Chapter 2
Data Fusion in WSN

Abstract WSN is intended to be deployed in environments where sensors can be exposed to circumstances that might interfere with measurements provided. Such circumstances include strong variations of pressure, temperature, radiation, and electromagnetic noise. Thus, measurements may be imprecise in such scenarios. Data fusion is used to overcome sensor failures, technological limitations, and spatial and temporal coverage problems. Data fusion is generally defined as the use of techniques that combine data from multiple sources and gather this information in order to achieve inferences, which will be more efficient and potentially more accurate than if they were achieved by means of a single source. The term efficient, in this case, can mean more reliable delivery of accurate information, more complete, and more dependable. The data fusion can be implemented in both centralized and distributed systems. In a centralized system, all raw sensor data would be sent to one node, and the data fusion would all occur at the same location. In a distributed system, the different fusion modules would be implemented on distributed components. Data fusion occurs at each node using its own data and data from the neighbors. This chapter briefly discusses the data fusion and a comprehensive survey of the existing data fusion techniques, methods and algorithms.

2.1 Introduction

A Wireless Sensor Network (WSN) may be designed with different objectives. It may be designed to gather and process data from the environment in order to have a better understanding of the behavior of the monitored area. It may also be designed to watch an environment for the occurrence of a set of possible events, thus the proper action may be taken whenever needed. A fundamental issue in WSN is the way to process the collected data. In this situation, data fusion arises as a discipline that is concerned with how data collected by sensors can be processed to increase the significance of such a mass of data [1]. Thus, data fusion can be defined as the combination of multiple sources to obtain improved data i.e., cheaper, greater

A. Abdelgawad and M. Bayoumi, *Resource-Aware Data Fusion Algorithms for Wireless Sensor Networks*, Lecture Notes in Electrical Engineering 118,
DOI 10.1007/978-1-4614-1350-9_2,

quality, or greater relevance. Data fusion is commonly used in detection and classification tasks in different application domains, such as military applications and robotics [2]. Within the WSN domain, simple aggregation techniques i.e., maximum, minimum, and average have been used to reduce the overall data traffic to save energy [3, 4]. Additionally, data fusion techniques have been applied to WSNs to improve location estimates of sensor nodes, detect routing failures, and collect link statistics for routing protocols [5].

WSN is intended to be deployed in environments where sensors can be exposed to circumstances that might interfere with measurements provided. Such circumstances include strong variations of pressure and temperature, radiation and electromagnetic noise. Thus, measurements may be imprecise in such scenarios. Even when environmental conditions are ideal, sensors may not give perfect measurements. Basically, a sensor is a measurement device, and vagueness is usually associated with its observation. Such imprecision represents the imperfections of the technology and methods used to measure a physical incident. Failures are not an exception in WSN. For example, consider a WSN that monitors a jungle to detect an event, such as fire or the presence of an animal. Sensor nodes can be destroyed by fire, animals, or even human beings; they might present manufacturing problems; and they might stop working due to a lack of energy. Each node that becomes inoperable might compromise the overall perception and/or the communication capability of the network. Here, perception ability is equivalent to the exposure concept. Both spatial and temporal coverage also pose limitations to WSN. The sensing capability of a node is restricted to a limited area. For example, a thermometer in a room reports the temperature near the device but it might not represent fairly the overall temperature inside the room. Spatial coverage in WSN has been explored in different scenarios, such as node scheduling, target tracking, and sensor placement. Temporal coverage can be understood as the ability to fulfill the network purpose during its lifetime. For example, in a WSN for event detection, temporal coverage aims at assuring that no relevant event will be missed because there was no sensor perceiving the region at the specific time the event occurred. Thus, temporal coverage depends on the sensor's sampling rate, node's duty cycle, and communication delays. To overcome sensor failures, technological limitations, and spatial and temporal coverage problems, three properties must be ensured:

1. Cooperation.
2. Redundancy
3. Complementarily

Usually, the area of interest can only be completely covered by the use of several sensor nodes, each cooperating with a partial view of the scene; and data fusion can be used to create the complete view from the pieces provided by each node. Redundancy makes the WSN less vulnerable to failure of a single node, and overlapping measurements can be fused to obtain more precise data. Complementarily can be achieved by using sensors that observe different properties of the environment; data fusion can be used to combine complementary data so the resultant data allows inferences that might be not possible to be obtained from the individual measurements, e.g., angle and distance of an imminent threat can be fused to obtain its position. Due to redundancy and cooperation properties, WSN is often composed of a large number of sensor nodes posing a new scalability challenge caused by possible collisions and transmissions of redundant data. Regarding the energy restrictions, communication should be reduced to increase the lifetime of the sensor nodes. Hence, data fusion is also important to reduce the overall communication load in the network by avoiding the transmission of redundant messages. In addition, any task in the network that handles signals or needs to make inferences can potentially use data fusion. Data fusion should be considered a critical step in designing a wireless sensor network. The reason is that data fusion can be used to extend the network lifetime and is commonly used to fulfill the application objectives, such as event detection, target tracking, and decision making. Hence, careless data fusion may result in waste of resources and misleading assessments. Therefore, we must be aware of possible limitations of data fusion to avoid blundering situations. Because of the resource rationalization needs of WSN, data processing is commonly implemented as in-network algorithms. Hence, data fusion should be performed in a distributed fashion to extend the network lifetime. Even so, we must be aware of the limitations of distributed implementations of data fusion. Thus, regarding the communication load, a centralized fusion system may outperform a distributed one. The reason is that centralized fusion has a global knowledge in the sense that all measured data is available, whereas distributed fusion is incremental and localized since it fuses measurements provided by a set of neighbor nodes and the result might be further fused by intermediate nodes until a sink node is reached. Such a drawback of decentralized fusion might often be present in WSN wherein, due to resource limitations, distributed and localized algorithms are preferable to centralized ones.

Data fusion has established itself as an independent research area over the last decades, but a general formal theoretical framework to describe data fusion systems is still missing. One reason for this is the huge number of disparate research areas that utilize and illustrate some form of data fusion in their context of theory. For example, the concept of data or feature fusion, which forms together with classifier and decision fusion the three main divisions of fusion levels, initially occurred in multi-sensor processing. By now several other research fields found its application useful. Besides the more classical data fusion approaches in statistics, control, robotics, computer vision, geosciences and remote sensing, artificial intelligence, and digital image/signal processing, the data retrieval community discovered some years ago its power in combining multiple data sources.

2.2 Information Fusion, Sensor Fusion, and Data Fusion

Several different terms have been used to illustrate the aspects regarding the fusion subject, e.g. information fusion, sensor fusion, and data fusion. The expressions related to systems, applications, methods, architectures, and theories about the fusion of data from multiple sources are not unified yet. Different terms have been adopted, usually associated with particular aspects that characterize the fusion i.e., sensor fusion is commonly used to specify that sensors provide the data being fused. Despite the theoretical issues about the difference between information and data, the terms information fusion and data fusion are usually accepted as overall terms. Many definitions of data fusion have been provided along the years, most of them were found in military and remote sensing fields. The data fusion work group of the Joint Directors of Laboratories (JDL) organized an effort to define a dictionary with some terms of reference for data fusion [6]. They define data fusion as a multilevel process dealing with the automatic detection, estimation, association, correlation, and combination of data and data from several sources. The JDL data fusion model deals with quality improvement. Hall defines data fusion as a combination of data from multiple sensors to accomplish improved accuracy and more specific inferences that could be achieved by the use of a single sensor alone [7]. All the previous definitions are focused on means, methods and sensors. Wald in [8] changes the attention of fuse data to the used framework. He defines data fusion as a formal framework in which is expressed means and tools for the alliance of data originating from different sources. He considers data taken from the same source at different instants as separate sources. For WSN, data can be fused with at least two objectives: accuracy improvement and energy saving.

Multisensor integration is another expression used in computer vision and industrial automation. Luo [9] defines multisensor integration as a synergistic use of data provided by multiple sensory devices to help in the accomplishment of a task by a system. However, multisensor fusion deals with the combination of different sources of sensory data into one representational format during any stage in the integration process. Multisensor integration is a broader term than multisensor fusion. It makes clear how the fused data is used by the whole system to interact with the environment. However, it might suggest that only sensory data is used in the fusion and integration processes.

The term data aggregation term has become popular in the wireless sensor network community as a synonym for information fusion [10]. Data aggregation comprises the collection of raw data from pervasive data sources, the flexible, programmable composition of the raw data into less voluminous refined data, and the timely delivery of the refined data to data consumers. Aggregation is the ability to summarize data i.e., the amount of data is reduced. However, for applications that require original and accurate measurements, such summarization may represent an accuracy loss [11]. Although many applications might be interested only in summarized data, we cannot always state whether or not the summarized data is more precise than the original data set. Because of that, the use of data aggregation as a general term should be avoided because it also refers to one example of data fusion, which is summarization. Figure 2.1 shows the relationship among the concepts of multisensor/sensor fusion, multisensor integration, data aggregation, information fusion, and data fusion. Here, we understand that both terms, information fusion and data fusion, can be used with the

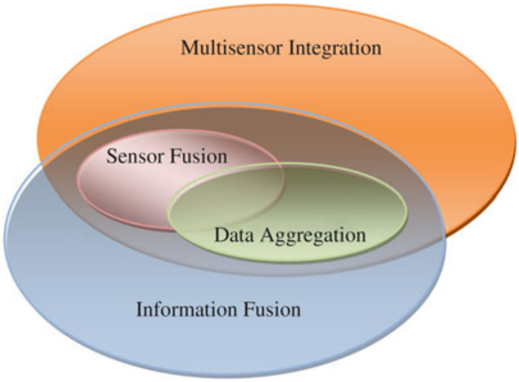


Fig. 2.1 The relationship among the fusion terms: multisensor/sensor fusion, multisensor integration, data aggregation, information fusion and data fusion

same meaning. Multisensor/sensor fusion is the subset that operates with sensory sources. Data aggregation defines another subset of information fusion that means to reduce the data volume, which can manipulate any type of information/data, including sensory data. Thus, multisensor integration is a slightly different term in the sense that it applies information fusion to make inferences using sensory devices and associated information to interact with the environment. Thus, multisensor/sensor fusion is fully contained in the intersection of multisensor integration and information/data fusion.

2.3 Data Fusion Classification

Data fusion can be classified based on several features. Relationships among the input data can be used to divide data fusion into:

1. Cooperative data
2. Redundant data
3. Complementary data.

The abstraction level of the manipulated data during the fusion process can be used to distinguish among fusion processes as:

1. Measurement
2. Signal
3. Feature
4. Decision

Another general classification considers the abstraction level, and it makes explicit the abstraction level of the input and output of a fusion process.

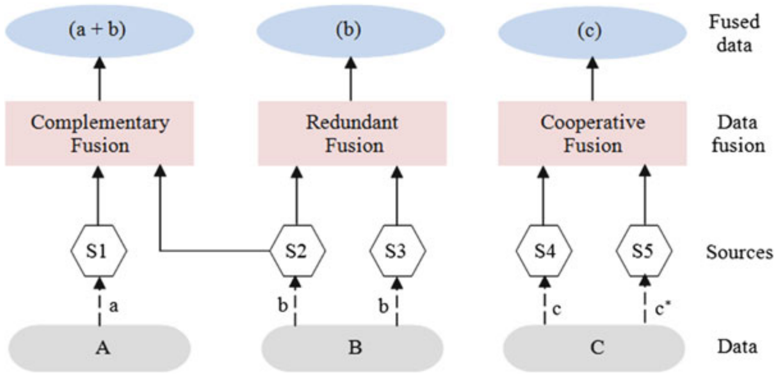


Fig. 2.2 Types of data fusion based on the relationship among the sources

2.3.1 Classification Based on Relationship Among the Sources

Data fusion can be classified, according to the relationship among the sources [9]. Thus, data fusion can be:

1. Complementary: Data provided by the sources represents different portions of a broader scene; data fusion can be applied to obtain a piece of data that is more complete. In Fig. 2.2, sources S1 and S2 provide different pieces of data (a and b) that can be fused to achieve a complete data (a + b) composed of non-redundant pieces a and b that refer to different parts of the environment. In general, complementary fusion searches for completeness by compounding new data from different pieces. Hoover [12] applies complementary fusion by using several cameras to observe different parts of the environment; then the video streams are fused into an occupancy map that is used to guide a mobile robot. An example of complementary fusion consists in fusing data from sensor nodes, e.g., a sample from the sensor field, into a feature map that describes the whole sensor field [13].
2. Redundant: If two or more independent sources provide the same piece of data, these pieces can be fused to increase the associated confidence. Sources S2 and S3 in Fig. 2.2 provide the same data (b). S2 and S3 are fused to obtain more accurate data (b). Redundant fusion might be used to increase the reliability, accuracy, and confidence of the data. In WSN, redundant fusion can provide high quality data and prevent sensor nodes from transmitting redundant data.
3. Cooperative: Independent sources are cooperative when the data provided by them is fused into new data that represents the reality. Sources S4 and S5 in Fig. 2.2, provide different data, c and c*, that are fused into (c), which better describes the scene compared to c and c* individually. A traditional example of

cooperative fusion is the computation of a target location based on angle and distance data. Cooperative fusion should be carefully applied since the resultant data is subject to the inaccuracies and imperfections of all participating sources.

2.3.2 Classification Based on Levels of Abstraction

Luo in [14] applied four levels of abstraction to classify data fusion:

1. Signal level fusion: It deals with single sensors and can be used in real-time applications or as an intermediate step for further fusions.
2. Pixel level fusion: It operates on images and can be used to improve image-processing tasks.
3. Feature level fusion: Deals with features or attributes extracted from signals or images, such as speed and shape.
4. Symbol level fusion: Data is a symbol that represents a decision, and it is also referred to a decision level.

In general, the feature and symbol fusions are used in object recognition applications. This classification presents some disadvantages and is not suitable for all data fusion applications. First, both images and signals are considered raw data and are usually provided by sensors, so they should be included in the same class. Second, raw data may not be only from sensors, because data fusion systems might also fuse data provided by databases or human interaction. Third, it proposes that a fusion process cannot deal with all levels at the same time.

According to the level of abstraction of the manipulated data, data fusion can be classified into four categories:

1. Low-level fusion: Raw data are provided as inputs and combined into new data that are more accurate than the individual inputs. Polastre in [15] gave an example of low-level fusion by applying a moving average filter to estimate ambient noise and determine whether or not the communication channel is clear.
2. Medium-level fusion: Features and attributes of an entity are fused to obtain a feature map that may be used for other tasks. It is also known as feature/attribute level fusion.
3. High-level fusion: It is known as symbol or decision level fusion. It takes decisions or symbolic representations as input and combines them to obtain a more confident and/or a global decision. An example of high-level fusion is the Bayesian approach for binary event detection proposed by Krishnamachari in [16] that detects and corrects measurement faults.
4. Multilevel fusion: Fusion process encompasses data of different abstraction levels and both input and output of fusion can be of any level. For example, a measurement is fused with a feature to provide a decision.

2.3.3 Classification Based on Input and Output

Dasarathy introduced another classification that considers the abstraction level. Data fusion processes are categorized based on the level of abstraction of the input and output data [17]. He identifies five categories:

1. Data in – data out (DAI-DAO): In this class, data fusion deals with raw data and the result is also raw data, possibly more accurate or reliable.
2. Data in – feature out (DAI-FEO): Data fusion uses raw data from sources to extract features or attributes that describe an entity. Entity here means any object, situation, or world abstraction.
3. Feature in – feature out (FEI-FEO): It works on a set of features to improve/refine a feature, or extract new ones.
4. Feature in – decision out (FEI-DEO): Data fusion takes a set of features of an entity generating a symbolic representation or a decision.
5. Decision in – decision out (DEI-DEO): Decisions can be fused in order to obtain new decisions or give emphasis on previous ones.

In comparison to the classification presented before, this classification can be seen as an extension of the earlier one with a finer granularity where DAI-DAO corresponds to Low Level Fusion, FEI-FEO to Medium Level Fusion, DEI-DEO to High Level Fusion, DAI-FEO and FEI-DEO are included in Multi-level Fusion.

2.4 Data Fusion: Techniques, Methods, and Algorithms

Techniques, methods, and algorithms used to fuse data can be classified based on several criteria, such as the data abstraction level, parameters, mathematical foundation, purpose, and type of data. Data fusion can be performed with different objectives such as inference, estimation, feature maps, aggregation, abstract sensors, classification, and compression.

2.4.1 Inference

Inference method is applied in decision fusion. The decision is taken based on the knowledge of the perceived situation. At this point, inference refers to the transition from one likely true proposition to another, which its truth is believed to result from the previous one. Classical inference methods are based on the Bayesian inference and the Dempster-Shafer belief about accumulation theory.

1. Bayesian inference: Data fusion based on Bayesian Inference provides a formalism to merge evidence according to rules of probability theory. The uncertainty is represented in terms of conditional probabilities describing the belief, and it can assume values in the [0, 1] interval, where 0 is the absolute disbelief and 1 is the absolute belief. Within the WSN domain, Bayesian inference has been used to solve the localization problem. Sichitiu in [18] uses the Bayesian inference to process data from a mobile beacon and determine the most likely geographical location of each node, as an alternative of finding a unique point for each node location.
2. Dempster-Shafer inference: The Dempster-Shafer inference is based on the Dempster-Shafer belief accumulation, which is a mathematical theory introduced by Dempster [19] and Shafer [20] that generalizes the Bayesian theory. It deals with beliefs or mass functions just as Bayes' rule does with probabilities. The Dempster-Shafer theory introduced a formalism that can be used for incomplete knowledge representation and evidence combination. Pinto discussed in-network implementations of the Dempster-Shafer and the Bayesian inference in such a way that event detection and data routing are combined into a single algorithm [21]. By using a WSN composed of Unmanned Aerial Vehicle (UAV) as sensor nodes, Yu uses the Dempster-Shafer inference to build dynamic operational pictures of battlefields for situation evaluation. However, the particular challenges of in-network fusion in such a mobile network are not evaluated [22].
3. Fuzzy logic: Fuzzy logic generalizes probability and, therefore, is able to deal with approximate reasoning to draw conclusions from imprecise premises. Each quantitative input is fuzzyfied by a membership function. The fuzzy rules of an inference system generate fuzzy outputs which, in turn, are defuzzyfied by a set of output rules. This structure has been successfully used in real world situations that defy exact modeling, from rice cookers to complex control systems. Gupta uses fuzzy reasoning for deciding the best cluster-heads in a WSN [23].
4. Neural networks: Neural Networks represent an alternative to Bayesian and Dempster-Shafer theories, being used by classification and recognition tasks in the data fusion domain. A key feature of neural networks is the capability of learning from examples of input/output pairs in a supervised fashion. For that reason, neural networks can be used in learning systems while fuzzy logic is used to control its learning rate. Neural networks have been applied to data fusion mainly for automatic target recognition using multiple complementary sensors.
5. Semantic data fusion: In semantic data fusion, raw sensor data is processed so that nodes exchange only the resulting semantic interpretations. The semantic abstraction allows a WSN to optimize its resource utilization when storing, collecting, and processing data. Semantic data fusion usually comprises two phases: pattern matching and knowledge-base construction. Friedlander [24] introduced the concept of semantic data fusion, which was applied for target classification.

2.4.2 Estimation

Estimation method was inherited from control theory and used the laws of probability to compute a process state vector from a measurement vector or a sequence of measurement vectors. We present, in this section, the estimation methods known as: Least Squares, Maximum Likelihood (ML), Moving Average filter, Kalman filter, and Particle filter.

1. Least squares: Least Squares method is a mathematical optimization technique that searches for a function that best fits a set of input measurements. This is accomplished by minimizing the sum of the square error between points generated by the function and the input measurements. The Least Squares method is suitable when the parameter to be estimated is considered fixed. Least Square method does not assume any prior probability.
2. Maximum likelihood: Estimation methods based on Likelihood are suitable when the state being estimated is not the outcome of a random variable. Xiao proposes a distributed and localized Maximum Likelihood that is robust to the unreliable communication links of WSN. In this method, every node computes a local unbiased estimate that converges towards the global Maximum Likelihood solution [25]. Xiao further extended this method to support asynchronous and timely delivered measurements, i.e., measurements taken at different time steps that happen asynchronously in the network. Other distributed implementations of ML for WSN include the Decentralized Expectation Maximization (DEM) algorithm and the local Maximum Likelihood estimator that relax the requirement of sharing all the data [26].
3. Moving average filter: Moving average filter is broadly adopted in digital signal processing (DSP) solutions as it is simple to understand and use. Moreover, this filter is optimal for reducing random white noise while retaining a sharp step response. This is the reason that makes the moving average the major filter for processing encoded signals in the time domain. As the name suggests, this filter computes the arithmetic mean of a number of input measurements to produce each point of the output signal. Yang uses the Moving Average filter on target locations to reduce errors of tracking applications in WSNs [27].
4. Kalman filter: Kalman filter is a very popular fusion method. It was originally proposed in 1960 by Kalman [28] and it has been extensively studied since then. Kalman filter is used to fuse low-level redundant data. If a linear model can describe the system and the error can be modeled as Gaussian noise, the Kalman filter recursively retrieves statistically optimal estimates. On the other hand, to deal with non-linear dynamics and non-linear measurement models other methods should be adopted. In WSN, we can find schemes to approximate distributed Kalman filter, in which the solution is computed based on reaching an average consensus among sensor nodes [29].
5. Particle filter: The Particle filter is a recursive implementation of a statistical signal processing known as sequential Monte Carlo methods. Although

Kalman filter is a classical approach for state estimation, particle filters represent an alternative for applications with non-Gaussian noise, especially when computational power is rather cheap and sampling rate is slow. The particles are propagated over time, sequentially combining, sampling, and resampling steps. At each time step, the resampling is used to discard some particles, increasing the relevance of regions with high posterior probability. Target tracking is currently the principal research problem wherein particle filters have been used.

2.5 Data Fusion: Architectures and Models

Many architectures and models have been introduced to serve as guidelines to design data fusion systems. These models evolved from data-based models to role-based models. These models are useful for guiding the specification, proposal, and usage of data fusion in WSN. Some of these models, such as the JDL and Frankel-Bedworth, provide a systemic view of data fusion, whereas others, such as the Intelligent Cycle and the Boyd Control Loop, provide a task view of data fusion.

2.5.1 Data-Based Models

Models proposed to design data fusion systems can be centered on the abstraction of the data generated during fusion. This section introduces the models that specify their stages based on the abstraction levels of data manipulated by the fusion system [1].

1. JDL model: JDL is a well-known model in the fusion research area. It was originally proposed by the U.S. Joint Directors of Laboratories (JDL) and the U.S. Department of Defense (DoD). The model consists of five processing levels, an associated database, and a data bus connecting all components as shown in Fig. 2.3. Its components are described as follows:

   - Sources: It is responsible for providing the input data and can be sensors, a prior knowledge, databases, or human input.
   - Database management system: It supports the maintenance of the data used and provided by the data fusion system. This is a critical function, as it supposedly handles a large and varied amount of data. In WSNs, this function might be simplified to fit the sensors' restrictions of resources.
   - Human computer interaction (HCI): It is a mechanism that allows human input, such as commands and queries, and the notification of fusion results through alarms, displays, graphics, and sounds. Commonly, human interaction with WSNs occurs through the query-based interfaces.

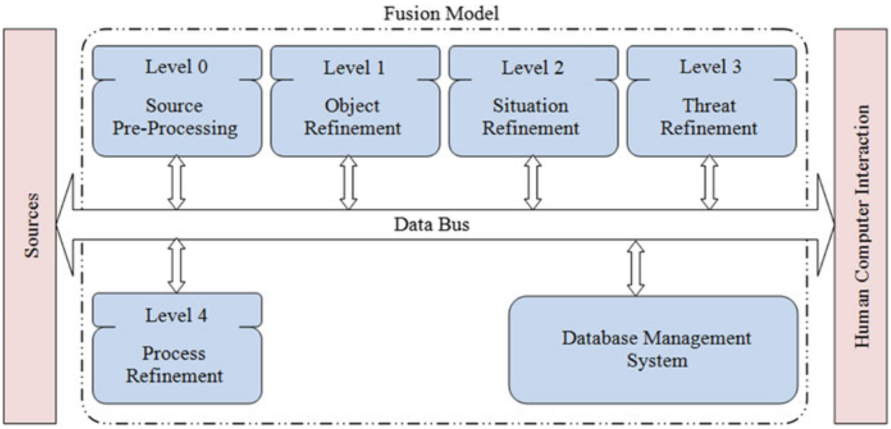


Fig. 2.3 The JDL model

- Level 0 (source preprocessing): It is also referred to as process alignment, this level aims to reduce the processing load by allocating data to appropriate processes and selecting appropriate sources.
- Level 1 (object refinement): It converts the data into a consistent structure. Source localization, and therefore, all tracking algorithms are in Level 1, since they transform different types of data, such as images, angles, and acoustic data, into a target location.
- Level 2 (situation refinement): It attempts to provide a contextual description of the relationship between objects and observed events. It uses a prior knowledge and environmental data to identify a situation.
- Level 3 (threat refinement): It estimates the current situation, projecting it in the future to identify possible threats, vulnerabilities, and opportunities for operations. This is a difficult task because it deals with computation complexities and enemies intent assessment.
- Level 4 (process refinement): It is responsible for monitoring the system performance and allocating the sources according to the specified goals. This function may be outside the domain of specific data fusion functions.

2. Dasarathy model: The Dasarathy model [17] is a fine-grained data-centered model in which the elements of data fusion are specified based on their inputs and outputs. It is known also as Data-Feature-Decision (DFD) [17]. Figure 2.4 depicts the DFD model.

The primary input is raw data and the main output is a decision. The components responsible for the several fusion stages are the elements DAI-DAO, DAI-FEO, FEIFEO, FEI-DEO and DEI-DEO, described before. The DFD model is successful in specifying the main types of fusion regarding their input and output data. For this reason it is also used to classify data fusion. In contrast to the JDL model, the DFD model does not provide a systemic view; instead it provides a fine-grained way to

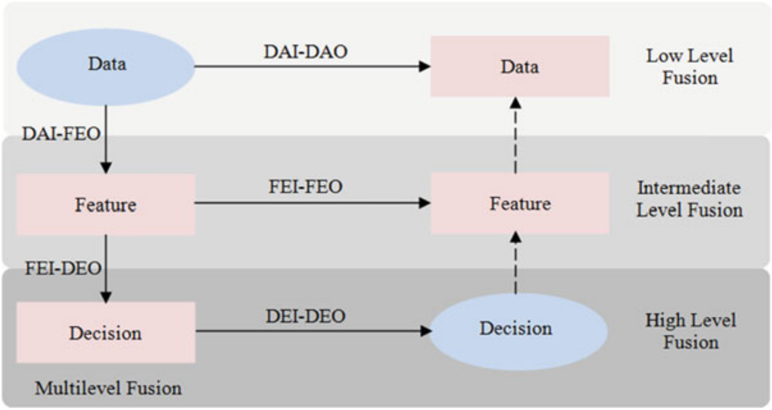


Fig. 2.4 The DFD model

specify fusion tasks by means of the expected input and output data. Therefore, the DFD model is useful for specifying and designing fusion algorithms in WSNs with different purposes such as ambient noise estimation.

2.5.2 Activity-Based Models

Some models are specified based on the activities that must be performed by the data fusion system. The activities and their correct sequence of execution, in such models, are explicitly specified.

1. Boyd control loop: The Boyd Control Loop is a cyclic model composed of four stages. It is known also as the Observe, Orient, Decide, Act (OODA) Loop. This model is a representation of the classic decision-support mechanism of military data systems, and because such systems are strongly coupled with fusion systems, the OODA loop has been used to design data fusion systems. The stages of the OODA loop define the major activities related to the fusion process as shown in Fig. 2.5, which are:

   - Observe: Data gathering from the available sources. It corresponds to level 0 of the JDL model.
   - Orient: Gathered data is fused to obtain an interpretation of the current situation. It encompasses levels 1, 2, and 3 of the JDL model.
   - Decide: Specify an action plan in response to the understanding of the situation. It matches level 4 of JDL model.
   - Act: The plan is executed. It is not dealt by the JDL model.

   The OODA loop is a wide model that allows the specification and visualization of the system tasks in an ample way. It allows the modeling of the main tasks of a system. The OODA fails to provide a proper representation of specific tasks of a data fusion system.

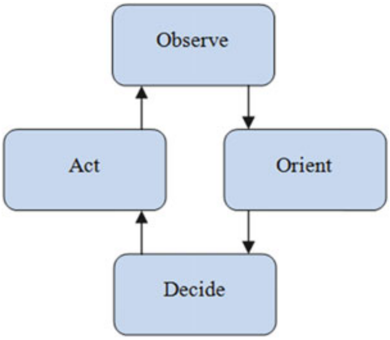


Fig. 2.5 The OODA loop

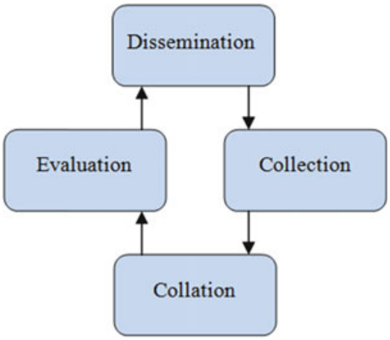


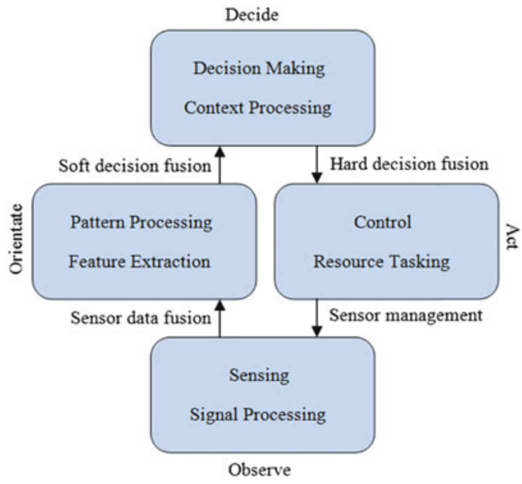
Fig. 2.6 The intelligence cycle

2. Intelligence cycle: The intelligence process is a four-stage cycle, which is called Intelligence Cycle. Figure 2.6, shows the process of developing raw data into finished intelligence used in decision-making and action. The activities of the Intelligence Cycle are:

   - Collection: Raw data is collected from the environment. It matches level 0 of the JDL model.
   - Collation: Collected data is compared, analyzed, and correlated. Irrelevant and unreliable data is discarded. Includes level 1 of the JDL model.
   - Evaluation: Collated data is fused and analyzed. It comprises levels 2 and 3 of the JDL model.
   - Dissemination: Fusion results are delivered to users who utilize the fused data to produce decisions and actions in response to the detected situation. It corresponds to level 4 of the JDL model.

   The Intelligence Cycle does not make explicit the planning (Decide) and the execution (Act) phases, which are most likely included in the Evaluation and Dissemination phases. The OODA and Intelligence Cycle are general and can be employed in any application domain. They do not fulfill the specific aspects of the fusion domain demanding, thus, experience and expertise to model fine-grained fusion tasks.

Fig. 2.7 Omnibus model



3. Omnibus model: The Omnibus model organizes the stages of a data fusion system in a cyclic sequence, just as the Intelligence Cycle and the OODA loop do [30]. The Omnibus model should be applied during the design phase of a data fusion system. Initially, it should be used to model the framework providing a general perception of the system. Then, the model can be used to design the subtasks, providing a fine-grained understanding of the system. Figure 2.7 shows the Omnibus model. The Omnibus model was originally proposed to deal with data collected by sensor devices. Some modifications can be suggested to make it more broad and suitable for other data fusion systems such as:

   - Sensing and signal processing can be replaced by data gathering and data preprocessing, respectively.
   - Sensor data fusion should be stated as raw data fusion.
   - Instead of Sensor management we should adopt source management.

   In this way, the Omnibus model will be suitable for data systems that deal with any kind of sources, including sensors.

2.5.3 Role-Based Model

Role-based model represents a change of focus on how data fusion systems can be modeled and designed. Data fusion systems are specified based on the fusion roles and the relationships among them providing a more fine-grained model for the fusion system. The two members of this generation are the Object-Oriented Model and the Frankel-Bedworth architecture [31]. The role-based model provides a systemic view of data fusion like the JDL model. However, it does not specify fusion tasks or activities. Instead, it provides a set of roles and specifies the relationships among them.

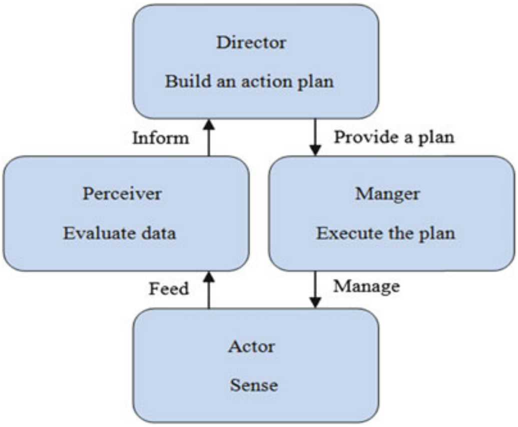


Fig. 2.8 The object-oriented model for data fusion

1. Object-oriented model: Kokar proposes an object-oriented model for data fusion systems. Figure 2.8 is a simplification of the object-oriented model in which four roles are identified as:

   - Actor: It is responsible for the interaction with the world, collecting data and acting on the environment.
   - Perceiver: After data is gathered, the perceiver assesses such data providing a contextualized analysis to the director.
   - Director: The director builds an action plan specifying the system's goals, based on the analysis provided by the perceiver.
   - Manager: It controls the actors to execute the plans formulated by the director.

2. Frankel-Bedworth architecture: Frankel described an architecture for human fusion composed of two self regulatory processes:

   - Local: The local estimation process manages the execution of the current activities based on goals and timetables provided by the global process.
   - Global: The global process updates the goals and timetables according to the feedback provided by the local process.

   Figure 2.9 shows the Frankel-Bedworth architecture. The local and global processes have different objectives and, consequently, different roles. The local process tries to achieve the specified goals and maintain the specified standards. The local process has the estimator role, which is similar to the previous fusion models and includes the following tasks:

   - Sense: Data is gathered by the data sources.
   - Perceive: Stimuli retrieved by sensing are dealt according to its relevance (focus), and the Controller is informed which stimuli are being used (awareness).

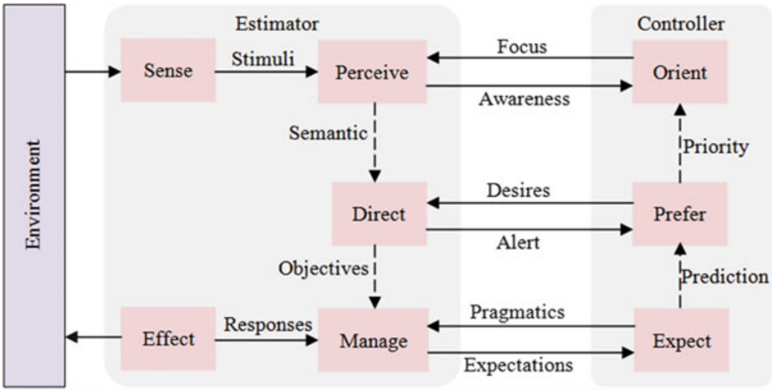


Fig. 2.9 The Frankel-Bedworth architecture

- Direct: Based on the comprehension of the perception (semantics), the Estimator can provide a feedback (alert) to the Controller. The disparity between current situation and desired situation is evaluated. Then, the Estimator is fed forward with desires that specify new goals and timetables.
- Manage: Based on the objectives, the Controller is activated to define what is practical (pragmatics) so the Estimator can provide an appropriate response. Then, the Estimator provides a feedback to the Controller by reporting the expectations about the provided decision (sensitivity).
- Effect selected decisions (responses) are applied and the control loop is closed by sensing the changes in the environment.

Global control process manages the goals of the system during the execution of the local process. The global process has the Controller role; it is responsible for controlling and managing the Estimator role and includes the following tasks:

- Orient: The relevance of sensed stimuli is configured.
- Prefer: Priority is given to the aspects that are most relevant to the goal-achieving behavior, detailing the local goals.
- Expect: Prediction is made and the intentional objective is filtered, determining what is practical to the estimator pragmatics.

The Frankel-Bedworth architecture introduces the notion of a global process separated from the local process. The global control process rules the local process by monitoring its performance and controlling its goals. Moreover, the local process is supposed to perform and implement fusion methods and algorithms to accomplish the system's objectives. This architecture expands the previous models that were concerned only with the local process aspects. In WSN, the global control process will most likely be performed by human beings who feed the network with operation guidelines, whereas the local estimation process should be implemented within the computational system.

Although these models provide a clear understanding of the fusion task, they do not explicitly consider the particularities of the WSN.

Bibliography

1. F. Nakamura, A.F. Loureiro, and C. Frery, “Data Fusion for Wireless Sensor Networks: Methods, Models, and Classifications,” *ACM Computing Surveys,* vol. 39, No. 3, Article 9, 2007, August 2007.
2. R.R. Brooks and S. Iyengar, *Multi-Sensor Fusion: Fundamentals and Applications with Software*: Prentice Hall PTR, Upper Saddle River, NJ, 2003.
3. C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: A scalable and robust communication paradigm for sensor networks,” in *Proceeding of the 6th ACM Annual International Conference on Mobile Computing and Networking,* Boston, MA, USA, August 2000, pp. 56–67.
4. L. Krishnamachari, D. Estrin, and S. Wicker, “The impact of data aggregation in wireless sensor networks,” in *Proceeding of the 22nd International Conference on Distributed Computing Systems Workshops*, Vienna, Austria, July 2002, pp. 575–578.
5. A. Woo, T. Tong, and D. Culler, “Taming the underlying challenges of reliable multihop routing in sensor networks,” in *Proceeding of the 1st International Conference on Embedded Network Sensor Systems*, Los Angeles, November 2003, pp. 14–27.
6. “Data fusion lexicon,” U. S. D. o. Defence, Ed.: Data Fusion Subpanel of the Joint Directors of Laboratories 1991.
7. J. Llinas and D.L. Hall, “An introduction to multi-sensor data fusion,” in *Proceeding of the IEEE International Symposium on Circuits and Systems*, Monterey, CA, USA, May 1998, pp. 537–540.
8. L. Wald, “Some terms of reference in data fusion,” *IEEE Transactions on Geoscience and Remote Sensing,* vol. 37, pp. 1190–1193, May 1999.
9. R.C. Luo and M.G. Kay, “Multisensor integration and fusion in intelligent systems,” *IEEE Transactions on Systems, Man and Cybernetics,* vol. 19, pp. 901–931, October 1989.
10. K. Kalpakis, K. Dasgupta, and P. Namjoshi, “Efficient algorithms for maximum lifetime data gathering and aggregation” *The International Journal of Computer and Telecommunications Networking,* vol. 42, pp. 697–716, August 2003.
11. A. Boulis, S. Ganeriwal, and M.B. Srivastava, “Aggregation in sensor networks: an energy-accuracy trade-off,” in *Proceeding of the 1st IEEE International Workshop on Sensor Network Protocols and Applications,* Anchorage, AK, USA, May 2003, pp. 128–138.
12. A. Hoover and B.D. Olsen, “A real-time occupancy map from multiple video streams,” in *Proceeding of the IEEE International Conference on Robotics and Automation*, Detroit, MI, USA, May 1999, pp. 2261–2266.
13. Y.J. Zhao, R. Govindan, and D. Estrin, “Residual energy scan for monitoring sensor networks,” in *Proceeding of the IEEE Wireless Communications and Networking Conference*, Orlando, Florida, USA, March 2002, pp. 356–362.
14. X. Luo, M. Dong, and Y. Huang, “On distributed fault-tolerant detection in wireless sensor networks,” *IEEE Transactions on Computers,* vol. 55, pp. 58–70, January 2006.
15. J. Polastre, J. Hill, and D. Culler, “Versatile low power media access for wireless sensor networks,” in *the 2nd ACM International Conference on Embedded Networked Sensor Systems,* Baltimore, USA, November 2004, pp. 95–107.
16. B. Krishnamachari and S. Iyengar, “Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks,” *IEEE Transactions on Computers,* vol. 53, pp. 241–250, March 2004.

17. B.V. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications," in *Proceedings of the IEEE,* vol. 85, pp. 24–38, January 1997.
18. M.L. Sichitiu and V. Ramadurai, "Localization of wireless sensor networks with a mobile beacon," in *Proceeding of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, Pasadena, CA, USA, May 2004, pp. 174–183.
19. A.P. Dempster, "A generalization of Bayesian inference," *Journal of the Royal Statistical Society,* pp. 205–247, March 1968.
20. G. Shafer, *A Mathematical Theory of Evidence* Princeton University Press, 1976.
21. A.J. Pinto, J.M. Stochero, and J.F. Rezende, "Aggregation-aware routing on wireless sensor networks," in *Proceeding of the 9th International Conference on Personal Wireless Communications*, Netherlands, September 2004, pp. 238–247.
22. B. Yu, J. Giampapa, S. Owens, and K. Sycara, "An evidential model of multisensor decision fusion for force aggregation and classification," in *Proceeding of the 8th International Conference on Information Fusion*, Philadelphia, USA. 25–29 July 2005, pp. 8–13.
23. G. Indranil, D. Riordan, and S. Srinivas, "Cluster-head election using fuzzy logic for wireless sensor networks," in *Proceeding of the 3rd Annual Communication Networks and Services Research Conference*, Halifax, Novia Scotia, Canada, May 2005, pp. 255–260.
24. D.S. Friedlander and S. Phoha, "Semantic data fusion for coordinated signal processing in mobile sensor networks," *International Journal of High Performance Computing Applications,* vol. 14, pp. 235–241, April 2002.
25. L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proceeding of the 4th International Symposium on Information Processing in Sensor Networks*, Los Angeles, California, USA, April 2005, pp. 63–70.
26. L. Xiao, S. Boyd, and S. Lai, "A space-time diffusion scheme for peer-to-peer least-squares estimation," in *Proceeding of the 5th International Conference on Information Processing in Sensor Networks*, Nashville, TN, USA, April 2006, pp. 168–176.
27. Y. Chin-Lung, S. Bagchi, and W. J. Chappell, "Location tracking with directional antennas in wireless sensor networks," in *Proceeding of the IEEE International Microwave Symposium Digest*, Long Beach, CA, USA, June 2005, pp. 4–10.
28. R.E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering,* vol. 3, pp. 35–45, May 1960.
29. R. Olfati-Saber, "Distributed Kalman Filter with Embedded Consensus Filters," in *Proceeding of the 44th IEEE Conference on Decision and Control*, Seville, Spain, December 2005, pp. 8179–8184.
30. M. Bedworth and J. O'Brien, "The Omnibus model: a new model of data fusion?," *IEEE Aerospace and Electronic Systems Magazine,* pp. 30–36, April 2000.
31. B. Frankel, "Control, estimation and abstraction in fusion architectures: Lessons from human data processing," in *Proceeding of the 3rd International Conference on Data Fusion,* Paris, France, May 2000, pp. 3–10.

http://www.springer.com/978-1-4614-1349-3

Resource-Aware Data Fusion Algorithms for Wireless
Sensor Networks
Abdelgawad, A.; Bayoumi, M.
2012, XVI, 108 p., Hardcover
ISBN: 978-1-4614-1349-3