Information Fusion Using Ontology-Based Communication between Agents

Tarek Sobh
Information Systems Department, Egyptian Armed Forces, Egypt

Abstract: *The distribution of on-line applications among network nodes may require obtaining acceptable results from data analysis of multiple sensors. Such sensors data is probably heterogeneous, inconsistent, and of different types. Therefore, multiple sensor data fusion is required. Here, there are many levels of information fusion (from low level signals to high level knowledge). Agents for monitoring application field events could be used to dynamically react to those events and to take appropriate actions. In a dynamic environment even a single agent may have varying capabilities to sense that environment. The situation becomes more complex when various heterogeneous agents need to communicate with each other. Ontologies offer significant benefits to multi-agent systems. The benefits as such are interoperability, reusability, support for multi-agent systems development activities such as system analysis and agent knowledge modeling. Ontologies support multi-agent systems operations such as agent communication and reasoning. The proposed agent based model in this paper can afford a promising model for obtaining acceptable information in case of multiple sensors.*

Keywords: *Information fusion, ontology, multi-agent system, agent communication language*

Received December 3, 2008; Accepted March 25, 2009

1. Introduction

Most modern organizations have a number of on-line applications and multiple data sources, distributed among the nodes of their information systems networks. In these applications, information fusion processes exploit a dynamic target situation picture produced by multi-sensor fusion, combining its information with relevant a priori information, in order to refine and interpret a battlefield situation picture [3, 4, 13]. Ultimately, this semi-automatic intelligence interpretation process aims at delivering a comprehensive picture of the opponents' options and, based on an evaluation of these options; suggest their likely intentions [3, 4, 13].

In real world applications software agents often have to be equipped with higher level cognitive functions that enable them to reason, act and perceive in changing, incompletely known and unpredictable environments [11, 12]. One of the major tasks in such circumstances is to fuse information from various data sources.

There are many levels of information fusion, ranging from the fusing of low level sensor signals to the fusing of high level, complex knowledge structures [11]. In a dynamically changing environment even a single agent may have varying abilities to perceive its environment which are dependent on particular conditions. The situation becomes even more complex when different agents have different perceptual capabilities and need to communicate with each other.

The proposed model integrates methods related to different fusion "levels" [2], specially multi-sensor and reactive multi-sensor management. The information fusion methodology integrated into proposed model rests on a few basic principles, i.e., cooperation between methods on different fusion levels with a coupling between a qualified synthetic environment and models of sensor behavior and communication between agents.

Also, reasoning about data with uncertainty is necessary in a growing number of applications and especially for such applications where the basic data sources correspond to sensors of multiple types and where the generated data are heterogeneous [14, 15, 16]. Data from such sensor sources are often associated with some level of uncertainty. To obtain acceptable results from the analysis of data from multiple sensors, multiple sensor data fusion is required. For this reason Chang [14] introduced acquired spatial/temporal information from the sensors. It was accomplished by means of a spatial/temporal query language.

The paper is structured as follows: Section 2 explains some definitions about data/information fusion, explains some background information about agent environment and some basic definitions. Section 3 illustrates the proposed model idea and architecture and finally section 4 contains conclusion.

2. Background

Data fusion according to Joint Directors of Laboratories (JDL) is a multifaceted process dealing with the automatic detection, association, correlation, estimation, and combination of data and information from single and multiple sources [2, 5].

Figure 1 comprises four levels, which form a hierarchy of processing. Level 1 is object refinement the product from this level is situation picture. Level 2 is situation refinement the product from this level is situation assessment. Level 3 is threat refinement the product from this level is threat assessment. Level 4 is process refinement the product from this level is performance assessment. It is clear from this model Information fusion, should be above the sensor level, to include situation refinement, impact assessment and process refinement [6, 7].

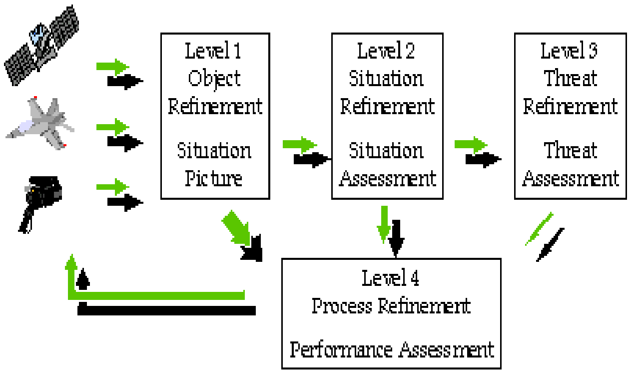

Figure 1. Levels of Data Fusion

Level 1, is performed "object refinement", which is an iterative process of fusing data to determine the identity and other attributes of entities and also to build tracks to represent their behavior. The term entity refers here to a distinct object. A track is usually directly based on detections of an entity, but can also be indirectly based on detecting its actions (i.e., tracking).

Functions of level 1 are data alignment, association, tracking, and identification. Data alignment means project data into a common reference frame. Association is sort or correlate observations into groups, each group represents single entity. Tracking refers to the estimation of the position and velocity of the entity. Identification seeks to better identify/ describe the entity. Level 2 performs "situation refinement", which is an iterative process of fusing the spatial and temporal relationships between entities to group them together and form an abstracted interpretation of the patterns in the order of battle data. Level 3 performs "threat refinement", which is an iterative process of fusing the combined activity and capability of enemy forces to infer their intentions and assess the threat that they pose. Level 4 performs "process refinement", which is an ongoing monitoring and assessment of the fusion process to refine the process itself and to regulate the acquisition of data to achieve optimal results. Level 4 interacts with each of the other levels.

Another definition of data fusion is as follows. Data fusion is a framework in which are expressed the means and tools for the amalgamation or alliance of data originating from different sources. It aims at obtaining best information; depending upon the business area. In some cases, information quality may be replaced by information efficiency. This definition emphasizes on the framework and on the fundamentals in remote sensing underlying data fusion instead of on the tools. Note that in this definition, the different observation modalities of one sensor, e.g., multispectral channels are to be considered as different sources [6, 7].

2.1. Data Fusion Domain

Efficient management of resources (i.e., sensing, computing, and communications) through data fusion domain is important to the success of on-line applications with multiple data sources. Traditionally, resource management in a data fusion process is treated as a separate component that is part of the so-called level 4 process refinement as defined in the JDL [2, 5] data fusion process model as shown in Figure 2. Consequently, the designs of resource management techniques are focused on ensuring the timely delivery and processing of data/information to support the data fusion processes for source processing and level 1–3 fusions.

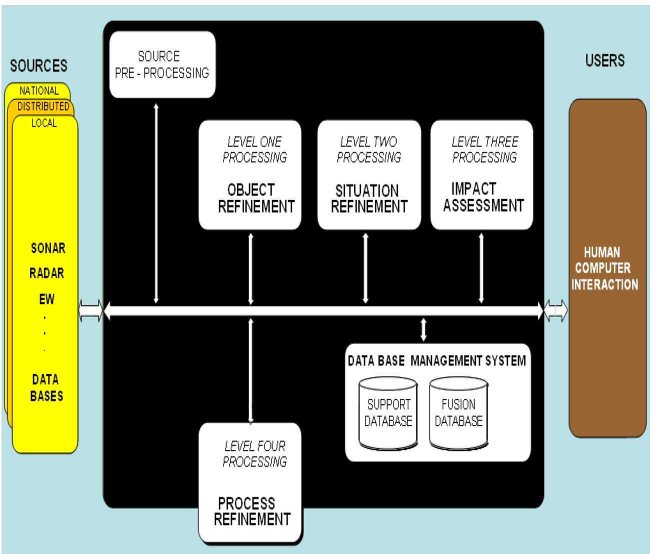

Figure 2. Data Fusion Levels and Resource Management

2.2. Properties of Data Fusion

In this section data fusion properties are introduced under the assumptions the sources of information are aligned and associated.

1. Fusion of attributes

   Fusion of attributes consists of merging the attributes of a same object, derived from two representations $(XS(1))^t$ and $(XS(2))^t$ at instant t obtained by means of the sources of information S(1) and S(2), in order to obtain new attributes in the space of sources S = S(1) U S(2).

2. Fusion of analysis
   Fusion of analysis consists of aggregating representations $(XS(1))^t$ and $(XS(2))^t$, into a new representation $(XS)^t$, then in generating an analysis or interpretation of the object for further use at instant (t+1), or at step i in an iterative process.
3. Fusion of representations
   Fusion of representations is defining and performing meta-operations applicable to representations $(XS(1))^t$ and $(XS(2))^t$, to obtain a new representation $(XS)^t$. Fusion of representations includes fusion of decisions. This fusion of representations may be performed at any moment, i.e. combined with other types of fusion.

This implies that fusion may operate at any of the three semantic levels: measurements (fusion of measurements), attributes (fusion of attributes) and rules (fusion of decision or rules), with possible crossings between levels. These properties take impact on the design of the architecture of a fusion system, on the selection of tools, suite of software and hardware (processing issues), communications (topological issues) and on the design of innovative procedures.

2.3. Definitions

An agent is an encapsulated computer system, situated in some environment, and capable of autonomous action in that environment in order to meet its design objectives [8]. Sometimes we said that Intelligence is to have the ability to solve problems and another thinking is consider any machine or program to be intelligent if it can act like human (turing test).

An intelligent agent is an encapsulated computer system, situated in some environment, and capable of flexible autonomous action in that environment in order to meet its design objectives. But sometimes humans take wrong decisions so we need another definition for agent. An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future. Autonomy means that agents are able to act without the intervention of humans or other systems: they have control both over their own internal state, and over their behavior. Mobile agent is a program that can migrate from a starting host to many other hosts in a network of heterogeneous computer systems and fulfill a task specified by its owner. During the self-initiated migration, the agent carries all its code and data, and in some systems also some kind of execution state. Multi-agent System (MAS) is such a system that consists of a group of agents that can potentially interact with each other. MAS characteristics are:

- Agents must communicate with each others.
- Agents must achieve their goals.
- Agents must have a strategic plan.

MAS Environments:

- Multi-agent environments must provide an infrastructure specifying how communications and interactions protocols will be.
- Multi-agent environment are most probably open and have no centralized design.
- Multi-agent environments contain agents that are autonomous and distributed, and may be self-interested or cooperative.

MAS environment provides an infrastructure specifying the regulations that agents follow to communicate and to understand each other, thereby enabling knowledge sharing [8].

Infrastructure deals with the following aspects: ontology, communication protocols and interaction protocols. Ontology is simply a specification of the objects, concepts, and relationships in an area of interest [18]. Communication protocols enable agents to exchange and understand messages. Interaction protocols enable agents to have conversations, which for our purposes are structured exchanges of messages.

The subject of ontology is the study of the categories of things that exist or may exist in some domain [20]. Ontology denotes an artifact that is designed for a purpose, which is to enable the modeling of knowledge about some domain, real or imagined. It is a pair O=(S, A) where S is an ontological signature (terms that lexicalize concepts and the relations between concepts) and A is an ontological axioms (restricting the intended meaning of the terms included in the signature) [19]. Also, we can say ontology is a catalog of the types of things that are assumed to exist in a domain of interest *D* from the perspective of a person who uses a language *L* for the purpose of talking about *D*. The types in the ontology represent the predicates, word senses, or concept and relation types of the language *L* when used to discuss topics in the domain *D*. An uninterpreted logic, such as predicate calculus, conceptual graphs, or KIF, is ontologically neutral. It imposes no constraints on the subject matter or the way the subject may be characterized. By itself, logic says nothing about anything, but the combination of logic with an ontology provides a language that can express relationships about the entities in the domain of interest. An **informal ontology** may be specified by a catalog of types that are either undefined or defined only by statements in a natural language [20]. A formal ontology is specified by a collection of names for concept and relation types organized in a partial ordering by the type-subtype relation. Formal ontologies are further distinguished by the way the subtypes are distinguished from their supertypes: an axiomatized ontology distinguishes subtypes by axioms and definitions stated in a formal language, such as logic or some computer-oriented notation that can be translated to logic; a prototype-based ontology distinguishes subtypes by a comparison with a typical member or prototype for each subtype [20].

Tom Gruber [19] said “ontologies are part of the W3C standards stack for the semantic web, in which they are used to specify standard conceptual vocabularies in which to exchange data among systems, provide services for answering queries, publish reusable knowledge bases, and offer services to facilitate interoperability across multiple, heterogeneous systems and databases. The key role of ontologies with respect to database systems is to specify a data modeling representation at a level of abstraction above specific database designs (logical or physical), so that data can be exported, translated, queried, and unified across independently developed systems and services. Successful applications to date include database interoperability, cross database search, and the integration of web services”.

3. The Proposed Model

The main objective of the proposed model in this work is to share information in real-time. Also, the proposed model emphasizes integrated fusion architectures for handling diversity of input sources rather than mathematical foundation only of information fusion. It is designed as a multi-agent system. This model consists of:

1. Data Warehousing for integrating schema
2. A spatial/temporal query agent to provide the retrieval and fusion of different information from real-time sources and existing databases. Textual query interface will be provided for users based on an SQL-Like query language, which allows the users to specify powerful spatial/temporal queries for multiple data sources [1].
3. The reasoner agent that requires information from ontology. Results from the reasoner agent can, however, are handled in various ways [11] [14] [18]. If the result has a high degree of certainty it is returned to the user. Otherwise a request is sent to the on-line application sensors to collect more information. It includes some element of fusion since data from new data sources must should be considered and integrated.
4. Some Data / Knowledge Stores

The prototype, as such, could be easily used to obtain a large-scale distributed intelligent system that can exploit both quantitative and qualitative data and/or information. This model considering the existing aspects of information fusion functions, object assessment and partly situation assessment are the two aspects of fusion that are jointly carried out by spatial/temporal query agent and reasoner agent. The system agents will communicate using KQML. Since KQML provides ontology, it represents an appropriate choice for specifications unification, concept integration and information fusion.

This model considering the existing aspects of information fusion functions, object assessment and partly situation assessment are the two aspects of fusion that are jointly carried out by spatial/temporal query agent and reasoner agent.

The proposed system, Figure 3, aims at providing information fusion by making use of ontology-based communication massages between agents. It consists of four levels real time data warehousing, spatial/temporal query agent, reasoner agent, and data / knowledge stores.

A spatial/temporal query agent acts as query processor. The query processor first performs a query to produce some initial results. If the initial results are uninformative then the reasoner agent guided by the user creates a more elaborate query by means of some rule and returns the query to the query processor. The query processor executes it and returns a more informative answer. Rules may be initially specified by the user and subsequently learned by the reasoner agent.

This model intended to handle objects that are uninformative belief values. To accomplish this, any information peculiar to an object in the application domain must be stored in the ontology, and any information peculiar to spatial/temporal reasoning must be stored in the rule base.

In a dynamically changing environment even a single agent may have varying abilities to perceive its environment which are dependent on particular conditions. The situation becomes more complex when different agents have different capabilities and need to communicate with each other.

This model proposes a framework that provides agents with the ability to fuse both low and high level approximate knowledge in the context of dynamically changing environments while taking account of heterogeneous and contextually limited perceptual capabilities.

3.1. Data Warehousing (Schema Integration)

The amount of digital information that is recorded and stored from sensor data in the distributed on-line applications such as tactical battlefield applications has been increasing at a tremendous rate [6, 7]. Common data formats for storage include commercial relational database engines. The one common element among all these applications is the fact that they must make use of data of multiple types and origins in order to function most effectively. This need emphasizes the demand for integration tools that allow such applications to make effective use of diverse data sets by supporting the querying of tailored information subsets.

In data warehousing, there is an initial setup phase during which relevant information is extracted from different sensor data sources, transformed and cleansed as necessary, fused with information from other sources, and then loaded into the proposed centralized data source.

In the proposed model data warehouses is a real time. Data warehouses at this stage are updated on a transaction or event basis, every time an operational system performs a transaction.

3.2. Spatial/temporal Query Agent

In the spatial/temporal query agent the query processor performs a query to produce some initial results. If the initial results are uninformative then the Reasoner agent creates a more elaborate query by means of some rule and returns the query to the spatial/temporal query agent. The query processor executes it and returns a more informative answer. Rules may be initially specified by the user and subsequently learned by the Reasoner agent.

The result of a query processor is generally the object types requested by the user including the attributes (e.g., color, size, *etc*.) and status values of these objects (e.g., position, orientation or speed, *etc*.), if requested. The difference between an attribute and a status value is basically that an attribute is not subject to change in the short range of time, that is, the color of a vehicle may change but not within the time frame of concern to the user. Status values may change within a very short time frame that may be less than seconds, such as the position and speed of a vehicle. The motivation for these two categories is to allow the system to use the attributes for object recognition and the status variables for reasoning about the object behavior.

Query processor returns all the information of the type attribute associated with a belief value. The belief values are the result of a matching process in the object recognition process. In the most general case the belief values are just given for the object types and from each type of sensor data and eventually there is also a belief value given as a result of the fusion process that takes place for the majority of the queries; this is due to the use of multiple sensor data sources. To determine the source data quality for a certain area of interest the corresponding meta-data will be required.

3.3. Reasoner Agent

In this model, an interactive reasoner agent with learning capability is presented. Results from the reasoner agent can, however, are handled in various ways. If the result has a high degree of certainty it is

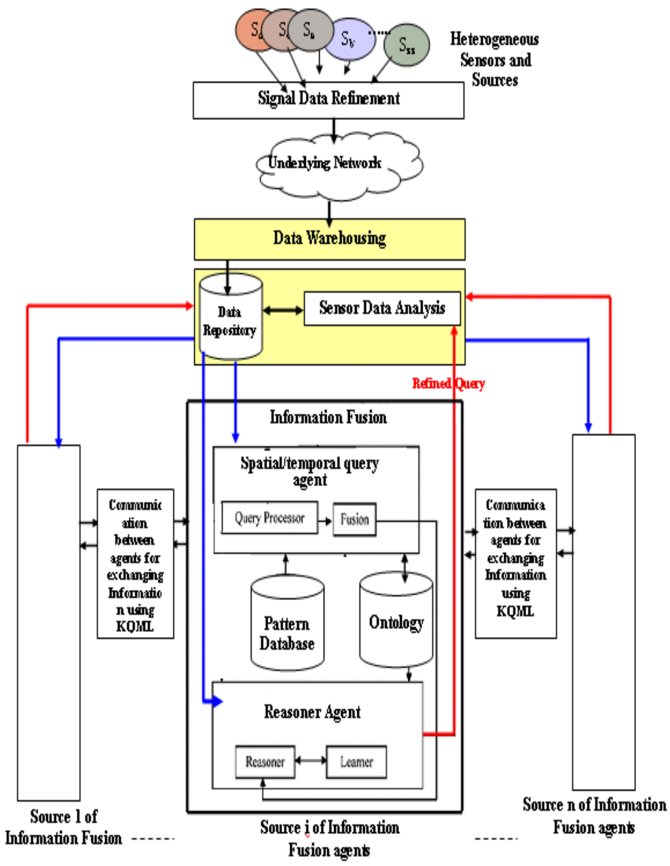

Figure 3. The proposed system

returned to the user [9, 17]. Otherwise a request is sent to the sensors to collect more information. This is followed by a rerun of the query. Besides this, background information may be accessed from other sources. Reasoner agent will take care of the query results that may be difficult to interpret while considering the context-dependent background information. Reasoner agent must also include some element of fusion since data from additional data sources should also be considered and integrated.

The reasoner agent accepts the output from the query agent, and selects a reasoning rule. The reasoner selects an applicable rule *R* on space *S*, which is the cartesian product of the sub-spaces. Sub-spaces include 1) sources, 2) objects to be recognized, 3) attributes of objects, 4) time, 5) location and 6) semantic relations.

3.4. Data / Knowledge Stores

In this section three different types of data/knowledge stores have been identified pattern database, meta database, and ontology. Here three tasks of data/knowledge stores 1) is concerned with how to improve the result of the original query by considering other aspects of the sensor data sources, 2) is concerned with how to associate different object instances with each other, and 3) is concerned with queries that need to inspect the dependency tree from the last user query. All previous tasks are related to iterative information fusion process that also includes empirical learning.

3.4.1. Pattern Database

In the proposed model we describe the typical tasks for information fusion, which can be grouped into different types. The query patterns are retrieved dynamically. The reasoning process as carried out here depends on whether the belief values that are output from any user query has got a value that is uninformative. The reasoner agent will match the query patterns in the set of patterns with the current query. The matched query patterns are sent to the query processor as well as the query result. Finally query patterns are synthesized by learner and saved in the pattern database.

Initially the system has no query patterns in the query pattern set. Once a query is completed, then a new query pattern is generated and appended to the query pattern set.

3.4.2. Meta Database

Meta data is needed for two reasons. First, meta data determines the background information. Second, it is used to determine the source data quality for a certain area of interest the corresponding meta-data will be required.

Input to information fusion reasoning step is mainly the output that may include the dependency tree information, from the query processor, the context information and the metadata. The meta-data is used to select the portion of the context information that corresponds to the area of interest.

3.4.3. Ontology

Besides being rule driven the reasoner agent also requires information from ontology. Any information peculiar to an object in the application domain must be stored in the ontology, and any information peculiar to spatial/ temporal reasoning must be stored in the rule base. The ontology basically is used for logical description of all possible data sources that can be used to deliver input data to the queries. Figure 4 presents ontology-based fusion use case diagram.

Ontology mapping can be defined as morphism from ontology to another. It is a collection of functions assigning the symbols used in one vocabulary to the symbols of the other [21].

$O_1 = (S_1, A_1)$ **to** $O_2 = (S_2, A_2)$

A morphism: f: $S_1 \rightarrow S_2$ such that $A_2 \vDash f(A_1)$,

A2 is a set of binary relations between the ontological signatures inclusion (⊑) and equivalence (≡) relations in this case all interpretations that satisfy O_2's axioms also satisfy O_1's translated axioms

Ontology alignment is the task of establishing a collection of binary relations between the vocabularies

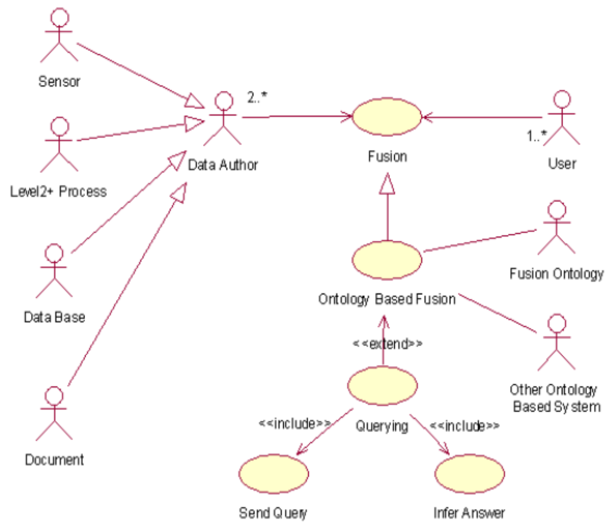

Figure 4. Ontology-based fusion use case diagram [10].

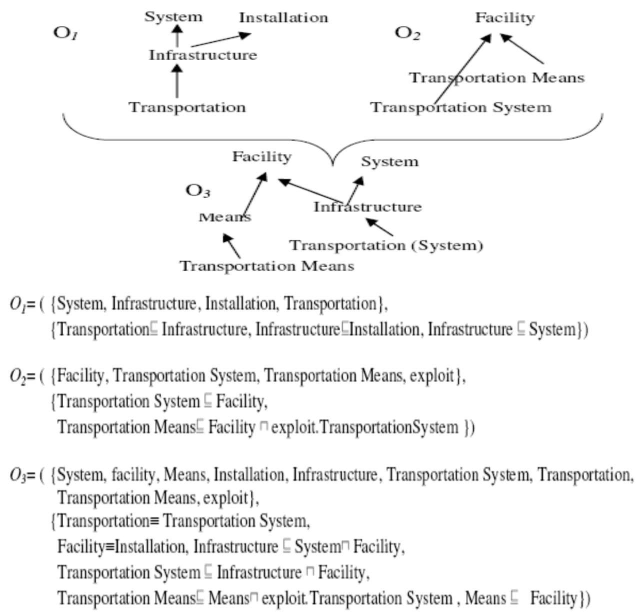

Figure 5. Ontology alignments [21].

of two ontologies, i.e., pairs of ontology mappings. Here ontology alignment articulates a set of binary relations (inclusion (⊑) and equivalence (≡)) between the ontological signatures (i.e. Signatures imply an alignment of the two ontologies). Instead of aligning two ontologies "directly" through their signatures, we may specify the alignment of two ontologies O_1 and O_2 by means of a pair of ontology mappings from intermediate source ontology O_3 as shown in Figure 5.

Ontology matching is the computation of similarity functions towards discovering similarities between ontology concepts or/and properties pairs using combinations of lexical, structural, and semantic knowledge.

4. Conclusion

In this work we tried to introduce a clear view of the information fusion using ontology based communication between agents. Having, this view of the problem, may give you an effective solutions to the problem of data fusion from multiple sensors.

This paper presents an ontology-based system using multi-agent. Spatial/temporal query agent performs a query to produce some initial results. Reasoner agent can be viewed as a progressive query language processor, since the input to this agent is the output from a query processor to process queries for data from multiple sensors.

Also, in this paper, we have proposed a framework to fuse both low and high level approximate knowledge in the context of dynamically changing environments as well as heterogeneous and contextually limited perceptual capabilities.

Finally, the proposed model provides a first step towards such an information fusion infrastructure. It has possible to be an environment capable to easy integration of new tools and modifying the behavior of some other existing tools to provide suitable analysis to various data sources.

References

[1] Narayanan A., Shaman T., "Iconic SQL: practical issues in the querying of database through structured iconic expressions, "*Journal of Visual Languages and Computing* 13 (6) (2002).

[2] Steinberg A., Bowman C., "Revisions to the JDL data fusion model, " in: D.L. Hall, J. Llinas (Eds.), Handbook of Multisensor Data Fusion, CRC Press, Boca Raton, FL, USA, 2001, pp. 2:1–19 (Chapter 2).

[3] DSTO (Defence Science and Technology Organization), "Data Fusion Special Interest Group, Data fusion lexicon". Department of Defence, Australia, 7 p., 21 September 1994.

[4] Hall D., "Mathematical Techniques in Multisensor Data Fusion", Artech House, Boston, London, 1992.

[5] JDL U.S. Department of Defense, "Data Fusion Subpanel of the Joint Directors of Laboratories", Technical Panel for C3, Data fusion lexicon, 1991.

[6] Klein L.. "Sensor and data fusion concepts and applications" Tutorial texts, vol. TT 14, SPIE Optical Engineering Press, USA, 131 p., 1993.

[7] Pau L., "Sensor data fusion". Journal of Intelligent and Robotics Systems, vol. 1, pp. 103-116, 1988.

[8] Zaki M., and Sobh T., "A Cooperative Agent-Based Model for Active Security Systems". The Journal of Network and Computer Applications, Volume 27, Issue 4,, pp. 201-220. November (2004)

[9] Zaki M., and Sobh T., "NCDS: Data Mining for Discovering Interesting Network Characteristics, " Information and Software Technology, vol. 47, Issue 3, Pages 189-198. (2005)

[10] Kokar M., Matheus C., Baclawski K., Letkowski J., Hinman M., and Salerno J., "Use Cases for Ontologies in Information Fusion, " *In Proc. Seventh Intern. Conf. Info. Fusion*, pages 415-421. (2004)

[11] Doherty P., Lukaszewicz W., and Szałas A., "Communication between agents with heterogeneous perceptual capabilities, " *Information Fusion*, vol. 8, 56–69, 2007.

[12] Doherty P.,Łukaszewicz W., Szałas A., "Approximative query techniques for agents using heterogenous ontologies, " *in: D. Dubois, C. Welty, M.-A. Williams (Eds.), Proceedings on the 9th International Conference on Principles of Knowledge Representation and Reasoning*, KR_2004, AAAI Press, 2004, pp. 459–468.

[13] Ahlberg S., Horling P., Johansson K., Jored K., Kjellstrom H., Martenson C., Neider G., Schubert J., Svenson P., Svensson, and Walter J., "An information fusion demonstrator for tactical intelligence processing in network-based defense, " *Information Fusion* 8 (2007) 84–107

[14] Chang S., Jungert E., and Li X., "A progressive query language and interactive reasoner for information fusion support, " *Information Fusion*, vol. 8 (2007) 56–69.

[15] Chang S., Jungert E., and Costagliola G., "Querying distributed multimedia databases and data sources for sensor data fusion, " *IEEE Transactions on Multimedia* 6 (5) (2004) 672–687.

[16] Chang S., Costagliola G. and ,Jungert E., "Multisensor Information Fusion by Query Refinement, "*Recent advances in Visual information systems, in: S.K. Chang, Z. Chen, S.-Y. Lee (Eds.), Lecture Notes in Computer Science*, vol. 2314, Springer-Verlag, Berlin, 2002, pp. 1–11.

[17] Sobh t., "Explanation-based learning to recognize network malfunctions, " *Information Knowledge Systems Management,* vol. 5, Pages 1–21, IOS Press, (2005/2006)

[18] Horney T., Jungert E., and Folkesson M., "An ontology controlled data fusion process for query language", *in: Proceedings of the International Conference on Information Fusion* (*Fusion_03*), Australia, 2003.

[19] http://tomgruber.org/writing/ontology-definition-2007.htm, Gruber T., to appear in the *Encyclopedia of Database Systems*, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2008,

[20] http://users.bestweb.net/~sowa/ontology/index.htm
[21] http://www.icsd.aegean.gr/kotis/

Tarek Sobh received his B.Sc. degree in computer engineering from Military Technical College, Cairo, Egypt in 1987. Both M.Sc. and Ph.D. degrees from Computer and System Engineering Department, Faculty of Engineering, Al-Azhar University, Cairo, Egypt. He has managed, designed and developed several package for business applications and security systems. He has authored/co-authored of many refereed journal/ conference papers and booklets. Some of the articles are available in the ScienceDirect Top 25 hottest articles. His research of interest includes computer networks, security systems, distributed systems, knowledge discovery, data mining, and software engineering.