Chemometrics and Intelligent Laboratory Systems 132 (2014) 91–102

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

journal homepage: www.elsevier.com/locate/chemolab

Active classification with arrays of tunable chemical sensors

Rakesh Gosangi, Ricardo Gutierrez-Osuna

Department of Computer Science and Engineering, Texas A&M University, United States

ARTICLE INFO

Article history:
Received 20 October 2013
Received in revised form 5 January 2014
Accepted 6 January 2014
Available online 15 January 2014

Keywords:
Active sensing
Sensor arrays
Chemical classification
Metal-oxide sensors
Fabry–Perot interferometers

ABSTRACT

This paper presents Posterior-Weighted Active Search (PWAS), an active-sensing algorithm for classification of volatile compounds with arrays of tunable chemical sensors. The algorithm combines concepts from feature subset selection and sequential Bayesian filtering to optimize the sensor array tunings on-the-fly based on information from previous measurements. Namely, the algorithm maintains an estimate of the posterior probability associated with each chemical class, and updates it sequentially upon arrival of each new sensor observations. The updated posteriors are then used to bias the selection of the next sensor tunings towards the most likely classes, in this way reducing the number of measurements required for discrimination. We characterized PWAS on a database of infrared absorption spectra with 250 analytes, and then validated it experimentally on an array of metal-oxide sensors. Our results show that PWAS outperforms passive-sensing approaches based on sequential forward selection, both in terms of classification performance and robustness to noise in sensor measurements.

1. Introduction

Chemical sensors are generally used as first-order devices, where one measures the sensor's response at a fixed setting, e.g., absorption of an optical sensor at a specific wavelength, or conductivity of a solid-state sensor at a specific operating temperature [1]. In many cases, additional information can be extracted by modulating some internal property of the sensor. As an example, measuring the conductivity of a metal-oxide chemical sensor at different temperatures can provide a wealth of discriminatory information [2]. However, this additional information comes at a cost, such as sensing times or power consumption. For this reason, feature subset selection (FSS) techniques are commonly used to identify a subset of the most informative sensor configurations.

Over the past decade, a handful of investigators in the chemical sensor community have explored active sensing as an alternative to FSS [3–7]. In contrast with FSS, where the sensor configurations are optimized off-line, active sensing adapts the sensor configurations in real-time based on information obtained from previous measurements. In previous work [6,7], we showed that active sensing can achieve higher classification performance than FSS with fewer measurements and provides a trade-off between sensing costs and classification performance. Unfortunately, these active-sensing methods were developed for individual sensors, and do not scale up to sensor arrays. First, the number of operating configurations for a sensor array grows exponentially with the size of the array; given an N-sensor array with D configurations per sensor, there exist D^N unique configurations. Second, chemical sensor arrays are notoriously collinear (i.e., their response across multiple chemicals is correlated), so additional strategies are needed to account for correlation among sensors.

This article proposes Posterior-Weighted Active Search (PWAS), an active-sensing algorithm for sensor arrays that addresses both issues (combinatorial explosion and sensor collinearity). PWAS performs active sensing by optimizing the sensors' tunings towards the most likely classes at each sensing step; for this purpose, PWAS uses the sequential Bayesian filter of our prior work [6,7] to update the posterior probability of each class upon arrival of each new measurement. To cope with the combinatorial explosion in sensor array tunings, PWAS uses local search to build the sensor array configurations incrementally (one sensor at a time). Finally, to cope with sensor collinearity PWAS uses one of the two objective functions we have developed for this purpose. The first objective function is a parametric filter derived from the multivariate Fisher score [8], and weighs the within-class and between-class scatter matrices according to the estimated class posteriors. The second objective function is a non-parametric information-theoretic filter that measures feature relevance and feature redundancy with respect to the class posteriors. PWAS operates following a 'search-sense-update' sequence. During the 'search' step, the algorithm uses the local search and objective functions to build the next sensor array configuration. During the 'sense' step, the algorithm takes sensor measurements using the selected configuration. During the final 'update' step, the algorithm re-estimates the class posteriors by feeding the measured sensor responses to a sequential Bayesian update equation. This search-sense-update process is continued until a predefined stopping

E-mail addresses: rakesh@cse.tamu.edu (R. Gosangi), rgutier@cse.tamu.edu (R. Gutierrez-Osuna).

0169-7439/$ – see front matter © 2014 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.chemolab.2014.01.001

criterion is met, at which point the final class label is declared based on maximum a posteriori (MAP) criterion.

The rest of the paper is organized as follows. Section 2 provides background material on active sensing, and its applications to chemical sensors. Section 3 describes the proposed PWAS algorithm with a focus on the two objective functions, which are novel contributions of this paper. Section 4 provides a thorough evaluation of PWAS against FSS and random feature selection on a database of low-resolution absorption spectra containing 250 chemicals. Section 5 describes the experimental setup and results from validating the approach on an array of commercial MOX sensors. The article concludes with a discussion of results and directions for future work.

2. Background

The idea of active sensing originates from the theory of 'active perception' [9,10], which states that an organism actively probes the environment to enhance its ability to extract behaviorally relevant information. The concept caught on during the 1980s in the robotics and vision community [11], where it was used to denote control strategies that dynamically adapted sensing configurations as the sensor interacted with its environment. Since then, active-sensing principles have been used widely in vision, robotics and target tracking to address various computational problems such as classification, detection, estimation, sampling, and tracking. This prior work has shown that active sensing can manage sensing resources more efficiently than passive sensing, and can also provide a balance between sensing costs and sensing accuracy [12].

In a classic paper on active vision, Aloimonos et al. [13] showed that several computer vision problems that are ill-posed and non-linear with passive observers become well-posed and linear by use of an active observer (i.e., one that can control the parameters of its apparatus, such as focal length or orientation). Over the last two decades, active sensing has also been used for motion tracking [14], scene exploration and reconstruction [15], face recognition [16], vision-based localization and mapping [17], and scene segmentation [18].

Active-sensing strategies have also been broadly used in robotic navigation [19], localization [20,21], simultaneous localization and mapping [17], and robotic exploration [22]. A classical active-sensing problem in robotics is to decide where to move the robot (location decisions) and how to reconfigure its sensors (sensing decisions) [23]. These problems arise from the exploration-exploitation dilemma, which involves a trade-off between immediate rewards (exploitation) such as bringing the robot closer to its goal, and long-term effects (exploration) such as gathering information through landmarks, surrounding obstacles, or reading signs.

Along these lines, active sensing has also received attention for use in military scenarios, specifically for tracking dynamic targets with stationary [24] and mobile sensors [25–27]. The target-tracking problem involves estimating locations and velocities of multiple moving targets (e.g., ground vehicles) using surveillance sensors such as radars, sonars, or electro-optical sensors. One of the central challenges in target tracking is selecting the next sensing action; this involves choosing sensors, setting their configurations (such as pointing angles, dwell lengths, etc.), or possibly moving them to another location.

2.1. Prior work in active chemical sensing

Though not as broadly as in vision, robotics and target tracking, active-sensing principles have been applied to various chemical sensing problems as well, including odor generation, chemical discrimination, and data collection. To our knowledge, the earliest use of active sensing in the chemical/olfaction domain is the work of Nakamoto et al. [28,29] on odor generation. The objective of this work was to reproduce an odor blend by creating a mixture from its individual components. The authors developed an active-control algorithm that adjusted the mixture ratio so that the response of a gas sensor array to the mixture matched the response of the array to the odor blend.

Active sensing has also been used for chemical discrimination problems. As an example, Priebe et al. [3] developed a statistical pattern classification method termed Integrated Sensing and Processing Decision Trees (ISPDT). This method builds a decision tree to partition feature space hierarchically; nodes close to the root provide good clustering of examples regardless of class labels, whereas nodes at the leaves seek to discriminate examples from different classes. Each internal node defines a sensor configuration (a feature) and its children the possible observations. The decision tree is used to guide the sensing process as follows. First, the sensor is operated according to the feature at the root node. The resulting observation falls into one of the child nodes, which determines the next step: either acquire new measurements (if it is an internal node), or to classify the sample and terminate sensing (in case of a leaf node). The authors evaluated ISPDT on a dataset containing the response of an optical sensor array to trichloroethylene (a carcinogenic industrial solvent) in complex backgrounds; ISPDT reduced misclassification rates by 50%, while requiring only 20% of all the sensors to make any individual classification.

More recently, Lomasky et al. [30] developed an "active class selection" method to optimize the generation of training datasets for e-nose applications. Their approach was based on principles from active learning, a machine-learning technique where the learning algorithm chooses the training samples from which it learns. Active learning assumes that many training instances are readily available and that the cost lies in labeling them (e.g. through human annotation). However, in e-nose applications the costs are not associated with labeling existing samples but with the more laborious process of collecting new ones. Therefore, the active class selection problem involves choosing the class of the next training instance, whereas the active learning problem deals with choosing the next training instance to be labeled. Lomasky's approach consists of generating the next set of n training instances in proportion to the instability of class boundaries, measured in terms of the number of test instances whose classification labels change upon inclusion of the previous set of n training instances. The authors validated the approach on an experimental dataset from an array of fluorescent micro-bead sensors exposed to six organic chemicals and their mixtures. The results show that active class selection can minimize the number of new training instances needed to obtain the maximal classification performance.

An optical implementation of active-sensing principles was proposed by Dinakarababu et al. [4] for rapid identification of chemicals. In this work, the authors developed an Adaptive Feature Specific Spectrometer (AFSS), a digital micro-mirror device capable of multiplexing certain spectral bands and directing them onto a photo-detector. In this fashion, the system is able to measure the projection of the incoming spectral density onto a set of basis vectors, rather than measure the spectral density directly. The basis vectors are the eigenvectors of a probabilistically-weighted covariance matrix, with the probabilities corresponding to the likelihoods of different classes based on previous measurements.

Our early investigations of active sensing focused on the problem of discriminating M chemicals at fixed concentration with a single temperature-modulated metal-oxide sensor. In [6], we presented a partially observable Markov decision process (POMDP) solution to this problem, and proposed a myopic policy that selected sensing actions based on the expected reduction in Bayesian risk. In subsequent work [5], we reformulated the problem to not only identify chemical samples but also estimate their concentrations using Fabry–Perot interferometers. This new approach used nonnegative matrix factorization [31] to create concentration-independent absorption profiles of different chemicals, and linear least squares to fit sensor observations to the response profiles. In latter work, we extended the active-sensing method to estimate the concentration of mixtures with known components [32], and the more challenging problem of estimating concentrations of mixtures with unknown components [33].

3. Methods

Consider an array of N tunable chemical sensors $\boldsymbol{S} = \langle s_1, s_2, \ldots, s_N \rangle$, where each sensor S_i can be operated at D distinct configurations $\boldsymbol{\rho} = \langle \rho_1, \rho_2, \ldots, \rho_D \rangle$.[1] As an example, in the case of a Fabry–Perot interferometer (see Section 4) each configuration corresponds to a particular absorption wavelength, whereas in a metal-oxide chemoresistor (see Section 5) each configuration corresponds to a pulse at a particular operating temperature (see Fig. 7).

The sensor array is exposed to a chemical, and we seek to determine the identity of this chemical from a list of M possible targets $\boldsymbol{\omega} = \langle \omega_1, \omega_2, \ldots, \omega_M \rangle$. For this purpose, the array is operated by a centralized controller, which can adjust the configuration of each sensor individually. The goal of the controller is to optimize the sequence of T action vectors $\boldsymbol{A_T} = \{\boldsymbol{a}_1, \boldsymbol{a}_2 \ldots \boldsymbol{a}_T\}$, where $\boldsymbol{a}_t$ is an N-dimensional vector denoting the configuration at time t for each sensor in the array. As an example, for an array with $N \geq 3$ sensors and $D \geq 6$ configurations, the action vector $\langle (s_1, \rho_2), (s_2, \rho_6), (s_3, \rho_5) \rangle$ would correspond to simultaneously operating sensor s_1 at configuration ρ_2, s_2 at ρ_6, and s_3 at ρ_5.

The conventional solution to this problem is off-line optimization; given a training set containing the response of each sensor at every configuration for different chemicals, one would find the best TN sensing configurations with the constraint that exactly T configurations are chosen for each sensor. These configurations would then be organized into a *fixed* sequence of T action vectors, and used at sensing time. However, this 'passive' approach is limited because it uses the same sequence of action vectors for any test sample, regardless of the information arriving from the sensors at each time step.

In contrast, our proposed approach (Posterior-Weighted Active Search; PWAS) adapts the action vectors based on information acquired at each measurement cycle. The approach follows a 'search-sense-update' sequence, as illustrated in Fig. 1.

- In the first step (*search*), PWAS uses local search to build the next action vector $\boldsymbol{a}_{t+1}$ incrementally (one sensor at a time). During this process, sensor configurations are selected based on their ability to (i) discriminate the classes with highest posterior $p(\omega_i|\boldsymbol{O_t})$, where $\boldsymbol{O_t}$ denotes the history of observations up to time t, and (ii) provide information that complements that of other configurations already included in the action vector.
- In the second step (*sense*), PWAS applies the action vector $\boldsymbol{a}_{t+1}$ to the sensor array and measures the corresponding sensor response vector $\boldsymbol{O}_{t+1}$.
- In the final step (*update*), PWAS estimates the posterior $p(\omega_i|\boldsymbol{o}_{t+1})$ to incorporate evidence from the latest measurement vector $\boldsymbol{O}_{t+1}$; this is done using a sequential Bayesian update equation [34].

The critical step in this process (and the main contribution of this work) is how to quantify the discriminatory information of a given action vector while considering the class posteriors. In the sections that follow we propose two objective measures that are suitable for this purpose, and describe the remaining components of the PWAS algorithm.

3.1. Posterior-weighted filters for active feature selection

A number of objective measures have been proposed to evaluate the discriminatory information of chemical sensor arrays, such as signal-to-noise ratio [35], Fisher scores [36], and mutual information [37]. However, these objective measures assume a passive-sensing scenario where the sensor array is optimized off-line, and therefore do not adapt as additional information from previous measurements becomes available. We propose two objective measures to address this need. The first measure, Weighted Fisher Score (WFS), evaluates action vectors based on their ability to discriminate the most likely classes, measured as the ratio of between-class to within-class scatter weighted by the class posteriors. The second method, Dynamic Mutual Information (DMI), evaluates action vectors based on information-theoretic measures of feature relevance and feature redundancy relative to the class posteriors. WFS is a parametric measure (it assumes class conditional densities are Gaussian), whereas DMI is non-parametric. Detailed descriptions of these two objective functions are presented next.

3.1.1. Weighted Fisher Score

WFS is a generalization of the traditional Fisher score (FS) used in linear discriminant analysis [38], a supervised dimensionality reduction method that seeks to preserve class discriminatory information. Given a feature set $\mathcal{F} = \rho_1, \rho_2, \ldots, \rho_j$, the FS is defined as the determinant of the ratio of between-class scatter to within-class scatter corresponding to features in $\mathcal{F}$:

$$\mathrm{FS}(\mathcal{F}) = \det\left(W^{-1}B\right) \tag{1}$$

where W and B are the between-class and within-class scatter matrices, respectively:

$$B = \sum_{i=1}^{M} B_i; W = \sum_{i=1}^{M} W_i \tag{2}$$

Matrix B_i is defined as:

$$B_i = N_i(\mu - \mu_i)(\mu - \mu_i)^T \tag{3}$$

where μ is the sample mean vector for all features in $\mathcal{F}$, estimated on the training set[2] $\boldsymbol{X}$: $\mu = \frac{1}{(N_1+N_2+\ldots+N_M)} \sum_{x \in \boldsymbol{X}} x(\mathcal{F})^T$, and μ_i is the sample mean for class ω_i: $\mu_i = \frac{1}{N_i} \sum x \in \omega_i x(\mathcal{F})^T$. Similarly, W_i is defined as:

$$W_i = \sum_{x \in \omega_i} (x(\mathcal{F}) - \mu_i)^T (x(\mathcal{F}) - \mu_i) \tag{4}$$

When used for feature selection, FS helps find a set of features that maximizes the distance between training instances from different classes relative to the distance among instances of the same class. In this process, FS weighs all classes equally, or in proportion to their number of examples in the training set as in Eqs. (3) and (4).

Now consider an active-sensing scenario where the sensor has been driven for t-1 sensing steps with action vectors $\boldsymbol{A_t} = \{\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_{t-1}\}$, resulting in observations vectors $\boldsymbol{O_t} = \{\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots \boldsymbol{o}_{t-1}\}$. Based on these observations, the class posteriors have been estimated to be $\boldsymbol{p}_t = \{p_t(\omega_1), p_t(\omega_2), \ldots, p_t(\omega_M)\}$, where $p_t(\omega_i) = p(\omega_i|\boldsymbol{O}, \boldsymbol{A})$, and $\Sigma_{i=1}^{M} p_t(\omega_i) = 1$. To contextualize, assume a problem with $M = 3$ classes and posterior distribution $\boldsymbol{p}_t = \{0.1, 0.4, 0.5\}$. How could $\boldsymbol{p}_t$ inform the selection of the next action vector $\boldsymbol{a}_t$? In this case, classes ω_2 and ω_3 are 4–5 times more likely than class ω_1, which suggests that additional sensing resources should be allocated to breaking the tie between them rather than furthering the gap with ω_1. The solution proposed here (WFS) redefines the total scatter matrix by weighing the individual scatter matrices according to the class posteriors:

$$B = \sum_{i=1}^{M} p_t(\omega_i) B_i; \qquad W = \sum_{i=1}^{M} p_t(\omega_i) W_i \tag{5}$$

[1] For notational convenience, we assume all the sensors have the same configuration space.

[2] $\boldsymbol{X}$ is a matrix of D columns and $(N_1 + N_2 + N_3 \ldots + N_M)$ rows, where N_i is the number of training instances from class ω_i. x denotes a single row vector of $\boldsymbol{X}$, and $x(\mathcal{F})$ denotes a row vector (of size $1 \times |\mathcal{F}|$) containing the entries corresponding to $\mathcal{F}$.

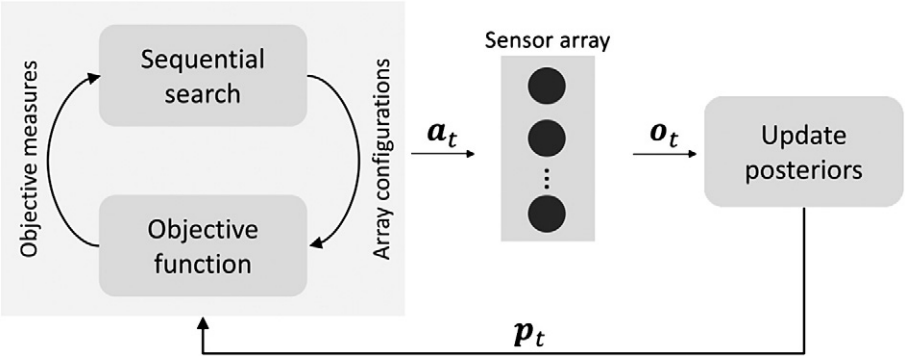

Fig. 1. An overview of the PWAS method. PWAS uses a sequential search to generate various array configurations and the objective function evaluates them. The best such configuration is chosen to drive the sensor array and the resulting observations are used to update the class posteriors.

Thus, for a given action vector $\boldsymbol{a}_t$, the WFS is defined as:

$$J_F(\boldsymbol{a}_t) = \det\left(\left(\sum_{i=1}^{M} p_t(\omega_i)W_i(\boldsymbol{a}_t)\right)^{-1}\sum_{i=1}^{M} p_t(\omega_i)B_i(\boldsymbol{a}_t)\right) \tag{6}$$

where $\det(\cdot)$ is the determinant of a matrix, $B_i(\boldsymbol{a}_t)$ is based on Eq. (3) with mean vectors $\mu = \frac{1}{(N_1+N_2+\ldots+N_M)}\sum_{x\in\boldsymbol{x}} x(\boldsymbol{a}_t)^T$ and $\mu_i = \frac{1}{N_i}\sum_{x\in\omega_i} x(\boldsymbol{a}_t)^T$ estimated *only* for those features included in $\boldsymbol{a}_t$. Likewise, $W_i(\boldsymbol{a}_t)$ is estimated as $\sum_{x\in\omega_i}(x(\boldsymbol{a}_t)-\mu_i)^T(x(\boldsymbol{a}_t)-\mu_i)$. In this way, WFS favors sensor configurations that have higher between-to-within scatter *among* the more likely classes.

It is important to note that the full scatter matrices $B_i(\mathcal{F})$ and $W_i(\mathcal{F})$ do not have to be recalculated at every sensing step, since they are independent of the class posteriors. Instead, these two scatter matrices can be computed off-line and stored for later use; at each sensing step, PWAS generates the sub-matrices $B_i(\boldsymbol{a}_t)$ and $W_i(\boldsymbol{a}_t)$ by selecting the rows and columns corresponding to the features in $\boldsymbol{a}_t$. This reduces computational costs significantly at run time.

3.1.2. Dynamic mutual information (DMI)

The second objective function we propose here can be considered as a generalization of information-theoretic filters used for supervised feature subset selection [39–42]. These filters typically measure the information content of a feature set as a combination of feature relevance to feature redundancy, where *relevance* is defined in terms of the mutual information between each feature and the class labels, and *redundancy* is defined in terms of the mutual information among the features themselves. A classic example is Hall's information-theoretic objective filter [42], which estimates the information content of a feature set $\mathcal{F} = \rho_1, \rho_2,\ldots,\rho_j$, as a ratio of relevance $\mathcal{R}$ to redundancy D. In Halls' method, $\mathcal{R}$ is defined as:

$$\mathcal{R} = \sum_{\rho_i\in\mathcal{F}} U(\rho_i, \boldsymbol{\omega}) \tag{7}$$

where $U(\rho_i,\boldsymbol{\omega})$ is the symmetrical uncertainty[3] between feature ρ_i and the class label $\boldsymbol{\omega}$ [43]:

$$U(\rho_i, \boldsymbol{\omega}) = 2\times\left(\frac{I(\boldsymbol{\omega};\rho_i)}{H(\boldsymbol{\omega})+H(\rho_i)}\right) \tag{8}$$

Likewise, Halls's redundancy measure D is defined as:

$$D = \sqrt{|\mathcal{F}| + \sum_{\rho_i\in\mathcal{F}}\sum_{\rho_j\in\mathcal{F},\rho_i\neq\rho_j} U\left(\rho_i,\rho_j\right)} \tag{9}$$

[3] Symmetrical uncertainty is a normalized variant of mutual information. Mutual information has an inherent bias in favor of variables with more values. Symmetrical uncertainty compensates for this bias by normalizing the values to be in the range [0, 1].

where $U(\rho_i,\rho_j)$ is the symmetrical uncertainty between features ρ_i and ρ_j:

$$U\left(\rho_i,\rho_j\right) = 2\times\left(\frac{I\left(\rho_i;\rho_j\right)}{H(\rho_i)+H\left(\rho_j\right)}\right) \tag{10}$$

$I(\boldsymbol{\omega};\rho_i)$ is the mutual information between $\boldsymbol{\omega}$ and ρ_i, $I(\rho_i;\rho_j)$ [4] is the mutual information between features ρ_i and ρ_j, and $H(\boldsymbol{\omega})$ and $H(\rho_i)$ are the entropy of $\boldsymbol{\omega}$ and ρ_i respectively.

As before, consider an active-sensing scenario where after t-1 sensing steps the class posterior distribution is $\boldsymbol{P}_t$. How can knowledge of $\boldsymbol{P}_t$ be incorporated into an information-theoretic filter to maximize the separability between the more likely classes? A conventional measure such as Hall's filter cannot be used for this purpose because it evaluates features with respect to the distribution of classes in training data—as shown in Eq. (7). Our solution (Dynamic Mutual Information; DMI) redefines feature relevance in terms of symmetrical uncertainty between each feature and the posterior distribution $\boldsymbol{P}_t$. Namely, given an action vector $\boldsymbol{a}_t$, we define $\mathcal{R}$ as:

$$\mathcal{R} = \sum_{\rho_i\in\boldsymbol{a}_t} U(\rho_i, \boldsymbol{p}_t) \tag{11}$$

where $U(\rho_i,\boldsymbol{p}_t)$ is the symmetrical uncertainty between feature ρ_i and posteriors $\boldsymbol{p}_t$:

$$U(\rho_i, \boldsymbol{p}_t) = 2\times\left(\frac{H(\boldsymbol{p}_t)-H(\boldsymbol{p}_{t+1}|\rho_i)}{H(\boldsymbol{p}_t)+H(\rho_i)}\right) \tag{12}$$

$H(\boldsymbol{p}_t) = -\sum \boldsymbol{p}_t \log(\boldsymbol{p}_t)$ is the entropy of the posterior distribution, and $H(\boldsymbol{p}_{t+1}|\rho_i)$ is the expected entropy in the posterior distribution upon observing ρ_i. Estimating $H(p_{t+1}|\rho_i)$ can be interpreted as projecting uncertainty into the next sensing step[4].

With the modified definition of feature relevance of Eq. (11) in place, the DMI of action vector $\boldsymbol{a}_t$ becomes:

$$J_M(\boldsymbol{a}_t) = \frac{\sum_{\rho_i\in\boldsymbol{a}_t} U(\rho_i, \boldsymbol{p}_t)}{\sqrt{|\boldsymbol{a}_t| + \sum_{\rho_i\in\boldsymbol{a}_t}\sum_{\rho_j\in\boldsymbol{a}_t,\rho_i\neq\rho_j} U\left(\rho_i,\rho_j\right)}} \tag{13}$$

Thus, by estimating symmetric uncertainty *relative to* the posteriors, DMI favors features that have high mutual dependence with the more likely classes. When the posterior distribution is uniform $\boldsymbol{p}_t(\omega_i) = \frac{1}{M}\forall_i$, DMI is equivalent to Hall's objective function. Note that the symmetrical uncertainty between every pair of features does not have to be recalculated at every sensing step. Instead, these values can be estimated

[4] Details on estimating $I(\rho_i;\rho_j)$ and $H(p_{t+1}|\rho_i)$ are provided in Appendix 2.

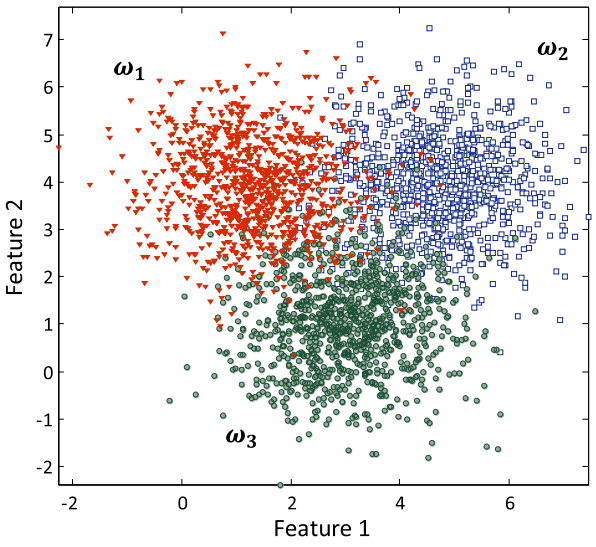

Fig. 2. The training set used for illustration purposes consists of 3000 samples (1000 samples per class) generated from 2D Gaussian distributions with parameters: $\boldsymbol{\mu}_1 = [1.26 \quad 4]$, $\boldsymbol{\mu}_2 = [4.73 \quad 4]$, $\boldsymbol{\mu}_3 = [3 \quad 1]$, and $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. The three class means form an equilateral triangle in the two-dimensional feature space.

before sensing begins to significantly reduce computational costs at run time.

3.1.3. Illustration

We conclude this section with an example that illustrates how the two proposed measures (WFS and DMI) adapt in response to information in the class posteriors. Consider a three-class problem (ω_1, ω_2, ω_3) with the training data of Fig. 2. The goal is to determine which of the two features (f_1, f_2) should be chosen based on the posterior distribution $\boldsymbol{p}_t$.

Consider a first scenario where the three classes are equally likely: $p_t(\omega_1) = p_t(\omega_2) = p_t(\omega_3) = 1/3$. Using Eq. (6), the WFS of the two features is $J_F(f_1) = J_F(f_2) = 2.03$; using Eq. (13), the respective DMI scores are $J_M(f_1) = J_M(f_2) = 0.33$. Lacking additional information from previous measurements, both methods rate features f_1 and f_2 as equally informative. Now, consider a second scenario with posterior distribution $\boldsymbol{p}_t = [0.4 \quad 0.4 \quad 0.2]$. Using Eqs. (6) and (13), the scores become $J_F(f_1) = 2.45$ and $J_F(f_2) = 1.62$ for WFS, and $J_M(f_1) = 0.35$ and $J_M(f_2) = 0.29$ for DMI. Thus both methods rate f_1 as more informative than f_2, in agreement with the distribution in Fig. 2, which shows that the two majority classes (ω_1 and ω_2) differ only along the first dimension. As a third and final scenario, consider the posterior distribution $\boldsymbol{p}_t = [0.2 \quad 0.4 \quad 0.4]$. In this case, the scores become $J_F(f_1) = 1.82$ and $J_F(f_2) = 2.25$ for WFS, and $J_M(f_1) = 0.30$ and $J_M(f_2) = 0.35$ for DMI. Thus, both methods rate f_2 as more useful than f_1, in agreement with the distribution in Fig. 2, which shows that separability between the majority classes (ω_2 and ω_3) is largest among the second dimension.

3.2. Search strategy and posterior update

Having defined objective functions that can adapt to new information *and* account for sensor collinearity, we now turn our attention to the problem of searching for suitable sensor configurations. Given the exponential number of possible configurations for a sensor array with N sensors and D configurations, $O(D^N)$, exhaustive enumeration is unfeasible except for toy problems. Instead, a search algorithm is needed to efficiently search through this large space at every sensing step, a process that is similar to that of feature subset selection [38]. In this paper, we use sequential forward search for its simplicity, though a number of alternative strategies may be used.[5] Namely, PWAS generates the next action vector $\boldsymbol{a}_t$ by adding the configuration of each sensor one at a time. Starting with an empty action vector $\boldsymbol{a}_t = \phi$, at each step PWAS selects the configuration for sensor n that maximizes the objective score when combined with the configurations of the previous $n-1$ sensors: $\rho_n = \underset{\rho}{\operatorname{argmax}}\, J_F(\boldsymbol{a}_t \cup \rho)$, with $\boldsymbol{a}_t = \{\rho_1, \rho_2, \ldots \rho_{n-1}\}$. This search is executed for N iterations to construct an action vector $\boldsymbol{a}_t$ of cardinality N. While doing so, we track the configurations that were used in the previous iterations and sensing steps to avoid repetition. The pseudo-code for the search process is included in Table 1.

Table 1
Pseudo-code for the algorithm used by PWAS to construct action vectors $\boldsymbol{a}_t$.

```
construct_action_vector (N, ρ, b_t)
    - a_t = φ
    - for i = 1 to N
            - best = 0
            - for j = 1 to |ρ|
                    - score = J_F(a_t ∪ ρ_j)
                    - if score > best
                            - best = score
                            - ρ_sel = ρ_j
            - ρ = ρ − ρ_sel
            - a_t = a_t ∪ ρ_sel
    - return a_t
```

Once the action vector $\boldsymbol{a}_t$ is constructed, PWAS drives the sensor array with the corresponding configurations and measures the sensor response vector $\boldsymbol{o}_t$, which is used to re-estimate the class posteriors $\boldsymbol{p}_{t+1}(\omega_k)$ via sequential Bayesian updating:

$$\boldsymbol{p}_{t+1}(\omega_k) = \frac{\boldsymbol{p}_t(\omega_k) p(\boldsymbol{o}_t|\omega_k, \boldsymbol{a}_t)}{p(\boldsymbol{o}_t)} \tag{14}$$

where the denominator $p(\boldsymbol{o}_t)$ is a normalization constant that ensures the posteriors add up to one, and the likelihood term $p(\boldsymbol{o}_t|\omega_k, \boldsymbol{a}_t)$ is the probability of obtaining observation vector $\boldsymbol{O}_t$ when the sensor array is exposed to chemical ω_k and driven with action vector $\boldsymbol{a}_t$, which we estimate under the assumption that sensor measurements are class-conditionally independent:

$$p(\boldsymbol{o}_t|\omega_k, \boldsymbol{a}_t) = \prod_{i=1}^{N} p(o_i|\omega_k, s_i, \rho_i) \tag{15}$$

Once the allocated T sensing steps have been completed, PWAS brings the sensing process to a halt and declares a final class label $\omega_{out} = \underset{1 \le k \le M}{\arg\max}\, \boldsymbol{p}_T(\omega_k)$ using the MAP criterion.[6] Pseudo-code for the complete PWAS algorithm is included in Table 2.

4. Validation on synthetic data

In a first series of experiments, we validated PWAS on simulated data from an array of tunable Fabry–Perot interferometers (FPI) [45]. FPIs are optical devices comprised of two partially-reflective parallel mirrors forming an optical resonance cavity [46]. When a light beam reaches the outer surface of the first mirror, it undergoes several reflections and refractions between the two mirrors, resulting in a number of parallel beams (of decreasing amplitudes) emerging from the second mirror.

[5] The list includes other variants of sequential search such as sequential backward search, floating search methods, etc., as well as meta-heuristics such as genetic algorithms, simulated annealing, or tabu search, among others [38].

[6] As shown in Eq. (14), the class posteriors are updated based on the naïve Bayes assumption. Though features are rarely independent, research in machine learning shows that in practice the naïve Bayes assumption is very effective because the final classification decisions are often correct even if the probability estimates are inaccurate [44]. This assumption can easily be relaxed by incorporating the feature covariance matrix in the posterior update, at the expense of increasing computational costs at run time.

Table 2
Pseudo-code for Posterior-Weighted Active Search (PWAS).

1) Initialization
- $p_0(\omega_k) = 1/M, \forall_k$
- $t = 1$
2) Search
-Construct action vector $\boldsymbol{a}_t$ using the procedure in Table 1
3) Sensing
-Apply $\boldsymbol{a}_t$ to the sensor array
-Measure observation vector $\boldsymbol{o}_t$
4) Posterior update
- $\boldsymbol{p}_{t+1}(\omega_k) = \boldsymbol{p}_t(\omega_k)p(\boldsymbol{o}_t|\omega_k, \boldsymbol{a}_t)/p(\boldsymbol{o}_t)$
- $t = t + 1$
5) Termination
-If $t \leq T$ go to step 2
-Else classify sample as $\omega_{out} = \arg\max_{1\leq k\leq M} \boldsymbol{p}_T(\omega_k)$

When combined with an infrared (IR) source and a detector FPIs can be used as tunable chemical sensors: by adjusting the distance between the mirrors one can measure IR absorption at a wavelength of interest.

To simulate the response of an FPI to different chemicals, we used IR absorption spectra in the NIST chemistry WebBook (http://webbook.nist.gov/chemistry/), a database containing high resolution (250 points per μm) IR spectra in the range 3–21 μm for over 16,000 chemicals in the gas phase. We chose 250 compounds from this database that have strong absorption peaks in the range 8–10.5 μm, a typical operating range of commercially-available FPI sensors (http://www.infratec.de/en.html). To simulate the low resolution of FPI sensors, each spectrum was down-sampled to 116 wavelengths; each one of these wavelengths was treated as a sensor configuration; i.e., $\boldsymbol{\rho} = \langle \rho_1, \rho_2, \ldots, \rho_{116} \rangle$, following the notation in Section 3. Fig. 3 illustrates the absorption spectra for 50 of the 250 compounds in our dataset.

Using these spectra, we generated 10 samples for each chemical by adding Gaussian noise of standard deviation $\sigma = 0.05$ at each wavelength and used the resulting dataset of 2500 (250×10) samples for training purposes. Similarly, we generated another dataset containing 5000 samples (20 samples per chemical) for testing purposes; this test set contained additive Gaussian noise with $\sigma = 0.15$, three times the noise level in the training data.

Following our prior work [6], we modeled the distribution of sensor responses (i.e., absorption) at configuration ρ_i (i.e., wavelength) to chemical ω_j with a Gaussian mixture model (GMM) [47] as:

$$p(o|\omega_k, \rho_i) = \sum_{m=1}^{M} \alpha_m N(o|\mu_m, \Sigma_m) \tag{16}$$

where m is the number of Gaussian components, and α_m, μ_m, Σ_m are the mixing coefficients, mean and covariance of each Gaussian, respectively; these parameters were optimized via Expectation-Maximization to maximize the likelihood of the 2500 training samples. The 2500 training samples were also used to calculate the total within-scatter and between-scatter matrices of WFS in Eqs. (2)–(4), and the symmetrical uncertainty for DMI in Eq. (10). Once the GMM, WFS and DMI models were built on training data, the 5000 test samples were used to run the PWAS algorithm in Table 2.

We characterized PWAS through a series of experiments. In a first experiment, we evaluate the ability of both objective functions (WFS, DMI) to handle increasing dimensionality. In a second experiment, we compared PWAS against two passive-sensing strategies, sequential forward selection and a naïve random feature selection. In a final experiment, we assessed the performance of all the methods at increasing levels of additive noise.

4.1. Experiment 1: performance vs. dimensionality

In the first experiment, we characterized the performance of PWAS for sensor arrays of size $N = \{1,2,3,4\}$ and $T = \{1,2\ldots10\}$ action steps using both objective functions. When using DMI as the objective

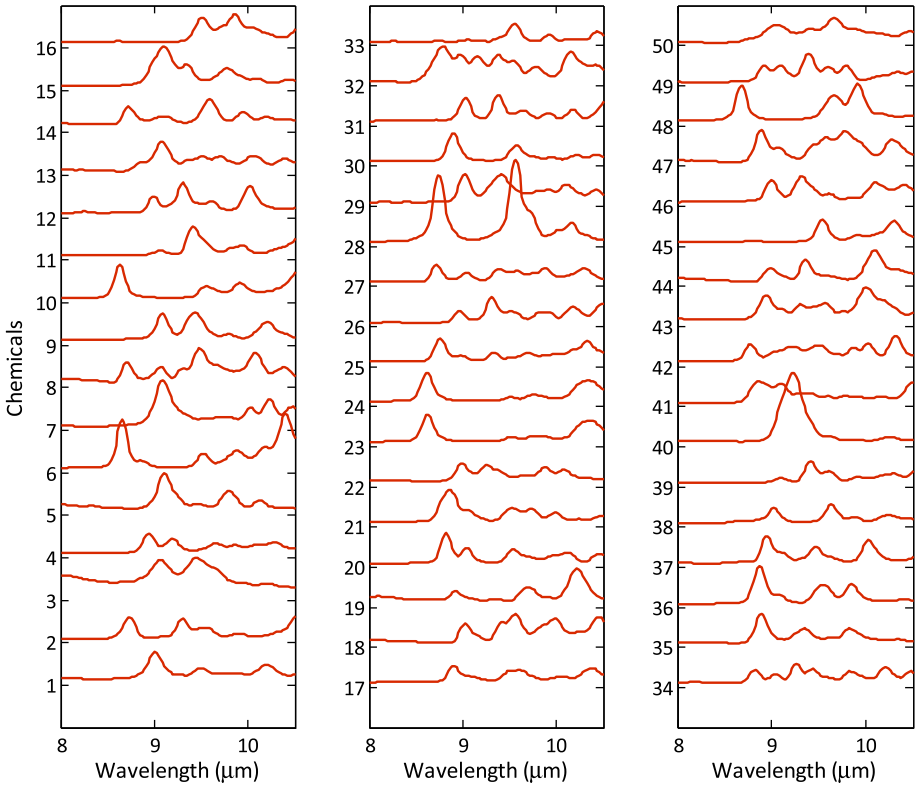

Fig. 3. Infrared absorption spectra of 50 compounds as a function of wavelength. For visualization purposes, the spectra are plotted in three columns with an offset along the y-axis.

function, we discretized each feature into 10 bins (value was chosen empirically) using k-means clustering for the purpose of estimating mutual information. No discretization was needed when using WFS as the objective function. Results are summarized in Fig. 4. Classification performance increases with the number of sensing steps for both objective functions. At any given sensing step, performance also increases with the number of sensors in the array. Both results are to be expected since the number of sensor measurements provided to the classifier increases with the number of sensing steps and the number of sensors in the array. Fig. 4 also shows that WFS outperforms DMI, the difference being more evident for larger N. This result suggests that WFS can account for sensor redundancy more effectively than DMI.

4.2. Experiment 2: active vs. passive sensing

In the second experiment, we compared the two objective functions under active-sensing and passive-sensing scenarios, also with varying array sizes ($N = 2,3,4$) and sensing steps ($T = 1,2…10$). For passive sensing we performed sequential forward selection (SFS) in combination with either the un-weighted Fisher Score (FS) or Hall's mutual information (MI) as the objective function.[7] To gauge the complexity of the discrimination problem we also include results from a naïve Random Selection (RS) algorithm that chooses features randomly at each sensing step, with the constraint that features used in the previous sensing steps cannot be used again; classification results for RS were averaged over 10 separate runs.

Overall results for the five methods are summarized in Fig. 5. In all cases, classification increases monotonically with the number of sensing steps and array size. In the initial sensing step, active and passive methods achieve the same performance, a reasonable result considering that no previous measurements are available. More importantly, the two active methods (WFS and DMI) consistently outperform their passive counterparts (FS and MI), and also RS as expected. The difference between active and passive methods is more significant when using Fisher scores as the selection criteria (WFS ≫ FS) rather than mutual information (DMI > MI), a result that is consistent with the findings in the first experiment.

4.3. Experiment 3: performance vs. noise

In the third experiment, we evaluated the five methods (WFS, DMI, FS, MI, and RS) at varying signal-to-noise ratios (SNR) in the test data. For this purpose, we generated datasets with additive Gaussian noise in the range $\sigma = 0.12$ to $\sigma = 0.3$ in steps of $\Delta\sigma = 0.02$. Then, we tested the methods on an array with $N = 2$ sensors and $T = 10$ time steps. Results are summarized in Fig. 6. As expected, classification degrades for all methods with decreasing SNR, but the two active methods (WFS, DMI) outperform their passive counterparts (FS, MI) at each SNR. As we had found in the second experiment, the difference between the active and passive methods becomes more evident when the Fisher Score is used as the selection criterion.

5. Validation on experimental data

Finally, we validated PWAS experimentally on an array of MOX sensors[8] (Figaro TGS 2620, TGS 2602, and TGS 2610) exposed to five household chemicals (mineral spirits, acetone, ammonia, denatured alcohol, and isopropyl alcohol). The chemicals were placed in a 30 ml vial and their headspace vapors delivered to the sensor chamber using an air pump connected downstream. The concentration of each chemical was controlled with a gas diluter (Custom Sensor Solutions, Inc.), and the entire apparatus (sensors, diluters, pump, and valves) was interfaced through two NI data acquisition cards and controlled with Matlab.

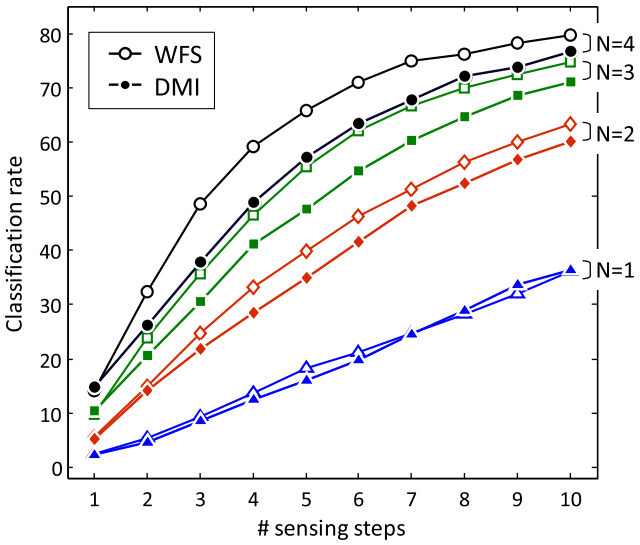

Fig. 4. Classification performance of WFS (hollow markers) and DMI (solid markers) for various array sizes and sensing steps.

As a first step, we conducted a preliminary study to determine concentrations for each chemical that made the discrimination problem non-trivial. Namely, we measured the isothermal response of each sensor to the five analytes at five concentrations (4% to 20%, in steps of 4% v/v), and identified concentrations at which the isothermal responses were similar. To obtain the isothermal response, the sensors were exposed to each analyte for 100 s under a constant heater voltage of 5 V. Our goal was to ensure the samples could not be distinguished based solely on the amplitude of the responses and that temperature modulation would be necessary. The final concentrations were: xylene — 12%, denatured alcohol — 4%, mineral spirits — 4%, turpentine — 8%, and ammonia — 16% v/v.

To generate experimental data for each analyte, the sensor array was driven with 20 sequences, each sensor with a different sequence and each sequence consisting of 10 heater voltages ($3 \le V_H \le 7.5$ V in steps of 0.5 V) in a randomized order. This resulted in a dataset with 100 samples (5 chemicals × 20 sequences). Analytes were presented in a randomized order to avoid systematic errors. At each heater voltage, the sensors were pulsed for 20 s; between consecutive pulses, the sensors were reset to a baseline heater voltage (0 V) for 10 s. This form of temperature programming helped reduce variance in the responses due to thermal dynamics [48]. Driving the sensor with a pulse at each of the 10 heater voltages (i.e., $V_H = 3 + 0.5 \times k$; $k = 0…9$) was treated as a sensor configuration; i.e., $\boldsymbol{\rho} = \langle \rho_1, \rho_2, \ldots, \rho_{10} \rangle$. Fig. 7 shows the response of the three MOX sensors to a sequence of five voltage pulses when exposed to mineral spirits. In this example, the three sensors were driven with the same temperature sequence.

Following our prior work [32], we used Principal Component Analysis (PCA) to extract features from the sensor transients. Namely, for each of the 10 voltage settings and for each sensor, we collected the transient responses to all analytes and then applied PCA to obtain the loadings (eigenvectors) and scores. Fig. 8(a) shows the transient responses of the TGS 2610 sensor to the five chemicals (5 transients per chemical) when driven with a 20-second pulse at 3.5 V; Fig. 8(b) shows the first

[7] For each value of N and T, we used SFS to find the best subset of $N \times T$ features out of $N \times D$ possibilities ($T \ll D$), then trained a naïve Bayes classifier with class conditional features distributions $p(\rho_i|\omega_j)$ modeled as Gaussian mixtures. To ensure that exactly T features are selected for each sensor, we enforce a constraint that in the j^{th} iteration of SFS we choose a configuration for the $(1 + mod(j-1, N))^{th}$ sensor, where mod is the modulo operator.

[8] These sensors have broad and overlapping selectivity: TGS 2620 is marketed for the detection of organic vapors, TGS 2602 is sensitive to ammonia and hydrogen-sulphide, and TGS 2610 is sensitive to hydrocarbons (propane and butane).

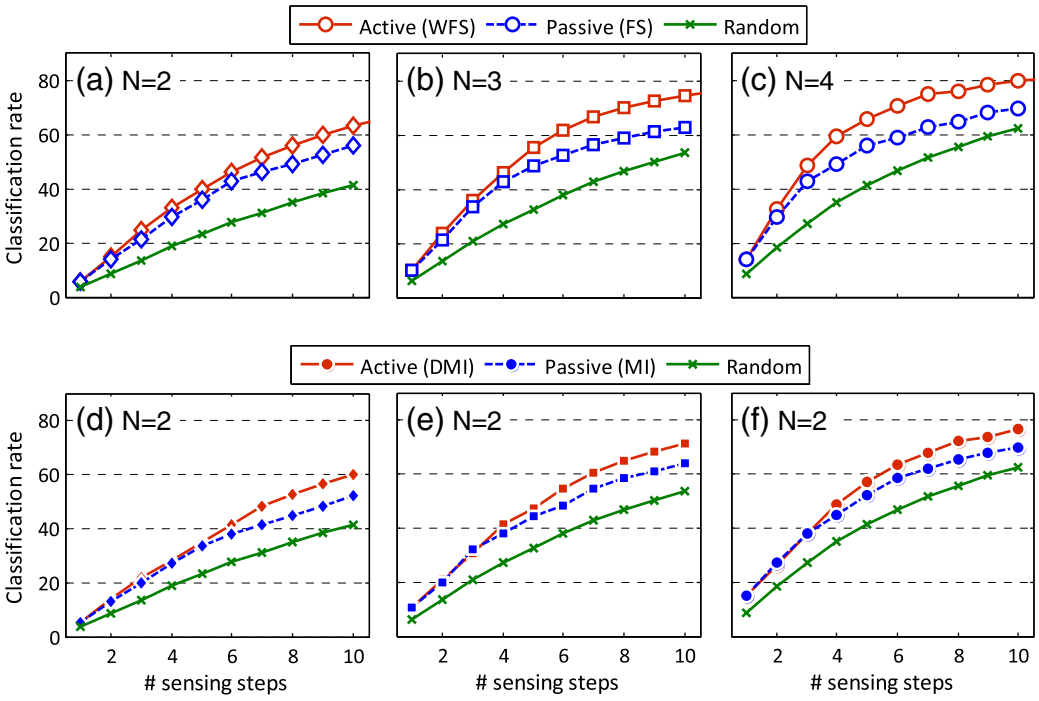

Fig. 5. Comparison between active (WFS, DMI) and passive sensing (FS, MI), for various array sizes and sensing steps: (a–c) WFS vs. FS, (d–e) DMI vs. MI. RS is included as a reference to gauge the complexity of the problem.

three loadings. This process is repeated for each sensor and voltage setting to obtain the corresponding loadings and scores. In all cases, the first three principal components were sufficient to capture more than 99% of the variance. Thus, PCA allowed us to compress the transient response down to three features (i.e., the PCA scores). Then, we used three one-dimensional GMMs (one per principal component) to create the sensor models, resulting in 90 GMMs (3 principal components × 10 temperatures × 3 sensors). To generate observations during testing, the sensor was driven with a particular pulse and the resulting transient response was multiplied with the corresponding PCA loadings; the resulting scores were treated as the observations.

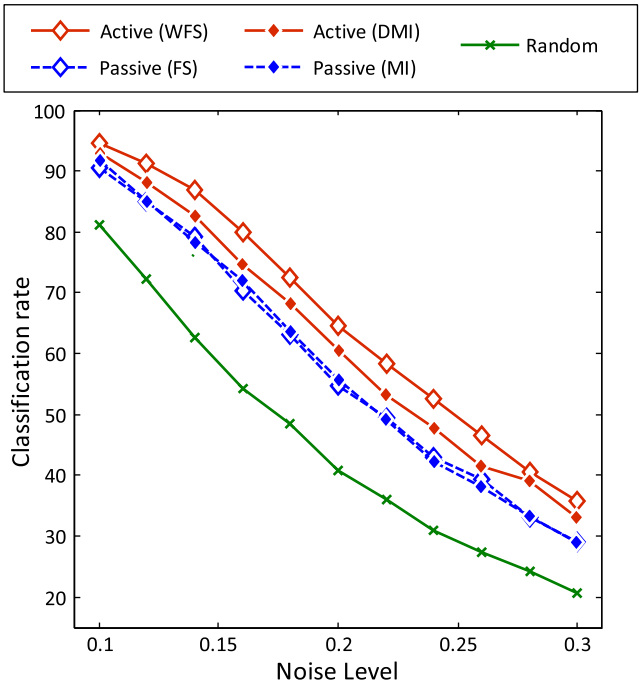

Fig. 6. Classification performance of active, passive and random sensing as a function of noise in test data.

5.1. Results

We present the results of comparing active sensing (WFS, DMI) against passive sensing (FS, MI) for arrays with $N = 2$ (TGS 2620, 2602) and $N = 3$ (TGS 2620, 2602, and 2610) sensors and varying number of sensing steps ($T = 1…5$). The evaluation process followed a 20-fold cross-validation loop. For each fold, the experimental data (100 samples) was randomly divided into two subsets: a training dataset containing 40 samples (8 per chemical), and a test dataset containing 60 samples (12 per chemical). For the passive approaches (FS and MI), we used sequential forward selection to select $N \times T$ features, with the constraint that T features were chosen for each sensor, then trained a naïve Bayes classifier on the $N \times T$ features and tested the classifier on holdout data; this process was repeated for each of value of N and T. We used the same cross-validation loop to test the active approaches (WFS and DMI).

Fig. 9 summarizes the results in terms of average classification rates. These results are largely consistent with those on simulated data: classification improves with increasing number of sensors and sensing steps and, more importantly, the active methods outperform their passive counterparts. In addition, as with simulated data, the choice of the objective function significantly influenced the difference between the active and passive methods. For example, at $N = 2$, WFS outperformed FS by 3.2% (averaged over all the sensing steps) but DMI bettered MI by only 0.5%.

Compared to the results on simulated data, the differences between active and passive methods are relatively small. This is partly due to the reduced complexity of the classification problem (5 classes vs. 250 classes), to where a single sensing step is often sufficient to obtain close to 85% classification rate, leaving a smaller margin for improvement when using active sensing. In such cases it is more meaningful

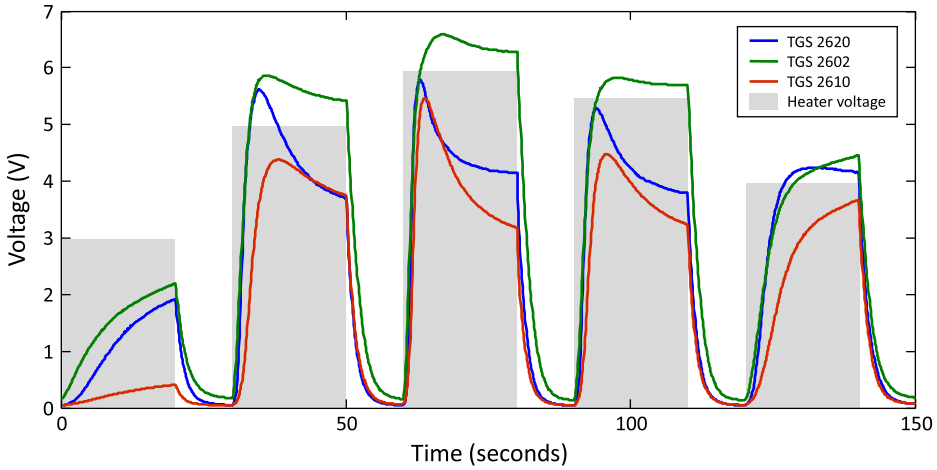

Fig. 7. Transient response of the MOX sensors to mineral spirits. The sensors were driven with a sequence of 5 voltage pulses, each pulse 20-seconds long.

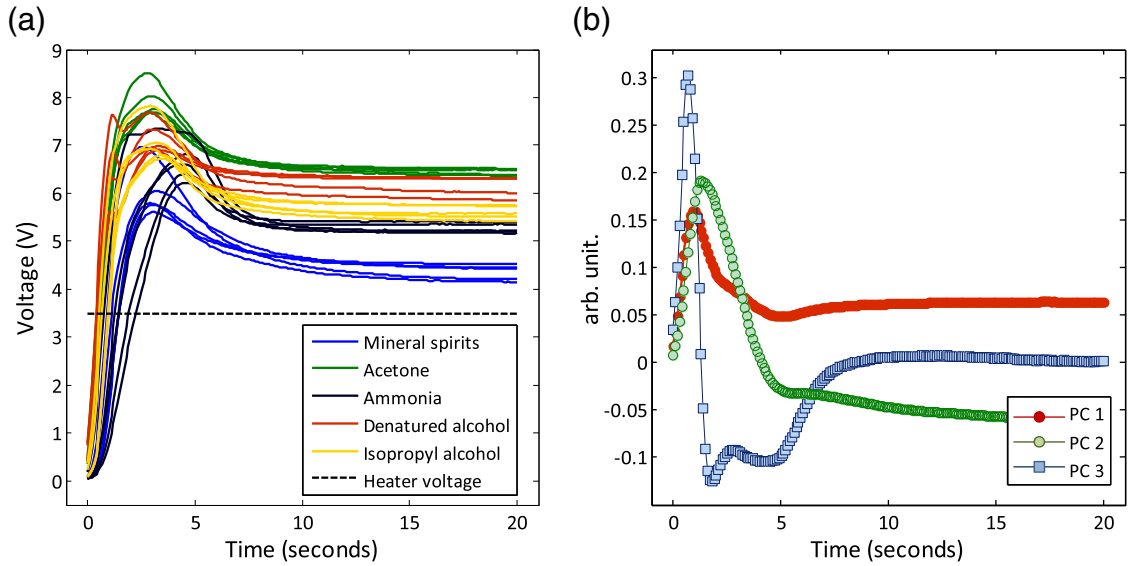

Fig. 8. (a) Sensor transients to the five chemicals (5 repetitions per chemical) in response to a 3.5 V pulse in heater voltage. This pulse was preceded and succeeded by a 10-second pulse at 0 V. (b) The first three principal components extracted from the transients in (a).

to analyze classification errors – the opposite of classification rates.[9] When considering this, the results in Fig. 9 show that active sensing can reduce classification errors significantly, by as much as 30% for $N = 2$ sensors and $T = 5$ actions. Altogether, these results are consistent with those on synthetic data and the general hypothesis that active-sensing methods obtains better classification performance than passive methods [5,6].

6. Conclusions and discussion

We have presented a Posterior-Weighed Active Search method for active classification with chemical sensor arrays. PWAS addresses a major limitation of previous active-sensing algorithms — their poor scalability to higher-dimensional problems, by using a local search to build the action vectors incrementally. We have also presented two objective functions (Weighted Fisher Scores, WFS; Dynamic Mutual information, DMI) that account for sensor collinearity and also adapt as additional information from previous measurements becomes available.

We evaluated our approach on two datasets; one containing low-resolution infrared absorption spectra from 250 chemicals, the second dataset containing temperature-modulated MOX responses to 5 chemicals. Results from both datasets lead to two main conclusions. First, active sensing achieves better classification performance than passive methods for a given sensing budget. This can be attributed to the adaptive nature of active sensing, which allows PWAS to modify the sensing programs according to the test sample at hand. Second, active sensing is more robust to measurement noise than passive. This again can be attributed to the fact that active sensing selects features at measurement time, which allows PWAS to adapt to noise, whereas passive sensing uses a pre-specified set of features computed off-line based on training data.

Results on both datasets also show that WFS outperforms DMI systematically, which indicates that WFS can account for feature redundancy more effectively than DMI — note that both objective functions use the same search strategy. On synthetic data (Section 4), one may

[9] As an example, improving classification rate from 98% to 99% (a seemingly small increment of 1%) can be difficult as it requires reducing errors by 50%.

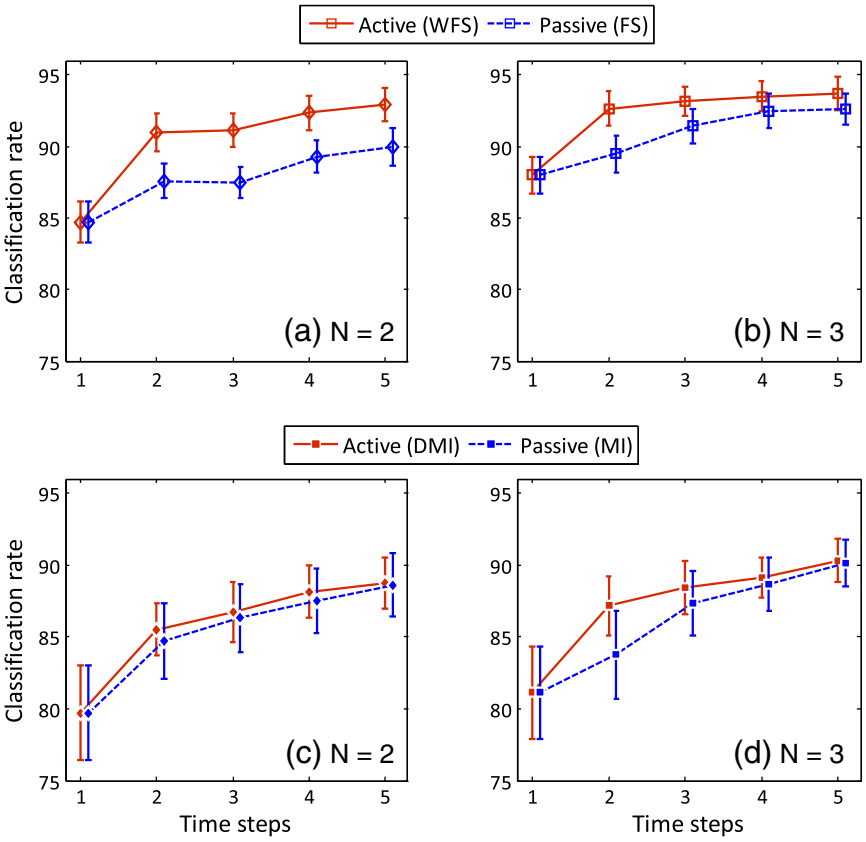

Fig. 9. Classification performance of WFS and FS for (a) $\mathbf{N} = 2$, and (b) $\mathbf{N} = 3$ sensors, DMI and MI for (c) $\mathbf{N} = 2$, and (d) $\mathbf{N} = 3$ sensors. For clarity, the curves corresponding to SFS were shifted slightly along the x-axis. Error bars denote standard deviations.

be tempted to attribute the higher performance of WFS to the fact that the dataset was generated by adding Gaussian noise to the NIST spectra; thus, the argument goes, WFS may have been ideally suited for this dataset since it assumes that the classes are Gaussian. However, WFS also outperforms DMI on experimental data, so an explanation for its higher performance lays elsewhere. A more likely factor is that DMI only considers the posterior distribution when calculating feature relevance but not feature redundancy.[10] Therefore, as the number of measurements increases, DMI becomes less effective in accounting for feature redundancy. On experimental data (Section 5), an added factor is the limited number of training samples. For DMI to be effective, the joint probability distributions need to be estimated accurately, which is challenging considering that there were only 40 training samples for each round of cross-validation.

Another likely factor for the subdued performance of DMI is the discretization step. The symmetrical uncertainty measures in Eqs. (10) and (12) are sensitive to the discretization process: both to the discretization algorithm and to the resolution (number of bins and bin sizes). Uniform discretization, though easy to implement, will fail to capture the nuances of a continuous distribution and is also sensitive to outliers. Therefore, a clustering-based discretization is the better option. Likewise, the number of bins can also influence the symmetrical uncertainty measures. If the distribution is approximated with too few bins, the details of the distribution will not be captured, whereas too many bins will lead to a very sparse distribution (and also increase computational complexity at run time). For the experiments in this paper, we chose the bin counts empirically, but optimizing the number of bins admits a more systematic approach. The discretization step serves the only function of providing a non-parametric estimate (e.g., a histogram) of the probability density function, from which to calculate entropy. Other forms of entropy estimation may be used to circumvent the discretization step altogether. As an example, Huber et al. [49] have presented close-form approximations of the true entropy for Gaussian mixtures. Alternatively, Xu et al. [50] have developed kernel-density-estimation methods that allow the entropy to be computed directly from data samples.

Active sensing yielded smaller performance gains (compared to passive sensing) on experimental data than on synthetic data, particularly for large N. As discussed in Section 5.1, this is largely due to the relative complexity of the two problems. The experimental dataset contained only 5 chemical compounds and 10 features (heater voltages), so a single measurement is able to achieve classification rates around 85% — see Fig. 9. In contrast, the synthetic dataset contained a much larger problem with 250 chemical compounds and 116 sensing actions (wavenumbers), to where classification performance with a single measurement is below 20% — see Fig. 5. Likewise, classification performance on the experimental dataset converges after 3–4 measurements (particularly for $N = 3$ sensors), whereas on the synthetic dataset it continues to increase after 10 measurements. These results suggest that active sensing is most beneficial in situations that require discriminating a large number of chemical targets, including the backgrounds and interferents that are likely to be present in practical settings. Our results also show that, given a sufficient number of measurements, the performance of active and passive sensing becomes comparable. However, for

[10] Though it is possible to consider the class posteriors when calculating feature redundancy, it requires estimating 3-dimensional joint probability distributions, which increases the computational complexity of the method and also the amount of training data required to learn the distributions.

a particular classification rate, active sensing generally requires far fewer measurements than passive sensing. As an example, the results in Fig. 9(a) show that active sensing can achieve classification rates above 90% with only two measurements, whereas passive sensing requires five measurements. These results suggest that active sensing is also likely to be most beneficial in time-critical situations, such as in the analysis of transient targets (e.g., plumes), or whenever sensors have a large settling/recovery times, in which case a long sequence of measurements becomes prohibitive.

6.1. Future work

In this paper, we tested PWAS on arrays with moderate number of sensors (in the range of $N = 2$ to $N = 6$). When applied to larger arrays ($N > 10$), PWAS could be modified to select joint configurations for groups of collinear sensors. As an example, consider a sensor array with N identical sensors. In this case, the sensors may be organized into M groups, each group containing N/M sensors. At each sensing step, PWAS would select M operating configurations, one for each group of sensors, and operate the sensors accordingly. The resulting observations would be averaged by the groups, and this process would reduce noise in the observations. Finally, the averaged values would be used to update the posteriors. In such an approach, the number of groups M would act as a trade-off between the noise in the observations and the number of unique features acquired at each sensing step.

Our current implementation of PWAS assumes no prior knowledge about the target chemicals (i.e., all sensing actions are initialized to be equiprobable) but could easily be extended to take advantage of domain information. As an example, prior knowledge about the distribution of target chemicals could be used to bias the selection process towards spectral lines that show strong absorption for the main functional groups. In this fashion, PWAS could be used to eliminate certain functional groups completely early in the sensing process, simplifying the classification problem at later stages and allowing the system to focus on fine discrimination of chemicals.

Acknowledgments

This work was supported in part by the National Science Foundation under Award #1002028.

Appendix 1. Computational complexity of PWAS

The computational complexity of PWAS depends on the choice of objective function. The computational costs of WFS at run time come from calculating the weighted sum of scatter matrices and then calculating the determinants. At each sensing step, the forward search goes through N iterations (one per sensor). At each iteration, PWAS searches through a maximum of D configurations. For each configuration, it computes the sum of M square matrices of size $N \times N$, which is $O(N^2M)$, and then calculates its determinant, which is $O(N^3)$ steps using LU decomposition. Therefore, the worst-case computation complexity of PWAS per sensing step is the higher of $O(N^4D)$ and $O(N^3MD)$.

The computational costs of DMI at run time come from estimating the expected entropy reductions in Eq. (12). As with WFS, the forward search goes through N iterations. At every iteration, DMI searches through a maximum of D configurations, and for each configuration calculates the entropy reduction. At each iteration, it also calculates the sum of $O(N^2)$ elements to estimate the denominator in Eq. (13). The complexity of Eq. (12) depends on the number of discrete values into which the observation spaces have been discretized. If each observation space is discretized into r bins, then Eq. (12) takes $O(rM)$ steps. Therefore, the worst-case computation complexity of PWAS, per sensing step, is the higher of $O(N^3D)$ and $O(NDrM)$.

Appendix 2. Mutual information and expected entropy

In Eq. (12), the expected entropy $H(\boldsymbol{p}_{t+1}|\rho_i)$ is defined as:

$$H(\boldsymbol{p}_{t+1}|\rho_i) = \sum_{o_k} p(o_k|\rho_i) H(\boldsymbol{p}_{t+1}|\rho_i, o_k)$$

$H(\boldsymbol{p}_{t+1}|\rho_i)$ is computed in three steps. First, for each possible discretized observation o_k from ρ_i, we estimate the expected posterior distribution $\boldsymbol{p}_{t+1}$ using Eq. (14). Second, we compute the entropy of each expected posterior distribution $H(\boldsymbol{p}_{t+1}|\rho_i, o_k)$. Third, we sum these entropies weighted by the corresponding observation probabilities $p(o_k|\rho_i)$, which are obtained from the GMMs.

In Eq. (10), the mutual information $I(\rho_i,\rho_j)$ between configurations ρ_i and ρ_j is defined as:

$$I\left(\rho_i;\rho_j\right) = \sum_{o_i \in \rho_i} \sum_{o_j \in \rho_j} p\left(o_i, o_j\right) \log\left(\frac{p\left(o_i, o_j\right)}{p(o_i)p\left(o_j\right)}\right) \tag{17}$$

where $p(o_i,o_j)$ is the probability of jointly obtaining o_i from configuration ρ_i, and o_j from configuration ρ_j. To estimate this joint distribution, we discretize the observation space of all features using k-means clustering and then estimate the joint probability of ρ_i and ρ_j by counting the number of samples for every pair of observation.

References

[1] A. Hierlemann, R. Gutierrez-Osuna, Higher-order chemical sensing, Chem. Rev. 108 (2008) 563–613.
[2] A.P. Lee, B.J. Reedy, Temperature modulation in semiconductor gas sensing, Sensors Actuators B Chem. 60 (1999) 35–42.
[3] C.E. Priebe, D.J. Marchette, D.M. Healy, Integrated sensing and processing decision trees, IEEE Trans. Pattern. Anal. Mach. Intell. 26 (2004) 699–708.
[4] D.V. Dinakarababu, D.R. Golish, M.E. Gehm, Adaptive feature specific spectroscopy for rapid chemical identification, Opt. Express 19 (2011) 4595–4610.
[5] J. Huang, R. Gosangi, R. Gutierrez-Osuna, Active concentration-independent chemical identification with a tunable infrared sensor, IEEE Sensors J. 12 (2012) 3135–3142.
[6] R. Gosangi, R. Gutierrez-Osuna, Active temperature programming for metal-oxide chemoresistors, IEEE Sensors J. 10 (2010) 1075–1082.
[7] R. Gosangi, R. Gutierrez-Osuna, R. Gosangi, R. Gutierrez-Osuna, Energy-aware active chemical sensing, Proc. IEEE Intl Conf. Sensors, 2010, pp. 1094–1099.
[8] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, second ed. Wiley, New York, 2001.
[9] J.J. Gibson, The Ecological Approach to Visual Perception, first ed. Lawrence Erlbaum Associates, New Jersey, 1986.
[10] J.J. Gibson, Observations on active touch, Psychol. Rev. 69 (1962) 477–491.
[11] R. Bajcsy, Active perception, Proc. IEEE 76 (1988) 966–1005.
[12] A.O. Hero, D.A. Castanon, D. Cochran, K. Kastella, Foundations and Applications of Sensor Management, first ed. Springer, New York, 2007.
[13] J. Aloimonos, I. Weiss, A. Bandyopadhyay, Active vision, Int. J. Comput. Vis. 1 (1988) 333–356.
[14] D. Murray, A. Basu, Motion tracking with an active camera, IEEE Trans. Pattern. Anal. Mach. Intell. 16 (1994) 449–459.
[15] E. Marchand, F. Chaumette, Active vision for complete scene reconstruction and exploration, IEEE Trans. Pattern. Anal. Mach. Intell. 21 (1999) 65–72.
[16] M. Tistarelli, E. Grosso, Active vision-based face authentication, Image Vis. Comput. 18 (2000) 299–314.
[17] A.J. Davison, D.W. Murray, Simultaneous localization and map-building using active vision, IEEE Trans. Pattern. Anal. Mach. Intell. 24 (2002) 865–880.
[18] A. Mishra, Y. Aloimonos, C.L. Fah, Active segmentation with fixation, Proc. IEEE 12th Intl. Conf. Computer Vision, 2009, pp. 468–475.
[19] R. Simmons, S. Koenig, Probabilistic robot navigation in partially observable environments, *Proc. Intl. Joint Conf. Artificial Intelligence* (*IJCAI*), Montreal, Quebec, Canada, 1995, pp. 1080–1087.
[20] H. Zhou, S. Sakane, Mobile robot localization using active sensing based on Bayesian network inference, Robot. Auton. Syst. 55 (2007) 292–305.
[21] D. Fox, W. Burgard, S. Thrun, Markov localization for mobile robots in dynamic environments, Journal of Artificial Intelligence Research 11 (1999).
[22] L. Pedersen, M. Wagner, D. Apostolopoulos, W.R. Whittaker, Autonomous robotic meteorite identification in Antarctica, Proc. IEEE Intl. Conf. Robotics and Automation (ICRA), 2001, pp. 4158–4165.
[23] L. Mihaylova, T. Lefebvre, H. Bruyninckx, K. Gadeyne, J.D. Schutter, Active sensing for robotics-a survey, Proc. Int. Conf. on Numerical Methods and Applications, 2002, pp. 316–324.
[24] C. Kreucher, K. Kastella, A.O. Hero, Sensor management using an active sensing approach, Signal Process. 85 (2005) 607–624.

[25] T.H. Chung, V. Gupta, J.W. Burdick, R.M. Murray, On a decentralized active sensing strategy using mobile sensor platforms in a network, Proc. 43rd IEEE Conf. Decision and Control (CDC), 2004, pp. 1914–1919.
[26] P. Yang, R.A. Freeman, K.M. Lynch, Multi-agent coordination by decentralized estimation and control, IEEE Trans. Autom. Control 53 (2008) 2480–2496.
[27] P. Yang, R.A. Freeman, K.M. Lynch, Distributed cooperative active sensing using consensus filters, Proc. IEEE Intl. Conf. Robotics and Automation (ICRA), 2007, pp. 405–410.
[28] T. Nakamoto, S. Ustumi, N. Yamashita, T. Moriizumi, Y. Sonoda, Active gas/odor sensing system using automatically controlled gas blender and numerical optimization technique, Sensors Actuators B Chem. 20 (1994) 131–137.
[29] T. Nakamoto, N. Okazaki, H. Matsushita, Improvement of optimization algorithm in active gas/odor sensing system, Sensors Actuators A Phys. 50 (1995) 191–196.
[30] R. Lomasky, C. Brodley, M. Aernecke, D. Walt, M. Friedl, Active class selection, Proc. European Conf. Machine Learning, 2007, pp. 640–647.
[31] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.
[32] R. Gosangi, R. Gutierrez-Osuna, Active temperature modulation of metal-oxide sensors for quantitative analysis of gas mixtures, Sensors Actuators B Chem. 185 (2013) 201–210.
[33] J. Huang, R. Gutierrez-Osuna, Active analysis of chemical mixtures with multi-modal sparse non-negative least squares, presented at the International Conference on Acoustics, Speech, and Signal Processing (In proceeding), 2013.
[34] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, first ed. The MIT Press, Cambridge, 2005.
[35] T.C. Pearce, M. Sanchez-Montanes, Chemical sensor array optimization: geometric and information theoretic approaches, Handbook of Artificial Olfaction Machines. 2003. 347–376.
[36] B. Raman, D.C. Meier, J.K. Evju, S. Semancik, Designing and optimizing microsensor arrays for recognizing chemical hazards in complex environments, Sensors Actuators B Chem. 137 (2009) 617–629.
[37] J. Fonollosa, L. Fernández, R. Huerta, A. Gutiérrez-Gálvez, S. Marco, Temperature optimization of metal oxide sensor arrays using mutual information, Sensors Actuators B Chem. 187 (2012) 331–339.
[38] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, 2006.
[39] D.D. Lewis, Feature selection and feature extraction for text categorization, Proceedings of the workshop on Speech and Natural Language, 1992, pp. 212–217.
[40] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Trans. Neural Netw. 5 (1994) 537–550.
[41] P. Hanchuan, L. Fuhui, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern. Anal. Mach. Intell. 27 (2005) 1226–1238.
[42] M.A. Hall, Correlation-based feature selection for discrete and numeric class machine learning, Proc. Intl. Conf. Machine Learning (ICML), 2000, pp. 359–366.
[43] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, second ed. Morgan Kaufmann, San Francisco, 2005.
[44] I. Rish, An empirical study of the naive Bayes classifier, Proc. IJCAI Workshop on Empirical Methods in Artificial Intelligence, 2001, pp. 41–46.
[45] H. Mai, A. Albrecht, C. Woidt, X. Wang, V. Daneker, O. Setyawati, et al., 3D nanoimprinted Fabry–Pérot filter arrays and methodologies for optical characterization, Appl. Phys. B Lasers Opt. (2012) 1–10.
[46] J.F. Mulligan, Who were Fabry and Perot? Am. J. Phys. 66 (1998) 797–801.
[47] D.A. Reynolds, R.C. Rose, Robust text-independent speaker identification using Gaussian mixture speaker models, IEEE Trans. Speech Audio Process. 3 (1995) 72–83.
[48] T.A. Kunt, T.J. McAvoy, R.E. Cavicchi, S. Semancik, Optimization of temperature programmed sensing for gas identification using micro-hotplate sensors, Sensors Actuators B Chem. 53 (1998) 24–43.
[49] M.F. Huber, T. Bailey, H. Durrant-Whyte, U.D. Hanebeck, On entropy approximation for Gaussian mixture random vectors, IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, 2008, pp. 181–188.
[50] D. Xu, J.C. Principe, J. Fisher III, H.-C. Wu, A novel measure for independent component analysis (ICA), *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, 1998, 1998, pp. 1161–1164.