

Full Paper

Multi-agent Negotiation Mechanisms for Statistical Target Classification in Wireless Multimedia Sensor Networks

Xue Wang *, Dao-wei Bi [†], Liang Ding [‡] and Sheng Wang [§]

State Key Laboratory of Precision Measurement Technology and Instrument, Tsinghua University, Beijing 100084, P. R. China; E-mails: [†]bdw02@mails.tsinghua.edu.cn;

[‡]dintliang03@mails.tsinghua.edu.cn ; [§]wang_sheng00@mails.tsinghua.edu.cn

* Author to whom correspondence should be addressed; E-mail: wangxue@mail.tsinghua.edu.cn

Received: 24 September 2007 / Accepted: 9 October 2007 / Published: 11 October 2007

Abstract: The recent availability of low cost and miniaturized hardware has allowed wireless sensor networks (WSNs) to retrieve audio and video data in real world applications, which has fostered the development of wireless multimedia sensor networks (WMSNs). Resource constraints and challenging multimedia data volume make development of efficient algorithms to perform in-network processing of multimedia contents imperative. This paper proposes solving problems in the domain of WMSNs from the perspective of multi-agent systems. The multi-agent framework enables flexible network configuration and efficient collaborative in-network processing. The focus is placed on target classification in WMSNs where audio information is retrieved by microphones. To deal with the uncertainties related to audio information retrieval, the statistical approaches of power spectral density estimates, principal component analysis and Gaussian process classification are employed. A multi-agent negotiation mechanism is specially developed to efficiently utilize limited resources and simultaneously enhance classification accuracy and reliability. The negotiation is composed of two phases, where an auction based approach is first exploited to allocate the classification task among the agents and then individual agent decisions are combined by the committee decision mechanism. Simulation experiments with real world data are conducted and the results show that the proposed statistical approaches and negotiation mechanism not only reduce memory and computation requirements in WMSNs but also significantly enhance classification accuracy and reliability.

Keywords: Wireless multimedia sensor networks, multi-agent negotiation, committee decision, Gaussian process classification, principal component analysis.

1. Introduction

Recent advances in micro-electro-mechanical systems and wireless communications have enabled the development of tiny, low-cost and low-power sensor nodes [1]. Wireless sensor networks (WSNs) are composed of large number of such sensor nodes that autonomously form wireless networks and can collaboratively monitor the environment [2]. Over the last few years, WSNs have drawn much attention in the research community and have been extensively investigated [2, 3]. Application fields of WSNs are highly diversified and include military counterterrorist operations [4], target tracking and classification [5], agriculture and food industry [6], underwater surveillance [7], structure health monitoring [8], industrial maintenance [9] and medical care [10].

More recently, the availability of inexpensive hardware such as CMOS cameras and microphones able to capture multimedia information from the environment has led to the development of wireless multimedia sensor networks (WMSNs) [11]. WMSNs are a newly emerging type of WSNs that comprise sensor nodes equipped with cameras, microphones, and other sensors retrieving audio, video and other scalar data. This new type of WSNs, apart from boosting existent WSN applications as stated above, will enable a whole new range of applications, which include multimedia surveillance sensor networks, traffic congestion avoidance and control system, and industrial process control [11], and so forth.

The mainstream concerns of traditional WSN research have been focused on decreasing energy consumption to extend network longevity under resource constraints such as battery, memory and processing capability. In contrast, WMSNs have another objective that is as important as (if not more than) reduction of resource consumption. That objective is efficient delivery of application level quality of service (QoS) and the mapping of these requirements to network layer metrics such as latency and jitter [11]. This goal is a challenging and largely unexplored task due to issues like resource constraints, variable channel capacity and multimedia in-network processing.

As identified in [11], flexible architecture, multimedia source coding techniques and multimedia in-network processing are among the key elements of WMSN designs. WMSN architectures have to support heterogeneous and independent applications with diversified requirements. Subsequently it is imperative to develop flexible and hierarchical architectures that can accommodate various applications in the same infrastructure. Uncompressed raw audio and video data require excessive memory and bandwidth which are scarce resources in WMSNs. For this reason, it is apparent that efficient processing techniques for raw data compression are indispensable. WMSNs allow performing multimedia in-network processing algorithms on the raw data acquired from the environment. New architectures are required for collaborative, distributed, and resource-constrained processing, which may enhance system scalability by reducing the transmission of redundant information and merging data from various sources.

Based upon our previous work [12], in this paper we propose hierarchical multi-agent architectures for WMSNs and enable collaborative multimedia in-network processing by multi-agent cooperation. Multi-agent systems (MAS) are computational systems where two or more agents interact and work together to satisfy some set of goals [12, 13]. In MAS, an agent is considered as a computational mechanism that exhibits a high degree of autonomy, performing actions in its environment based on information extracted from the environment (e.g. by sensors) [13]. The autonomous and social natures

of MAS make it extremely easy to satisfy these WMSN requirements such as flexible architecture and collaborative in-network processing [12].

In this paper, we focus on target classification problems in WMSNs where audio information is retrieved from the environment by microphone sensors. To deal with the statistical nature of audio information retrieval, a set of statistical methods are employed. As memory and bandwidth are critical limitations for WMSNs, features are first extracted from raw audio data by power spectrum analysis and then further compressed through principal component analysis. The highly compressed features are used by the statistical Gaussian process classifier, which determines the class information of the observed target. Energy preservation is a persistent and essential issue for both WSNs and WMSNs, as a consequence, multi-agent negotiation mechanisms are specially designed to allocate classification tasks among agents by means of auction and combine individual decisions in a committee manner, with the aim to extend network longevity and accomplish efficient collaborative multimedia in-network processing. Simulated experiments with real world audio data are performed to evaluate the proposed multi-agent negotiation mechanisms and statistical methods.

The rest of this paper is organized as follows. In Section 2, related work is surveyed on multi-agent WSN architectures and agent negotiation mechanisms. Statistical classification and dimension reduction techniques are also overviewed in this section. Section 3 presents the proposed negotiation mechanism for target classification in WMSNs. In the section that follows, relevant statistical approaches are introduced, including power spectral density estimation, principal component analysis and Gaussian process classification. Section 5 presents the simulation experiments with real world data. In the final section, conclusions and future work are provided.

2. Related Work

Solving problems in WSN domains from the perspective of multi-agent systems has recently drawn much attention within the research community. Multi-agent frameworks offer scalability and efficient collaborative processing capability to WSNs, which is desirable for WMSNs as well. Negotiation is one of the frequently employed mechanisms to achieve efficient collaborative processing. Dimension reduction has special significance, because multimedia data volume is usually prohibitively huge for limited memory and bandwidth in WMSNs. Gaussian process classification is a promising practical Bayesian approach, and an attempt is made in this paper to explore its application in WMSNs. Work related to these issues is overviewed as follows.

2.1. Multi-agent architecture of WSNs

One of the desirable characteristics for WSNs is autonomous operation, which essentially means the network demonstrates some kind of artificial intelligence. Among various approaches in the domain of artificial intelligence, multi-agent systems (MAS) have been recognized as a promising one to imbue WSNs with intelligence [14], due to its distributed nature.

Strong correspondence between autonomous WSNs and MAS has been identified in [14], [15] and [16]. In [14] it was shown that MAS can empower the sensor nodes with autonomous self management. MAS are applied to WSNs for lighting control in [15], to accomplish the required self configuration

and address the uncertainty in making decisions. Power management is critical for WSNs and agent deliberation is proposed in [16] to achieve adaptive WSN control.

The work above focuses on the justification of deploying MAS in WSNs, and consequently few details on MAS architecture designs are provided. By contrast, [17], [18] and [19] propose several deployment architectures in detail. Generic agent architectures for WSNs with built-in security components are proposed in [17], but this work focuses on the architecture of a single agent instead of MAS. A multi-agent architecture is reported in [18], which divides agents in the WSNs into four categories according to the tasks they perform. In [19], peer to peer MAS architectures for WSNs are developed. MAS are also deployed for visual sensor networks [20], but the application scenario is wired networks other than wireless networks.

In this paper, we attempt to extend MAS architectures to WMSNs and focus on detailed MAS implementations, such as agent mental state models and collaboration mechanisms, in addition to high level multi-agent architecture designs.

2.2. Multi-agent negotiation

As noted above, one of the benefits offered by MAS is intelligent collaborative processing. There are various approaches to achieve collaboration between agents, and negotiation is a prevalent form of interaction that enables groups of agents to arrive at a mutual agreement regarding some belief, goal or plan, and so forth.

Diversified negotiation mechanisms have been reported in the literature. Multi-issue negotiation problems are investigated in [21], [22] and [23], but focus on different aspects. In [21], effort is made to address the problems of limited computational resources and tough deadlines. Bilateral multi-issue negotiation in an incomplete information setting is studied in [22]. In [23], they investigated negotiation in the scenarios of interdependent multiple issues, where nonlinear utility functions are used.

Relatively more work has been reported regarding single issue than multi-issue negotiation. A comprehensive survey of popular negotiation mechanisms can be found in [24]. It presents an auction based market architecture system for multi-agent contract negotiation in [25]. In [26] a general approach for negotiation is developed and a generic theory of strategy in negotiation interactions is proposed in [27].

Little work has been reported on negotiation in WSNs. In [28] agent negotiation is employed to solve the resource allocation problem in radar sensor networks for target tracking. It takes real time issues into serious consideration, and integrates a real time Belief-Desire-Intention (BDI) model and a temporal logic model. It establishes a virtual market in [29] to enable agent negotiation so as to efficiently allocate limited energy, radio bandwidth and other resources in sensor networks.

Multi-agent negotiation for problem solution in WSNs (applicable to WMSNs too), as emphasized in these works, is application specific. Thus available negotiation mechanisms should be carefully chosen and adapted to accommodate particular applications in WMSNs.

2.3. Statistical dimension reduction and classification

In the statistical classification domain, one of the most active directions is the development of practical Bayesian methods [30]. Gaussian processes classification (GPC) represents one of the most important practical Bayesian classification methods [30, 31]. Overviews and principles of Gaussian process and its classification applications can be found in [30] and [31]. Several approximation schemes have been suggested for GPC, which include, among others, Laplace's method [32], variational approximations [33], mean field methods [34] and expectation propagation [35]. GPC has found applications in gender classification [36], biomarker discovery [37], target recognition [38] and multi-user detection in CDMA receivers [39].

As noted, in WMSNs it is imperative to significantly reduce multimedia data volume to satisfy memory and computation constraints. This is generally achieved by feature extraction and dimension reduction [5, 40, 41]. Feature extraction is more application specific [42], but dimension reduction is a general issue. Publications [43] and [44] review traditional and current state-of-the-art dimension reduction methods. Among these available methods, principal component analysis (PCA) is one of the simple but effective dimension reduction techniques [43], which has found applications in feature detection [45], disease diagnosis [46], biomedical sample identification [47], shape retrieval [48] and remote sensing [49].

Our investigation is inspired by the above literature, and initiatives are taken to implant these approaches into WMSN scenarios which have not been explored much. Efforts are made to tailor or improve these approaches to address many of the new challenges presented by the emerging WMSNs.

3. Negotiation Mechanisms for Target Classification

It is pointed out in [11] that WMSN architectures have to support heterogeneous applications with different requirements and consequently it is necessary to develop flexible hierarchical architectures that can accommodate the requirements of all these applications in the same infrastructure. In this paper, a hierarchical architecture is developed to deploy MAS in WMSNs. There are two principal objectives in WMSN target classification, i.e. guaranteeing reliable classification accuracy and accommodating to constraints such as limited energy, bandwidth and computation capabilities in WMSNs. To satisfy these two goals concurrently, the mechanism of negotiation is incorporated and specially designed.

3.1. Hierarchical multi-agent architecture for WMSNs

Multi-agent architectures for WSNs have been extensively investigated in [12], [17], [18] and [19]. As far as system architecture is concerned, there is essentially no difference between WSNs and WMSNs; therefore those proposed architectures are equally applicable to WMSNs. Based on these reported works, we propose the following hierarchical architecture for WMSNs. The multi-agent WMSN architecture comprises four layers. At the top layer is the front-end interface agent responsible for accepting user requests, dispatching user directions and providing feedbacks in the form of static images, video or audio. To facilitate network management, the network is divided into several regions

managed by corresponding regional agents, based on geographical or similar criteria. A regional agent is usually provided with rich power supplies and computational resources, managing agents within its supervised region. To improve in-network processing efficiency and reduce communication load, a region is further divided into several sub-regions which are usually called clusters and managed by respective cluster agents. At the bottom layer is the query agent which exactly corresponds to a sensor node. Obviously a query agent is responsible for audio and video information acquisition and further processing. Evidently the proposed hierarchical multi-agent WMSN architecture is scalable, and readily enables the flexible configurations required by WMSNs.

3.2. Agent reasoning model and communication language

One of the dominant and most appealing advantages offered by MAS deployment in WMSNs is intelligent and collaborative in-network processing. As far as collaboration between agents is concerned, there are at least two important issues that have to be considered, namely how an agent reasons about the action to take and how the agents communicate with each other.

There are diversified reasoning models for agents and we have chosen the Belief-Desire-Intention (BDI) model [50] because it has been well developed in theory [50, 51, 52, 53] and adopted in several sensor network applications [16, 20, 28].

The foundation for most implemented BDI systems is the abstract interpreter proposed by Rao and Georgeff [50]. BDI focuses on representing the agent's mental states in a way imitating human beings. The beliefs represent knowledge of the world and describe the state of the world from the point of view of an agent. For WMSN applications, beliefs are essential [20] because the environment is dynamic and therefore past events need to be remembered. Moreover each agent in WMSNs only has a local view of the environment; consequently events outside its sphere of perception need to be memorized. Desires refer to such objectives as the agent would like to accomplish. Intentions represent what the agent has chosen to do. They are effectively the desires that the agent has to some extent committed to. From the point of view of implementation, intentions are the set of plans that have been adopted by the agent. Plans refer to a set of sequential actions that an agent performs to achieve one or more of its intentions.

Agent interaction requires some kind of communication, and the most accepted agent communication languages (ACL) are those based on speech-act theory, for example, FIPA-ACL developed by the Foundation for Intelligent Physical Agents (FIPA) [54]. FIPA-ACL presumes two agents share a common ontology, which ensures the agents ascribe the same meaning to the symbols used in the message. It also defines the individual message types that are central to the ACL specification. In particular, the form of the messages and meaning of the message types are defined.

Though many realizations of the BDI interpreter have been developed, the release of JADEX is gaining more acceptances recently [55, 56, 57]. JADEX is based on the BDI model and uses FIPA-ACL for communication. It is a promising technology and has been used for sensor network implementation in [20]. In this paper, we employ JADEX to implement the negotiation mechanism within the multi-agent WMSN framework.

3.3. Two phase negotiation mechanisms

Collaborative processing is crucial for WMSN applications where energy supply and computation capabilities are severely limited. As stated above, multi-agent negotiation is an effective mechanism to achieve collaboration and cooperation in MAS. Formally we propose the following definition [58]: negotiation is a form of interaction in which a group of agents, with conflicting interests, try to come to a mutually acceptable agreement on the division of scarce resources.

In WMSN scenarios, the scarce resources primarily include energy, memory, communication and computational capability. The objective of the negotiation is to guarantee reliable and high classification accuracy and in the meanwhile satisfy such constraints as limited power and low bandwidth. Each agent has the desire to classify the target detected in the WMSN sensing field. But not all agents can realize their desires due to limited resources. Only the agents enabling efficient resource usage are allowed to participate in the classification tasks and consequently realize their desires. Essentially each agent possesses its own resources of energy and computation, so in this paper, resource allocation actually means whether the agent should engage in a classification task that necessitates usage of these resources.

Thus the negotiation in this context refers to the mechanism where the agents come to a mutually acceptable agreement on how to allocate classification tasks among the agents and how to make the most reliable and accurate decision from individual decisions of the involved agents. Consequently the negotiation can be intuitively divided into two phases, namely task allocation and individual decision combination.

3.3.1. Phase one: task allocation

We first investigate the negotiation mechanisms for efficient task allocation among all the agents. Obviously there are usually many different possible allocations, so negotiation can be seen as a “distributed search through a space of potential agreements” as proposed by Jennings in [58]. Suppose the deployed WMSNs, following the proposed hierarchical multi-agent architecture, are represented by $A = \{a_1, a_2, \dots, a_m\}$, where m is the number of deployed sensor nodes and a_i denotes an agent or equivalently a sensor node. The search space can be seen as a set of deals $\Psi = \{\Omega_1, \Omega_2, \dots, \Omega_n\}$ where n is the size of the search space and Ω_i represents one possible deal [59, 60]. For task allocation problems, a deal Ω_i can be expressed in the form of an m -dimensional vector whose j -th component indicates whether the task is assigned to the agent a_j . How to indicate such an assignment is a trivial issue, and a common choice is to use +1 for assignment and -1 for the opposite. For example, in a given scenario where $m=4$, the deal $\Omega_1 = [+1, -1, +1, -1]$ means the task is assigned to a_1 and a_3 .

Since the goal is to find the most appropriate task allocation strategy, then it naturally leads to the problem of deciding whether one deal is better than another. In agent methodology, this is determined by a binary preference relationship \succeq [59]. If it holds $\Omega_i \succeq_g \Omega_j$, then we know that for agent g , the deal Ω_i is at least as good as Ω_j . It is also common to use a similar notation \succ to express the concept of strict preference. By $\Omega_i \succ_g \Omega_j$ it means for agent g the deal Ω_i is better than Ω_j [59, 60]. For convenience, the preference relationship is often described in terms of a utility function $U_g : \Psi \rightarrow R$,

which assigns a real number to each possible deal, evaluated by agent g . By means of utility functions, we have the following relationship [59]:

$$U_g(\Omega_i) \geq U_g(\Omega_j) \Leftrightarrow \Omega_i \succeq_g \Omega_j \quad (1)$$

Each agent may have its own preference and therefore a deal may be preferred by one agent but disliked by another. As far as MAS are concerned, the objective should be to find a deal that is preferred by as many agents as possible. The frequently used criterion to determine such a deal is Pareto optimal [59]. A deal is Pareto optimal if it is not dominated by any other deal. A deal Ω_i is said to dominate a deal Ω_j if for any agent g in the MAS, $U_g(\Omega_i) > U_g(\Omega_j)$ holds true. There are a variety of mechanisms to find the Pareto optimal deal, such as game-theoretic approaches, argumentation and auctions [59, 60].

These approaches have all been extensively investigated and have found application in a wide range of fields. Different applications may have distinct requirements, and consequently the design objectives of negotiation mechanism should be identified first before choosing the most suitable mechanism.

In addition to the requirements of efficient resource usage and reliable classification accuracy in our investigated problem, the negotiation should be conducted in a real-time manner for WMSN applications in a general sense. In [28], it outlines the following design objectives for negotiation in a real time, dynamic and distributed environment.

Objective 1: A negotiation should be bounded by time, which means whether successful or not, a negotiation should complete within a predefined time window.

Objective 2: Each step of the negotiation should be fast, so that the negotiation process that consists of multiple steps will be finished quickly.

Objective 3: A negotiation should be kept short, that is, the number of iterations should be minimized.

Objective 4: The negotiation-related messages should be kept short so as to reduce loss and improve communication speed.

These design objectives serve as our design guidelines too, because our applications also fall into the domain of sensor networks and present even stronger real-time requirements. Taking these design objectives into account, we choose the auction based negotiation mechanism for collaborative target classification in WMSNs. Auction is simple and easy to implement [59], therefore it can easily satisfy the real-time requirements. But traditional auction mechanisms have to be modified to accommodate the task allocation problem in WMSNs.

As is well known, an auction system consists of three components, namely, the auctioneer, bidders and goods (items). In the task allocation problem, many agents bid to try to get involved in the classification tasks. There is actually only one item (i.e. the classification task) to be auctioned, however it is desired in the investigated target classification applications that several agents may win the bidding simultaneously. This is contradictory to commonly used auction mechanisms where an item can only be sold to a single bidder [59]. Moreover, in traditional auction, it may require several rounds to come to a final deal, which is time consuming and undesirable for real-time processing required in WMSNs.

To address these problems, a modified version of the auction mechanism called One Shot Dummy Multi-Item Auction (OSDMIA) is particularly proposed. OSDMIA reaches an agreement on task allocation in a single bidding round and thus guarantees real-time in-network processing. By dummy multi-item auction, it is intended to mean that allocation of the single classification task to several agents is treated as allocation of several different tasks (essentially the same task) among the agents. In other words, selling the same item to several buyers simultaneously is seen as selling several dummy duplicates of the item to several buyers. Since only one item is actually sold, therefore it is called *dummy* multi-item auction. By OSDMIA, this special task allocation issue is transformed into a traditional on shot multi-item auction problem.

In the following, the details of OSDMIA are presented. The OSDMIA auction system can be represented by the tuple $\{Auctioneer, Bidders, Items\}$. In the tuple, the auctioneer is responsible for supervising the auction and the bidders are the agents trying to get involved in the classification task. Suppose the task is predefined to assign to N_i agents, then the items exactly refer to the N_i dummy classification tasks. Recall the MAS are represented by $A = \{a_1, a_2, \dots, a_m\}$. Suppose the agent a_{Auc} acts as the auctioneer. Then the bidders are actually the agents $A \setminus \{a_{Auc}\}$. Denote the items by $\{T_1, T_2, \dots, T_{N_i}\}$. In this way, the negotiation system can be expressed as:

$$OSDMIA = \{a_{Auc}, A \setminus \{a_{Auc}\}, \{T_1, T_2, \dots, T_{N_i}\}\} \quad (2)$$

Two of the most important issues concerning OSDMIA can be easily identified, namely specification of the auctioneer a_{Auc} and determination of N_i . These issues are essentially open and difficult to find optimal solutions. However if they are viewed from the perspective of engineering rather than theory, then things get much easier. Engineering applications don't necessarily require the solutions to be optimal. Instead as long as the solution is sufficient, then it is desirable and applicable. Consequently some sufficient solutions to the two issues are suggested.

Recall that MAS are deployed on WMSNs in a hierarchical manner, where the network is divided into a set of clusters coordinated by cluster agents. So a straightforward choice is to specify the cluster agent as the auctioneer a_{Auc} . It must be clarified that in this sense the MAS denote the cluster only instead of the whole WMSNs. In other words, the MAS considered in the negotiation process are only part of the whole MAS representing the WMSNs. But the notations are not changed, therefore in the following m is used to denote the number of agents in the cluster and the multi-agent system $A = \{a_1, a_2, \dots, a_m\}$ represents the cluster.

By contrast, determination of N_i is comparatively more complicated than the choice of the auctioneer. On one hand, scarce resources make it necessary to set N_i as small as possible, however on the other hand, a larger N_i is desirable to enhance the classification accuracy which is one of the dominant objectives in target classification applications. To reach a compromise between them, in this paper it is proposed to set N_i as 3.

Now that the OSDMIA system is established, we proceed to consider some other important issues related to temporal performance in the auction. A fundamental issue is when OSDMIA will start. This is important but in some sense simple. When a target is detected, the auctioneer a_{Auc} starts OSDMIA by sending a message to all the agents in the cluster to ask for bidding. If the agents are in critical energy state or engaged in some other tasks, they may simply refuse to bid. Otherwise, they retrieve audio information about the target and decide how to bid. As stated above, OSDMIA is a one shot

auction; therefore each agent only has the chance to bid once. Finishing bidding in one round satisfies the design objective that a negotiation should be kept short. Another issue related to time is the negotiation time bound. The auctioneer cannot wait endlessly for the agents to bid. The solution is to specify a time window for the bidding process. Bids made within the time window are valid, but if they are made out of it, they will be simply rejected or discarded. Since the allowable time window varies with system requirements, therefore its specification is application specific.

In any auction, to buy an item, the agent has to bid some price. Consequently it is necessary to investigate the form of price in OSDMIA. As required for efficient and reliable in-network processing, the agent reporting higher classification accuracy and more available resources should be more likely to win the auction. Additionally with other factors being the same, if the agent is closer to the target, its classification result should be more reliable, because the signals are less contaminated by noises. Based on these observations, the bidding price should be a combination of these quantities. Hence the bidding is assumed to take the following form.

$$Bidding = \{C_a, A_r, S_s\} \quad (3)$$

In (3) C_a denotes a priori classification accuracy; A_r represents available resources and S_s indicates the strength of the observed signals which implicitly indicates the distance between the agent and the detected target. To simplify the discussion, A_r is restricted to represent available energy only, that is to say, the percentage of the remaining energy. Suppose the detected audio signal $x[n]$ contains N_s samples. Under such assumption the signal strength S_s is calculated by

$$S_s = \sum_{n=1}^{N_s} |x[n]|^2 \quad (4)$$

Given the form of the bidding, the price can be obtained simply by means of the utility function

$$U : C_a \times A_r \times S_s \rightarrow R \quad (5)$$

Careful examination of (5) shows the three quantities (i.e. C_a , A_r and S_s) are expressed in different scales. Classification accuracy C_a and available resource A_r are expressed in percentage (relative value), but signal strength S_s is expressed in absolute value. To make them consistent and comparable, it is proposed to normalize S_s to the range between 0 and one hundred percent. The normalization is performed by the auctioneer. It finds the largest S_s denoted by S_{smax} from all the bids and normalize it to unit, while other signal strength is normalized accordingly. Subsequently the bids are evaluated by the auctioneer a_{Auc} following the utility function

$$U_{Auc} : C_a \times A_r \times (S_s / S_{smax}) \rightarrow R \quad (6)$$

The utility function U_{Auc} may take different forms but in this paper we simply propose the following one

$$U_{Auc}(C_a, A_r, S_s) = C_a^2 A_r (S_s / S_{smax})^2 \quad (7)$$

Using this utility function, available energy contributes more to the price, because it has the unity exponent. Note that for numbers less than one, the larger their exponents are, the smaller their evaluated values are. For example, if A_r and C_a are equal and less than one, then C_a^2 is smaller than A_r .

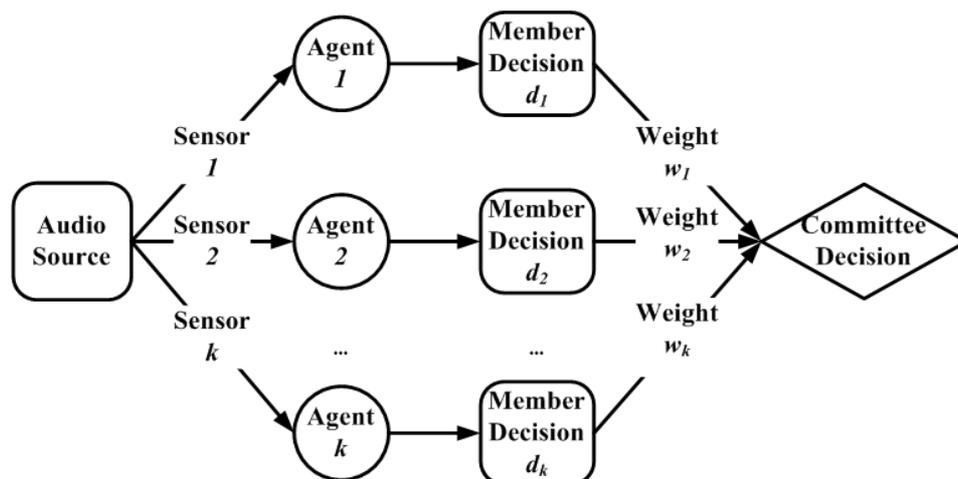
A priori classification accuracy and signal strength are treated as equally important. The reason to put larger weight on A_r is very straightforward, because energy is critical to prolong the lifetime of WMSNs.

Suppose there are N_r buyers bidding in OSDMIA. Usually N_r is less than $\text{card}(A \setminus \{a_{\text{Auc}}\})$ which is actually $m-1$. The reason is at least twofold. On one hand, some bids may be made beyond the time window specified for the auction, and therefore they will be considered as invalid bids. On the other hand, some agents may refuse to bid because, for example, they are in critical shortage of power and can not possibly get involved in the classification task. The auctioneer determines the price of each bidding agent by (7) and chooses the three agents that bidding the highest price as the winner of OSDMIA. The auctioneer notifies these agents of their winning the auction and these winners start to be engaged in classification of the detected target immediately.

3.3.2. Phase two: combination of individual decisions

The reason that several agents are selected by OSDMIA to jointly participate in the classification is principally due to the uncertainties related to audio signal acquisition and the predictions the classifiers make. After the winners of OSDMIA make their individual decisions, they should be combined to make a more reliable decision. There are various approaches to such combination, and we adopt the committee decision mechanism proposed in [61] and [62]. This approach views each winner as a member of the committee, where the final decision is made based on member decisions. There is a variety of benefits of committee decision, for example, the committee decision can, to some degree, cancel out errors of the individual committee members. Usually the decision of the committee is obtained by a weighted combination of the decisions of the committee members [61, 62].

Figure 1. Illustration of target classification committee decision in WMSNs.



The weighted committee decision used in this paper is schematically illustrated in Figure 1. The winner agents (i.e. committee members), retrieve audio information by corresponding sensors and make individual decision. The committee decision D is made by a weighted combination of member decisions:

$$D = \frac{\sum w_i d_i}{\sum w_i} \quad (8)$$

In (8), individual decision d_i is made by the member i . It must be clarified that the classification decision made by the winning agents in this paper indicates the probability that the target belongs to a certain category. In other words, decision d_i is within the range between 0 and 1. Weight w_i is employed to indicate the significance of the member i . The introduction of the term $\sum w_i$ is meant to normalize the weights. This is no unified approach to set weight for each committee member. In this paper, similar to the determination of the utility functions, we propose to set the weight as

$$w = C_a S_s / \hat{S}_{s \max} \quad (9)$$

where $\hat{S}_{s \max}$ denotes the largest signal strength of all the committee members, which differs from $S_{s \max}$ used in (7).

Setting the weight this way, it means that classification accuracy C_a and signal strength S_s make equally significant contributions to the final committee decision. It is necessary to note that (9) is only one of the possible settings. Obviously it is an open issue, which requires further investigation to find the optimal setting. Nevertheless for real world applications, there is no need for optimal setting and sufficient setting is enough.

In summary, the two phase multi-agent negotiation mechanism is proposed to enable efficient collaborative in-network processing for target classification in WMSNs. The target classification task is efficiently allocated to guarantee classification accuracy and meanwhile reduce consumption of scarce resources. To enhance the reliability and accuracy of classification, weighted committee decision is introduced to combine individual decisions. In this section, it is assumed that the classifier is already known. In the section that follows, the classifier and related processing approaches will be discussed.

4. Statistical Dimension Reduction and Classification

In WMSNs, the volume of raw multimedia data is generally prohibitively large, and consequently it is imperative to employ data compression or dimension reduction techniques to reduce memory and computation requirements. Taking the statistical nature of retrieved audio information into account, a set of statistical approaches is proposed for dimension reduction and classification in this section. Note that in the context of this paper, data compression and dimension reduction take the same meaning and are used interchangeably.

4.1 Feature extraction

Feature extraction is one of the simplest but effective data compression approaches. It helps to reduce the data volume by extracting the most useful information from raw data. In [5] compression of acquired acoustic signals by means of spectral analysis is proposed. In this paper, we propose to extract features using power spectral density (PSD) estimate, which is briefly explained in the following.

Suppose the observed acoustic signal $x[n]$ is an N_s point series. It is required that N_s takes the form of $N_s = 2^\beta$ where β is an integer. In cases where the acquired signals fail to meet this requirement, simply pad them with zeros to satisfy it. The power spectrum of $x[n]$ is denoted by $S_{xx}[k]$ which is given by:

$$S_{xx}[k] = \frac{1}{N_s} |X(k)|^2 \quad (10)$$

where $X(k)$ is the Fourier transformation of $x[n]$, determined by:

$$X(k) = \sum_{n=0}^{N_s-1} x(n) e^{-j\omega_k n}, \quad \omega_k = \frac{2\pi}{N_s} k \quad (11)$$

The power spectrum $S_{xx}[k]$ is divided into $M=2^\gamma$ adjoining segments (without overlapping) where γ is less than β . In other words, the derived p -th segment SG_p ($1 \leq p \leq M$) is a set containing the following elements:

$$SG_p = \{S_{xx}[N_s(p-1)/M + 1], \dots, S_{xx}[N_s(p-1)/M + N_s/M]\} \quad (12)$$

Then, the average \overline{SG}_p of SG_p is calculated:

$$\overline{SG}_p = \frac{1}{M} \sum_{S_{xx}(r) \in SG_p} S_{xx}(r) \quad (13)$$

Combine the M average power, and we get a feature vector:

$$\hat{F} = [\overline{SG}_1, \overline{SG}_2, \dots, \overline{SG}_M] \quad (14)$$

Usually \hat{F} is normalized to make it possible to compare the features extracted from various samples:

$$F = \hat{F} / \|\hat{F}\|_2 \quad (15)$$

where $\|\hat{F}\|_2$ is the vector-2 norm of \hat{F} determined by

$$\|\hat{F}\|_2 = \sqrt{\sum_{t=1}^M |\overline{SG}_t|^2} \quad (16)$$

In this way, the maximum element in the feature vector F is normalized to one. When the features are extracted from raw data samples, a Gaussian process classifier can be learned from them. However not all the information contained in the feature vector will contribute much to the final classification decision, therefore it is desirable to discard these less useful elements. This can be done by the dimension reduction technique of principal component analysis covered in the following section.

4.2. PCA dimension reduction

As is well known, problems will arise when performing classification in a high-dimensional space, which is often referred to as the “curse of dimensionality”. The problems are even more serious for WMSN applications where high dimensional feature vectors require more memory and computation expenditure. So it is desirable to reduce the dimension while preserve the useful information of the extracted features.

Principal component analysis (PCA) provides us with an ideal solution. It reduces dimensions and meanwhile preserves as much information as possible. In statistics, the goal of PCA is to reduce the dimensionality of the data while retaining as much as possible of the variation present in the original dataset.

Simply put, PCA just transforms $\mathbf{X}=[a_1, a_2, \dots, a_N]^T$ in N dimensional space into $\mathbf{Y}=[b_1, b_2, \dots, b_K]^T$ in K dimensional space where $K \ll N$. Suppose the basis for N dimensional space is $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N\}$ and that for K dimensional space is $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K\}$. Suppose there are M samples $\{\mathbf{X}_i\}_{i=1}^M$, and their principle components can be calculated following the procedures [43, 46, 63] as listed in Figure 2.

There arises a problem related to the selection of K . Denote the variance of \mathbf{X}_j in $\{\mathbf{X}_i\}_{i=1}^M$ by σ_j^2 . It can be shown (see [63]) that the total variance of the N components of the given samples $\{\mathbf{X}_i\}_{i=1}^M$ is governed by

$$\sum_n \sigma_j^2 = \sum_n \lambda_j \quad (17)$$

It has been shown larger eigenvalues corresponds to components with larger variance; therefore we can keep the principal components that account for the total variance to a desirable extent. Suppose we desire the variance of the principal components contribute no less than α of the total. Then K is determined by:

$$K = \min \{m\} \quad (18)$$

where m satisfies the following formula [63]

$$\frac{\sum_n^m \lambda_j}{\sum_n \lambda_j} \geq \alpha \quad (19)$$

In this way K , the number of principal components, is determined.

Figure 2. Algorithm to compute the principal components of given samples.

The principal component analysis algorithm to reduce the dimension of the M N -dimensional samples $\{\mathbf{X}_i\}_{i=1}^M$ from N to K where K is much less than N .

Step 1: Find the mean $\bar{\mathbf{X}}$ of $\{\mathbf{X}_i\}_{i=1}^M$:

$$\bar{\mathbf{X}} = \frac{1}{M} \sum_{i=1}^M \mathbf{X}_i \quad (20)$$

Step 2: Transform $\{\mathbf{X}_i\}_{i=1}^M$ to zero-mean samples $\{\mathbf{Z}_i\}_{i=1}^M$

$$\mathbf{Z}_i = \mathbf{X}_i - \bar{\mathbf{X}} \quad (21)$$

Step 3: Combine \mathbf{Z}_i to form a matrix $\mathbf{A} = [\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_M]$, then compute the sample covariance matrix \mathbf{C}

$$\mathbf{C} = \frac{1}{M} \sum_{i=1}^M \mathbf{Z}_i \mathbf{Z}_i^T = \mathbf{A} \mathbf{A}^T \quad (22)$$

Step 4: Compute eigenvalue λ_i and eigenvector \mathbf{u}_i of \mathbf{C} :

$$\mathbf{C} \mathbf{u}_i = \lambda_i \mathbf{u}_i \quad (23)$$

where $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N$

Step 5: Express the sample \mathbf{X}_i in the basis of $\{\mathbf{u}_i\}_{i=1}^N$

$$\mathbf{X}_i - \bar{\mathbf{X}} = \sum_{i=1}^N b_i \mathbf{u}_i \quad (24)$$

Step 6: Approximate \mathbf{X}_i with $\hat{\mathbf{X}}_i$ by retaining the components corresponding to the K largest eigenvalues:

$$\hat{\mathbf{X}}_i - \bar{\mathbf{X}} = \sum_{i=1}^K b_i \mathbf{u}_i \quad (K \ll N) \quad (25)$$

The resulted

$$\mathbf{Y} = [b_1, b_2, \dots, b_K]^T \quad (26)$$

is the so called the principal components of \mathbf{X}_i .

4.3. Gaussian process classification

A Gaussian process is a generalization of the Gaussian probability distribution [30, 32]. A probability distribution describes random variables which are scalars or vectors, but a stochastic process governs the properties of functions. Gaussian process classification (GPC) is a promising statistical classifier for both binary and multi-category classification. In this paper we focus on binary classification, that is, it is assumed the target intruding the deployed WMSNs belongs to one of the two known types.

A binary classification problem can be simply formulated as follows [30, 32]: Given the set of m observed data $D = \{(\mathbf{x}_i, y_i) | i = 1, 2, \dots, m\}$, where $\mathbf{x}_i \in R^d$ is the input and in the meanwhile $y_i \in \{+1, -1\}$ denotes its class label, determine the probability distribution $p(y|\mathbf{x})$ from the given data so that it can be used to predict the label of new inputs.

Since it invariantly holds $p(+1|\mathbf{x}) + p(-1|\mathbf{x}) \equiv 1$, we focus on the probability distribution of $p(y = +1|\mathbf{x})$ only. For GPC, this probability is related to a latent function $f(\mathbf{x})$ which is mapped to the interval of $[0, 1]$ by a sigmoid transformation σ . For example the logistic function [30]

$$\lambda(z) = \frac{1}{1 + \exp(-z)} \quad (27)$$

can achieve such sigmoid transformation, which squashes its argument in the domain of $[-\infty, +\infty]$ into the range of $[0, 1]$.

By the sigmoid transformation σ , the model $p(y=+1|\mathbf{x})$ is replaced by $p(y=+1|\mathbf{x}) = \sigma(f(\mathbf{x}))$. Therefore in GPC, the Bayesian inference is essentially closely related to the latent function $f(\mathbf{x})$. Let $f_i = f(\mathbf{x}_i)$ and $\mathbf{f} = [f_1, \dots, f_m]^T$ be shorthand for the values of the latent function. Moreover, $\mathbf{y} = [y_1, \dots, y_m]^T$ and $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m]^T$ denote the class labels and corresponding inputs, respectively. Given the latent function, the class labels are independent; therefore the joint likelihood $p(\mathbf{y}|\mathbf{f})$ can be factorized as [32]

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^m p(y_i|f_i) \quad (28)$$

Here we introduce the relationship $p(y_i|f_i) = \sigma(y_i f_i)$ without proof, and detailed derivation can be found in [30]. By Gaussian process, the joint distribution of latent function values corresponding to any set of inputs \mathbf{X} is a multivariate Gaussian distribution $p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{0}, \mathbf{K})$ with the mean of $\mathbf{0}$. The covariance matrix \mathbf{K} is parameterized by $\boldsymbol{\theta}$ which is generally referred to as *hyper-parameter*. In other words, the covariance matrix is defined by its elements $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j, \boldsymbol{\theta})$, where k is a covariance function. Following Bayes' rule, the posterior distribution of the latent function for given $\boldsymbol{\theta}$ and D can be expressed as [32]

$$p(\mathbf{f}|D, \boldsymbol{\theta}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta})}{p(D|\boldsymbol{\theta})} = \frac{\mathcal{N}(\mathbf{0}, \mathbf{K})}{p(D|\boldsymbol{\theta})} \prod_{i=1}^m \sigma(y_i f_i) \quad (29)$$

Note that $p(\mathbf{f}|D, \boldsymbol{\theta})$ is not Gaussian. The objective of GPC is to predict the class label y_* of a new input \mathbf{x}_* , which can be achieved by computing [32]

$$p(y_*|D, \boldsymbol{\theta}, \mathbf{x}_*) = \int p(y_*|f_*)p(f_*|D, \boldsymbol{\theta}, \mathbf{x}_*)df_* \quad (30)$$

where $p(f_*|D, \boldsymbol{\theta}, \mathbf{x}_*)$ is determined through [32]

$$p(f_*|D, \boldsymbol{\theta}, \mathbf{x}_*) = \int p(f_*|\mathbf{f}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{x}_*)p(\mathbf{f}|D, \boldsymbol{\theta})d\mathbf{f} \quad (31)$$

However, none of the distributions, i.e. posterior $p(\mathbf{f}|D, \boldsymbol{\theta})$, predicted posterior distribution $p(y_*|D, \boldsymbol{\theta}, \mathbf{x}_*)$ or the marginal likelihood $p(D|\boldsymbol{\theta})$ can be analytically calculated [30, 32]. Therefore Gaussian approximations are employed instead to calculate these distributions. By means of Gaussian approximations, the posterior $p(f_*|D, \boldsymbol{\theta}, \mathbf{x}_*)$ is approximated by the following Gaussian distribution [32]

$$q(f_*|D, \boldsymbol{\theta}, \mathbf{x}_*) = \mathcal{N}(f_*|\mu_*, \Sigma_*^2) \quad (32)$$

The mean μ_* of the Gaussian distribution is $\mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{m}$ while the variance Σ_*^2 is $k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T (\mathbf{K}^{-1} - \mathbf{K}^{-1} \mathbf{A} \mathbf{K}^{-1}) \mathbf{k}_*$, where $\mathbf{k}_* = [k(\mathbf{x}_1, \mathbf{x}_*), \dots, k(\mathbf{x}_m, \mathbf{x}_*)]^T$ is a vector of prior covariance between \mathbf{x}_* and the training input \mathbf{X} [32]. Determination of the parameters \mathbf{m} and \mathbf{A} is to be expounded later. With the Gaussian approximation, the probability that a new input \mathbf{x}_* belongs to class +1 can be analytically computed as [32]

$$q(y_* = +1 | D, \boldsymbol{\theta}, \mathbf{x}_*) = \int \sigma(f_*) \mathcal{N}(f_* | \mu_*, \Sigma_*^2) df_* \quad (33)$$

To simplify the expression, Let $\ln \mathcal{L}(\mathbf{f}) = \ln p(\mathbf{y} | \mathbf{f})$ and

$$\ln Q(\mathbf{f} | D, \boldsymbol{\theta}) = \ln \mathcal{L}(\mathbf{f}) - \frac{1}{2} \ln |\mathbf{K}| - \frac{1}{2} \mathbf{f}^T \mathbf{K}^{-1} \mathbf{f} - \frac{m}{2} \ln(2\pi) \quad (34)$$

Under these settings, the parameters \mathbf{m} and \mathbf{A} can be found by the Laplace's method [30, 32]

$$\mathbf{m} = \arg \max_{\mathbf{f} \in \mathbb{R}} \ln Q(\mathbf{f} | D, \boldsymbol{\theta}) \quad (35)$$

$$\mathbf{A} = (\mathbf{K}^{-1} - \nabla \nabla_{\mathbf{f}} \ln \mathcal{L})^{-1} \quad (36)$$

Note that in GPC, the only parameter undetermined is the hyper-parameter $\boldsymbol{\theta}$ used by covariance function. Take the frequently used isotropic squared exponential (isoSE) covariance function [30] for example. The function takes the form of

$$k(\mathbf{x}_p, \mathbf{x}_q) = \sigma_f^2 \exp\left(-\frac{1}{2} (\mathbf{x}_p - \mathbf{x}_q)^T \mathbf{M}^{-1} (\mathbf{x}_p - \mathbf{x}_q)\right) \quad (37)$$

where $\mathbf{M} = \ell^{-2} \mathbf{I}$ (\mathbf{I} is the unit matrix). It is evident the hyper parameter is $\boldsymbol{\theta} = [\sigma_f, \ell]$ for the isoSE covariance function.

Obviously the classification accuracy is closely related to the value of $\boldsymbol{\theta}$. It is shown in [30] and [32] that the optimal hyper parameter is the $\boldsymbol{\theta}$ maximizing the marginal likelihood $p(D | \boldsymbol{\theta})$. Using Laplace's method, the marginal likelihood can be approximated as [32]

$$\ln q(D | \boldsymbol{\theta}) = \ln Q(\mathbf{m}) + \frac{m}{2} \ln(2\pi) + \frac{1}{2} \ln |\mathbf{A}| \quad (38)$$

Conjugate gradient methods can be employed [30, 32] to optimize (38) which will determine the optimal hyper parameter $\boldsymbol{\theta}$.

All of these introduced statistical approaches have been well developed and have proved effective in relevant applications. Nevertheless their effectiveness and efficiency for target tracking in WMSNs still need experimental evaluation.

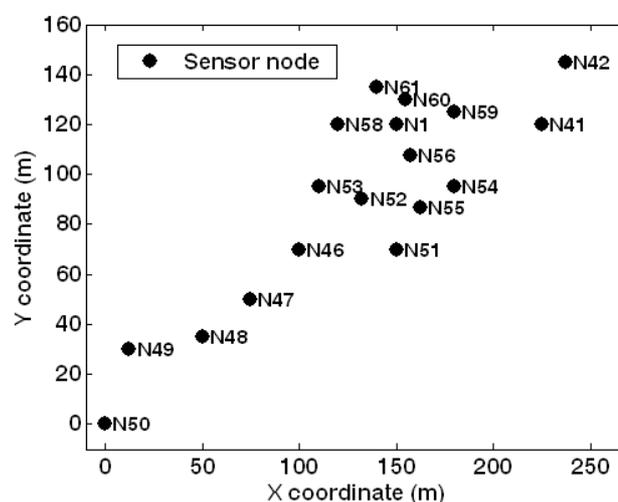
5. Experiments

5.1. Experimental setup

The proposed statistical processing approaches and agent negotiation mechanisms for target classification in WMSNs are evaluated by simulation with the real world data reported in [64]. In [64] both acoustic and seismic signals are collected by the deployed wireless sensor nodes, but in our WMSN simulation only the acoustic signals are used. These real world acoustic data are collected by microphone sensors at the sampling frequency of 4.96 kHz. The objective in [64] is to evaluate the performance of WSNs for vehicle classification in real world deployment. Four classes of vehicles are investigated in [64], but only the data collected from Assault Amphibian Vehicle (AAV) and Dragon Wagon (DW) are publicly accessible. Therefore in our simulation, the objective is to apply the proposed approaches to classify two classes of vehicles (i.e. AAV and DW) in WMSNs. Data collected in several runs are available, but in this paper we use the data of run 3 exclusively. In run 3, 18 sensor nodes are used, whose deployment is illustrated in Figure 3. In the figure, N_x denotes the x -th sensor nodes deployed in the experiment, which is identical to the naming convention used in [64]. The deployed sensor field covers an area of approximately 250x150 meters and the separation of adjacent sensors ranges from 20 to 40 meters.

Following the proposed hierarchical multi-agent architecture, all the sensor nodes in Figure 3 form a cluster, of which N52 is nominated as the cluster agent according to their geographic distribution. Though the number of deployed sensor nodes is relatively small, yet it can be viewed as a miniature of the proposed multi-agent architecture, because most of the required processing is actually performed within the cluster. The proposed multi-agent architecture is implemented by means of the BDI agent engine of JADEX. Using the JADEX engine, each sensor node is represented by a BDI agent. These agents may be deployed on a single or several computers. In our simulation, all the agents in the cluster are deployed on a single computer. Agent perception or equivalently data acquisition is simulated by feeding the real world data to the agents. Relevant statistical processing and agent negotiation mechanisms are implemented by programming using the JADEX engine.

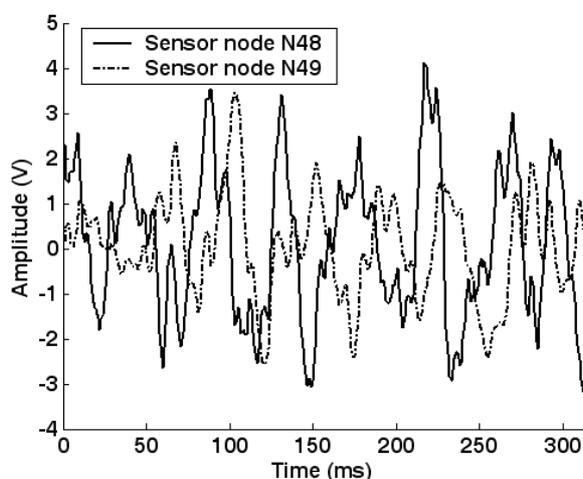
Figure 3. Sensor node deployment for target classification in the simulation.



5.2. Feature extraction and dimension reduction

In real world applications, the acoustic signals emitted from the vehicles are inevitably affected by noises from various sources. Moreover the acoustic signals will attenuate as they propagate in the air, which means that the signals detected by sensors located at different sites generally show distinct amplitudes. These phenomena are clearly illustrated in Figure 4. The corresponding scenario is that an AAV vehicle moves in the sensor fields, where N48 and N49 both record the signals they detect. As the figure illustrates, the signals measured by the two sensors are similar on the whole, but the amplitudes are different. It can be noted that the signal of N49 lags behind that of N48 by approximately twenty milliseconds. Such lag corresponds to phase shift, which can be exploited to locate the target, but this issue is not our concern in this paper. If examined in detail, it is evident that the signals differ much at any corresponding intervals, even if taking the phase shift into consideration. This is primarily due to contamination by noises, but also partially attributed to the variety of microphone sensor quality. Obviously the noises occur randomly, and therefore statistical approaches should be exploited to counteract such undesired interference. Moreover it also justifies the in-network collaborative processing required by WMSNs, which can significantly enhance the reliability by reducing statistical uncertainty.

Figure 4. Acoustic signals (of the same AAV) observed by different sensors.



Despite of all these uncertainties resulting from noise contamination and sensor quality diversity, the acoustic signals are suitable and sufficient to discriminate one kind of vehicle from another. Figure 5 illustrates the AAV and DW acoustic signals recorded by N49 in the same run. The illustrated signals contain 512 points, which last for 103.2 ms (recall the sampling frequency is 4096Hz). Evidently, the pattern of the AAV acoustic signal is different from DW. Though such contrast can be easily noticed, yet it is not so easy to describe mathematically. In other words discrimination of signals in the time domain is intractable, and this is partially why feature extraction is needed. In addition, for WMSN applications, it also helps to reduce memory requirement. As proposed, power spectral density (PSD) estimate is used to extract features from the time series signals. PSD estimate of the signals shown in Figure 5 is reported in Figure 6. Since FFT is performed on a sample of 512 points, therefore the frequency resolution of PSD estimate is 9.6875 Hz. As Figure 6 illustrates, the resulting PSD,

which is represented in logarithmic scale, is normalized so that the largest component is unitary. The normalization is intended to make it more convenient to compare between them. From Figure 6, it can be seen that for most frequencies, the PSD of AAV is higher than that of DW.

To accommodate WMSN requirements related to memory, the whole PSD estimate is divided into 16 segments (i.e. setting $M=16$), following (12) as proposed in the section discussing feature extraction. Choosing to divide it into 16 segments is not an arbitrary choice. In fact, such a choice is based upon the tradeoff between information loss and memory requirement. Smaller M means less memory and computation requirements, but more information will be lost, and vice versa. The 16-segmented features extracted from the PSD estimates shown in Figure 6 are illustrated in Figure 7. The frequency band indices represented by x-axis exactly correspond to the 16 segments. *Note* that to more explicitly demonstrate their difference, the logarithms of these PSD estimates are used instead of the original PSD estimates. Obviously it is easier to discriminate AAV from DW by the extracted features, because the PSD distribution in the 16 frequency bandwidths demonstrates distinct patterns for AAV and DW.

Figure 5. Signals of AAV and DW observed by the sensor node N49.

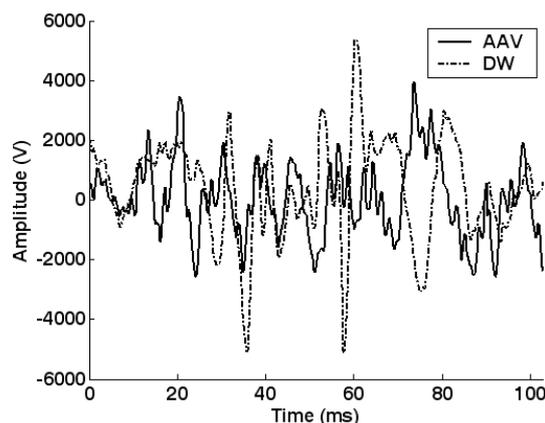
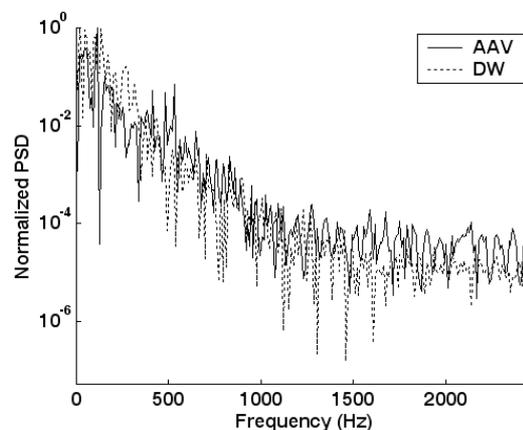


Figure 6. Normalized power spectral density of the acoustic signals illustrated in Figure 5.



The data collected by N49 is divided into smaller samples of 512 points from which features are accordingly extracted. Then PCA is performed to further reduce feature dimensions. The key issue

then is to specify α (the variance contribution) required by (19) to determine K (the number of principal components). Recall that the parameter α specifies the least that the principal components contribute to the total variance. The graph in Figure 8 shows K (the number of principal components) as a function of α (contribution to the total variance). The analysis is performed on the data collected by several sensor nodes (i.e. N60, N54, N49 and N48), and consequently the result can be generalized to other observed data. As expected, when α is set to 1, all the 16 components of the features are principal components. As variance contribution α declines, the principal component number K first decreases drastically but remains steady when α is less than about 0.9 for all the sensors nodes.

Similar to the selection of M (the number of PSD division), determining the parameter α necessitates compromise between accuracy and memory requirement. For N49, when α is 0.98, K is already reduced to 5, which makes 68.75% compression of the original features. Obviously setting α as 0.98 can concurrently offer satisfactory dimension reduction and retain as much information as possible. Under this setting, the 16 component features shown in Figure 7 is reduced to 5 components by PCA, which is illustrated in Figure 9. Note that the 16 elements of the original features are all positive, but when reduced to 5 elements by PCA, both positive and negative elements exist. It can be explained by observing (21) where the data are so transformed that the mean is zero; therefore after PCA reduction, both positive and negative components are possible.

Figure 7. Features extracted from the PSD shown in Figure 6 for AAV and DW.

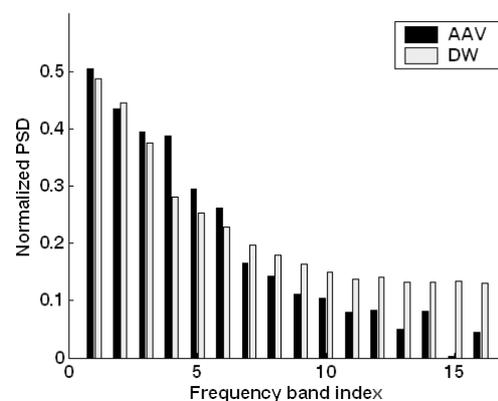


Figure 8. Relationship between the principal component number K and the specified contribution to total variance α .

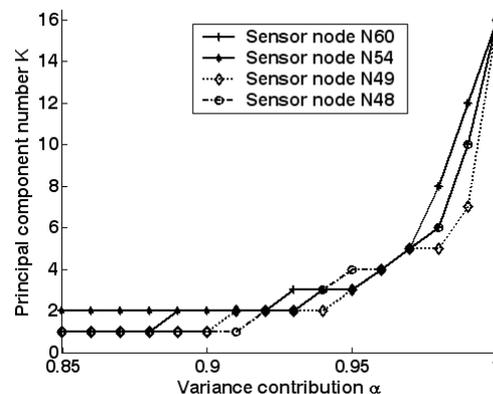
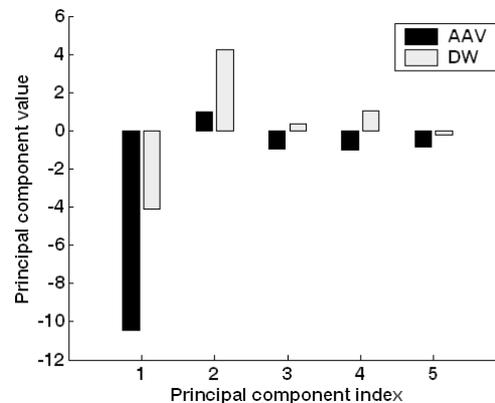


Figure 9. Principal components of the features illustrated in Figure 7. These principal components account for no less than 98% of the total variance.



The above discussion is primarily focused on the signals collected by the sensor node N49, but the approaches are applicable to other sensor nodes too. In other words, feature extraction and dimension reduction are implemented as follows. PSD estimate uses 512 points and the resolution is 9.6875Hz. The whole PSD spectrum is divided into 16 segments to obtain the features whose dimensions are further reduced by PCA. In its implementation, the number of principal components is so selected that they will contribute no less than 98% to the total variance.

By PSD feature extraction and PCA dimension reduction, the volume of raw data is extremely compressed, which not only satisfies the memory requirement in WMSNs, but also reduces the computation load of GPC. Training and testing of GPC with these reduced features will be detailed in the following section.

5.3. Training and testing of GPC

As stated in Section 4, GPC is essentially a supervised learning method, which means that the GPC classifiers should be derived by learning from given samples. The samples to be used are exactly those features extracted in the approaches illustrated in the above section. In our simulations, the samples are prepared in the following ways. The whole data collected by a sensor node is split into such segments as contain 512 points. These segments are consecutive but don't overlap. A third of these segments are chosen as training samples and the remaining are used for testing. In this section, the data collected by N49 is used to exemplify GPC training and testing. Following the segment splitting schemes, there are 294 samples for AAV, with 98 samples for training and 196 samples for testing. For DW, there are 178 samples in all, of which 60 are used for training and the remaining 118 samples for testing. As discussed in the above section, each of these samples contains 5 elements after PCA is performed.

When implementing GPC, some important decisions have to be made, including covariance function specification, hyper-parameter determination and approximation method selection. In our GPC implementation, we have chosen the isotropic squared exponential covariance function(37), because there are only two parameters (i.e. l and σ_f) to specify. There are several approximation approaches to GPC, and in this paper, we have chosen Laplace's approximation, because it is very straightforward and easy to implement. Usually the hyper-parameter $\theta=[\sigma_f, l]$ is arbitrarily initialized

and then the optimal hyper-parameter is obtained by optimizing (38). Since the hyper-parameter generally takes the form of $\ln\sigma_f$ and $\ln l$ in detailed implementation, therefore in the following we refer to $\ln\sigma_f$ and $\ln l$ as the parameters.

GPC is first applied to classify AAV and DW with reduced features with 5 components. The parameters are initialized as $\ln\sigma_f=0$ and $\ln l=0$. Using these initial parameters, GPC classifiers are trained and tested, and the results are illustrated in Figure 10 and Figure 11. Figure 10 shows the classification result using the initial parameter and Figure 11 illustrates the result obtained with optimal parameters (i.e. $\ln\sigma_f=1.3708$ and $\ln l=1.5584$). The x-axis represents the indices of the testing samples, and the samples corresponding to AAV and DW are intentionally grouped for better visualization. The vertical dotted line separates the two groups, the left and right of which corresponds to the AAV and DW samples, respectively. As said above, there are 196 and 118 samples for AAV and DW, and consequently the separating line is not in the middle. In the implementation, AAV is labeled as +1 and DW is labeled as -1. The y-axis indicates the probability that a sample belongs to the AAV class. Therefore if the predicted probability is greater than 0.5 (indicated by the solid horizontal line which is called the watershed line), then the corresponding sample is believed to belong to the AAV, otherwise it falls into the category of DW. These dots in the figure indicate the predicted probability that the corresponding samples belong to the AAV class.

Figure 10. GPC with features reduced by PCA for the sensor node N49; the parameters are arbitrarily specified.

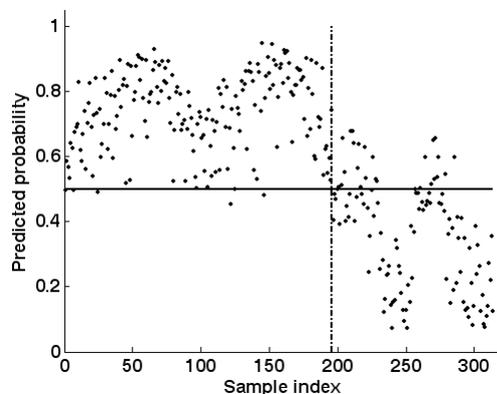
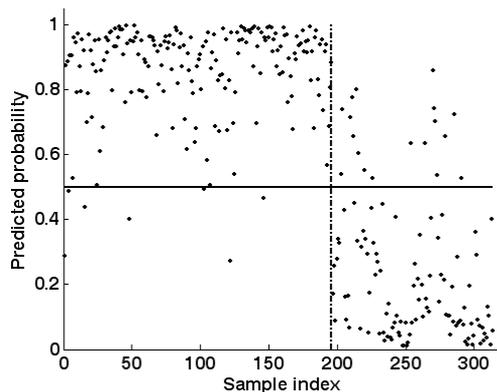


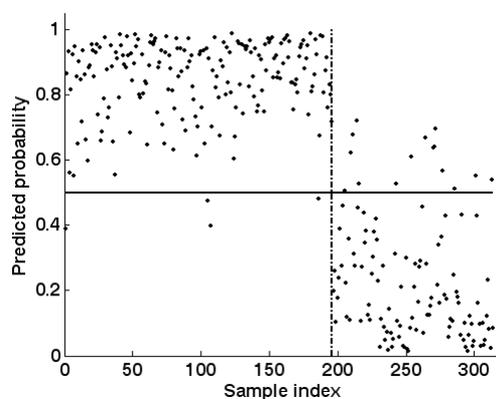
Figure 11. GPC with features reduced by PCA for the sensor node N49; the parameters are optimized.



If GPC can classify these samples perfectly, or in other words, if all the testing samples are correctly classified, then all the dots denoting predicted probability should be located in either the left upper or the right lower quadrants defined by the vertical dotted line and the horizontal watershed line. However practically some samples will be misclassified, and subsequently several dots will be located in the quadrants other than the above ones. Overall classification accuracy can thus be determined by calculating the ratio of the dots located in the left upper and right lower quadrants to all dots in the figure. If classification accuracy of AAV or DW is to be individually calculated, only dots located on the left or the right side of the vertical dotted line need to be considered respectively. For example, the classification accuracy of AAV is determined by calculating the ratio of the dots in the left upper quadrant to all the dots on the left of the vertical dotted line.

Examining the two figures, some important conclusions relevant to GPC accuracy can be drawn. As illustrated in Figure 10, the overall classification accuracy is 90.13%, while individual accuracy is 96.94% and 78.81% for AAV and DW respectively. In contrast, in Figure 11 the overall classification accuracy is 92.04%, while individual accuracy is 96.42% and 84.74% for AAV and DW respectively. Recall that the former figure represents GPC results with arbitrarily selected parameters, while the latter corresponds to the classification using optimized parameters. From the perspective of overall classification accuracy, GPC with optimized parameters achieves the accuracy that is approximately 2% higher than that with arbitrary parameters. It means that optimal parameters will guarantee higher classification accuracy, which is as expected by intuition. As to individual classification accuracy, the above conclusion will not hold. It can be easily verified that for GPC using optimized parameters, though DW accuracy is much higher than GPC with arbitrary parameters, however AAV accuracy is slightly less than the latter. This phenomenon is not difficult to understand, since the objective of parameter optimization is to maximize the overall classification accuracy rather than individual accuracy.

Figure 12. GPC using features without PCA for the sensor node N49; the parameters are optimized.



The advantage of GPC with parameter optimization is much more than improvement in overall classification accuracy. The most appealing benefit lies in the enhancement of reliability of the predicted results. This can be confirmed by carefully examining and comparing between Figure 10 and Figure 11. In Figure 11 where the parameters are optimized, these dots are much closer to the desired

probability of 1 for AAV and 0 for DW than Figure 10 using arbitrary parameters. In Figure 10, there are a considerable number of dots lying near the watershed line (i.e. the solid horizontal line), which means that the prediction is not reliable.

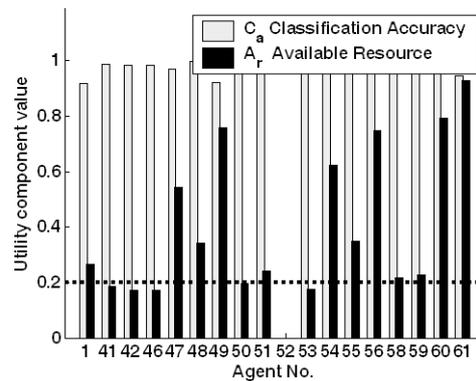
Experiments are also conducted to evaluate the influence of dimension reduction on classification accuracy. Figure 12 illustrates the results obtained from GPC with parameter optimization (initial parameters are $\ln\sigma_f=0$ and $\ln l=0$; the optimized parameters are $\ln\sigma_f=-1.9945$ and $\ln l=1.6277$), where the 16-component features are used (i.e. without PCA dimension reduction). In such case, the overall classification accuracy is 93.95%, while individual accuracy is 97.96% and 87.29% for AAV and DW respectively. As expected, GPC with 16-component features gives higher accuracy (for both overall and individual classification accuracy), than GPC using the features reduced by PCA. The results are reasonable, because the 16-component features are more informative than those features reduced by PCA where some information is inevitably lost. However for WMSN applications, accuracy decrease is well compensated by reduced memory and computation requirements. The significance is that vehicle classification by GPC and PCA is able to achieve good enough classification with minimum resource requirements, meeting the challenges presented by WMSNs.

In summary, GPC with parameter optimization is both effective and reliable for the purpose of vehicle classification in WMSNs. Meanwhile, PCA dimension reduction significantly brings down memory and computation requirements, meeting the constraints of WMSNs.

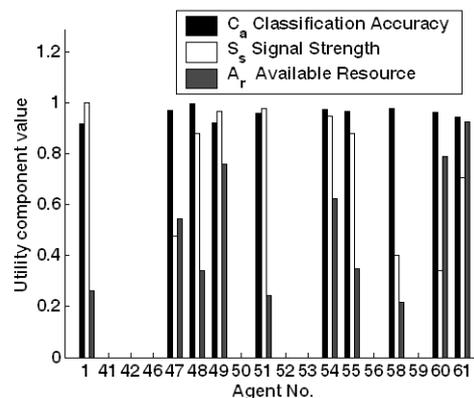
5.4. OSDMIA mechanism for target classification

As stated in Section 3, when a target is detected in the field where the WMSNs are deployed, three agents will be selected by the OSDMIA mechanism to perform efficient collaborative in-network classification to determine whether the target is AAV or DW. Before their deployments, each of the agents learns a Gaussian process classifier from given samples (i.e. the data from [64]) following the same approaches described above. Using the BDI model, the learned classifier is essentially the belief that an agent holds about the world, and it has the desire to classify a detected target. However the desire can turn into a goal only if it wins the OSDMIA auction. The auction process to determine whether an agent will win the auction and fulfill its desires will be detailed as follows.

The agent N52, as stated in the experimental setup, is designated the cluster agent; therefore it serves as the auctioneer. When the auctioneer detects or is informed of an intruding target, the OSDMIA auction is triggered. It sends a message to all the agents in this cluster to ask for bidding. This message also specifies the time window for the auction, which is assumed to be 300ms in our simulation. The FIPA-ACL equivalent to this action is *cfp* (i.e. call for proposal). On receiving the message, an agent first checks its energy level before elaborating on whether to bid or not. If in critical shortage of energy, it will not participate in the auction. In our experiments, the energy level is simulated by random assignments, namely random numbers are generated and assigned to the agents to represent their energy levels. The larger the number is, the more energy is available, where the largest number is 1 and the least is zero. In our simulation, it is assumed that if the energy level is below 0.2, the agent will refuse to be engaged in the auction in order to preserve energy.

Figure 13. Classification accuracy and energy levels of all the agents before the bidding.

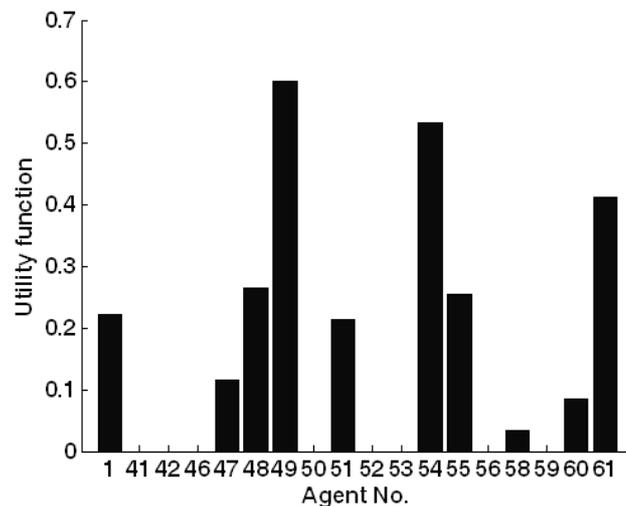
Actually both energy level and the learned classifiers are the beliefs of the agents. Figure 13 illustrates the beliefs these agents hold about the world before bidding. The data for N52 is missing, because it is the auctioneer and thus won't take part in the bidding. Obviously the classification accuracy doesn't vary much for different agents, but the energy level differs a lot. It must be clarified that available resource in the figure refers to energy level exclusively and other resources are *not* considered in the simulation. Note that N41, N42, N46 and N53 are in critical energy level (the dotted horizontal line corresponds to 0.2); therefore they will deny to bid in the auction.

Figure 14. Bids made by the agents in the auction. Some refuse to bid and some fail to bid within the given time window.

As a consequence, all the agents except the four in critical energy status are involved in the auction. Each of them will sample the signals emitted by the target vehicle for 100ms (i.e. 496 points), from which the energy or the signal strength is calculated by(4). In our simulation, the energy is calculated from corresponding segments (i.e. the same time interval) of the data collected by the sensor nodes as reported in [64]. Then they will bid to the auctioneer by providing the parameters of classification accuracy C_a , available resource A_r and signal strength S_s . When the auctioneer receives these bids, S_s will be normalized following(6). The bidding action corresponds to the *propose* message in the FIPA-ACL. The bids made by the agents are shown in Figure 14. Note that the bids corresponding to N50 and 59 are all zeros, which is the same as N41, N42, N46 and N53 which refuse to bid due to critical

energy levels. This is explained by the fact that the two agents fail to bid within the given time window of 300ms.

Figure 15. Utility functions of all the bidding agents.



Immediately after receiving these bids, the auctioneer starts to calculate their utility functions by (7) and chooses the three agents that have the largest utility functions. The utility functions are shown in Figure 15. Evidently N49, N54 and N61 give the largest utility functions, and therefore they are the winners of the auction and are assigned the resources to perform classification of the intruding target. The winners are notified of their winning by the auctioneer. This action is implemented again by a *propose* message in FIPA-ACL, which is intended to instruct the agents to engage in the collaborative classification of the target vehicle. The winning agents respond to the auctioneer by the *accept-proposal* message in FIPA-ACL to confirm that they are committed to the assignments. Then they perform feature extraction and reduction on the perceived acoustic signals and subsequently perform classification using their learned Gaussian process classifiers.

In the experiment, the perceived acoustic signals are all simulated by the data collected by N47. Though in real world, it is impossible for the three sensor nodes to perceive the same signals from a give source, however for the purpose of simulation it is reasonable. The rationale is actually simple. In [64] the signals observed by all the sensor nodes differ from each other, and therefore the classifiers learned by them differ from each other too. Employing different Gaussian process classifiers to classify the same signal doesn't make much difference from classifying several different signals. Using the simulated perception, the bidding agents report their individually predicted probability to the auctioneer by an *inform* message in FIPA-ACL, from which the auctioneer makes a final prediction by the mechanism of committee decision. Such decision making will be discussed in the following section.

5.5. Committee decision mechanism

When the auctioneer receives individual classification decisions from all of the winning agents, it combines these decisions following the committee decision mechanism proposed in Section 3. In this

section, it is assumed that the winning agents are still N49, N54 and N61 and the signals to be classified are simulated by the data collected by N47, following the results and assumption made in the preceding section.

In (9), the weight is proposed as the product of classification accuracy C_a and signal strength S_s . Of course this is an open issue, but from a practitioner's point of view, as long as the weight setting is reasonable and good enough, it is well acceptable and applicable. The optimal setting is not necessarily the unique objective. It will be shown that the setting is good enough later, but before that the details concerning committee decision are examined first.

Table 1. The committee decision made by three committee members.

Committee Member	Member Decision d_i	Weight Component		Member Weight $C_a S_s$	Committee Decision D
		C_a	S_s		
N49	0.3911	0.9204	1.0000	0.9204	
N54	0.6750	0.9746	0.9824	0.9574	0.6293
N61	0.8837	0.9440	0.7310	0.6900	

Table 1 demonstrates an instance of committee decision, where the decisions made by committee member N49, N54 and N61 are combined into a committee decision by N52 (the auctioneer and cluster agent). Note the signal strength S_s is calculated in the same manner as described in the section immediately above. In our simulation, the signal to be classified is extracted from the AAV data collected by N47, and thus we can know that the target is AAV. However as shown in the table, of the three committee member, N49 makes the wrong decision, because a decision (i.e. the predicted probability) less than 0.5 means that the target is DW.

If the proposed negotiation mechanisms are not employed and it happens that N49 is used for classification of the target, then misclassification occurs. But if the negotiation mechanisms are used, the result is totally different. Though N49 misclassifies the target, but the other two members correctly identify it. The committee decision is dominated by N54 and N61 and the final decision is AAV, which cancels out the wrong decision made by N49. Even from this sole instance, the advantage of committee decision may be appreciated.

As said above, the weight is determined by the product of classification accuracy C_a and signal strength S_s . Such choice is essentially arbitrary and a more theoretical approach is to assume the weight takes the form of (39) and find the optimal parameters σ and λ .

$$w = [C_a]^\sigma [S_s / \hat{S}_{s\max}]^\lambda \quad (39)$$

Optimization of parameters σ and λ is not practical in real world applications, but in this paper we will investigate the influence of σ and λ specifications on committee decision performance. It is supposed that σ and λ both range from 0 to 20 and all the data segments from N47 are classified by the three agents and a final decision is correspondingly made by the committee. Figure 16 and Figure 17

show the committee decision (overall classification accuracy) as a function of the parameters σ and λ , for AAV and DW respectively. The step size is 0.5 for both parameters.

Figure 16. Influence of parameters σ and λ on committee decision accuracy for AAV classification.

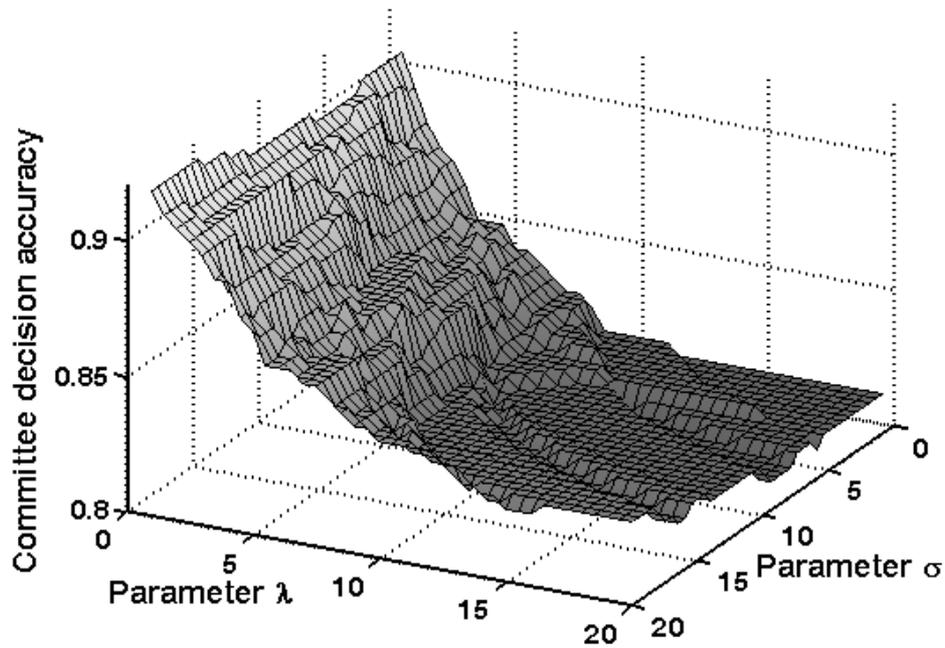


Figure 17. Influence of parameters σ and λ on committee decision accuracy for DW classification.

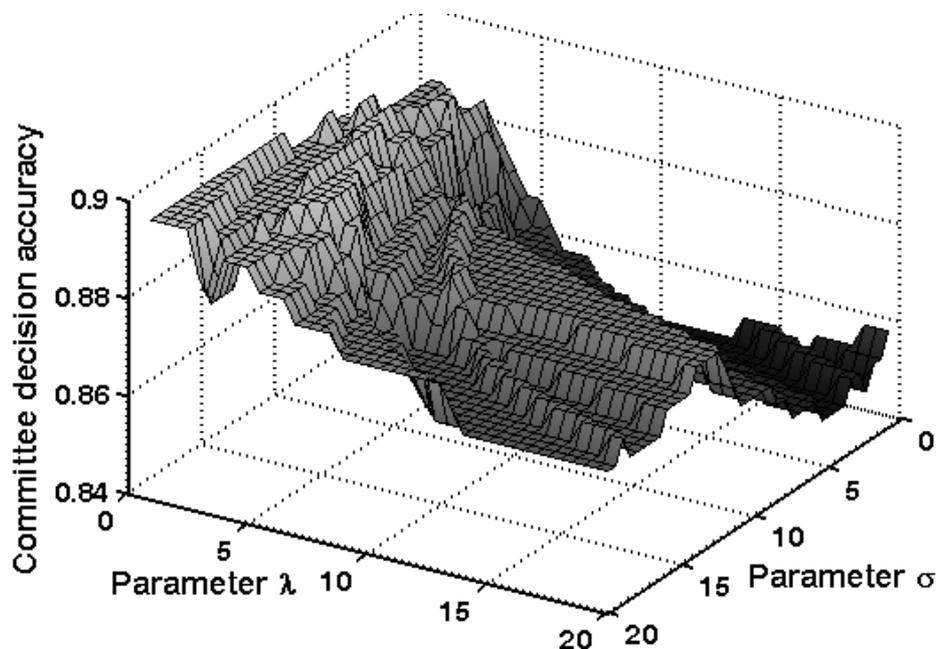
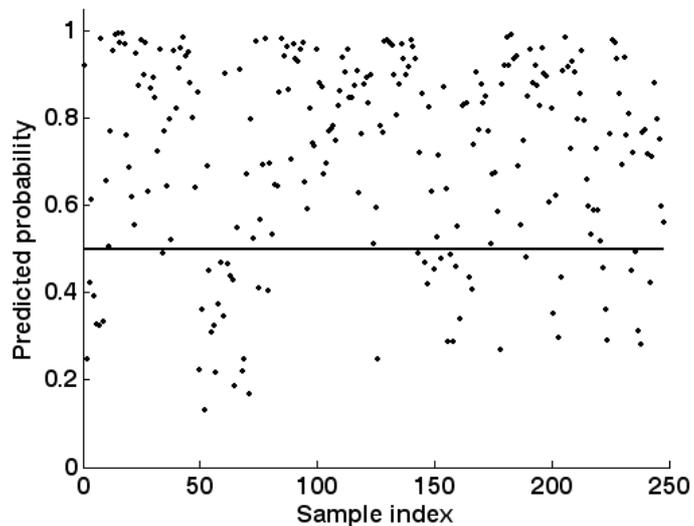
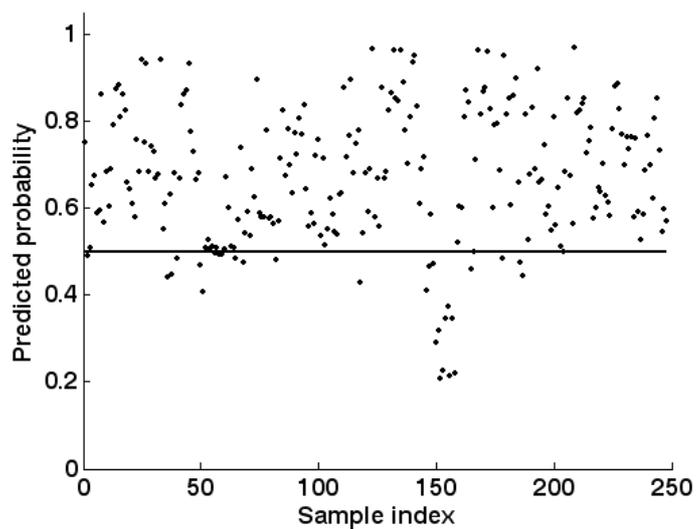


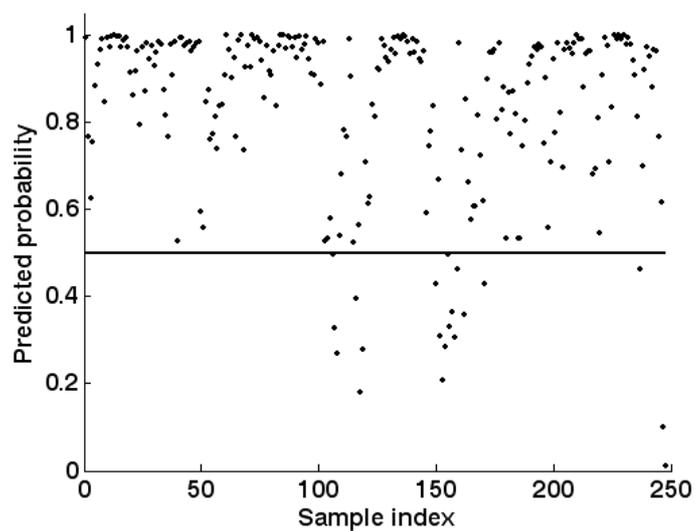
Figure 18. Comparison of individual and committee decisions
(a) Member decision made by N49, with the accuracy of 79.03%



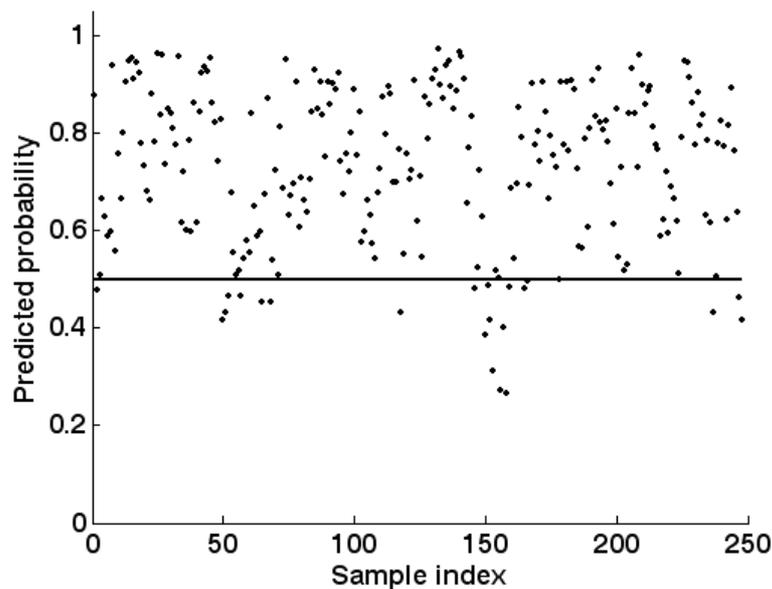
(b) Member decision made by N54, with the accuracy of 87.50%



(c) Member decision made by N61, with the accuracy of 91.93%



(d) Committee decision results, with the accuracy of 90.73%



From these figures, it can be seen that σ has little influence on the committee decision accuracy, while the accuracy is greatly affected by λ . This is quite reasonable. The parameter σ corresponds to classification accuracy which varies not much, and therefore it affects little. However signal strength differs much and is related to noise contamination, therefore the parameter λ significantly influence the final decision accuracy. More importantly, we are concerned with the performance of the parameter $\lambda=\sigma=1$. Evidently such setting gives the best accuracy in neither figure. But in both figures, corresponding accuracy is actually suboptimal but it suffices to be used for real world applications. Therefore our proposed weight calculation by (9) is reasonable and sufficient.

Now we proceed to investigate the performance of committee decision in dealing with the uncertainties relevant to target classification in WMSNs. There are a variety of uncertainties and several of them are considered here. The prediction uncertainty concerning a Gaussian process classifier is primarily because the samples to be classified usually deviate from the samples used to train the classifier. More formally, this is usually referred to as generalization error. On the other hand, the perceived signals may be badly contaminated, or the target might be operating in a total different situation. The uncertainties of individual prediction decisions and the uncertainty reduction achieved by committee decision are illustrated in Figure 18(a) through 18(d). In these figures, all the data segments of AAV collected by N47 are used, from which individual and committee decisions are made by N49, N54 and N61.

Prediction uncertainties can be easily observed in the figures. For instance, the samples with indices around 50 are correctly classified by N61, but misclassified by the other two agents (i.e. N49 and N54). Such uncertainties are significantly reduced by the committee decision mechanism, as is observed in Figure 18(d). Moreover, it can be verified that the target is misclassified by the committee decision only when most of the committee members make wrong classifications. Though the classification accuracy achieved by the committee decision is not necessarily higher than the members (in fact in this experiment, the committee decision accuracy 90.73% is less than the highest member accuracy

91.93%) , yet its decision is much more reliable. For DW classification, the results are similar where uncertainties are significantly reduced by committee decision.

In summary, the proposed feature extraction and dimension reduction can efficiently deal with the statistical natures of the investigated problems and meanwhile reduce the memory and computation requirements in WMSNs. The two phase negotiation mechanism enhances the classification decision accuracy at the minimum communication expenditure, which efficiently achieves collaborative in-network processing for WMSN target classification.

6. Conclusions and Future Work

The newly emerging technology of WMSNs has presented a variety of new challenges, including resource constraints, flexible architecture and multimedia in-network processing. Furthermore target classification using audio information in WMSNs is complicated by the uncertainties caused by noise contamination and inherent generalization error of GP classifiers. As verified by the simulation experiments, the proposed statistical processing and negotiation mechanism are capable of meeting the challenges and reduce the uncertainties concurrently. PSD feature extraction and PCA dimension reduction essentially perform lossy compression of raw audio data. Though some information is, to some extent, lost due to compression, yet it has great significance for WMSN applications, because memory and computation capability requirements drastically decrease. As revealed by experiments, the Gaussian process classifiers learned from training samples produce lower classification accuracy for new samples than testing samples, for the reason that the pattern of new samples usually deviate from that of the samples used for the classifier training. This uncertainty is reduced by choosing several agents to collaboratively engage in observation and classification of the target. To reach a compromise between resource consumption and accuracy, the agents are selected by the proposed auction mechanism OSDMIA. The auction finishes with one bidding round, which is so designed to ensure its real time performance. The committee decision mechanism proves to considerably reduce the uncertainty of individual classifier prediction and enhance the overall classification reliability.

The proposed approaches are evaluated by simulation with real world data, and prove to be efficient. However, details such as CPU percentage, memory availableness and communication delay can not be precisely simulated. Therefore this work should be extended by real world deployment on such platforms as MICAz. In addition, deployment of multi-agent systems on real multimedia sensor nodes is more than a trivial task. Gaussian process classification proves to produce good classification accuracy and the approximation methods such as Laplace's method radically reduces computation complexity. Nevertheless its real time performance still needs evaluation in real world deployment.

In the proposed auction mechanism, the utility function is essentially arbitrarily specified, which calls for further in-depth investigation to be of general significance. The situation is the same for the weight determination of committee decision. More often than not these problems are application specific, but it is desirable if a guideline can be found that can invariantly guarantee good enough and sufficient performance. In addition, as far as negotiation is concerned, there are also some important issues (such as fraudulence in bidding and auctioneer selection mechanism) that require further investigation.

Acknowledgements

This paper is supported by the National Grand Fundamental Research 973 Program of China under Grant No.2006CB303000 and National Natural Science Foundation of China under Grant No.60673176, No.60373014 and No.50175056.

References and Notes

1. Baronti P.; Pillai P.; Chook V.W.C. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. *Computer Communication*, **2007**, *30*, 1655-1695.
2. Akyildiz I.F.; Su W.L.; Sankarasubramaniam Y.; Cayirci E. A Survey on sensor networks. *IEEE Communications Magazine* **2002**, *40(8)*, 102-114.
3. Akyildiz I.F.; Su W.; Sankarasubramaniam Y.; Cayirci E. Wireless sensor networks: a survey. *Computer Networks*, **2002**, *38*, 393-422.
4. Lédeczi Á.; Nádas A.; Völgyesi P. Countersniper system for urban warfare. *ACM Transactions on Sensor Networks*, **2005**, *1(2)*, 153-177.
5. Li D.; Wong K.; Hu Y. Detection, classification and tracking of targets in distributed sensor networks. *IEEE Signal Processing Magazine*, **2002**, *19(2)*, pp. 17-30.
6. Wang N.; Zhang N.Q.; Wang M.H. Wireless sensors in agriculture and food industry—Recent development and future perspective. *Computers and Electronics in Agriculture*, **2006**, *50*, 1-16.
7. Cayirci E.; Tezcan H.; Dogan Y.; Coskun V. Wireless sensor networks for underwater surveillance systems. *Ad Hoc Networks*, **2006**, *4*, 431-446.
8. Paek J.; Chintalapudi K.; Govindan R. A wireless sensor network for structural health monitoring: performance and experience. *Proceedings of the Second IEEE Workshop on Embedded Networked Sensors*, **2005**, 1-10.
9. Krishnamurthy L.; Adler R.; Buonadonna P. Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the North Sea. *Proc. of the 3rd International Conference on Embedded Networked Sensor Systems*, **2005**, 64-75.
10. Shnayder V.; Chen B.R.; Lorincz K. Sensor networks for medical care. *Technical Report TR-08-05, Division of Engineering and Applied Sciences, Harvard University*, **2005**.
11. Akyildiz I.F.; Melodia T.; Chowdhury K.R. A survey on wireless multimedia sensor networks. *Computer Networks*, **2007**, *51*, 921-960.
12. Wang X.; Bi D.W.; Ding L.; Wang S. Agent collaborative target localization and classification in wireless sensor networks. *Sensors*, **2007**, *7(8)*, 1359-1386.
13. Lesser V.R. Cooperative multiagent systems: a personal view of the state of the art. *IEEE Transactions on Knowledge and Data Engineering*, **1999**, *11(1)*, 133-142.
14. Marsh D.; Tynan R.; O’Kane D. Autonomic wireless sensor networks. *Engineering Applications of Artificial Intelligence*, **2004**, *17*, 741-748.

15. Sandhu J.S.; Agogino A.M.; Agogino A.K. Wireless sensor networks for commercial lighting control: decision making with multi-agent systems. *AAAI Workshop on Sensor Networks*, **2004**.
16. Marsh D.; O’Kane D.; O’Hare G.M.P. Agents for wireless sensor network power management. *Proceedings of the 2005 International Conference on Parallel Processing Workshops*, **2005**, 413-418.
17. Liu Z.Y.; Wang Y.G. A secure agent architecture for sensor networks,” *Proc. of the 2003 International Conference on Artificial Intelligence - Intelligent Pervasive Computing Workshop (IC-AI’03)*, **2003**.
18. Shakshuki E.; Ghenniwa H.; Kamel M. Agent-based system architecture for dynamic and open environments. *Journal of Information Technology and Decision Making*, **2003**, 2(1), 105-133.
19. Shakshuki E.; Hussain S.; Matin A.W.; Matin A.R. Agent-based peer-to-peer layered architecture for data transfer in wireless sensor networks. *Proc. of 2006 IEEE International Conference on Granular Computing*, **2006**, 490-493.
20. Patricio M.A.; Carbo J.; Perez O.; Garcia J.; Molina J.M. Multi-agent framework in visual sensor networks. *EURASIP Journal on Advances in Signal Processing*, **2007**, 2007, 1-21.
21. Lau R.Y.K. Towards genetically optimised multi-agent multi-issue negotiations. *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, **2005**, 35c-35c.
22. Fatima S.; Wooldridge M.; Jennings N.R. Optimal negotiation of multiple issues in incomplete information settings. *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, **2004**, 1080- 1087.
23. Ito T.; Hattori H.; Klein M. Multi-issue negotiation protocol for agents: exploring nonlinear utility spaces. *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI07)*, **2007**, 1347-1353.
24. Kraus S. *Mutli-agents systems and applications*, New York, NY: Springer-Verlag New York, Inc, **2001**, 150 – 172.
25. Collins J.; Gini M.; Mobasher B. Multi-agent negotiation using combinatorial auctions with precedence constraints. *Technical Report 02-009, Department of Computer Science and Engineering, University of Minnesota*, **2002**.
26. Bartolini C.; Preist C. A generic software framework for automated negotiation. *Technical Report HPL-2002-2, HP Labs*, **2002**.
27. Rahwan I.; McBurney P.; Sonenberg L. Towards a theory of negotiation strategy (a preliminary report). *Proceedings of the 5th Workshop on Game Theoretic and Decision Theoretic Agents (GTDT-2003)*, **2003**, 73-80.
28. Soh L.K.; Tsatsoulisa C. Real-time negotiation model and a multi-agent sensor network implementation. *Autonomous Agents and Multi-Agent Systems*, **2005**, 11, 215-271.

29. Mainland G.; Parkes D.C.; Welsh M. Decentralized, adaptive resource allocation for sensor networks,” *Proceedings of the 2nd conference on Symposium on Networked Systems Design and Implementation - Volume 2*, **2005**.
30. Rasmussen C.E.; Williams C.K.I. *Gaussian Processes for Machine Learning*, Massachusetts London: the MIT Press, **2006**, 2-72.
31. Seeger M. Gaussian processes for machine learning. *International Journal of Neural Systems*, **2004**, *14*(2), 1-38.
32. Kuss M.; Rasmussen C.E. Assessing approximate inference for binary Gaussian process classification. *Journal of Machine Learning Research*, **2005**, *6*, 1679-1704.
33. Gibbs M.N.; Mackay D.J.C. Variational Gaussian process classifiers. *IEEE Transactions on Neural Networks*, **2000**, *11*(6), 1458-1464.
34. Opper M.; Winther O. Gaussian processes for classification: mean-field algorithms. *Neural Computation*, **2000**, *12*(11), 2655-2684.
35. Kim H.C.; Hahramani Z.B. Bayesian Gaussian process classification with the EM-EP algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2006**, *28*(12), 1948-1959.
36. Kim H.C.; Kim D.J.; Ghahramani Z.B. Appearance-based gender classification with Gaussian processes. *Pattern Recognition Letters*, **2006**, *27*, 618-626.
37. Chu W.; Ghahramani Z.B.; Falciani F. Biomarker discovery in microarray gene expression data with Gaussian processes. *Bioinformatics*, **2005**, *21*(16), 3385-3393.
38. Williams D.P. Gaussian process classification using image deformation. *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. ICASSP 2007, Volume 2*, **2007**, 605- 608.
39. Murillo-Fuentes J. J.; Caro S.; Perez-Cruz F. Gaussian processes for multiuser detection in CDMA receivers. *Advances in Neural Information Processing Systems*, Weiss Y.; Scholkopf B.; Platt J. editors, Cambridge, MA: The MIT Press, **2006**, 939-946.
40. Huynh Q.Q.; Cooper L.N.; Intrator N. Classification of underwater mammals using feature extraction based on time-frequency analysis and BCM theory. *IEEE Transactions on Signal Processing*, **1998**, *46*(5), 1202-1207.
41. Ishizuka K.; Miyazaki N. Speech feature extraction method representing periodicity and aperiodicity in sub bands for robust speech recognition. *Proceedings of Acoustics, Speech, and Signal Processing, 2004 (ICASSP '04), Volume 1*, **2004**, 141-144.
42. Lee C.; Hyun D.; Choi E. Optimizing feature extraction for speech recognition. *IEEE Transactions on Speech and Audio Processing*, **2003**, *11*(1), 80-87.
43. Carreira-Perpinan M.A. A review of dimension reduction techniques. *Technical Report CS-96-09, Department of Computer Science, University of Sheffield*, **1997**.
44. Fodor I.K. A survey of dimension reduction techniques. *Technical Report UCRL. ID-148494, Lawrence Livermore National Laboratory*, **2002**.

45. Hu J.; Si J.; Olson B.P.; He J.P. Feature detection in motor cortical spikes by principal component analysis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **2005**, *13*(3), 256-262.
46. Polat K.; Günes S. An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. *Digital Signal Processing*, **2007**, *17*, 702-710.
47. Ye Z.M.; Auner G. Principal component analysis approach for biomedical sample identification. *Proc. 2004 IEEE International Conference on of Systems, Man and Cybernetics, Volume 2*, **2004**, 1348-1353.
48. Wang B.H.; Bangham J.A. PCA based shape descriptors for shape retrieval and the evaluations. *Proceedings of 2006 International Conference on Computational Intelligence and Security, Volume 2*, **2006**, 1401-1406.
49. Zubko V.; Kaufman Y.J.; Burg R.I.; Martins J.V. Principal component analysis of remote sensing of aerosols over oceans. *IEEE Transactions on Geoscience and Remote Sensing*, **2007**, *45*(3), 730-745.
50. Rao A.S.; Georgeff M.P. BDI agents: from theory to practice. *Proc. of First International Conference on Multi-agent Systems*, **1995**, 312-319.
51. Thangarajah J.; Lin P.; Harland J. Representation and reasoning for goals in BDI agents. *Proceedings of the Twenty-Fifth Australasian Computer Science Conference (ASCS2002)*, **2002**, 259-245.
52. Georgeff M.P.; Pell B.; Pollack M.E.; Tambe M.; Wooldridge M. The Belief-Desire-Intention model of agency. *Proceedings of the 5th International Workshop on Intelligent Agents V, Agent Theories, Architectures, and Languages*, **1998**, 1-10.
53. Braubach L.; Pokahr A.; Lamersdorf W.; Moldt D. Goal representation for BDI agent systems. *Second International Workshop on Programming Multiagent*, **2004**, 9-20.
54. Labrou Y.; Finin T.; Peng Y. The current landscape of agent communication languages. *IEEE Intelligent Systems*, **1999**, *14*(2), 45-52.
55. Pokahr A.; Braubach L.; Lamersdorf W. Jadex: implementing a BDI-infrastructure for JADE agents. *EXP In Search of Innovation (Special Issue on JADE)*, **2003**, *3*(3), 76-85.
56. Bordini R.; Dastani M.; Dix J. Jadex: A BDI Reasoning Engine. *Multi-Agent Programming, Springer Science+Business Media Inc.*, **2005**, 149-174.
57. Bellifemine F.; Poggi A.; Rimassa G. Developing multiagent systems with JADE. *Proceedings of the 7th International Workshop Agent Theories Architectures and Languages*, **2000**, 89-103.
58. Jennings N.R. An agent-based approach for building complex software systems. *Communications of the ACM*, **2001**, *44*(4), 35-41.
59. M. Wooldridge. *Introduction to Multiagent Systems*, New York, NY: John Wiley & Sons, Inc., **2001**.

60. Rahwan I. Interest-based negotiation in multi-agent systems. *PhD thesis, Dept. of Information Systems, the University of Melbourne, 2004.*
61. Shi M.H.; Amine B. Committee machine with over 95% classification accuracy for combustible gas identification. *Proc. of the 13th IEEE International Conference on Electronics, Circuits and Systems, 2006. ICECS '06, 2006*, 862-865.
62. Tang H.M.; Lyu M.R.; King I. Face recognition committee machines: dynamic vs. static structures. *Proceedings of 12th International Conference on Image Analysis and Processing, 2003*, 121-126.
63. Haykin S. *Neural Networks*, New Jersey: Prentice Hall, **1999**, 392-435.
64. Marco F.D.; Yu H.H. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing* **2004**, *64*, 826-838.

© 2007 by MDPI (<http://www.mdpi.org>). Reproduction is permitted for noncommercial purposes.