Monitoring Boats in Marine Reserves: A MAS Solution

Andrea Addis
Dept. of Electrical and Electronic Engineering
University of Cagliari
Italy
addis@diee.unica.it

Giuliano Armano
Dept. of Electrical and Electronic Engineering
University of Cagliari
Italy
armano@diee.unica.it

Eloisa Vargiu
Dept. of Electrical and Electronic Engineering
University of Cagliari
Italy
vargiu@diee.unica.it

ABSTRACT

The application of agent methodologies in process monitoring and control is a relatively novel approach, particularly suitable for distributed and dislocated systems. As for marine reservers, access monitoring is aimed at avoiding intrusions of unauthorized boats – also considering that, typically, marine reserves are located in areas not easily accessible. Typical solutions consist of using radar systems or suitable cameras activated by movement sensors. In this paper, we present a multiagent system aimed at monitoring boats in marine reserves. The goal of the proposed system is to discriminate between authorized and unauthorized boats – the formers being equipped with GPS+GSM devices. Boats are tracked by a digital radar that detects their positions. The system has been used to monitor boats in a marine reserve located in the North of Sardinia. Results show that adopting the proposed approach facilitates the system administrator, as well as staff operators, in the task of identifying intrusions.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*multiagent systems*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering*

General Terms

marine reserves, multiagent systems

1. INTRODUCTION

In the summertime, in Sardinia and in its small archipelago a lot of tourists often sail in protected or forbidden areas and/or close to the coast. Monitoring such areas is quite complicated since the corresponding scenario requires to discriminate between authorized and unauthorized boats.

Along Sardinia coasts, there are two-hundred tourist harbors with about thirteen thousand places available for boats and several services for boat owners. Small ports typically have to monitor large areas in order to guarantee the access to authorized boats without suitable resources (e.g., radars). In these areas, staff operators have to directly patrol the surface in an uneconomic way. A typical solution consists of using a radar system controlled by a central unit located ashore in a strategical position. Radar signals allow to detect the positions of the boats that sail in the controlled area. The main problem is that it is needed to distinguish among authorized and unauthorized boats.

Since the multiagent technology may help in deploying real applications both from a software engineering [11] and a technological perspective [4], we propose a multiagent approach to monitor boats in marine reserves. To our knowledge no agent-based solutions have been proposed in the literature to monitor and signal intrusion in marine reserves. The system has been implemented by using X.MAS, a generic multiagent architecture devised to implement information retrieval and information filtering applications [2].

The corresponding scenario requires to discriminate between authorized and unauthorized boats – the formers being equipped with GPS+GSM devices. Boats are tracked by a digital radar that detects their positions. The system has been used to monitor boats in a marine reserve located in the North of Sardinia. Results show that adopting the proposed approach allows system administrator and staff operators to easily identify intrusions.

The remainder of the paper is organized as follows: first, selected related work on agent-based systems for monitoring and surveillance are presented. Subsequently, the proposed MAS solution is described, focusing on the corresponding macro-architecture and on agent capabilities that have been implemented. The current deployed prototype is then presented together with some experimental results. Conclusions end the paper.

2. RELATED WORK

Applying agent methodologies in process monitoring and control is a relatively new approach, particularly suitable for distributed and dislocated systems. In particular, agent-based solutions have been proposed to monitor: (i) control systems [6], (ii) distant control experimentation systems [10], and forest fires [5].

In [6], a methodology for designing agent-based production control systems, which can be successfully applied by an engineer with no prior experience in agent technology, has been proposed. Authors propose a design method for identifying the agents of a production control system. The identification of agents allows the designer to move from pure domain concepts (such as production processes), to agent-oriented concepts (such as agents and decision responsibilities). In addition, the identification of agents provides a basis for all other subsequent design steps, such as interaction design or agent programming.

In [10], an approach to the realization of distant control experimentation system has been described, the main task being to monitor temperature and humidity and to control experiments with the possibility of video monitoring of the experimental area. The underlying idea is to integrate an existing laboratory greenhouse model and an existing video system. A multiagent system is then developed that involves five specific software agents. Agents are responsible of all operations: from user-system communication to telecontrol and video monitoring.

In [5], a TCP/IP-based system conceived of sensor networks, central server units for collecting, processing and storing all data, has been presented. Each sensor network has several monitoring units, each including: (i) pan, tilt, zoom-controlled video camera connected to a network-embedded video web server, (ii) mini meteorological stations connected to network-embedded data web servers, and (iii) a wireless communication unit. Agents designed for forest fire monitoring strictly follow these guidelines, each module of the system being autonomous, aware of its environment and capable of active behavior if alarmed. Environment awareness is accomplished by connecting numerous meteorological sensors to a network-embedded microcontroller unit. Network-embedded microcontroller units are responsible for collecting data from sensors, formatting and preprocessing it and passing it to the central server agent if needed.

Agent-based solutions have also been proposed to develop video surveillance systems. Video surveillance is an active area of research. In this field, researchers are concerned with detection and tracking based on a security issue. In particular, researchers are mainly interested in autonomous system configuration [1], object identification [9], and multimodal systems [8].

In [1], a video-based multi-agent system for traffic surveillance, called Monitorix, is presented. Agent interactions are controlled by a BDI-like architecture. Agents communicate using FIPA-ACL messages with SL contents. Each agent defines a set of predicates, functions and actions that may be referred in the messages that it receives. Vehicles are tracked across cameras by a suitable agent, using a traffic model whose parameters are continuously updated by learning algorithms. The classification of mobile objects uses competitive learning algorithms. The computation of typical trajectories uses statistical adaptation. The tracking of mobile objects, from one camera into the next one, updates the parameters of its prediction model, using a combination of symbolic learning and genetic algorithms.

In [9], an architecture for implementing scene understanding algorithms in the visual surveillance domain has been presented. The agent paradigm is adopted to provide a framework in which inter-related and event-driven processes can be managed in order to achieve a high level description of events observed by multiple cameras. Each camera has an associated agent, which detects and tracks moving regions of interest, and is calibrated to transform image co-ordinates into ground plane locations. By comparing properties, two agents can infer that they have the same referent, i.e. that two cameras are observing the same entity, and, as a consequence, merge identities. Agents store a hidden Markov model of learned activity patterns.

In [8], a coordinated video surveillance system is presented, which can minimize the spatial limitation and can precisely extract the 3D position of objects. The proposed system uses an agent based system and also tracks normalized objects using active wide-baseline stereo method. The system is composed of two parts: multiple camera agents (CAs) and a support module (SM). Each CA treats image processing and camera controlling. SM is devoted to manage communication between CAs. The system extracts object positions independent of environment via the collaboration of CAs and a SM.

3. THE PROPOSED MAS SOLUTION

As we are interested in monitoring and signaling intrusions in marine reserves, we decided to supply authorized boats with suitable devices able to transmit (through the GSM technology) their position (through the GPS technology). In this way, the corresponding scenario encompasses two kinds of boats: authorized, recognizable by the GPS+GSM devices, and unauthorized. Both kinds of boats will be identified by a digital radar able to detect the position of all boats located in the protected area. Comparing the positions sent by boats with those detected by the radar will easily help in identifying unauthorized boats, signaling intrusions to the staff operators. Furthermore, this approach allows communications from the central server to the boats (e.g., sending information about weather forecasting) or vice versa (e.g., issuing a request of assistance).

From a conceptual perspective, the problem of monitoring and signaling intrusions in marine reserves can be seen as an information retrieval task. In fact, a typical information retrieval task has to take into account the following issues:

i. how to deal with different information sources and to integrate new information sources without re-writing significant parts of it;

ii. how to suitably encode data in order to put into evidence the informative content useful to discriminate among categories;

iii. how to allow the user to specify her / his preferences;

iv. how to exploit the user feedback to improve the overall performance of the system.

As for the considered scenario, information sources are radar and GPS+GSM devices; the categories among discriminate with are authorized and unauthorized boats; user preferences depend on the specific role of the user (i.e., system administrator or staff operators); and the feedback could be adopted by the system administrator to signal possible errors in the intrusion detection. The above issues are typically strongly interdependent in information retrieval systems. To promote the decoupling among relevant aspects, we adopted the agent-based technology. In fact, resorting to a multiagent solution may help in dealing with the underlying distributed environment, which requires light-weight task allocation, flexibility, and scalability.

The adopted multiagent system consists of a society of software agents, each embodying heterogeneous characteristics and responsibilities. Each agent exhibits an intelligent behavior while providing a useful support to the task of boat monitoring. To this end, several kinds of agents have been devised: (i) agents aimed at extracting information from the information sources (i.e. the radar and the mobile devices); (ii) agents devoted to encode such information; (iii) cooperative agents that perform the role of domain experts and are aimed at following boats signaling intrusions; and (iv) interface agent through the user can interact with.

The proposed system, called SEA.MAS, has been devised by customizing X.MAS [2], a generic multiagent architecture built upon JADE [3], devised to make it easier the implementation of information retrieval and information filtering applications. For the sake of completeness, in this section we first summarize X.MAS (the interested reader may consult [2] for further information) and then illustrate its customization to monitor boats in marine reserves.

3.1 X.MAS

The X.MAS architecture encompasses four main levels: information, filter, task, and interface.

At the *information level*, agents are entrusted with extracting data from the information sources. Each information agent is associated to one information source, playing the role of wrapper.

At the *filter level*, agents are aimed at selecting information deemed relevant to the users, and at cooperating to prevent information from being overloaded and/or redundant. In general, two filtering strategies can be adopted: generic and personalized.

At the *task level*, agents arrange data according to users personal needs and preferences. Task agents are devoted to achieve user goals by cooperating together and adapting themselves to the changes of the underlying environment.

At the *interface level*, a suitable interface agent is associated with each different user interface. In fact, a user can generally interact with an application through several interfaces and devices (e.g., pc, pda, mobile phones, etc.).

At the *mid-span level*, agents are aimed at establishing communication among requesters and providers. In the literature, several solutions have been proposed: e.g., blackboard agents, matchmaker or yellow page agents, and broker agents (see [7] for further details). In X.MAS, agents at the mid-span level can be implemented as matchmakers or brokers, depending on the specific application.

X.MAS agents are JADE [3] agents that can (i) interact by exchanging FIPA-ACL messages, (ii) share a common ontology in accordance with the actual application, and (iii) exhibit a specific behavior according to their role. As for agent internals (see Figure 1), each agent encompasses a scheduler devoted to control the information flow between adjacent levels. Information and interface agents embody information sources and specific devices, respectively. Filter and task agents encompass an actuator that depends on the actual application. Middle agents contain a dispatcher aimed at handling interactions among requesters and providers.

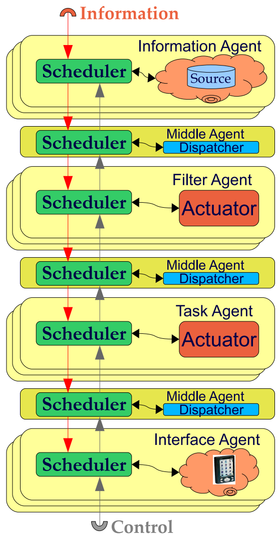

Figure 1: Agent internals.

3.2 SEA.MAS

Figure 2 sketches the macro-architecture of SEA.MAS, the implemented customization of X.MAS for monitoring boats in marine reserves.

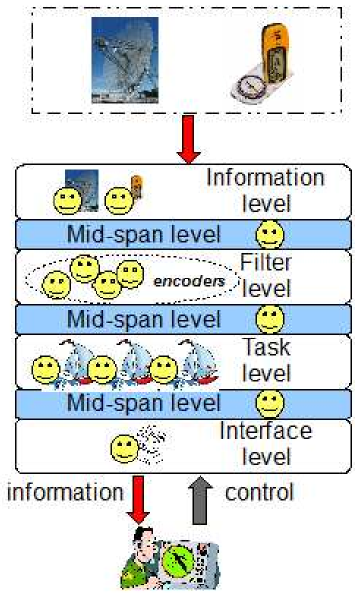

Figure 2: SEA.MAS macro-architecture

Information level. In this particular scenario, information sources are the digital radar and GPS+GSM devices. For each information source, a suitable information agent is devoted to embody the information provided therein. To this end, we implemented a wrapper for the digital radar and a wrapper for the GPS+GSM devices. The information generated by wrappers becomes available to the other agents of the system by invoking the middle agent corresponding to the middle-span level "Information-Filter".

Filter level. Filter agents are aimed at encoding the information extracted by the information agents, which is used to create events containing the position of the detected boats and their identification code, when available. Moreover, filter agents are devoted to avoid two kinds of redundancy: information detected more than once from the same device (caching) or throughout different devices (information overloading). Then, the middle agent corresponding to the middle-span level "Filter-Task" forwards the event to the corresponding task agent, assuming the role of yellow pages if the identification code is available or the role of broker otherwise. If a detected event is not related to an authorized boat, the middle agent creates a task agent able to handle the event.

Task level. Task agents are devoted to achieve user goals by cooperating together and adapting themselves to the changes of the underlying environment. A task agent corresponds to each boat, the underlying motivation being the necessity to centralize the knowledge regarding the position of a boat, its state and other further communication channels. As for the position, events are classified as follows: events belonging to anonymous detection systems and events belonging to known detection systems (i.e., with an identification code). As for the state, it depends on the identification code and/or on possible information provided by the corresponding boat. As for further communication channels, a bidirectional message exchange between the boats and the server could be also available (e.g., to provide weather information). The main tasks of the agents belonging to this architectural level are: (i) to follow a boat position during its navigation, also dealing with any temporary lack of signal; (ii) to promptly identify unauthorized boats alerting the interface agents; and (iii) to handle messages coming from the interface level in order to notify the involved devices.

Interface level. A suitable interface agent allows users to interact with the system. Final users are the system administrator and staff operators. In particular, the interface agent is aimed at providing user profiling and getting a feedback from the user –which can be exploited to improve the overall ability of discriminating among authorized and unauthorized boats. In this way, the interface agent will pass information from user to agents, for instance to notify the interested agents about changes occurred in the environment or about faulty that might occur in the devices located on the authorized boats. So far, user's feedback is performed through a simple solution based on the k-NN technology[1]. When either a false positive or a false negative is evidenced by the user, it is immediately embedded in the training set of the k-NN classifier that implements the feedback. A check performed on this training set after inserting the negative example allows to trigger a procedure entrusted with keeping the number of negative and positive examples balanced. In particular, when the ratio between negative and positive examples exceeds a given threshold (by default set to 1.1), some examples are randomly extracted from the set of "true" positive examples and embedded in the above training set.

Let us note that the corresponding system involves a number of agents that depends on the number of boats. In fact, whereas the number of middle agents (one for each middle-span level) and information-, filter-, and interface-agents is fixed (i.e., two information-, one filter, and one interface-agent), for each boat a task agent is instantiated. This is not a problem for two main reasons: agent can be distributed on several nodes, and, typically, marine reserves can host a maximum of 20 boats at a time.

As for the capabilities that X.MAS agents can exhibit in this particular scenario, the main ones are: (i) cooperation, (ii) mobility, (iii) personalization, (iv) adaptivity, and (v) deliberative capability.

Cooperation. Cooperation is the main requirement to be implemented in SEA.MAS agents. In particular, agents must cooperate to coordinate their actions in order to achieve their goals. In SEA.MAS cooperation may occur both horizontally and vertically. The former kind of communication supports cooperation among agents belonging to a specific level, whereas the latter supports the flow of information and/or control between adjacent levels through suitable middle-agents. Cooperation is implemented in accordance with the following modes: centralized composition, pipeline, and distributed composition: (i) centralized compositions can be used for integrating different capabilities, so that the resulting behavior actually depends on the combination activity; (ii) pipelines can be used to distribute information at different levels of abstraction, so that data can be increasingly refined and adapted to the user needs; and (iii) distributed compositions can be used to model a cooperation among the involved components aimed at processing interlaced information. Communication among agents is performed by the FIPA-ACL support provided by JADE.

Mobility. All involved agents can be mobile, if needed. In fact, in case of a large number of agents (i.e. boats) this requirement becomes mandatory in order to handle the computational complexity. Thus, mobility permits the runtime deployment of agents in a distributed architecture. Let us recall that X.MAS agents are in fact JADE agents; for this reason, it is very easy to build mobile agents able to migrate or copy themselves across a network –as JADE mobile agent can navigate across different containers and platforms. Of course, mobile agents must be location aware in order to decide when and where to move. Therefore, X.MAS provides a suitable ontology that holds the required concepts and actions.

Personalization. Personalization is provided to perform profiling at the interface level. In fact, the system is able to provide a different interface depending on the actual operator (e.g., system administrator and several kinds of staff operators). Information about user profiles is stored by agents belonging to the interface level, although to support personalization, filter and task agents may need to store it either. It flows up from the interface level to the other levels through the middle-span levels. In particular, agents be-

[1]The k-nearest neighbor is a classification method based upon observable features. The algorithm selects a set which contains the k nearest neighbors and assigns the class label to the new data point based upon the most numerous class with the set.

longing to mid-span levels (i.e., middle agents) take care of handling synchronization and avoiding potential inconsistencies. Moreover, the user behavior is tracked during the execution of the application to support explicit feedback, in order to improve her/his profile.

Adaptivity. Currently, SEA.MAS agents exhibit a trivial adaptive capability. Task agents, in fact, are able to adapt their behavior in order to avoid to lose boats in case of signal absence (i.e., areas devoid of GSM signal). In a future release of the system, we are planning to implement multiagent learning strategies, such as evolutionary computation, to allow task agents to self-adapt to further changes that may occur in the environment.

Deliberative capability. Currently, SEA.MAS agents do not exhibit deliberative capabilities. In a future release of the system, we are planning to implement reasoning algorithms that allow task agents to autonomously replan their actions if needed.

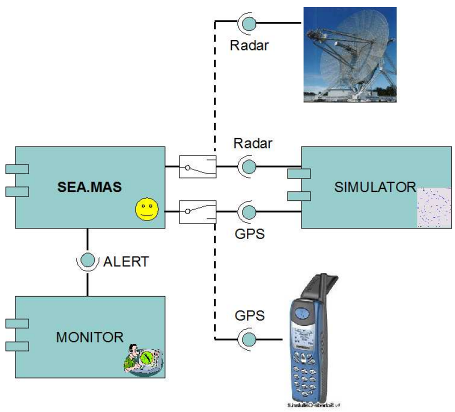

Figure 3: Schema of the overall system

4. EXPERIMENTS AND RESULTS

Figure 3 illustrates the components involved in the system. First, a simulator has been implemented for testing SEA.MAS. Such component simulates both the radar and the GPS+GSM devices. In particular, it simulates signals belonging to GPS and radar by randomizing temporal displacement and by adding detection errors. It is worth pointing out that the simulator mimics the real world behavior of boats, such as accelerations and changing course. The presence of GPS devices on the boats have also been simulated. To test the robustness of the approach, the prototype has been tested scaling up the number of boats from 20 to 100. Let us recall that, in the worst case (i.e., 100 boats) the system is composed by 107 agents. Performances put into evidence that, also considering 100 boats, the system is able to signal intrusions in a very short amount of time. To assess the capability of SEA.MAS in detecting intrusions, we performed experiments considering first a scenario with less than 40 boats, and then a scenario with a number of boats varying from 40 to 100. In the former scenario, being boats enough spaced out, the filter agent is able to easily distinguish among authorized and unauthorized boats and, consequently, task agents are correctly instantiated and able to signal the detected intrusions as soon as they occur. In the latter scenario, the signals provided by boats could overlap, and, as a consequence, the filter agent could not correctly distinguish among authorized and unauthorized boats. In this case, some task agents could be associated to unauthorized boats even if they actually correspond to an authorized one. To measure such error, we calculate the confusion matrix, where: true positives (TP) are intruders, true negatives (TN) are authorized boats, false positives (FP) are authorized boats recognized as intruders; and false negatives (FN) are intruders recognized as authorized boats. Starting from these values we then calculated accuracy (α), precision (π), and recall (ρ), whose definitions are reported below for the sake of completeness:

$$\alpha = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\pi = \frac{TP}{TP + FP} \quad (2)$$

$$\rho = \frac{TP}{TP + FN} \quad (3)$$

Experiments pointed out that, on average, system accuracy is 95%. As for precision and recall, the system exhibits the same propensity to make mistakes in FP and FN, on the other hand the overall error is quite low. On average, precision and recall are 93%. Furthermore, for both scenarios we tested the system varying the percentage between intruders (TP) and authorized boats (TN) examples. In both cases, the system is not influenced by those variations in the inputs, meaning that the behavior of the system is independent from the involved boats.

Once SEA.MAS has been tested, the simulator has been switched off and the system has been used to monitor boats in a marine reserve located in the North of Sardinia. GPS- and radar-signals are retrieved by suitable information agents that are able to extract the actual position according to GPS and NMEA standards. As for the interface agent (see Figure 4), it represents in different colors different states: authorized, unauthorized, not-detected, under verification. In case of intrusion, an acoustic sound is generated together with the position of the unauthorized boat(s). Such signal is sent to the security patrol, whose primary goal is to catch intruders. The maximum number of boats in the selected marine reserve was 20. Experiments performed on the real scenario showed results comparable with the ones performed during the simulation.

Summarizing, results show that adopting the proposed approach allows system administrator and staff operators to easily identify intrusions. Results are quite interesting, also considering that the system is down-market.

5. CONCLUSIONS

In this paper, SEA.MAS, a multiagent system for monitoring boats in marine reserves, has been presented. The system has been built upon X.MAS, a generic multiagent architecture devoted to support the implementation of information

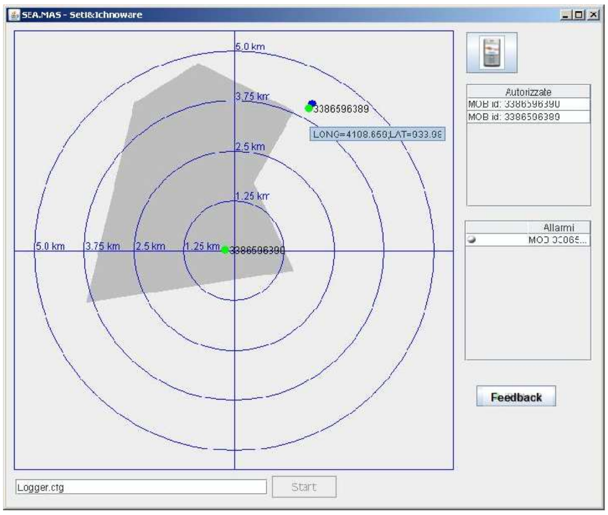

Figure 4: A snapshot of the user interface

retrieval and information filtering applications. The system has been used to monitor boats in a marine reserve located in the North of Sardinia. Results show that adopting the proposed approach allows the system administrator and staff operators to easily identify intrusions.

6. ACKNOWLEDGMENTS

This work has been supported by Regione Autonoma della Sardegna, under the project "A Multiagent System for Monitoring Intrusions in Marine Reserves" (POR Sardegna 2000/2006, Asse 3 - Misura 3.13). The prototype has been developed and deployed together with SETI snc and ICHNOWARE sas.

7. REFERENCES

[1] B. Abreu, L. Botelho, A. Cavallaro, D. Douxchamps, T. Ebrahimi, P. Figueiredo, B. Macq, B. Mory, L. Nunes, J. Orri, M. Trigueiros, and A. Violante. Video-based multi-agent traffic surveillance system. *Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE*, pages 457–462, 2000.

[2] A. Addis, G. Armano, and E. Vargiu. From a generic multiagent architecture to multiagent information retrieval systems. In *AT2AI-6, Sixth International Workshop, From Agent Theory to Agent Implementation*, pages 3–9, 2008.

[3] F. Bellifemine, G. Caire, and D. Greenwood. *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*. John Wiley and Sons, 2007.

[4] F. Bergenti, G. Rimassa, A. Poggi, and P. Turci. Middleware and programming support for agent systems. In *Proceedings of the 2nd International Symposium from Agent Theory to Agent Implementation*, pages 617–622, 2002.

[5] L. Bodrozic, D. Stipanicev, and M. Stula. Agent based data collecting in a forest fire monitoring system. *Software in Telecommunications and Computer Networks, 2006. SoftCOM 2006. International Conference on*, pages 326–330, 29 2006-Oct. 1 2006.

[6] S. Bussmann, N. R. Jennings, and M. Wooldridge. On the identification of agents in the design of production control systems. In *Agent-Oriented Software Engineering, LNCS*, pages 141–162. Springer-Verlag, 2001.

[7] K. Decker, K. Sycara, and M. Williamson. Middle-agents for the internet. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI 97)*, pages 578–583, 1997.

[8] N. Kim, I. jae Kim, and H. gon Kim. Video surveillance using dynamic configuration of mutiple active cameras. *Image Processing, 2006 IEEE International Conference on*, pages 1761–1764, Oct. 2006.

[9] J. Orwell, S. Massey, P. Remagnino, D. Greenhill, and G. A. Jones. A multi-agent framework for visual surveillance. In *ICIAP '99: Proceedings of the 10th International Conference on Image Analysis and Processing*, page 1104, Washington, DC, USA, 1999. IEEE Computer Society.

[10] D. Stipanicev, M. Stula, and L. Bodrozic. Multiagent based greenhouse telecontrol system as a tool for distance experimentation. *Industrial Electronics, 2005. ISIE 2005. Proceedings of the IEEE International Symposium on*, 4:1691–1696, 20-23, 2005.

[11] F. Zambonelli and A. Omicini. Challenges and research directions in agent-oriented software engineering. *Journal of Autonomous Agents and Multiagent Systems*, 9:253–283, 2004.