Chapter 15
Algorithms for Context Learning and Information Representation for Multi-Sensor Teams

Nurali Virani, Soumalya Sarkar, Ji-Woong Lee, Shashi Phoha and Asok Ray

Abstract Sensor measurements of the state of a system are affected by natural and man-made operating conditions that are not accounted for in the definition of system states. It is postulated that these conditions, called contexts, are such that the measurements from individual sensors are independent conditioned on each pair of system state and context. This postulation leads to kernel-based unsupervised learning of a measurement model that defines a common context set for all different sensor modalities and automatically takes into account known and unknown contextual effects. The resulting measurement model is used to develop a context-aware sensor fusion technique for multi-modal sensor teams performing state estimation. Moreover, a symbolic compression technique, which replaces raw measurement data with their low-dimensional features in real time, makes the proposed context learning approach scalable to large amounts of data from heterogeneous sensors. The developed approach is tested with field experiments for multi-modal unattended ground sensors performing human walking style classification.

Keywords Context awareness · Feature extraction · Machine learning · Pattern recognition · Support vector regression · Sensor fusion

N. Virani · S. Sarkar · A. Ray
Department of Mechanical and Nuclear Engineering, Pennsylvania State University, University Park, PA, USA

J.-W. Lee
State College, University Park, PA, USA

S. Phoha (✉)
Applied Research Laboratory, Pennsylvania State University, University Park, PA, USA
e-mail: sxp26@arl.psu.edu

© Springer International Publishing Switzerland (outside the USA) 2016

L. Snidaro et al. (eds.), *Context-Enhanced Information Fusion*,
Advances in Computer Vision and Pattern Recognition,
DOI 10.1007/978-3-319-28971-7_15

15.1 Introduction

In realistic scenarios with data-driven systems, sensor measurements and their interpretation are affected by various environmental factors and operational conditions, which we call contexts [1–3]. For example, factors that determine ground conditions—such as the soil type, moisture content, permeability, and porosity—form the set of contexts for seismic sensor measurements, because they affect the propagation of surface and sub-surface seismic waves [4]. A reliable, high-performance inference engine for pattern recognition, state estimation, etc., must therefore be based on a sensor measurement model that takes into account the effects of the context. For example, in dynamic data-driven application systems (DDDAS) [5], modeling context helps not only in the information fusion as a part of the forward problem, but it is also relevant for obtaining the value of information for selecting relevant sources of information in the inverse problem. However, it is an often onerous and arbitrary task to identify the context set for every sensing modality in a multi-sensor team or to develop a physics-based measurement model that accounts for all contextual effects. This chapter focuses on the forward problem of multi-modal sensor fusion in the DDDAS framework, and develops a systematic machine learning method for the context.

The notion of context is task-specific in nature, and often differs across sensing modalities. For example, research in image processing generally assumes the visual scene to be the context for object recognition [6]; for natural language processing tasks such as speech recognition, handwriting recognition, and machine translation, the intended meaning of an ambiguous word might depend on the text which precedes the word in question, thus the preceding text would be considered as context [7]; and, for ubiquitous or mobile computing, the context set consists of the user location as well as activity attributes [8]. In a multi-sensor operational environment, involving both hard and soft sensing modalities, a broad unified notion of context is needed. This notion should characterize situations in the physical, electronic, and tactical environments that affect the acquisition and interpretation of heterogeneous sensor data for machine perception and adaptation. Furthermore, it is often necessary to iteratively update the belief about the spatio-temporal context automatically and treat it as a latent variable to be estimated.

Different clustering techniques [1, 9] and mixture modeling methods [10] were previously developed and used to identify the context set from measurements. In [1], the authors presented a supervised context learning technique via finding all maximal cliques from an undirected graph [11, 12]. An unsupervised context learning approach using the concept of community detection as in social networks [13] was also presented in [1]. These approaches, however, push the burdensome task of characterizing the size of the context set to the user, the resulting context set is different for each modality in the system, and the context model is not suitable for sequential decision-making and multi-modal fusion problems [3].

The main focus of this chapter is to present an unsupervised context-learning approach that addresses, or mitigates, the aforementioned issues. This approach is

based on the postulation that the context of a system, along with the system state, completely conditions sensor measurements. That is, extending the common, but often incorrect, assumption that the measurements are conditionally independent given the system state, we hypothesize that the sensor measurements are independent conditioned on the state-context pair. This postulation allows for a definition of context that is application-specific, and yet uniform across different sensor modalities. Moreover, the arbitrary nature of clustering and mixture modeling approaches is avoided through a kernel-based unsupervised context learning, where the context set and a context-aware measurement model are automatically generated by the machine. In particular, the machine-generated measurement model automatically guarantees the required conditional independence of sensor measurements, which is crucial for tractable sequential inference.

Aside from sequential inference and multi-modal sensor fusion with heterogeneous sensor teams, the developed context-aware measurement model finds application in the problem of in situ measurement system adaptation for improved state estimation performance [3]. In addition to cheap, persistent sources of information, it allows more expensive, higher-fidelity sensors to be activated and added to a team of active sensors in a sequential manner. Changes in the sensor team are tantamount to adjusting decision boundaries in accordance with the contextual interpretation of data, and to exploiting the expected complementarity between available and new sensor measurements, in order to optimally trade off the accuracy of situation assessment against the cost of sensor activation. Realistic scenarios in multi-modal surveillance, health monitoring, target localization, etc., can employ these context-aware techniques for improved system performance. In this work, the context-aware decision-making framework in which the forward process leads to state estimation and the inverse process involves measurement system adaptation was developed as a dynamic data-driven application system (DDDAS) [5] and a schematic view of the system is shown in Fig. 15.1.

In order for the overall system to handle large amounts of data from multiple sources in real time, raw measurements are normally replaced with their

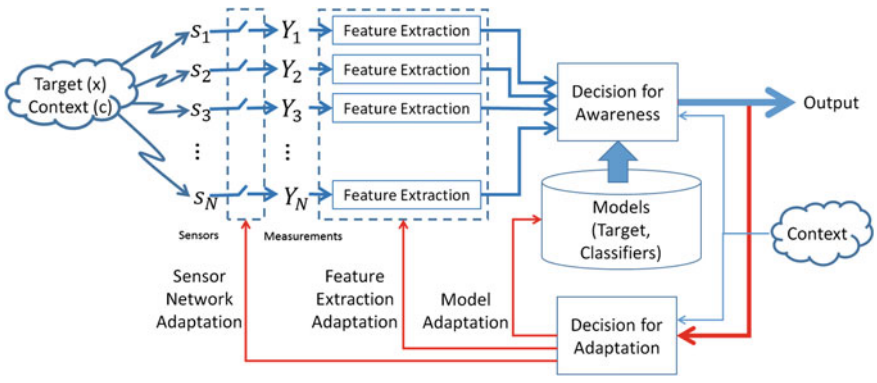

Fig. 15.1 Schematic of a dynamic data-driven application system (DDDAS) with in situ, context-aware, sequential decision-making

low-dimensional features, which are in the form of probabilistic finite state automata (PFSA) and their synchronous compositions and cross machines; otherwise, the context learning algorithm and resulting measurement model remain valid. A realistic numerical example verifies the effectiveness of the context learning and context-aware sensor fusion approaches in combination with the PFSA feature extraction technique.

The organization of this chapter is as follows. Section 15.2 mathematically formalizes the notion of context, presents an approach to automatically identify the context set from heterogeneous sensor data, and shows a context-aware technique that can be used for sequential and multi-modal information fusion and decision adaptation. Section 15.3 presents powerful tools for extracting, refining, and combining features from data. These tools enable the application of the context-aware approach in Sect. 15.2 to realistic situations. A realistic numerical example in Sect. 15.4 assesses the performance of the proposed approach. Lastly, concluding remarks are made in Sect. 15.5.

15.2 Context Learning

Existing context modeling techniques [1, 9, 10] do not guarantee that the measurement sequences from a single or multiple sensors are conditionally independent given the system state and context pair. The inability to guarantee conditional independence of measurements limits the applicability of these techniques in sequential analysis and decision-making. In this section, we mathematically formalize the notion of context and present a context-learning approach that automatically guarantees conditional independence of measurements.

15.2.1 Mathematical Formalization of Context

Let $\mathcal{S}$ be a nonempty finite set of sensors, possibly with different modalities, and let X be the random system state that takes values in a finite set $\mathcal{X}$. For each sensing modality $s \in \mathcal{S}$, let $Y(s)$ be the random measurement, or the feature vector obtained as in Sect. 15.3, associated with the observation of the system state X from sensor s. Before introducing a modality-independent context notion suitable for unsupervised, machine-generation of the context set, let us present a modality-specific context definition, and context types (i.e., intrinsic and extrinsic contexts), that are suitable for supervised learning.

Definition 1 (*Context Elements*) For each $s \in \mathcal{S}$, let $\mathcal{L}(s)$ be a nonempty finite set of labels. Each element of $\mathcal{L}(s)$ is called a *context element*. Every context element is a natural or man-made physical phenomenon, which is relevant to the sensing modality s used to observe the system state. It is assumed that the context elements

are enumerated in $\mathcal{L}(s)$ in such a way that no two elements can occur simultaneously.

The assumption in this definition is not restrictive. If it is possible for two context elements l and m to occur simultaneously, then a new context element k representing l and m occurring together can be added to $\mathcal{L}(s)$. For $s \in \mathcal{S}$ and $l \in \mathcal{L}(s)$, let $p(Y(s)|X, l)$ be the probability density of sensor measurements of modality s for the state X under a given context element l.

Definition 2 (*Extrinsic and Intrinsic Subsets of Contexts*) For $s \in \mathcal{S}$, a nonempty set $\tilde{\mathcal{C}} \subseteq \mathcal{L}(s)$ is called *extrinsic* relative to the state $X = x$ and its measurement $Y(s) = y$ if

$$p(y|x, l) = p(y|x, \tilde{l}) \quad \text{for all } l, \tilde{l} \in \tilde{\mathcal{C}}.$$

Otherwise, the set $\tilde{\mathcal{C}}$ is called *intrinsic* relative to the state $X = x$ and its measurement $Y(s) = y$.

It is sometime impractical to precisely distinguish extrinsic context elements from intrinsic ones. If the observation densities are overlapping and very close to each other under different context elements, then it is deduced that these context elements have nearly the same effect on the sensor data. Thus, an alternative approach is to obtain sets of context elements that are approximately indistinguishable for a given threshold parameter $\varepsilon > 0$ and a metric $d(\cdot, \cdot)$ on the space of observation densities, and let them define contexts.

Definition 3 (*Modality-Specific Context and Context Set*) For $s \in \mathcal{S}$ and $x \in \mathcal{X}$, let $\mathcal{C}(s, x)$ be a set cover of $\mathcal{L}(s)$. Then, the collection $\mathcal{C}(s, x)$ is called a *context set* and each (nonempty) set $c(s, x) \in \mathcal{C}(s, x)$ is called a *context* provided that $c(s, x)$ is a maximal set satisfying the following condition:

$$d(p(Y(s) \mid x, l), p(Y(s) \mid x, m)) < \varepsilon \quad \text{for all } l, m \in c(s, x).$$

In order to obtain a context set $\mathcal{C}(s, x)$ based on Definition 3, the set $\mathcal{L}(s)$ of all context elements must be known a priori, in which case a supervised context modeling approach [1] can be used to reduce the problem of context learning to that of finding all maximal cliques in an undirected graph [11]. However, in many cases, the set $\mathcal{L}(s)$ is unknown, and thus unsupervised context modeling techniques must be used to directly obtain $\mathcal{C}(s, x)$ from the data. In [1], a fast community detection algorithm for social networks [13] was used for unsupervised extraction of context. The resulting context sets are modality-specific as in Definition 3.

The rest of this subsection is aimed at presenting an alternative definition of contexts, which facilitates learning a unified, modality-independent, context set from a multi-modal sensor set [14]. This approach of context learning does not need a defined set of context elements and thus it is an unsupervised way of context

modeling. Let $Y_1 = Y(s_1)$ and $Y_2 = Y(s_2)$ be random measurements of the state X from sensors s_1, $s_2 \in \mathcal{S}$. Let $p(Y_1, Y_2|X)$ denote the joint likelihood function of the pair (Y_1, Y_2). For $i = 1, 2$, let $p_i(Y_i|X)$ denote the marginal likelihood function of Y_i. A common practice in sequential, statistical inference tasks is to assume, for the sake of convenience, that the measurements are statistically independent conditioned on the state [15]. Clearly, this assumption is incorrect unless the state X completely determines all factors that condition the measurements. That is, in general, we have

$$p(Y_1, Y_2 \,|\, X) \neq p_1(Y_1 \,|\, X) p_2(Y_2 \,|\, X).$$

For example, two seismic sensor measurements in binary location testing are expected to be correlated, even if they are conditioned on the true location of a target, because the location alone does not specify the target type, soil conditions (e.g., moisture and porosity), etc., that affect seismic sensor measurements.

Therefore, we define the context as a parameter that, together with the system state, completely conditions the measurements.

Definition 4 (*Context and Context Set*) Suppose that the measurements Y_1 and Y_2 take values in $\mathcal{Y}_1$ and $\mathcal{Y}_2$, respectively. Suppose that the state X takes values from a finite set $\mathcal{X}$. Then, a nonempty finite set $\mathcal{C}(X)$ is called the *context set* and each element $c \in \mathcal{C}(X)$ of the set is called a *context*, if the measurements Y_1 and Y_2 are mutually independent conditioned on the state-context pair (x, c) for all $x \in \mathcal{X}$ and for all $c \in \mathcal{C}(X)$.

According to this definition, the following relation holds:

$$p(Y_1, Y_2 \,|\, X, c) = p_1(Y_1 \,|\, X, c) p_2(Y_2 \,|\, X, c) \quad \text{for all } c \in \mathcal{C}(X). \tag{15.1}$$

Here, the left-hand side of (15.1) denotes the conditional density of (Y_1, Y_2) given (X, c), and the right-hand side gives the product of conditional densities of Y_1 and Y_2 given (X, c). It is now of interest to generate a context set $\mathcal{C}(x)$ for each $x \in \mathcal{X}$, so that (15.1) holds.

15.2.2 Learning Context-Aware Measurement Models

A novel machine learning approach to identifying contexts and determining their prior probabilities (which reflect one's prior knowledge about the true context) in a modality-independent manner is described in this subsection (See [14] for more details). The resulting model treats the context as a random variable and explicitly takes into account the effect of contexts on sensor measurements. The task of identifying all contexts is done by the machine in an unsupervised setting, and thus the extracted contexts need not have a human-understandable meaning associated with them.

15.2.2.1 Mathematical Formulation

Let $p(Y_1, Y_2 \mid X)$ denote the joint density of the pair (Y_1, Y_2) conditioned on the state X; for $i = 1, 2$, let $p_i(Y_i \mid X)$ denote the marginal density of Y_i conditioned on X. The measurement modeling problem that we are concerned with is to estimate these conditional densities, called likelihood functions, based on a training sample consisting of realizations of the triple (Y_1, Y_2, X). In view of Definition 4, a context-aware measurement model gives a likelihood function of the form

$$\begin{aligned} p(Y_1, Y_2|X) &= \sum_{c \in \mathcal{C}(X)} \pi_c(X) p(Y_1, Y_2|X, c) \\ &= \sum_{c \in \mathcal{C}(X)} \pi_c(X) p_1(Y_1|X, c) p_2(Y_2|X, c), \end{aligned} \tag{15.2}$$

where $\pi_c(X)$ is the prior probability that, conditioned on the state X, the true context is c. It is immediate from (15.2) that the marginal likelihoods are given as

$$p_i(Y_i|X) = \sum_{c \in \mathcal{C}(X)} \pi_c(X) p_i(Y_i|X, c) \quad \text{for } i = 1, 2.$$

In general, it is a difficult task to identify a nontrivial context set and a probability distribution on it, so that the prior information about all possible contexts is correctly represented by the measurement model. This task is addressed using a special type of mixture models, where each component density is a product of marginal component densities. For example, Gaussian mixture models with block diagonal covariance matrices are of this type. More specifically, we propose that mixture models of the form (15.2) be used conditioned on the state X, where the context set $\mathcal{C}(x)$ is finite for all $x \in \mathcal{X}$:

$$\mathcal{C}(X) = \{1, 2, \ldots, N(X)\}. \tag{15.3}$$

Conditioned on the state X, the latent variable plays the role of a machine-defined context variable C that takes values in $\mathcal{C}(X)$ and satisfies the conditional independence requirement (15.1) by construction. Here, $N(X)$ is the cardinality of the finite context set $\mathcal{C}(X)$.

15.2.2.2 Kernel-Based Approach

If the marginal component densities $p_i(Y_i \mid X, C)$ are assumed Gaussian, then the expectation maximization algorithm [16] or the variational Bayesian method [17] can be used to obtain a mixture model of the form (2). In this case, the number of contexts $N(x)$ may be determined for each state value $x \in \mathcal{X}$ based on a model selection criterion such as the Akaike and Bayesian information criteria [18, 19]. Alternatively, a Dirichlet process prior can be put over $N(X)$ and then a Gaussian

mixture density model can be estimated together with the optimal number of component densities [20]. However, these parametric estimation approaches do not scale up to high-dimensional measurement spaces, especially with small sample sizes, and also their applicability is limited to Gaussian component densities.

We suggest that a kernel-based nonparametric method be used to overcome this limitation. A kernel function defines an inner product on an implicit, possibly infinite-dimensional, feature space. The standard topology of such a feature space is that of the reproducing kernel Hilbert space induced by a (continuous) Mercer kernel [21, 22]. On the other hand, it is shown in [23] that, if one uses a discontinuous kernel, the resulting feature space can be taken to be the space ℓ^2 (of square-summable sequences) endowed with its weak topology [24]. Let $K : (\mathcal{Y}_1 \times \mathcal{Y}_2)^2 \to \mathbb{R}$ be a kernel function of the form

$$K\left(\begin{bmatrix} s_1 \\ s_2 \end{bmatrix}, \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}\right) = K_1(s_1, y_1) K_2(s_2, y_2), \tag{15.4}$$

with

$$\int_{\mathcal{Y}_i} K_i(s_i, z_i)\, dz_i = 1 \quad \text{for } i = 1, 2 \text{ and } s_i, y_i \in \mathcal{Y}_i. \tag{15.5}$$

Then, conditioned on the state X, a support-vector regression method [25, 26] with the kernel K leads to a mixture model of the form

$$p(Y_1, Y_2 | X) = \sum_{c=1}^{N(X)} \pi_c(X) K_1(s_1^{(c)}(X), Y_1) K_2(s_2^{(c)}(X), Y_2) \tag{15.6}$$

where $(s_1^{(c)}(X), s_2^{(c)}(X))$, $c = 1, \ldots, N(X)$, are the support vectors chosen by the machine from the available data, and the number of support vectors $N(X)$ can be controlled by tuning the underlying insensitivity factor [27]. Note that, with (15.3) and

$$K_i(s_i^{(C)}(X), Y_i) = p_i(Y_i \,|\, X, C) \quad \text{for } i = 1, 2, \tag{15.7}$$

the kernel-based model (15.6) leads to a mixture model of the desired form (15.2) and the support vectors can be taken to be the machine-defined contexts, provided that the following extra constraints are satisfied in addition to (15.4) and (15.5):

$$\sum_{c=1}^{N(X)} \pi_c(X) = 1, \quad \pi_c(X) \geq 0, \quad c = 1, \ldots, N(X). \tag{15.8}$$

15.2.2.3 Support Vector Density Estimation

For the purpose of learning a context-aware measurement model, support vector regression has a clear advantage over other nonparametric approaches like the Parzen density estimation method [28]. Depending on the insensitivity factor utilized in support vector regression, it is possible that only a few key data points contribute to the density estimate and become the support vectors, resulting in a sparse representation without much loss in accuracy. Support vector density estimation (SVDE) [29, 30] is a version of the support vector regression method appropriate for our purpose. Since the cumulative distribution of (Y_1, Y_2) conditioned on X is unknown at the outset, one cannot directly estimate the likelihood function. Instead, one approximates the cumulative distribution function with its empirical approximation formed by the sample of measurements $(y_1^{(1)}, y_2^{(1)})$, …, $(y_1^{(L)}, y_2^{(L)})$ available for the given value of X [29]. For example, if $\mathcal{Y}_1 = \mathcal{Y}_2 = \mathbb{R}$, then the true distribution F and the empirical distribution $\widetilde{F}$ are

$$F(y_1, y_2|X) = \int_{-\infty}^{y_1} \int_{-\infty}^{y_2} p(z_1, z_2|X)\, dz_1 dz_2,$$

$$\widetilde{F}(y_1, y_2|X) = \frac{1}{L}\sum_{j=1}^{L} \theta(y_1 - y_1^{(j)})\theta(y_2 - y_2^{(j)}),$$

where $\theta(\cdot)$ is the unit step function. In order for the empirical distribution to be a consistent estimator of the true distribution (i.e., for the convergence of $\widetilde{F}$ to F as the sample size L tends to infinity), it is assumed in the literature that the available data $(y_1^{(1)}, y_2^{(1)})$, …, $(y_1^{(L)}, y_2^{(L)})$ form an i.i.d. sample of the pair (Y_1, Y_2) conditioned on X [29]. Note that this assumption is a reasonable one even if Y_1 and Y_2 are correlated conditioned on X.

For simplicity, assume $\mathcal{Y}_1 = \mathcal{Y}_2 = \mathbb{R}$. Let $\mathbf{G} = (G_{ij})$ be a matrix whose entry (i, j) is

$$G_{ij} = K_1(y_1^{(i)}, y_1^{(j)})K_2(y_2^{(i)}, y_2^{(j)})$$

for $i, j = 1, \ldots, L$. Let

$$\widetilde{F}_i = \widetilde{F}(y_1^{(i)}, y_2^{(i)}\,|\,X),$$

$$K_{ij} = \int_{-\infty}^{y_1^{(i)}} \int_{-\infty}^{y_2^{(i)}} K_1(y_1^{(j)}, z_1)K_2(y_2^{(j)}, z_2)\, dz_1 dz_2$$

for $i, j = 1, \ldots, L$. Then, taking note of the extra constraint (15.8), and introducing an insensitivity factor $\sigma > 0$, our SVDE problem is translated to the following constrained optimization problem:

Minimize the cost

$$\pi^{\mathrm{T}}\mathbf{G}\pi$$

over column vectors $\pi = (\pi_i)$ subject to the constraints

$$\left|\widetilde{F}_i - \sum_{j=1}^{L} \pi_j K_{ij}\right| \leq \sigma,$$

$$\pi_i \geq 0, \quad \sum_{j=1}^{L} \pi_j = 1, \quad i = 1, \ldots, L.$$

Matrix $\mathbf{G}$ is symmetric and positive definite, and thus this is a convex optimization problem with a quadratic cost function and affine constraints. If the problem is feasible, then a unique solution is guaranteed. If there are a few kernel parameters to be tuned, then the admissible set of these parameters is identified by checking the feasibility of the problem. One can perform a grid search over this admissible set to find the parameters that minimize the cost function. Conditioned on X, the set of support vectors obtained by solving the above optimization problem is the context set. The product form of the kernel guarantees conditional independence of Y_1 and Y_2 given X for each support vector.

15.2.2.4 Extension to Multiple Measurements

It is straightforward to extend the proposed approach to the case of $M(> 2)$ sensor measurements. In this case, the context-aware measurement model (15.2) becomes

$$\begin{aligned} p(Y_1, \ldots, Y_M \,|\, X) &= \sum_{c \in \mathcal{C}(X)} \pi_c(X) p(Y_1, \ldots, Y_M \,|\, X, c) \\ &= \sum_{c \in \mathcal{C}(X)} \pi_c(X) \prod_{k=1}^{M} p_k(Y_k \,|\, X, c), \end{aligned}$$

and its kernel-based approximation (15.6) will be of the form

$$p(Y_1, \ldots, Y_M \,|\, X) = \sum_{c=1}^{N(X)} \pi_c(X) \prod_{k=1}^{M} K_k(s_k^{(c)}(X), Y_k).$$

As in the case of $M = 2$, these equations are related via (15.3) and (15.7).

15.2.3 Context-Aware In Situ Decision Adaptation

In this subsection, an in situ decision adaptation scheme with multi-modal sensor fusion and sensor selection is proposed as a major application of the context-aware measurement model. The key enabler of the proposed application system is that the measurement model guarantees the conditional independence of sensor measurements given the state and context of the system.

15.2.3.1 Context-Aware Sensor Fusion for Multi-Sensor Teams

In the context-aware sensor fusion approach, the following relation holds for multi-sensor teams with M (possibly heterogeneous) measurements:

$$p(Y_1, \ldots, Y_M \,|\, X, C) = \prod_{i=1}^{M} p_i(Y_i \,|\, X, C).$$

If the state space $\mathcal{X}$ is finite, then the following sequential update rule for the posterior distribution of the state-context pair (X, C) is used:

$$P(X, C|Y_1, \ldots, Y_{i-1}, Y_i) = \frac{p_i(Y_i|X, C)P(X, C|Y_1, \ldots, Y_{i-1})}{\sum_{x\in\mathcal{X}} \sum_{c\in\mathcal{C}(x)} p_i(Y_i|x, c)P(x, c|Y_1, \ldots, Y_{i-1})} \quad (15.9a)$$

for $i = 2, 3, \ldots, M$, where

$$P(X, C \mid Y_1) = \frac{p_1(Y_1|X, C)\pi_C(X)P(X)}{\sum_{x\in\mathcal{X}} \sum_{c\in\mathcal{C}(x)} p_1(Y_1|x, c)\pi_c(x)P(x)}. \quad (15.9b)$$

This update rule plays a crucial role in sequential inference and decision-making problems. In a sequential state estimation problem, for instance, one keeps track of the posterior probability of the state-context pair $P(X, C \mid Y_1, \ldots, Y_i)$, updates it to $P(X, C|Y_1, \ldots, Y_i, Y_{i+1})$ as a new sensor measurement Y_{i+1} becomes available, and marginalizes out the context variable to obtain the posterior probability of the state $P(X \mid Y_1, \ldots, Y_i, Y_{i+1})$, from which an updated state estimate can be deduced.

15.2.3.2 Multi-Modal Context-Aware Sensor Team Formation

Suppose now that a set of sensors of possibly different modalities are available for the purpose of sequential state estimation, where the state space $\mathcal{X}$ is finite. Some of these sensors are of high fidelity and generate quality measurements under most contexts, but are costly and need more computational power for operation. On the other hand, some of the sensors are inexpensive to operate, but yield relatively poor measurements. Sensor fidelity, however, is a relative measure. Under some

contexts, low-cost sensor measurements can be effective and show good reliability; likewise, sensors that are generally of high quality can be cost-ineffective and/or unreliable depending on the context. For example, while an inexpensive acoustic sensor on a calm summer day can give good human-vehicle classification results, an expensive camera may not be very useful in poor visibility conditions.

A dynamic sensor team formation framework was proposed in [3]. It integrates the aforementioned contextual effects and their impact on hypothesis testing performance in a systematic manner using dynamic programming. In this framework, the number and types of selected sensors are determined in a data-driven fashion in order to achieve an optimal compromise over estimation performance, cost effectiveness, and contextual awareness. The state as well as the context is assumed to be fixed and unknown and the aim of the sensor team is to estimate the state. The dynamic sensor selection framework enables us to sequentially select sensors and sample their measurements until either sufficient information about the state is gathered or adding an additional sensor is deemed too costly. What makes this framework unique is that the measurement model avoids, without significantly increasing computational burden, the often incorrect assumption that the measurements are independent conditioned on the state. Further details beyond the early conference presentation in [3] are currently being developed and will appear elsewhere.

15.3 Semantic Information Representation of Multi-Modal Signals

This section develops an efficient approach to extract low-dimensional features from heterogeneous signals. This approach facilitates the real-time applicability of the context-aware measurement and fusion models introduced in the previous section. PFSA, along with their Hilbert space framework, form the basis for the approach.

15.3.1 Structure of Probabilistic Finite State Automata

The generative, low-dimensional model to be discussed in this section is the probabilistic finite state automaton (PFSA). The rationale for having the PFSA structure as a semantic model, as opposed to other models such as hidden Markov models (HMM) [31], is that, in general, PFSA is easier to learn and may also perform better in practice. For example, experimental results [32] show that the usage of a PFSA structure could make learning of a pronunciation model for spoken words to be 10–100 times faster than a corresponding HMM, and yet the performance of PFSA is slightly better. Rao et al. [33] and Bahrampour et al. [34] have

shown that the performance of PFSA-based tools for feature extraction in statistical pattern recognition is comparable, and often superior, to that of other existing tools such as Bayesian filters, artificial neural networks, and principal component analysis. This leads to a very wide usage of PFSA in many areas such as pattern classification [35, 36] and anomaly detection [37, 38].

In formal language theory, an alphabet Σ is a (non-empty finite) set of symbols. A string s over Σ is a finite-length sequence of symbols in Σ. The length of a string s, denoted by $|s|$, represents the number of symbols in s. The Kleene closure of Σ, denoted by $\Sigma^{\star}$, is the set of all finite-length strings including the null string ε; the cardinality of $\Sigma^{\star}$ is $\aleph_0$. The set Σ^{ω} denotes the set of all strictly infinite-length strings over Σ; the cardinality of Σ^{ω} is $\aleph_1$. See [39, 40] for more details. The following is a formal definition of the PFSA.

Definition 5 (*PFSA*) A probabilistic finite state automaton is a tuple $G = (Q, \Sigma, \delta, q_0, \Pi)$, where

- Q is a (nonempty) finite set, called the set of states;
- Σ is a (nonempty) finite set, called the input alphabet;
- $\delta : Q \times \Sigma \rightarrow Q$ is the state transition function;
- $q_0 \in Q$ is the start state;
- $\pi : Q \times \Sigma \rightarrow [0, 1]$ is an output mapping which is known as a probability morph function and satisfies the condition $\sum_{\sigma \in \Sigma} \pi(q_j, \sigma) = 1$ for all $q_j \in Q$. The morph function π has a matrix representation Π, called the *(probability) morph matrix* $\Pi_{ij} = \pi(q_i, \sigma_j)$, $q_i \in Q$, $\sigma_j \in \Sigma$.

Note that Π is a $|Q|$-by-$|\Sigma|$ stochastic matrix; i.e., each element of Π is non-negative and each row sum of Π is equal to 1. While the morph matrix defines how a state sequence leads to a string of symbols, a PFSA gives rises to another stochastic matrix that defines how state sequences are formed. That is, every PFSA induces a Markov chain.

Definition 6 (*State Transition Probability Matrix)* Associated with every PFSA $G = (Q, \Sigma, \delta, q_i, \Pi)$ is a $|Q|$-by-$|Q|$ stochastic matrix P, called the *state transition (probability) matrix*, which is defined as follows:

$$P_{jk} = \sum_{\sigma : \delta(q_j, \sigma) = q_k} \pi(q_j, \sigma).$$

We are only interested in PFSA where all states are reachable (or accessible) from the initial state q_0. In particular, we focus on the following class of PFSA:

$$\mathscr{A} = \{(Q, \Sigma, \delta, q_0, \Pi) : \pi(q, \sigma) > 0 \text{ for all } q \in Q \text{ and for all } \sigma \in \Sigma\}.$$

We say that two PFSA are structurally similar if their graph representations have the same connectivity. Structurally similar PFSA only differ in the probabilities on the directed edges.

Definition 7 (*Structural Similarity*) Two PFSA $G_i = (Q_i, \Sigma, \delta_i, q_0^{(i)}, \Pi_i) \in \mathscr{A}$, $i = 1, 2$, are said to be *structurally similar* if $Q_1 = Q_2$, $q_0^1 = q_0^2$, and $\delta_1(q, \sigma) = \delta_2(q, \sigma)$ for all $q \in Q_1$ and for all $\sigma \in \Sigma$.

One can always bring two arbitrary PFSA into the common structure without loss of information by composing the two PFSA in a time-synchronous manner.

Definition 8 (*Synchronous Composition*) [35] The *synchronous composition* $G_1 \otimes G_2$ of two PFSA $G_i = (Q_i, \Sigma, \delta_i, q_0^{(i)}, \Pi_i) \in \mathscr{A}$, $i = 1, 2$, is defined as

$$G_1 \otimes G_2 = (Q_1 \times Q_2, \Sigma, \delta', (q_0^{(1)}, q_0^{(2)}), \Pi'),$$

where

$$\begin{aligned} \delta'((q_i, q_j), \sigma) &= (\delta_1(q_i, \sigma), \delta_2(q_j, \sigma)), \\ \Pi'((q_i, q_j), \sigma) &= \Pi_1(q_i, \sigma) \end{aligned}$$

for all $q_i \in Q_1$, $q_j \in Q_2$, and $\sigma \in \Sigma$.

It was shown in [35] that $G_1 \otimes G_2$ and $G_2 \otimes G_1$ describe the same stochastic process as G_1 and G_2, respectively, and yet $G_1 \otimes G_2$ and $G_2 \otimes G_1$ are structurally similar. Synchronous composition is an efficient procedure for fusing the information contents of individual PFSA into a single PFSA representation. It is, however, limited to PFSA sharing a common alphabet.

15.3.2 Hilbert Space Construction

This subsection describes the construction of a PFSA Hilbert space, which allows algebraic manipulations and comparison of PFSA. The space $\mathscr{A}$ of PFSA is a vector space with vector addition $\oplus$ and scalar multiplication $\odot$ defined as follows.

Definition 9 [40] For $G_i = (Q, \Sigma, \delta, q_0, \Pi_i) \in \mathscr{A}$, $i = 1, 2$ and for $k \in \mathbb{R}$, define operations $\oplus$ and $\odot$ as

- $G_1 \oplus G_2 = (Q, \Sigma, \delta, q_0, \Pi)$ where

$$\pi(q, \sigma) = \frac{\pi_1(q, \sigma)\pi_2(q, \sigma)}{\sum_{\alpha \in \Sigma} \pi_1(q, \alpha)\pi_2(q, \alpha)}; \tag{15.10a}$$

- $k \odot G_1 = (Q, \Sigma, \delta, q_0, \Pi')$ where

$$\pi'(q,\sigma) = \frac{(\pi_1(q,\sigma))^k}{\sum_{\alpha\in\Sigma}(\pi_1(q,\alpha))^k}. \tag{15.10b}$$

Theorem 1 [40] The triple $(\mathscr{A}, \oplus, \odot)$ forms a vector space over the real field $\mathbb{R}$.

In addition, the space $\Sigma^{\star}$ is measurable. A probability measure on $\Sigma^{\star}$ leads to a definition of inner product.

Definition 10 (*Measure* μ) [40] The triple $(\Sigma^{\star}, 2^{\Sigma^{\star}}, \mu)$ forms a measure space, where $\mu : 2^{\Sigma^{\star}} \to [0,1]$ is a finite measure satisfying the following:

- $\mu(\Sigma^{\star}) = 1$;
- $\mu(\bigcup_{k=1}^{\infty}\{s_k\}) = \sum_{k=1}^{\infty}\mu(\{s_k\})$ for all $s_k \in \Sigma^{\star}$.

For each PFSA $G = (Q, \Sigma, \delta, q_0, \Pi) \in \mathscr{A}$, denote the row vector of the morph matrix Π for a particular state q_i by Π_i, so that Π_i is a probability vector with $|\Sigma|$ components. Denote the componentwise natural logarithm of Π_i by

$$f(\Pi_i) = \begin{bmatrix} \log \Pi_{i1} & \cdots & \log \Pi_{i|\Sigma|} \end{bmatrix} \quad \text{for } i = 1, \ldots, |Q|.$$

Define $g : \mathbb{R}^{|\Sigma|} \to \mathbb{R}^{|\Sigma|-1}$ by

$$g(x) = x - \left(\frac{1}{|\Sigma|}\sum_{i=1}^{|\Sigma|} x_i\right) 1_{|\Sigma|} \quad \text{for } x \in \mathbb{R}^{|\Sigma|},$$

where $1_{|\Sigma|}$ denotes the vector in $\mathbb{R}^{|\Sigma|}$ whose components are all equal to 1. Then, overload the composition $F = g \circ f$ on the stochastic matrix as

$$F(\Pi) = \begin{bmatrix} F(\Pi_1) \\ \vdots \\ F(\Pi_{|Q|}), \end{bmatrix}$$

and define the set

$$\mathscr{H} = \{(Q, \Sigma, \delta, q_0, K) : (Q, \Sigma, \delta, q_0, \Pi) \in \mathscr{A}, K = F(\Pi)\}.$$

It is readily seen that the sets $\mathscr{H}$ and $\mathscr{A}$ are isomorphic to each other. According to (10), the linear operations on $\mathscr{A}$ involve normalization steps. We can avoid these steps if we work on $\mathscr{H}$ instead. The space $\mathscr{H}$ turns out to be a Hilbert space with inner product defined as follows.

Proposition 1 *For any* $h_i = (Q, \Sigma, \delta, q_0, K^i) \in \mathscr{H}$, $i = 1, 2$, *we have*

- $h_1 + h_2 = (Q, \Sigma, \delta, q_0, K^1 + K^2)$;
- $k \cdot h_1 = (Q, \Sigma, \delta, q_0, kK^1)$.

Proposition 2 *The function* $\langle \cdot, \cdot \rangle : \mathscr{H} \times \mathscr{H} \to \mathbb{R}$ *defined by*

$$\langle h_1, h_2 \rangle = \sum_{j=1}^{|Q|} \mu(q_j) \langle K_j^1, K_j^2 \rangle \quad \text{for } h_i = (Q, \Sigma, \delta, q_0, K^i) \in \mathcal{H},\ i = 1, 2,$$

is an inner product on $\mathscr{H}$.

The Hilbert space–structure of the space of PFSA makes it possible to speak of comparison, reduction, refinement, etc., of PFSA, which are essential operations for ensuring the scalability of PFSA-based features to data size.

15.3.3 Extension to Cross Machines

The construction in the previous subsection naturally extends to cross machines. Cross machines are obtained from two symbol sequences s_1 and s_2 associated with two different sensors, possibly of different modalities, and capture the symbol-level cross-dependence of sensor measurements.

Definition 11 (*Cross Machine*) The cross machine of two sensor measurements is defined as $(Q, \Sigma_1, \Sigma_2, \delta, q_0, \Psi)$, where

- Q is a (nonempty) finite set, called the set of states;
- Σ_1 is a (nonempty) finite set, called the alphabet of sensor 1;
- Σ_2 is a (nonempty) finite set, called the alphabet of sensor 2;
- $\delta : Q \times \Sigma_1 \to Q$ is the state transition function;
- $q_0 \in Q$ is the start state;
- $\psi : Q \times \Sigma_2 \to [0, 1]$ is the output morph function satisfying the condition $\sum_{\sigma \in \Sigma_2} \psi(q_j, \sigma) = 1$ for all $q_j \in Q$. The output morph function ψ has a matrix representation Ψ, called the *output (probability) morph matrix* $\Psi_{ij} = \psi(q_i, \sigma_j), q_i \in Q, \sigma_j \in \Sigma_2$.

One can define a Hilbert space of cross machines as well. Define

$$\mathscr{R} = \{R = (Q, \Sigma_1, \Sigma_2, \delta, q_0, \Psi) : \psi(q, \sigma) > 0 \text{ for all } q \in Q \text{ and all } \sigma \in \Sigma_2\}.$$

We will focus on the cross machines in $\mathscr{R}$.

Definition 12 (*Synchronous Composition*) The synchronous composition $R_1 \otimes R_2$ of two cross machines $R_j = (Q_j, \Sigma_1, \Sigma_2, \delta_j, q_0^{(j)}, \Psi_j) \in \mathscr{R}$, $j = 1, 2$, is defined as

$$R_1 \otimes R_2 = (Q_1 \times Q_2, \Sigma_1, \Sigma_2, \delta', (q_0^{(1)}, q_0^{(2)}), \Psi'),$$

where

$$\delta'((q_i, q_j), \sigma) = (\delta_1(q_i, \sigma), \delta_2(q_j, \sigma))$$
$$\Psi'((q_i, q_j), \tau) = \Psi_1(q_i, \tau)$$

for all $q_i \in Q_1$, $q_j \in Q_2$, $\sigma \in \Sigma_1$, and $\tau \in \Sigma_2$.

This definition ensures that $R_1 \otimes R_2$ is a non-minimal realization of R_1, and that $R_1 \otimes R_2$ and $R_2 \otimes R_1$ describe the same process. This also implies that, without loss of generality, we can consider structurally similar cross machines that only differ in their output morph matrices. As in Sect. 3.2, one can obtain a space $\mathscr{H}$ isomorphic to $\mathscr{R}$ by considering a mapping $\Psi \mapsto K$ that involves the logarithm of the output morph matrix, and then by defining the inner product of two cross machines as in Propositions 3.2 and 3.2. See [39, 40] for more details.

15.3.4 PFSA Feature Extraction: Construction of D-Markov Machine

This subsection briefly describes the procedure for constructing PFSA features from time series data. More details can be found in [38, 41].

15.3.4.1 Symbolization of Time Series

Time series data, generated from a physical system or its dynamical model, are symbolized by using a partitioning tool—e.g., maximum entropy partitioning (MEP)—based on an alphabet Σ whose cardinality $|\Sigma|$ is finite. MEP maximizes the entropy of the generated symbols, where the information-rich portions of a data set are partitioned finer and those with sparse information are partitioned coarser. That is, each cell contains (approximately) equal number of data points under MEP. The choice of $|\Sigma|$ largely depends on the specific data set and the trade-off between the loss of information and computational complexity.

15.3.4.2 *D*-Markov Machine Construction

A D-Markov Machine is a special class of PFSA [38], where a state is solely dependent on the most recent history of at most D symbols, where the positive integer D is called the depth of the machine. That is, each state of a D-Markov Machine is a string of D symbols, or less, in alphabet Σ. In general, we have $|Q| = |\Sigma|$ when $D = 1$, and $|Q| \leq |\Sigma|^{|D|}$ for $D \geq 1$.

The construction procedure for D-Markov Machines consists of two major steps; namely, *state splitting* and *state merging*. In general, state splitting increases the number of states to achieve more precision in representing the information content in the time series. Conceptually, state splitting should reduce the entropy rate, thereby, focusing on the critical states (i.e., those states that carry more information). On the other hand, state merging is the process of combining the states, often resulting from state splitting, that describe similar statistical behavior. The similarity of two states, $q, q' \in Q$, is measured in terms of the conditional probabilities of future symbol generation. A combination of state splitting and state merging is performed in order to trade off information content against feature complexity, and leads to the final form of the D-Markov Machine, possibly with $|Q| \ll |\Sigma|^{|D|}$.

15.3.4.3 Feature Extraction

Once a D-Markov Machine is constructed based on quasi-stationary time series data, the associated state probability vector is computed by frequency counting. Let $N(q)$ be the number of times state $q \in Q$ occurs in the state sequence associated with the constructed D-Markov Machine. Then the probability of state q is estimated as

$$\widehat{P}(q) = \frac{1 + N(q)}{|Q| + \sum\limits_{q' \in Q} N(q')} \quad \text{for all } q \in Q.$$

The resulting vectors $\widehat{P}(q_j)$ can be used as stationary features representing the sensor measurements for statistical inference and decision-making purposes. These feature vectors serve the role of low-dimensional versions of raw data. That is, they replace all the sensor measurements $Y_1,\ldots, Y_M$ (in the case of a team of M sensors) that appear in Sect. 15.2.

15.4 Experiments and Results

The methods presented in Sect. 15.2 for context learning, and in Sect. 15.3 for multi-modal feature extraction, were validated using a binary target classification scenario. The details are presented in this section.

15.4.1 Experimental Scenario and Data Collection

The experiment aims to identify the walking gait of a human target and classify it as normal walking or stealthy walking. The dataset used in this work was collected in

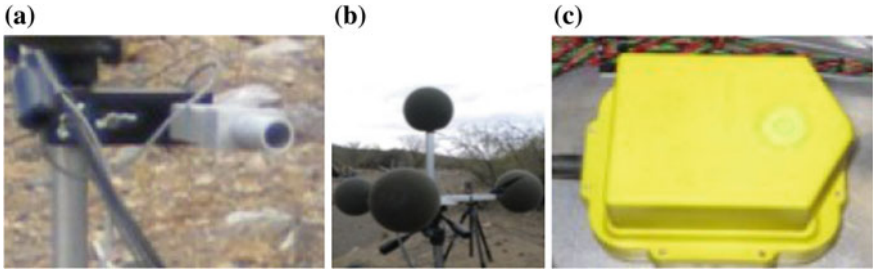

Fig. 15.2 Sensors used in the experiment. **a** PIR, **b** acoustic and **c** seismic

a field experiment conducted with the U.S. Army Research Lab. There were in total 160 observations, each collected at 4 kHz for 10 s; i.e., 40,000 data points in a time series. For each observation, the following 4 sensors recorded measurements at different locations: one passive infrared (PIR) sensor, one acoustic sensor, and two seismic sensors. These sensors are shown in Fig. 15.2.

Out of total 160 observations, there were 80 observations under each hypothesis. Hypothesis 1 (i.e., $X = 1$) corresponds to the event of a human walking with a normal gait, and hypothesis 2 (i.e., $X = 2$) corresponds to a human walking with a stealthy gait. Typical signals from the 4 sensors under each hypothesis are shown in Fig. 15.3. Out of the 80 samples for each hypothesis, 40 samples were collected in a region which had moist soil, and 40 samples in another region which had gravel soil. The response of the seismic sensors was affected by different soil conditions. If soil conditions at the sensor deployment site are unknown at the outset, then context estimation and context-aware pattern classification become important tools for measurement system modeling.

15.4.2 Data Preprocessing and Feature Extraction

In the signal preprocessing step, each time-series signal is down-sampled by a factor of 4 to give a time series of 10,000 data points. The DC component (i.e., the constant offset) of a seismic signal was eliminated by subtracting the average value, resulting in a zero mean signal. Then, the signal was partitioned using the MEP approach with a symbol size of 7. The maximum number of allowable states was varied, and the classification performance on a validation test set was found under each hypothesis using individual sensors. This process was repeated three times to obtain average error. The number of states was then chosen to be 10 for the PIR sensor and 14 for other sensors, as these numbers resulted in the minimum average error. After partitioning, symbolization was done and the feature vectors were extracted as explained in Sect. 15.3. The feature vectors constructed in this fashion replace the (down-sampled) raw sensor measurements for the purpose of target classification.

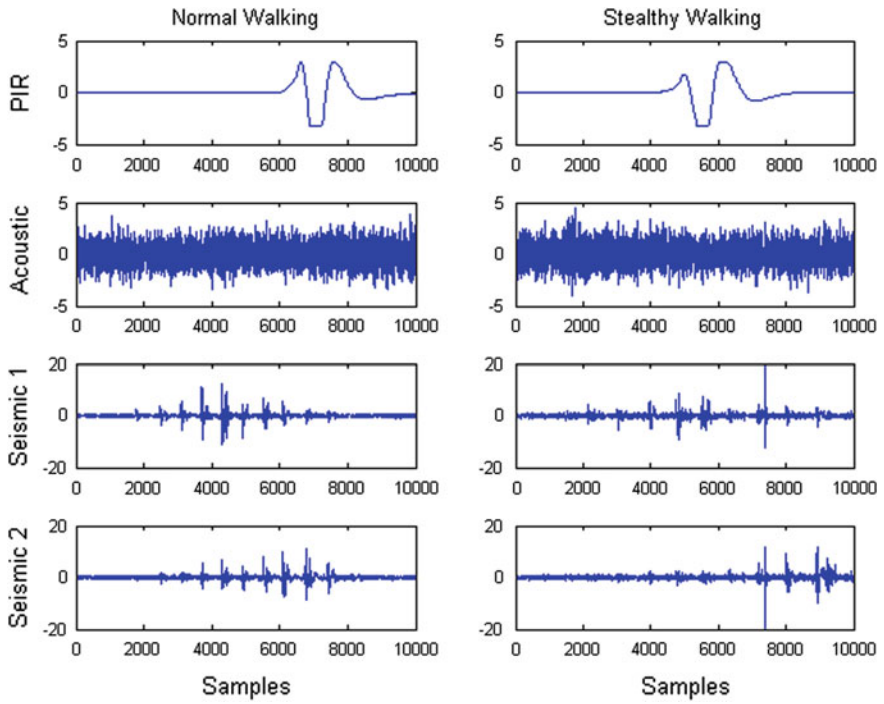

Fig. 15.3 Typical signals from the sensors used in the experiment

15.4.3 Performance Assessment

We now compare the following two approaches:

1. *Naïve Bayes approach*. This is the usual approach, where the sequential measurements are assumed to be independent conditioned on the state.
2. *Context-aware approach*. This is the proposed approach, where the sequential measurements are assumed to be independent conditioned on the state-context pair and the context is allowed to be dependent on the state.

The naïve Bayes approach assumes

$$p(Y_1, \ldots, Y_M|X) = \prod_{i=1}^{M} p_i(Y_i \,|\, X),$$

and so, as new measurements are being sampled sequentially, updates the posterior distribution of X via the standard update rule given by

Table 15.1 Confusion matrix result for target walking-type classification

(a) Naïve Bayes approach (Acc: 85.61 %)		
X	1	2
1	191	50
2	9	160
(b) Context-aware approach $\sigma = 0.01$, $\gamma = 0.01$ (Acc: 89.02 %)		
X	1	2
1	184	29
2	16	181
(c) Context-aware approach $\sigma = 0.01$, $\gamma = 0.05$ (Acc: 88.29 %)		
X	0	1
0	179	27
1	21	183

$$P(X \mid Y_1) = \frac{p_1(Y_1 \mid X)P(X)}{\sum_{x=0}^{1} p_1(Y_1 \mid x)P(x)}$$

and

$$P(X \mid Y_1, \ldots, Y_{i-1}, Y_i) = \frac{p_i(Y_i \mid X)P(X \mid Y_1, \ldots, Y_{i-1})}{\sum_{x=0}^{1} p_i(Y_i \mid x)P(x \mid Y_1, \ldots, Y_{i-1})}$$

for $i = 2, 3, M$, where $P(x)$ is the prior probability that $X = x$. The marginal likelihood $p_i(Y_i \mid X)$ for each i is estimated by computing the sample mean and sample covariance from the available data.

The context-aware approach fuses the information from sequential multi-modal measurements using the context-aware sensor fusion rule given in (15.9a, b). The context set $\mathcal{C}(X)$, as given in Definition 4, is constructed from Sect. 15.2.2.3. The SVDE approach has one user-defined parameter $\sigma > 0$ and also the kernel parameters can be chosen suitably. It was found that the results were sensitive to the choice of kernel parameters and the results shown here are not for the optimal choice of these parameters. The dataset was partitioned into randomly drawn training (75 %) and test (25 %) sets. The partitioning of the dataset was repeated 10 times and the overall classification performance is reported below. The D-Markov Machine construction-based feature extraction technique, as shown in Sect. 15.3.4, was used for extracting features from the sensor time-series data. Assuming that the D-Markov Machine feature vector from each modality has a multi-variate Gaussian distribution, the context-aware approach shows a 10.98 % error and the naïve Bayes approach gives 14.39 % error. Thus, the context-aware approach yields a 24 % reduction in classification error over that of the naïve Bayes approach. See Table 15.1 for a summary of the result.

15.5 Summary and Conclusions

In this chapter, the notion of context in the multi-modal sensor framework is mathematically formalized, and then a novel, kernel-based context learning approach is presented. Based on the resulting context-aware measurement model, a multi-modal sensor fusion approach for sequential statistical inference is discussed. A powerful feature extraction technique results in a low-dimensional feature of sensor measurements in the form of PFSA (or, in particular, the *D*-Markov Machine), which replaces the raw data set and yet captures the information from, and the dynamics of, the heterogeneous sensor system. The superior performance of the context learning approach is validated on a real-world sensor data set, whose feature is extracted through a *D*-Markov Machine.

Our major innovation consists of two sets of algorithms. One set of algorithms is for unsupervised learning of the context and for context-aware multi-modal sensor fusion. The context-learning algorithm is based on a kernel machine that yields an estimate of the joint density of multi-modal measurements, so that the support vectors serve the role of machine-generated contexts, and that the conditional independence of sensor measurements given the state-context pair, which is crucial for sequential inference, is automatically satisfied. The algorithm for context-aware sensor fusion suggests the potential of the proposed context-aware approach in realistic scenarios. The other set of algorithms is for symbolic compression–based feature extraction, which yields an attractive and scalable low-dimensional PFSA features. These features are attractive because they are endowed with a Hilbert-space structure, which enables us to replace the raw measurement data with their low-complexity features and carry out feature comparison (i.e., inner product), reduction (i.e., state merging), and refinement (i.e., state splitting) operations. Owing to these operations defined on the space of PFSA (and, in particular, the *D*-Markov Machine–based features), PFSA features are, in principle, scalable to any desired level of description.

A research direction in the immediate future is to fully develop the proposed sensor fusion and decision-making system, and demonstrate it in a scenario more complicated than those of binary hypothesis testing. The experimental validation described in Sect. 15.4 used a limited amount of data, which was obtained from a test performed by the Army Research Lab. A border-control testbed being set up at Penn State would allow collection of much larger amounts of data from several heterogeneous sensors and enable a more detailed and systematic validation of the presented techniques. Other future research paths include optimizing the proposed kernel machine for the purpose of unsupervised context learning (in terms of sparsity, generalization ability, etc.), and developing a method that unifies the context learning and feature extraction steps, which are currently separate and hence suboptimal.

Acknowledgments The work reported in this chapter has been supported in part by U.S. Air Force Office of Scientific Research (AFOSR) under Grant No. FA9550-12-1-0270 and by the Office of Naval Research (ONR) under Grant No N00014-11-1-0893. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

References

1. S. Phoha, N. Virani, P. Chattopadhyay, S. Sarkar, B. Smith, A. Ray, Context-aware dynamic data-driven pattern classification. Procedia Comput. Sci. **29**, 1324–1333 (2014)
2. N. Virani, S. Marcks, S. Sarkar, K. Mukherjee, A. Ray, S. Phoha, Dynamic data-driven sensor array fusion for target detection and classification. Procedia Comput. Sci. **18**, 2046–2055 (2013)
3. N. Virani, J.W. Lee, S. Phoha, A. Ray, Dynamic context-aware sensor selection for sequential hypothesis testing, in *2014 IEEE 53rd Annual Conference on Decision and Control (CDC),* 2014, pp. 6889–6894
4. D.K. Wilson, D. Marlin, S. Mackay, Acoustic/seismic signal propagation and sensor performance modeling, in *SPIE*, vol. 6562, 2007
5. F. Darema, Dynamic data driven applications systems: new capabilities for application simulations and measurements. In: Computational Science–ICCS 2005, Springer, 2005, pp. 610–615
6. A. Olivaa, A. Torralba, The role of context in object recognition. Trends Cogn. Sci. 520–527 (2007)
7. R. Rosenfield, Two decades of statistical language modeling: Where do we go from here? (2000)
8. B. Schilit, N. Adams, R. Want, Context-aware computing applications, in *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on, IEEE*, 1994, pp. 85–90
9. H. Frigui, P.D. Gader, A.C.B. Abdallah, A generic framework for context-dependent fusion with application to landmine detection, in *SPIE Defense and Security Symposium, International Society for Optics and Photonics*, 2008, pp. 69,531F–69,531F
10. C.R. Ratto, Nonparametric Bayesian context learning for buried threat detection. Ph.D. thesis, Duke University, (2012)
11. C. Bron, J. Kerbosch, Algorithm 457: Finding all cliques of an undirected graph. Commun. ACM **16**(9), 575–577 (1973)
12. E. Tomita, A. Tanaka, H. Takahashi, The worst-case time complexity for generating all maximal cliques and computational experiments. Theor. Comput. Sci. **363**(1), 28–42 (2006)
13. M. Newman, Fast algorithm for detecting community structure in networks. Phys. Rev. E. **69** (2003)
14. N. Virani, J.W. Lee, S. Phoha, A. Ray, Learning context-aware measurement models, in *Proceedings of the 2015 American Control Conference*, IEEE 2015, pp. 4491–4496
15. P.R. Kumar, P. Varaiya, Stochastic systems: estimation, identification and adaptive control, Prentice-Hall, Englewood Cliffs, NJ, 1986
16. A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Stat. Soc. B **39**(1), 1–38 (1977)
17. C.R. Ratto, P.A. Torrione, L.M. Collins Context-dependent feature selection using unsupervised contexts applied to GPR-based landmine detection, in *SPIE Defense, Security, and Sensing, International Society for Optics and Photonics*, 2010, pp. 76,642I–76,642I

18. H. Akaike, A new look at statistical model identification. IEEE Trans. Autom. Control **19**(6), 716–723 (1974)
19. G. Schwarz, Estimating the dimension of a model. Ann. Stat. **6**(2), 461–464 (1978)
20. C.R. Ratto, K.D. Morton, L.M. Collins, P.A. Torrione Contextual learning in ground-penetrating radar data using dirichlet process priors, in *Proceedings of SPIE, the International Society for Optical Engineering, Society of Photo-Optical Instrumentation Engineers*, 2011
21. F. Cucker, S. Smale On the mathematical foundations of learning. Bulletin (New Series) of the Am. Math. Soc. 39(1):1–49 (2001)
22. B. Schölkopf, C.J.C. Burges, A.J. Smola (eds.), *Advances in Kernel Methods: Support Vector Learning* (MIT Press, Cambridge, MA, 1999)
23. J.W. Lee, P.P. Khargonekar, Distribution-free consistency of empirical risk minimization and support vector regression. Math. Control Sig. Syst. **21**(2), 111–125 (2009)
24. J.L. Kelley et al., *Linear Topological Spaces* (Springer, New York, NY, 1976)
25. A.J. Smola, B. Schölkopf, A tutorial on support vector regression. Stat. Compu. **14**(3), 199–222 (2004)
26. V.N. Vapnik, *Statistical Learning Theory* (Wiley, New York, NY, 1998)
27. C.C. Chang, C.J. Lin, Training *v*-support vector regression: theory and algorithms. Neural Comput. **14**(8), 1959–1977 (2002)
28. E. Parzen, On estimation of a probability density function and mode. Ann. Math. Stat. 1065–1076 (1962)
29. S. Mukherjee, V. Vapnik, Support vector method for multivariate density estimation. Cent Bio. Comput. Learn. Dept Brain and Cogn. Sci., MIT CBCL **170** (1999)
30. J. Weston, A. Gammerman, M.O. Stitson, V. Vapnik, V. Vovk, C. Watkins, Support vector density estimation. (Advances in kernel methods, MIT Press, 1999), pp. 293–305
31. L. Rabiner, A tutorial on hidden markov models and selected applications in speech proccessing. Proc. IEEE **77**(2), 257–286 (1989)
32. D. Ron, Y. Singer, N. Tishby, On the learnability and usage of acyclic probabilistic finite automata. J. Comput. Syst. Sci. **56**(2), 133–152 (1998)
33. C. Rao, A. Ray, S. Sarkar, M. Yasar, Review and comparative evaluation of symbolic dynamic filtering for detection of anomaly patterns. SIViP **3**, 101–114 (2009)
34. S. Bahrampour, A. Ray, S. Sarkar, T. Damarla, N.M. Nasrabadi, Performance comparison of feature extraction algorithms for target detection and classification. Pattern Recogn. Lett. **34**, 2126–2134 (2013)
35. I. Chattopadhyay, A. Ray, Structural transformations of probabilistic finite state machines. Int. J. Control **81**(5), 820–835 (2008)
36. X. Jin, S. Sarkar, A. Ray, S. Gupta, T. Damarla, Target detection and classification using seismic and PIR sensors. IEEE Sens. J. **12**(6), 1709–1718 (2012)
37. S. Gupta, A. Ray, Statistical mechanics of complex systems for pattern identification. J. Stat. Phys. **134**(2), 337–364 (2009)
38. A. Ray, Symbolic dynamic analysis of complex systems for anomaly detection. Sig. Process. **84**(7), 1115–1130 (2004)
39. P. Adenis, Y. Wen, A. Ray, An inner product space on irreducible and synchronizable probabilistic finite state automata. Math. Control Sig. Syst. **23**(4), 281–310 (2012)

40. Y. Wen, S. Sarkar, A. Ray, X. Jin, T. Damarla, A unified framework for supervised learning of semantic models, in *Proceedings of the 2012 American Control Conference*, pp. 2183–2188, IEEE, (2012)
41. K. Mukherjee, A. Ray, State splitting and merging in probabilistic finite state automata for signal representation and analysis. Sig. Process. **104**, 105–119 (2014)