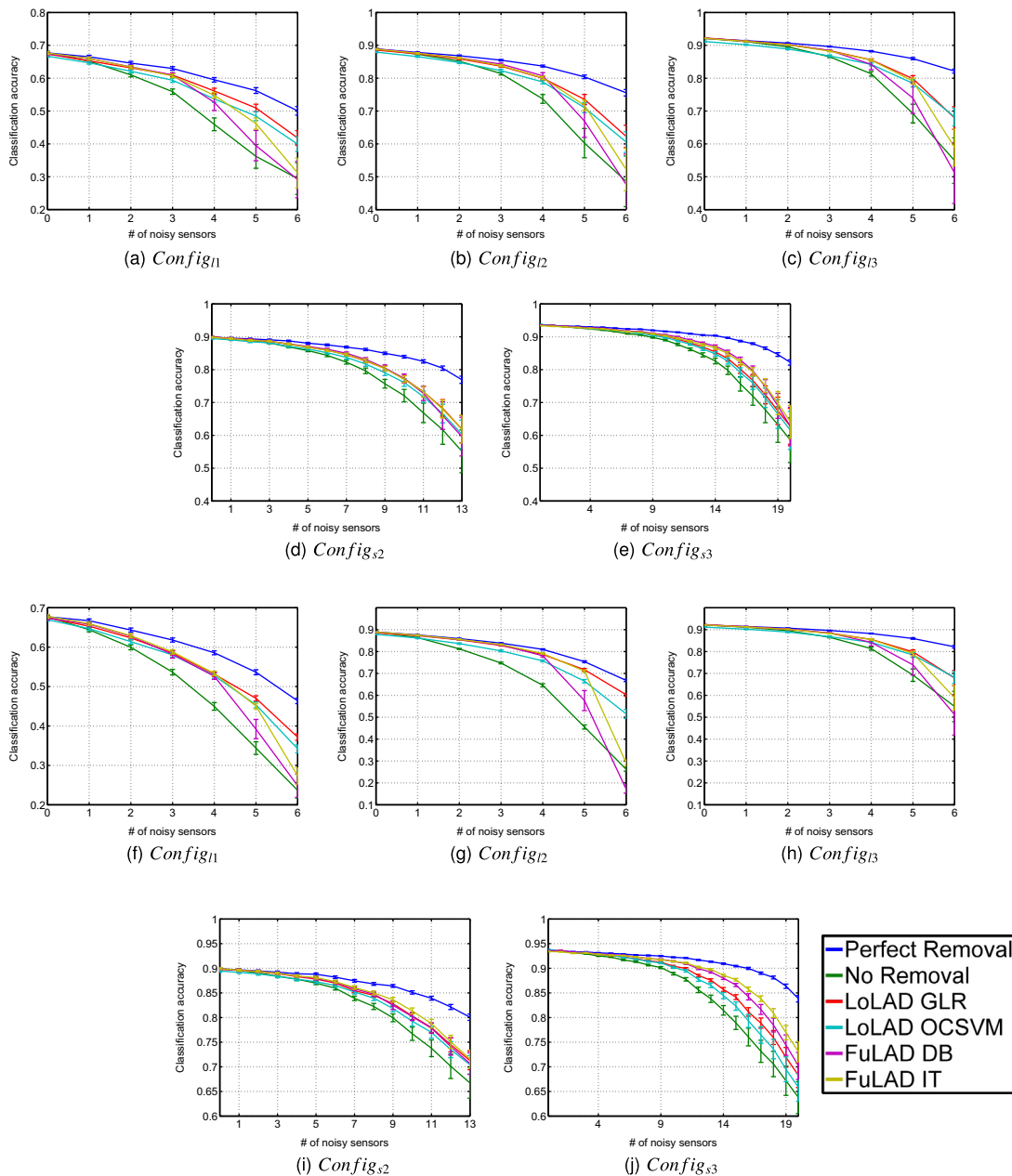

Fig. 4. Classification accuracy before and after anomaly detection. Car manufacturing dataset, (a)–(e) Rotational noise, (f)–(j) Additive noise. Note: (a) and (f) have a different scale on *y*-axis.
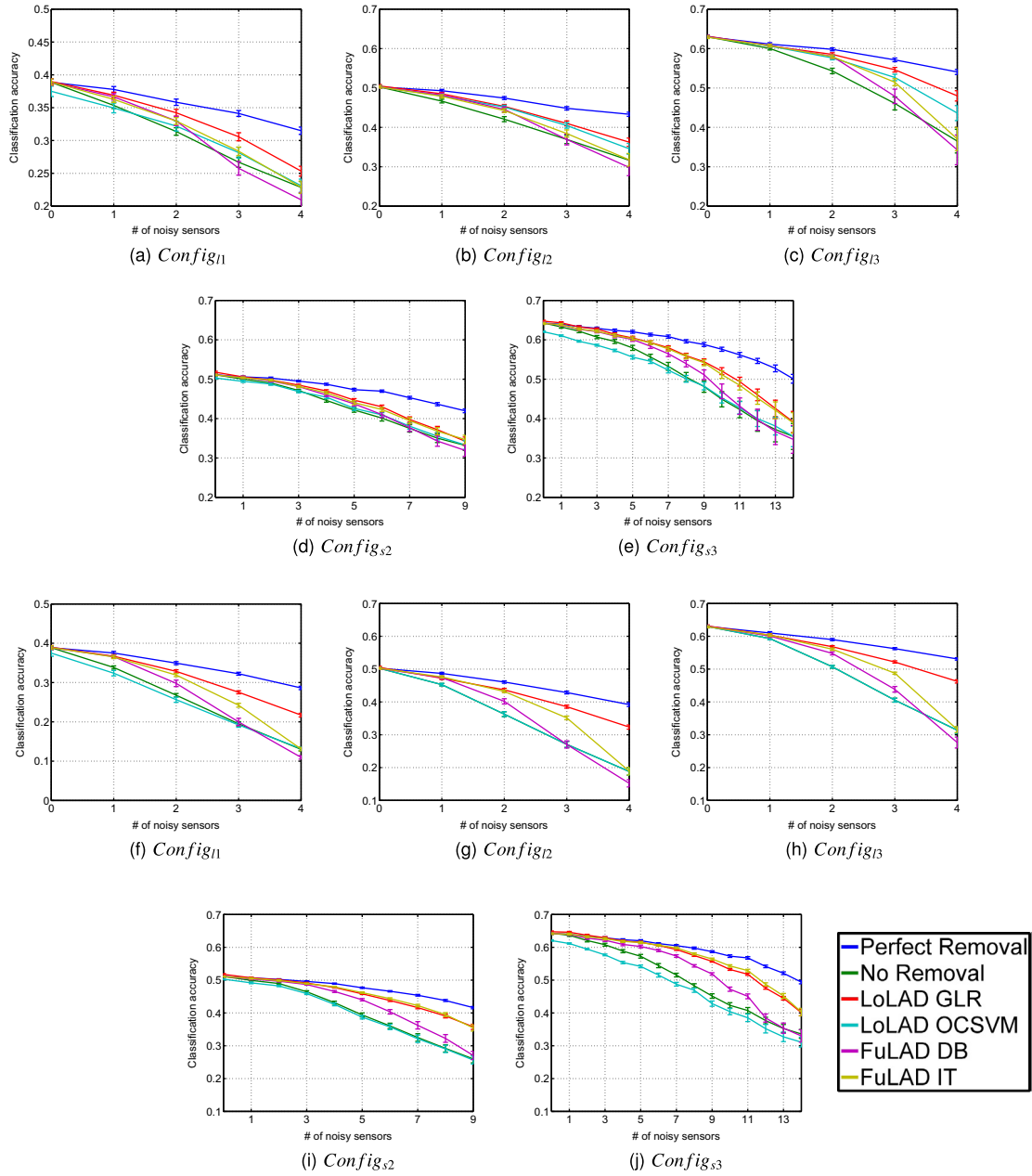
Fig. 5. Classification accuracy before and after anomaly detection. Opportunity dataset, (a)–(e) Rotational noise, (f)–(j) Additive noise. Note: (a) and (f) has a different scale on y-axis.

The performance of the DB measure is similar when less than half of the sensors are faulty. In turn, OCSVM reacts close to GLR for the car manufacturing dataset, but performs poorly on the second dataset. This results from its sensitivity to the number of features, and data distribution, highlighting the need for tuning the parameters for each configuration (we kept the parameters which was tuned for $config_{l2}$ for all other configurations).

Altogether, the results show that (a) both LoLAD and FuLAD succeed in increasing the classification accuracy, (b) There is a possibility of detecting anomalies at fusion level, with comparable performance to LoLAD.

As an example, Fig. 6 shows the average detection accuracy across subjects with respect to the level of noise, for LoLAD and FuLAD in the car manufacturing dataset using $Config_{l2}$. It is computed as:

$$DetectionAccuracy = \frac{TruePositive\ +\ TrueNegative}{\#of\ samples\ \times\ \#of\ sensors}, \tag{11}$$

where *TruePositive* is the correctly detected noisy sensors and *TrueNegative* is the correctly detected unaffected sensors and *# of samples* is the number of samples in the trial. They show that once the level of noise is low, the anomaly spotting is poor while by increasing the level of noise, it rises. This is due to the fact that these levels of noise do not affect the classifier decisions. Hence, the inability to spot them does not impact the overall system performance, see Fig. 2.

Fig. 7(a) and (c) show the average number of required iterations per pattern with respect to the number of noisy sensors and the level of rotational noise for both methods. During the DB iterations as anomalous classifiers are removed from the ensemble, the
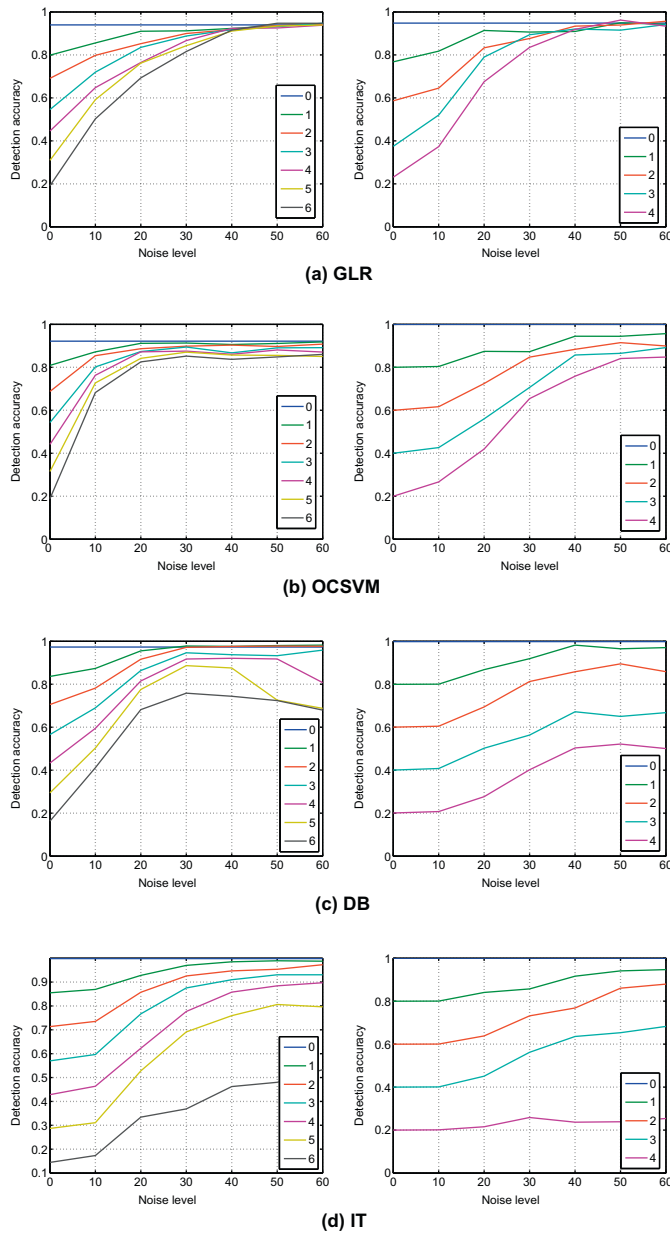
Fig. 6. Anomaly detection accuracy. Left: car manufacturing dataset, Right: opportunity dataset. *Config$_{I2}$*, rotational noise. Each curve corresponds to a different number of noisy sensors.

Fig. 7. The average number of iterations and changing decisions. Left: car manufacturing dataset, Right: opportunity dataset. *Config$_{I2}$*, rotational noise.

fusion output may change its decision; the average number of changes is shown in Fig. 7(b). Unsurprisingly, as the noise or the number of noisy sensors increases, the number of loops and changes of decisions also increases as more classifiers are recognized as anomalous.

The effect of the window length for DB and IT on the accuracy of detection (car manufacturing database) is shown in Fig. 8(a) and (b), respectively. Results show that a window length of 50 samples yields the best results. Of course, there is a trade-off of choosing the window length: lower values may lessen the performance while larger values impose a delay on the detection.

Fig. 8(c) and (d) illustrates how the threshold value affects the anomaly detection and classification accuracy, for DB method. It shows that a threshold value equal to one yields good results and it is reasonable since only distances in the range of the training dataset are permitted. Also, we empirically found that for this
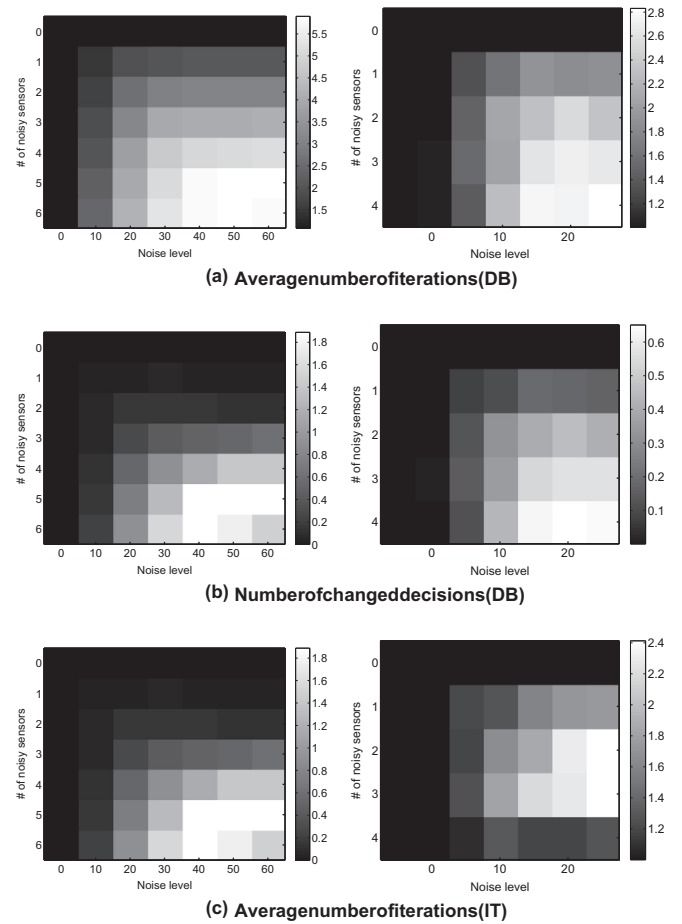
dataset, setting the value of Γ at least twice the threshold value and at most three times the threshold achieves the best results.

To demonstrate that the methods do not depend on the classification or fusion method we tested them when a Linear Discriminative Classifier (LDA) is used, as well as when Dempster–Shafer fusion (Kuncheva et al., 2001) is applied for combining the classifiers. Fig. 9 shows the results obtained on the car manufacturing dataset with rotational noise. It is obvious that the IT approach is not dependent on the fusion method, since it is not using its output. But as the results show, the DB approach also is not particularly sensitive to the type of classifier or fusion methods. Comparing the performance of QDA (Fig. 4(b)) and LDA (Fig. 9(a)), the former appears more robust to this kind of noise. As expected, Dempster–Shafer fusion (Fig. 9(c)) is also less sensitive to noise than the naïve fusion.

6. Discussion

Nowadays, multimodal sensor networks are increasingly used to gather information about the environment and improve context and activity recognition. Classifier ensembles can provide an efficient and flexible approach to achieve this. However, sensors are always prone to noise and anomalies that may degrade the system performance. Therefore, the ability of detect such anomalies may allow to dynamically adapt the system, bringing more robustness to the recognition chain.

Anomaly detection can be performed at the data/feature or fusion level. This choice depends on several factors, including the
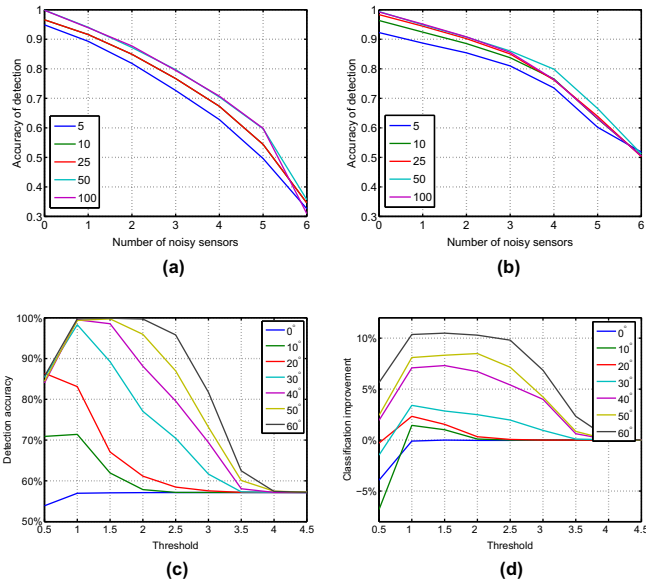
Fig. 8. (Top) Detection accuracy with respect to the length of moving average window. Car manufacturing dataset – $Config_{I2}$. Rotational noise. (a) IT measure, (b) DB measure. (Bottom) Effect of different threshold values – DB measure. Car manufacturing dataset, $config_{I2}$, $\Gamma = 5$, 3 noisy sensors out of 7. (c) Anomaly detection accuracy. (d) Increase in classification performance with different rotation levels.

application requirements and the available technology. Complementing existing methods for low level anomaly detection, in this paper we assess the possibility of detecting anomalies at the fusion level with a comparable performance.

The proposed method relies on a distance measure between the expected system behavior (based on training data), and the data collected at run time. We provide two metrics, one based on the Mahalanobis distance and the other one based on information theoretical measures. We characterize the proposed methods using both an artificially generated dataset and real data from two scenarios of human activity recognition using wearable sensors. The final results show the possibility of recognizing the anomalous behavior at the fusion level, resulting in graceful degradation of the classification performance.

Regarding the proposed measures, the IT approach has an advantage over the DB, as it does not take the fused decision into account. In that sense, it could be more suitable for dynamic classifier ensembles – where sensors (and their corresponding trained classifiers (Kurz et al., 2011a; Calatroni et al., 2011)) – may be added or removed from the network on runtime. Having said so, in this approach the length of the window should be carefully chosen in order to have a reasonable estimate of the mutual information.
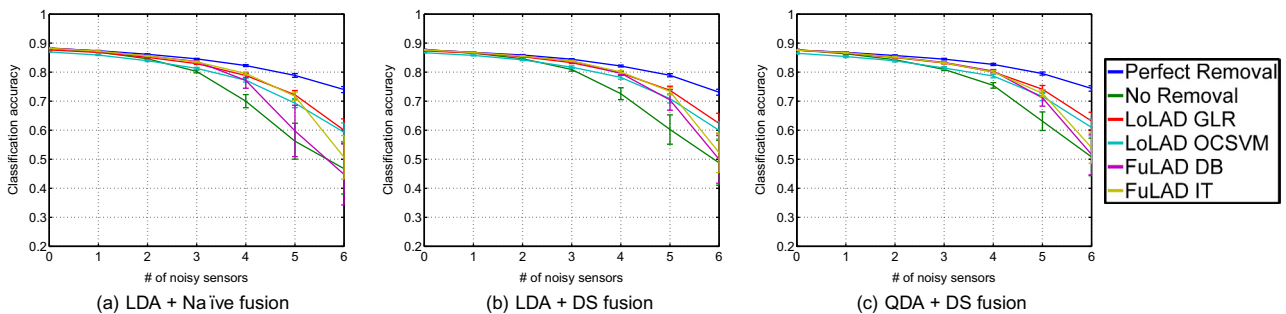
An interesting characteristic of FuLAD is that nodes in the ensemble will be removed only if the corresponding classifier is affected by the anomaly. For example, if an accelerometer is slightly rotated, the classifier decisions may not change, therefore it may still be included in the fusion. This property is not provided by LoLAD methods since do not take into account information about classification.

The conventional approaches of LoLAD create a model of data and detect anomalies when the new data does not follow the model. Even when the best model is chosen or the parameters are perfectly estimated, it may not cover all the possible anomalies. In contrast, FuLAD is independent of the type of anomaly. There is no need to investigate all the possible anomalies of each sensor. Only monitoring the classifier decisions could bring information if the sensor is affected or not. Moreover, in the case of multimodal setup, LoLAD requires the development of methods and parameters for each modality. In contrast, FuLAD is independent of the modality of the sensors used, because the working space is composed of the classifier outputs, independent of their input domain.

Another characteristic of FuLAD is that it puts the burden of the computation of anomaly detection on a central node (fusion node). This has consequences in the system design with respect to the amount of computational power at each node and the communication requirements (i.e., need to transfer classification labels or raw data). Noteworthy, this property, inherited from the classifier ensemble approach, allows for more flexibility in the case of dynamically changing, self-adaptive recognition systems (Roggen et al., 2009).

7. Conclusions

We propose a method for detecting anomalies in classifier ensembles. The detection process take place at the classifier fusion level (FuLAD), in contrast to the most common approach of doing it at the data/feature level (LoLAD). The proposed approach is independent of the sensor modality and the type of noise or anomalies it has to deal with. In this work, we thoroughly tested the method on datasets from activity recognition with on-body sensors, but it can be applied in any other application. Although there are many alternative approaches that consider fault and change detection at raw data or feature level, this work is the first one which explores them at fusion level.

We propose two different approaches: one is based on the Mahalanobis distance which compares the individual classifier decisions and the fused output, while the second approach is based on the mutual information between classifiers in the ensemble. They spot the faulty classifiers based on the behavior of the classifiers from the training set. Upon detection of anomalous classifiers, compensatory strategies (such as removing them from the ensemble) can be applied to improve the performance.

(a) LDA + Naïve fusion   (b) LDA + DS fusion   (c) QDA + DS fusion

Fig. 9. Performance of different classification and fusion methods. Car manufacturing dataset, rotational noise, $Config_{I2}$.

Besides testing on synthetic data, we applied our method to two activity recognition datasets, allowing us to test it in realistic conditions using different sensor modalities and configurations. The results show the feasibility of change and anomaly detection at the fusion level and the improvement of the classification for all the cases. Although the proposed method does not build explicit models for each sensor type or modality in the ensemble, its performance is comparable to that achieved with LoLAD. The IT measure can detect anomalies just as well as the feature level method, even when only two classifiers remain healthy. Whereas the DB measure works satisfactorily until half of the classifiers are affected.

Alternatively, the distance measures presented here can be used to quantify the confidence of each classifier in the ensemble at runtime. This measure can then be used to dynamically reconfigure the ensemble, taking into account different specifications including classification performance, as well as communication costs or energy consumption (Kurz et al., 2011b).

Acknowledgments

References

Alippi, C., Roveri, M., 2006a. An adaptive cusum-based test for signal change detection. In: IEEE Internat. Symposium on Circuits and Systems (ISCAS). IEEE.

Alippi, C., Roveri, M., 2006b. A computational intelligence-based criterion to detect non-stationarity trends. In: Internat. Joint Conf. on Neural Networks (IJCNN), pp. 5040–5044.

Alippi, C., Roveri, M., 2008. Just-in-time adaptive classifiers. Part I: Detecting nonstationary changes. IEEE Trans. Neural Networks 19, 1145–1153.

Anderson, J., 2008. A generalized likelihood ratio test for detecting land mines using multispectral images. IEEE Geosci. Remote Sens. Lett. 5, 547–551.

Andrieu, C., Doucet, A., Singh, S.S., Tadic, V.B., 2004. Particle methods for change detection, system identification, and control. Proc. IEEE 92, 423–438.

Balzano, L.K., 2007. Addressing fault and calibration in wireless sensor networks. Master's Thesis, University of California Los Angeles.

Basseville, M., Nikiforov, I.V., 1993. Detection of Abrupt Changes: Theory and Application. Prentice-Hall.

Betta, G., Pietrosanto, A., 1998. Instrument fault detection and isolation: state of the art and new research trends. In: IEEE Conference Proceedings Instrumentation and Measurement Technology Conference, pp. 483–489.

Boisgontier, H., Noblet, V., Heitz, F., Rumbach, L., Armspach, J.P., 2009. Generalized likelihood ratio tests for change detection in diffusion tensor images. In: IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 811–814.

Bolton, R.J., Hand, D.J., H, D.J., 2002. Statistical fraud detection: A review. Statist. Sci. 17, 235–255.

Bosc, M., Heitz, F., Armspach, J.P., Namer, I., Gounot, D., Rumbach, L., 2003. Automatic change detection in multimodal serial MRI: Application to multiple sclerosis lesion evolution. NeuroImage 20, 643–656.

Brotherton, T., Johnson, T., 2001. Anomaly detection for advanced military aircraft using neural networks. In: IEEE Proceedings of Aerospace Conference, pp. 3113–3123.

Calatroni, A., Roggen, D., Tröster, G., 2011. Automatic transfer of activity recognition capabilities between body-worn motion sensors: Training newcomers to recognize locomotion. In: 8th Internat. Conf. on Networked Sensing Systems (INSS), Penghu, Taiwan.

Camps-Valls, G., Gomez-Chova, L., Munoz-Mari, J., Rojo-Alvarez, J., Martinez-Ramon, M., 2008. Kernel-based framework for multitemporal and multisource remote sensing data classification and change detection. IEEE Transactions on Geoscience and Remote Sensing 46, 1822–1835.

Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly detection: A survey. ACM Comput. Surv. 41, 1–58.

Chang, C.C., Lin, C.J., 2011. LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol. 2, 27:1–27:27, Software available at: <http://www.csie.ntu.edu.tw/cjlin/libsvm>.

Chavarriaga, R., Sagha, H., Millán, J.d.R., 2011. Ensemble creation and reconfiguration for activity recognition: An information theoretic approach. In: IEEE Internat. Conf. on Systems, Man, and Cybernetics.

Chavarriaga, R., Bayati, H., Millán, J.d.R., 2013a. Unsupervised adaptation for acceleration-based activity recognition: robustness to sensor displacement and rotation. Pers. Ubiquitous Comput. 17, 479–490.

Chavarriaga, R., Sagha, H., Calatroni, A., Digumarti, S.T., Tröster, G., Millán, J.d.R., Roggen, D., 2013b. The opportunity challenge: A benchmark database for on-body sensor-based activity recognition. Pattern Recognition Lett. this issue.

Chen, J., Kher, S., Somani, A., 2006. Distributed fault detection of wireless sensor networks. In: Proc. Workshop on Dependability Issues in Wireless ad Hoc Networks and Sensor Networks. ACM, New York, NY, USA, pp. 65–72.

Das, K., Bhaduri, K., Votava, P., 2011. Distributed anomaly detection using 1-class SVM for vertically partitioned data. Statist. Anal. Data Min. 4, 393–406.

Demirbas, M., 2004. Scalable design for fault-tolerance for wireless sensor networks. Ph.D. Thesis, Ohio State University, Computer and Information Science.

Ding, S., 2008. Model-Based Fault Diagnosis Techniques. Springer.

El Falou, W., Khalil, M., Duchene, J., 2000. AR-based method for change detection using dynamic cumulative sum. In: 7th IEEE Internat. Conf. on Electronics, Circuits and Systems, ICECS, vol.1, pp. 157–160.

Förster, K., Brem, P., Roggen, D., Tröster, G., 2009. Evolving discriminative features robust to sensor displacement for activity recognition in body area sensor networks. In: 5th Internat. Conf. on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 43–48.

Fu, Y., Cao, L., Guo, G., Huang, T.S., 2008. Multiple feature fusion by subspace learning. In: Proc. Internat. Conf. on Content-Based Image and Video Retrieval. ACM, New York, NY, USA, pp. 127–134.

Gage, J., Murphy, R.R., 2010. Sensing assessment in unknown environments: A survey. IEEE Trans. Syst. Man Cybernet. Part A: Syst. Humans 40, 1–12.

Hwang, I., Kim, S., Kim, Y., Seah, C.E., 2010. A survey of fault detection, isolation, and reconfiguration methods. IEEE Trans. Control Syst. Technol. 18, 636–653.

Ide, T., Papadimitriou, S., Vlachos, M., 2007. Computing correlation anomaly scores using stochastic nearest neighbors. In: 7th IEEE Internat. Conf. on Data Mining, ICDM, pp. 523–528.

Jakubek, S., Strasser, T., 2002. Fault-diagnosis using neural networks with ellipsoidal basis functions. In: Proc. American Control Conference, pp. 3846–3851.

Koushanfar, F., Potkonjak, M., Sangiovanni-Vincentelli, A., 2002. On-line fault detection of sensor measurements. In: IEEE Sensors, pp. 69–85.

Krishnamachari, B., Iyengar, S.S., 2003. Efficient and fault-tolerant feature extraction in wireless sensor networks. In: Proc. 2nd Internat. Workshop on Information Processing in Sensor Networks.

Kuncheva, L.I., Bezdek, J.C., Duin, R.P.W., 2001. Decision templates for multiple classifier fusion: An experimental comparison. Pattern Recognition 34, 299–314.

Kunze, K., Lukowicz, P., 2008. Dealing with sensor displacement in motion-based onbody activity recognition systems. In: Proc. 10th Internat. Conf. on Ubiquitous Computing. ACM, New York, NY, USA, pp. 20–29.

Kurz, M., Hölzl, G., Ferscha, A., Calatroni, A., Roggen, D., Tröster, G., 2011a. Real-time transfer and evaluation of activity recognition capabilities in an opportunistic system. In: 3rd Internat. Conf. on Adaptive and Self-Adaptive Systems and Applications, pp. 73–78.

Kurz, M., Hölzl, G., Ferscha, A., Sagha, H., Millán, J.d.R., Chavarriaga, R., 2011b. Dynamic quantification of activity recognition capabilities in opportunistic systems. In: Fourth Conference on Context Awareness for Proactive Systems: CAPS.

Lane, T., Brodley, C.E., 1998. Approaches to online learning and concept drift for user identification in computer security. In: KDD. AAAI Press, pp. 259–263.

Li, W., Raghavan, H., Shah, S., 2003. Subspace identification of continuous time models for process fault detection and isolation. J. Process Control 13, 407–421.

Lorden, G., 1971. Procedures for reacting to a change in distribution. Ann. Math. Statist. 42, 1897–1908.

Lukashevich, H., Nowak, S., Dunker, P., 2009. Using one-class SVM outliers detection for verification of collaboratively tagged image training sets. In: IEEE Internat. Conf. on Multimedia and Expo, ICME 2009, pp. 682–685.

Luo, X., Dong, M., Huang, Y., 2006. On distributed fault-tolerant detection in wireless sensor networks. IEEE Trans. Comput. 55, 58–70.

Page, E.S., 1954. Continuous inspection schemes. Biometrika 41, 100–115.

Rajasegarar, S., Leckie, C., Palaniswami, M., Bezdek, J., 2007. Quarter sphere based distributed anomaly detection in wireless sensor networks. In: IEEE Internat. Conf. on Communications, ICC, pp. 3864–3869.

Rajasegarar, S., Leckie, C., Palaniswami, M., Bezdek, J.C., 2006. Distributed anomaly detection in wireless sensor networks. In: 10th IEEE Singapore Internat. Conf. on Communication systems, ICCS, pp. 1–5.

Roggen, D., Förster, K., Calatroni, A.G.T., 2011. The adARC pattern analysis architecture for adaptive human activity recognition systems. J. Ambient Intell. Human. Comput., 1–18.

Roggen, D., Förster, K., Calatroni, A., Holleczek, T., Fang, Y., Tröster, G., Lukowicz, P., Pirkl, G., Bannach, D., Kunze, K., Ferscha, A., Holzmann, C., Riener, A., Chavarriaga, R., Millán, J.d.R., 2009. OPPORTUNITY: Towards opportunistic activity and context recognition systems. In: 3rd IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications.

Ruiz, L.B., Siqueira, I.G., Oliveira, L.B.e., Wong, H.C., Nogueira, J.M.S., Loureiro, A.A.F., 2004. Fault management in event-driven wireless sensor networks. In: Proc. 7th ACM Internat. Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems. ACM, New York, NY, USA, pp. 149–156.

Ruta, D., Gabrys, B., 2000. An overview of classifier fusion methods. Comput. Inform. Syst. J. 7, 1–10.

Sagha, H., Chavarriaga, R., Millán, J.d.R., 2010. A probabilistic approach to handle missing data for multi-sensory activity recognition. In: 12th ACM Internat. Conf. on Ubiquitous Computing. Copenhagen, Denmark.

Sagha, H., Millán, J.d.R., Chavarriaga, R., 2011a. Detecting and rectifying anomalies in opportunistic sensor networks. In: Internat. Conf. on Body Sensor Networks (BSN).

Sagha, H., Millán, J.d.R., Chavarriaga, R., 2011b. Detecting anomalies to improve classification performance in an opportunistic sensor network. In: 7th IEEE Internat. Workshop on Sensor Networks and Systems for Pervasive Computing, PerSens, Seattle.

Sannen, D., Lughofer, E., Van Brussel, H., 2010. Towards incremental classifier fusion. Intell. Data Anal. 14, 3–30.

Schonlau, M., DuMouchel, W., Ju, W.H., Karr, A.F., Theus, M., Vardi, Y., 2001. Computer intrusion: detecting masquerades. Statist. Sci. 16, 58–74.

Severo, M., Gama, J., 2010. Change detection with kalman filter and cusum. In: May, M., Saitta, L. (Eds.), Ubiquitous Knowledge Discovery, Lecture Notes in Computer Science, vol. 6202. Springer, pp. 148–162.

Shannon, C.E., 2001. A mathematical theory of communication. ACM SIGMOBILE Mobile Comput. and Comm. Rev. 5, 3–55.

Sheth, A., Hartung, C., Han, R., 2005. A decentralized fault diagnosis system for wireless sensor networks. In: IEEE Internat. Conf. on Mobile Adhoc and Sensor Systems Conference.

Smyth, P., 1994. Hidden markov models for fault detection in dynamic systems. Pattern Recognition 27, 149–164.

Snoussi, H., Richard, C., 2007. Distributed Bayesian fault diagnosis of jump Markov systems in wireless sensor networks. Internat. J. Sensor Networks 2, 118–127.

Steinhoff, U., Schiele, B., 2010. Dead reckoning from the pocket – an experimental study. In: IEEE Internat. Conf. on Pervasive Computing and Communications (PerCom), pp. 162–170.

Stiefmeier, T., Roggen, D., Ogris, G., Lukowicz, P., Tröster, G., 2008. Wearable activity tracking in car manufacturing. IEEE Pervasive Comput. Mag., 42–50.

Wang, T.Y., Han, Y.S., Varshney, P.K., Chen, P.N., 2005. Distributed fault-tolerant classification in wireless sensor networks. IEEE J. Sel. Areas Comm. 23, 724–734.

Wei, T., Huang, Y., Chen, C., 2009. Adaptive sensor fault detection and identification using particle filter algorithms. IEEE Trans. Syst. Man Cybernet. Part C: Appl. Rev. 39, 201–213.

Wu, B., Chen, L., 2006. Use of partial cumulative sum to detect trends and change periods for nonlinear time series. J. Econ. Manage. 2, 123–145.

Xie, J., Yan, G., Xie, K., Lin, T.Y., 2007. Neuro-fuzzy model-based cusum method application in fault detection on an autonomous vehicle. In: Proceedings of the 2007 IEEE Internat. Conf. on Granular Computing. IEEE Computer Society, Washington, DC, USA, p. 548.

Xu, L., Li, J., 2007. Iterative generalized-likelihood ratio test for MIMO radar. IEEE Trans. Signal Process. 55, 2375–2385.

Yao, Y., Sharma, A., Golubchik, L., Govindan, R., 2010. Online anomaly detection for sensor systems: A simple and efficient approach. Perform. Eval. 67, 1059–1075.

Zhuang, L., Dai, H., 2006. Parameter optimization of kernel-based one-class classifier on imbalance text learning. In: Proc. 9th Pacific Rim Internat. Conf. on Artificial Intelligence. Springer-Verlag, Berlin, Heidelberg, pp. 434–443.