Surveillance System with SIS Controller for Incident Handling using a Situation-based Recommendations Handbook

Erland Jungert[1] and S.-K. Chang[2]

[1]CustodIT
S-582 26 Linköping, Sweden
erland @jungert.net

[2]University of Pittsburgh
Pittsburgh, PA, USA
chang@pitt.cs.edu

***Abstract*---Protection of critical infrastructures involves handling of incidents that may range from serious to quite harmless events. Such systems require means for surveillance that involves a type of sensor system that may identify entities that behave in an unusual way. However, this is not sufficient as means for determination of entities that seemingly are behaving in a normal way but whose activities somehow relate to the first category, must be determined. Means for the support of the operators must also be available by the surveillance system. In this work, an approach to a surveillance system with a Slow Intelligence systems controller for incident handling using a situation-based recommendations Handbook, is proposed and discussed.**

Keywords: security systems, surveillance systems, Slow Intelligence, recommendations handbook

I. INTRODUCTION

Critical infrastructures are to an increasing degree becoming the target for intruders with the intentions to either destroy such facilities or to take them over. For this reason, systems designed for the surveillance of such critical infrastructures have become necessary. An integrated part of such surveillance systems is the sensor system that may include multiple sensors of varying types, such as video cameras, IR-cameras and radars. The sensor systems are continuously collecting large amount of data that are analyzed by the surveillance system and made available to the operators. Collected data may be represented in various information structures. The generation of such extremely large data volumes will eventually lead to the determination of overwhelmingly large information quantities that must be handled and interpreted by the surveillance system to support its operators. Adequate handling of the incoming information by the operators is more or less impossible unless the information is organized and presented in a suitable way. The support for this should not only be carried out by the operational picture presentation system, the operator will also need recommendations on how to act under various circumstances as the situation that need to be handled may be quite complex and of unexpected nature. Important aspects here are, for example, relationships existing between entities within and around the facility and in which context the entities are acting and whether they could be determined as direct or indirect intruders. Operator support from the surveillance system is of great importance when dealing with serious incidents such as attacks from threatening intruders or antagonists. The approach taken here is to solve the problem based on an approach to Slow Intelligence [1] and the use of a situation-based recommendation handbook for crisis management [2] and the protection of critical infrastructures.

The main objective of the work discussed in this work concerns an approach to incident handling based upon Slow Intelligence systems (SIS) controller. Secondary to this, some details of a recommendations handbook to support the operator of the surveillance system will also be discussed together with the required information structures of the surveillance system.

This paper is organized as follows. The architecture of the surveillance system is described in section **II** together with its process steps. In section **III** the Slow Intelligence system controller is introduced together with various computation cycles of the Slow Intelligence system controller. The situation-based recommendation handbook is discussed in section IV. Section V describes a short scenario and section VI gives an overview of the identified information structures, section VII presents related works and conclusions of the work are discussed in section VIII.

II. SYSTEM ARCHTECTURE

The architecture of the surveillance system can be seen in Fig. 1 and it is made up by three basic modules i.e. the sensor system, the Slow Intelligence system controller (SISC) and the visual operations control (VOC). The sensor system and its sensors will not be dealt with further in this work but it is expected to be able to detect and identify entities of all relevant types and on command track

DOI reference number: 10.18293/DMS2015-037

entities entering, residing inside or leaving the facility. This will require a system with a large number of sensor types where the capacity for collection, analyses and storage of these data will be necessary. Such sensor systems will be feasible in the near future as the technology for the development of such systems already exists. The SISC module supported by the sensor system will have the capability to identify entities engaged in hostile activities during the entire period of an incident. Another capability of great importance to the surveillance system is to allow for early detection of hostile entities. To be able to carry out all its requirements SISC also needs to have direct access to all information collected, generated or pre-stored in the databases of the surveillance system. The VOC contains, besides the operational picture, the command and control unit, the recommended actions viewer module and the situation-based recommendation handbook. Attached to the SISC module are three different databases, i.e. the Surveillance information database, the Terrain database and the Normal states database, which stores context information and other descriptions related to the normal state of the facility.

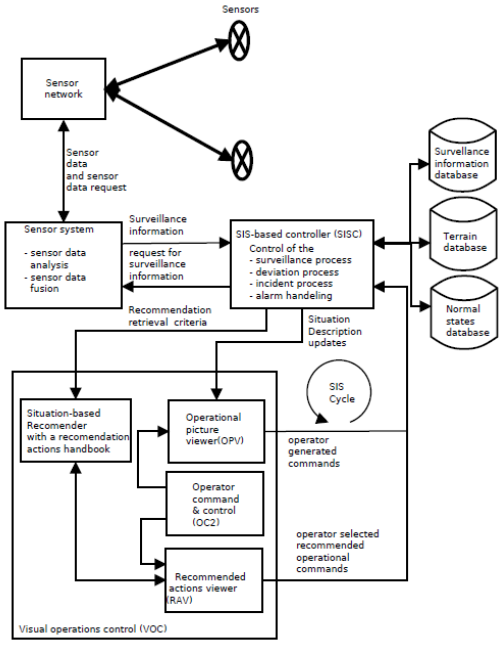

Figure 1. The surveillance system and its main modules, including the sensor system, SISC and the Visual operations control.

The system can be in one of three event states, i.e. *Normal*, *Deviation* and *Incident*. Besides, there is also an alarm state with three different states, i.e. *Inactive*, *Unverified* and *Verified***.** In the Normal state the surveillance system collects information that is used to determine whether the behavior of all appearing entities are normal. Once some type of deviation from the normal occurs, the Event status will be switched to *Deviation* and in some obvious cases directly to *Incident*. In the Deviation state one or more entities may be subject to further observations and an alarm is released, which will correspond to an unverified alarm that immediately must be verified either automatically by some sensors or manually by a guard. The alarm will be determined either as *false* or *true*. When false the system is switched back to *Normal* and when true is set to *Incident*. In the Incident state, the operator, the rest of the staff and other involved must be concerned with bringing the incident to its end. Once the incident is brought to its end the system is switched back to the *Normal* and the alarm state is set to *Inactive*. The various steps in these processes are in short described below.

Normal state

At all times: for all events collect and store Surveillance information;

Analyze all events; If deviation, set Event status to *Deviation* and Alarm status to *Unverified*; release unverified alarm and proceed to Deviation state; if alarm verified set Event status to *Incident*, Alarm status to *Verified*, release verified alarm and proceed to Incident state ;

If Event status equal to *Normal*: Proceed.

Deviation state

At all times: Collect and store Surveillance information for all events; analyze all events;

Handle unverified alarm: if alarm *false* set Event status to *Normal*, Alarm status to *Inactive* and switch to Normal state; if alarm *true* set Event status to *Incident*, Alarm status to *Verified*, release verified alarm and switch to Incident state;

Incident state

At all times: Collect and store Surveillance information for all entities; analyze all events, determine relations of all other entities;

Handle incident: if incident over set Event status to *Normal*, Alarm status to *Inactive* and switch to Normal state; If incident still going on, proceed.

III. THE SIS CONTROLER

SISC can be seen as an *information hub* of the Surveillance system as almost all information in the system flows through this module. Besides by coordinating the information flow of the system CISC also determines entity to entity and entity to context relations. A further aspect of concern is to determine which of the entities are related to any entity with the event state equal to *Incident***.** To determine this is the main task of the Slow Intelligence process. In short, this means that any entity with an identified relationship to an entity determined as an intruder and part of the incident

will be classified and handled as an intruder independently of whether this eventually is the case.

The Slow Intelligence process can be seen as a procedure where first all entities, including also their entity relations, that could be considered antagonist candidates are determined, i.e. whether they could be determined as antagonists that could be participating in an incident. In a second step some of the identified entities are eliminated as they are considered unlikely participants of the incident. This procedure can be described as follows in terms of a pseudo high level programming language.

Repeat for all **entities** in **Facility** when **Alarm** has been released or when **Event-status** *eql* **Incident**
(*if* **Alarm** is released for **Entity** E_j *then*
(*if* **Alarm** unverified *then* (verify **Alarm** for **Entity** E_j /* E.g. send out guard */
if **Alarm** verified for **Entity** E_j *then*
if ***Event*-status** *eql* **Normal** *then Exit*)
if ***Event*-status** *eql* **Incident** *then*
set ***Incident-entity*-set** to E_j
Repeat until no extension of **entities** in
***Incident-entity*-set** *then Exit*
/* enumeration step*/
for all **entities** in ***Incident-entity*-set**
D*etermine* all relations between all
***entities** present* in **Facility**
/* elimination step */
Eliminate all **entities** with harmless relation(s)
to **entities** in ***Incident-entity*-set**
Extend ***Incident-entity*-set** with **entities** with
relevant relations to **entities** in
Incident-entity-set
Extend ***Incident entity relations*** with
the relations between relations in
New-entities and the relation in
***Incident-entity*-set**
set ***Event*-status** to **Incident** to all new
entities in **Incident-entity-set**))

A. Computation Cycles of SIS Controller

The computational cycles of the SIS Controller described in terms of Slow Intelligence operators and operational cycles as can be seen below. The comments in the description are inserted to make the description more readable.

Variables: **Incident-entity-set** (IES), **New-entities (NE), Incident-entity-relations** (IER), **Entities-in-Facility** (EIF), **Observed-entity** (OE)
Initial state: IES = {}, NE = {}, IER = {}, EIF = {$OE_{1,....}OE_{k,...,}$ OE_n}, OE = {}
/**Description in terms of the abstract machine for Slow Intelligence**/
Cycle0: [guard 0,2] P_0 +adapt= P_1 =prop+ P_2
/**Information is received from the sensor system and propagated to the operational picture system which is updated; the information concerns entities entering and exiting the facility. Cycle0 proceeds receiving and propagating information as long as Event status equal to "Normal" and when it becomes "Deviation" the cycle terminates, if the Event status is "Incident" control is switched to Cycle 2*/*
Cycle1: [guard1,0] P_3 +adapt= P_4
/**This cycle is entered for Event status equal to "Deviation". SISC request the operator to verify this status. If the response is "Normal" or "Deviation" control is switched to Cycle0 otherwise if the status is "Incident" the cycle terminates/*

Cycle2: [guard 2,2] P_1 -enum< P_5 >elim- P_6
/**This cycle proceeds recursively when an incident has occurred to identify all entities that are part of the incident, i.e. entities related to the entity that caused the incident. It terminates when all entities associated with the incident are determined*/*
Cycle3: P_7 =prop+ P_8
/**An incident is going on and entities participating in the incident must be tracked; this request is propagated to the sensor system*/*
Cycle4: P_9 +adapt= P_{10} =prop+ P_{11}
/**Incident related information (basically tracks and entity information) from the sensor system is further propagated to the operational picture system/*
Cycle5: [guard5,9] P_{12} +adapt= P_{13}
/**This cycle request information, from the sensor system concerning entities entering the facility; if no new entity is available control is transferred to Cycle9 otherwise the cycle terminates*/*
Cycle6: [guard 6,9] P_{14} -enum< P_{15} >elim- P_{16}
/**In this cycle* P_{14} *is enumerated with respect to the latest acquired entities in Cycle 5 and all relations between the entities in the Incident-entity-set are determined and then the relations are eliminated with respect to whether they are non-incident related. If the relation set is empty control is transferred to Cycle9 otherwise the Incident-entity-set and the Incident-entity-relations are extended with the determined entity and its relations respectively and then the cycle terminates.*/*
Cycle7*:* P_{17} =prop+ P_{18}
/**Incident information is propagated to the operational picture that becomes updated*/*
Cycle8: P_{19} =prop+ P_{20}
/**The sensor system is notified that a new* incident *related entity should be tracked*/*
Cycle9: P_{21} +adapt= $P_{22.}$
/**SISC should be notified by the operator when incident related entities have been taken care of by the security staff, that is the entity has been captivated or has disappeared from the facility*/*

Cycle10*:* [guard10,5] $P_{22} > elim\text{-}\ P_{23}$
*/*captivated or disappeared incident related entities are eliminated from the Incident-entity-set and from the Incident-entity-relations. If Incident-entity-set becomes empty the cycle terminates, which means that the incident is over, otherwise control is switched to Cycle5 and the incident proceeds in a new loop*/*
Cycle11*:* [guard11, 0] P_{18} =prop+ P_{19}
*/*The incident has been terminated (Event status is set to "Normal", Alarm status to "Inactive") and all relevant information is saved in the databases for later use and control is transferred to the normal status loop, i.e. Cycle0*/*

B. Basic Computation Cycles

The algorithmic computational descriptions of the SIS controller described in terms of Slow Intelligence operators and operational cycles can in a simplified overview version be described as follows:

Cycle0: handles the situation when the Event status is *Normal* and Alarm status is *Inactive*. During the execution of this cycle these two variables may be switched to 1) *Deviation* and *Unverified* or 2) *Incident* and *Verified*. In case 1) SISC is transferred to Cycle1 and in 2) to Cycle2.

Cycle1: handles the situation when Event status has been set to *Deviation* and Alarm status to *Unverified*. When the alarm has been verified the state may change to 1) *Normal* and *Inactive* and control switched back to Cycle0 or 2) *Incident* and *Verified* then control to Cycle2.

Cycle2 – Cycle10: these cycles control the maintenance of the incident.

Cycle11: the incident has terminated and condition is set to *Normal* and *Inactive* and control is switched to Cycle0.

C. Details of computation Cycle 2

A more detailed description of the process in Cycle 2 above can be described as:

Cycle2: *variables:* **Incident-entity-set** (IES), **New-entities (NE), Incident-entity-relations** (IER), **Entities-in-Facility** (EIF), **Observed-entity** (OE)

P_1: IES = $\{OE_J\}$
EIF = $\{OE_{1,\ldots,} \ldots OE_j,.. OE_k,\ldots..,OE_{k+t,} \ldots.OE_n\}$
NE = {}
IER = {}

P_5: IES = {OEj}
IER = $\{OE_j$ *rel* $OE_1,\ldots ,OE_j$ *rel* $OE_k,\ldots ,OE_j$ *rel* $OE_{k+t},\ldots, OE_j$ *rel* $OE_n\}$

P_6: IES = $\{OE_j,.. OE_k,\ldots..,OE_{k+t,} \ldots.\}$
IER = $\{OE_j$ *rel* $OE_k,\ldots ,OE_j$ *rel* $OE_{k+t},\ldots\}$

The formal specification of the computation cycles provides a concise way to describe the SIS controller and also offers the possibility to mathematically derive certain properties such as the termination or non-termination of the SIS controller.

IV. SITUATION-BASED RECOMMENDATIONS HANDBOOK

In this section an overview of the situation-based Recommendation Handbook will be presented.

A. *Organization of the Handbook*

The Situation-based Recommendation Handbook will be engaged in a number of activities, i.e. to respond to the information received from SISC by looking up recommendations in the Handbook aimed at supporting the operator in the occurred situation. The received information can be of any of the three following alternative types:

- An entity including its Event type, properties, context and Event status set to *Deviation* and *Alarm* status to *Unverified.*
- An entity including its Event type, properties, context and Event status set to *Incident* and Alarm status to *Verified***.**
- Entities, their direct or indirect relations, their Event type and Event status set to *Incident* and Alarm status to *Verified.*

A consequence of the above input to the Handbook is a set of recommended actions to be carried out by the operator. The Handbook will include instructions of type *call the police* or *send out a guard to patrol the location.* However, those types of instructions are basically determined by local authorities at the facilities and will for this reason not be dealt with further here. Of importance to the work is the organization of the Handbook and the means to access its entries. The Handbook is basically split into two parts where the first is concerned with unverified alarms and deviating behavior while the second part is entirely focusing on incidents. The two parts are called the *Deviation part* and the *Incident part.* The search criteria of the two parts can simply be expressed as follows:

- The search criteria of the *Deviation part* are the Entity, Event types and Behavior of entity.
- The search criteria of the *Incident part* are the Entity, Event types and Behavior of entity where

- a single entity is in focus resulting in just a single look-up,
- multiple entities are in focus resulting in one look up for each entity.

The first cases are rather trivial. The last concerns multiple entities that may relate to, e.g. a *meeting* which involves at least two entities that may or may not be of different types but nevertheless will need one look-up for each entity so that the operator can handle them both separately and together.

Besides, recommendations to make, for example, phone calls to specific persons or organizations the Handbook must also give recommendations that concern the context of the operational situation. If an antagonist is walking through a forest around the facility a sent out guard cannot follow that person by car. If the antagonist is expected to carry weapons other precautions must be recommended. The list of special recommendations may be made quite long and cannot be completed here but must be seen as a task determined by the security staff at each specific facility.

B. Events corresponding to possible incidents

In Table 1 a series of possible events that may cause an incident are described; the number of incidents in the list is not complete and include just a few examples for illustration purposes. If an entity during an incident is acting accordingly its event status will be set to *Deviation* or *Incident.*

Table 1. Event types, their possible relations to other entities.

Event type	**relation to other entity or object**
Approaching a fence	An entity is approaching and acting unnatural close to a fence
Approaching a prohibited area	An entity is approaching a prohibited area or have been standing there for some time
Object picked up inside facility	Object picked up by an entity inside the facility
Object picked up outside facility	Object picked up by entity outside the facility
Object thrown over fence of facility	Object thrown over fence from outside or inside the facility by entity

The processing of these events may occur either in Cycles 1, 2 or 6 in SIS controller. Besides occurring during an incident as in 2 or 6, each of these events may either be the cause of an incident or a deviation, i.e. when any of these events occur in Cycle 1. For Cycles 2 and 6 verification of the alarm is not necessary because the incident is already going on, i.e. the alarm state is already set to *Verified.* Of importance to the events described is that they correspond to capabilities of surveillance applications, see e.g. [3], and carried out in conjunction with the sensor system and SISC.

V. A SCENARIO

The scenario given here can in short be described as follows:

A person is observed walking against a fence of a facility. At the fence the person stops and throws a package over the fence and walks away. After a while a second person comes on the inside of the facility and picks up the package and walks away against a prohibited area.

This short scenario includes, in sequence a number of events and for each one of them SISC generates instances of the Status and Context information relations. This information is then sent to the Handbook that looks up the corresponding recommendations.

Event 1

Status information: <person#3,14.45, *Deviation, Unverified*, approaching fence >
Context information: <person#3, fence, forest >
Hand book search criteria: Person, *Deviation, Behavior of entity*
Recommendations: *send out guard to verify alarm; instruct guard to inform on what is going on; operator should follow track of person approaching fence in operational picture*.

Event 2

Status information: <person#3, 14.54, *Incident, Verified*, object thrown over fence >
Context information: <person#3, fence, road>
Handbook search criteria: Person, *Incident, Behavior of entity*
Recommendations: *call police, set facility in safe mode*.

Event 3

Status information: <person#46, 15.23, *Incident, Verified*, object picked up inside >
Context information: <person#46, fence, road>
Handbook search criteria: Person, *Incident, Behavior of entity*
Recommendations: *the operator is instructed to follow track of person in operational picture; a pair of guards should be sent out to observe the person; Guards instructs to report on what is going on*.
Report from guard: *person outside fence is taken care of*.

Event 4

Status information: <person#46, 15.38, *Incident, Verified*, approaching prohibited area >

Context information: <person#46, fence, road>
Handbook search criteria: Person, *Incident, Behavior of entity*
Recommendations: *the operator is instructed to follow track of person in operational picture*.

Event 5

Status information: <person#46, 15.54, *Incident, Verified,* at prohibited area>
Context information: <person#46, prohibited area, road>
Handbook search criteria: Person, *Incident, Behavior of entity*
Recommendations: *instruct guards to arrest person at prohibited area*.
Report from guard: *person at prohibited area is arrested*.

As soon as the last report has come in the incident has been brought to its end and the state of the surveillance system will be set to *Normal* and *Inactive***.** However, this last step may need to include some further activities as the antagonists may have carried out activities whose effects have not yet been discovered and that consequently may cause problems later on. It is consequently necessary to inspect the facility for such perhaps dangerous threats even after the incident has been terminated. This is an activity that must be carried out by the staff of the facility.

VI. INFORMATION DESCRIPTION

In this section information structures that need to be used by the surveillance system are described.

A. Event related information

The purpose of event related information is to serve two capabilities of the surveillance system, i.e. to

- find relevant entries to the Handbook
- keep the operational picture updated at a current state.

That is, more or less the same information used to look up entries in the Handbook is also used to keep the operational picture updated. Consequently, Event dependent information used in these two activities corresponds to information acquired by means of the sensor system and in conjunction with the Slow Intelligence system controller that continuously is surveilling the facility and analyzing the acquired information; an activity that can be seen as the screening of a number of ongoing events that may correspond to various types of incidents. Incidents may consequently range from quite harmless behavior of different entities and up to really serious events carried out by terrorists. Furthermore, event related information can also be determined in part from historic events where the underlying data are captured over long periods in time essentially to allow statistical determination of what is a deviation from normal. Examples of such information could be tracks of observed objects that compared to historic data shows that the entity deviate from what can be considered normal. Other information that may be needed to improve the knowledge of an observed object and its general behavior could be the determination of relations to other entities. This information may be used to find new and relevant entries in the Handbook.

The information that needs to be collected by the sensor system and eventually stored belongs to classes that can be expressed as follows including also possible but not entirely complete value sets.

Facility entities

- Facility subarea:{outside facility boundary, inside facility boundary, facility boundary, facility airspace, restricted area, ...}
- Physical installation of facility: {fence, building, road, walk way, gateway, check point …}
- Facility terrain type: {forest, hill, park, plain, water front, urban area …}
- Sensor system: {sensor type, sensor location ...}
- Manually controlled sensor: {sensor type, sensor location...}

Event entity

- Event location: {coordinates /2D or 3D/}
- Event subarea: {perimeter, outside facility boundary, inside facility boundary, airspace, prohibited area ...}
- Physical installation at event: {fence, building, road, walk way, gateway, check point …}
- Event terrain type: {forest, hill, park, plain, water front, urban area}

Event condition

- State of event: {day, night}
- Time of event: {time}
- Weather condition: {rain, snow, fog, clear sky...}

Observed entity type

- Entity type: {Person, Car, Truck, Aircraft …}
- Person: {antagonist, police, fireman, guard …}

Behavior of entity

- Observed behavior: {walking, running, driving, still, climbing, entering, exiting, hiding …}
- Estimated direction: {N, NE ..... W, NW}
- Estimated Target: {/*facility dependent*/]
- Event installation type: {fence, building, roof, road, walk way, gateway, check point …}
- Event terrain type: {hill, park, plain, water front, urban area, prohibited area… }

Event situation

- Event status: {*Normal*, *Deviation*, *Incident*}
- Alarm status: {*Inactive*, *Unverified*, *Verified* (false alarm, system failure)}

The above set of classes can be seen as an ontology, see Fig. 2.

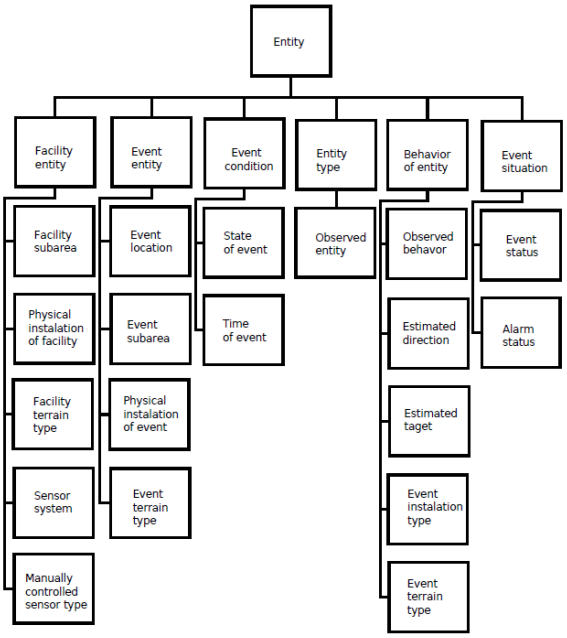

Figure 2.The ontology of Entities.

B. Acquired and complementary information

Acquired information in this context means information captured by the sensor system that relates to detected entities and their status and properties. That is, information originating from sensor data that have been analyzed, often fused and eventually transferred into SISC for further analyses. Besides this, there is also a need for stationary information such as geographical information, which here is called complementary information and relates in most cases to the context of the facility and where the detected entities may reside from time to time. To thoroughly describe the context in which an entity resides and its status at a specific time during an ongoing event both acquired and complementary information is required. This can be described in terms of two information sets (relations) where some of the information appears in both sets basically for identification. The relations are approximately described as follows:

- Status information: <Entity, Event location, Time of event, Event status, Event type, Alarm status, Observed object type, Observed behavior>
- Context information: <Entity, Event location, Time of event, Event status, Alarm status, Subarea, Event installation, Terrain type>

The first relation, Status information, describes the status of an event at a specific point in time (Time of event) and its location; further information concerns the actual event type that is normal, a deviation or an incident. Data are gathered periodically for every observed entity within the area covered by the sensor system. Altogether, for every observed entity a track of every observed entity can be determined although it seems more economical to just track entities with the event status *Deviation* or *Incident*. Events that are classified as normal will be analyzed for determination of whether the general behavior of the entity is normal; all this will obviously require analyses of massively large data sets. For normal events no alarm is activated, that is the alarm status is set to *Inactive* and for these events the Handbook is not consulted. However, for all observations the Operational picture is updated to give the operator a presentation of the current situation at the facility. If the event is classified as a deviation by the system the Handbook must be consulted, and the operator may be instructed to verify the alarm to determine whether the event is to be classified as an incident. In case of an incident the operator must bring the incident to an end by means of the recommendations from the Handbook and the views of the operational picture.

C. The event situation

The event situation concerns the status both of observed events and the current alarm status. The relationship between Event status and Alarm status can be described as in Table 2. To be observed here is also that events like threats must be considered as just deviations which can be seen as a situation with unverified alarms. This means that this alarm must be verified before an incident is at hand.

Table 2. Possible event situation

Event status /Alarm status	Inactive	Unverified	Verified
Normal	N	Error	Error
Deviation	Error	D	I
Incident	Error	Error	D

Whenever an error occurs the operator must deal with a system failure; in other words it is a serious event that immediately must be handled by special domain experts or technicians but it is not an

incident or deviation in the usual sense.

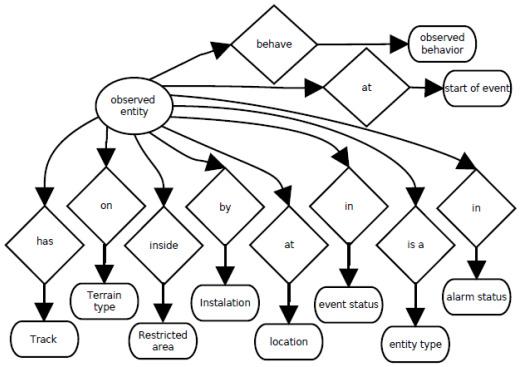

Figure 3. Properties and context of an observed entity.

D. *Entity relations*

All observed entities has relations to the environment in which they are acting but they also relate to the current event and leave tracks that especially during deviations and incidents must be acquired. As a consequence, the state description of any observed object may look like in Fig. 3 where both properties and contextual descriptive terms are included.

Of importance are not only the properties and the context of observed entities but also the relations between entities involved in deviations and incidents. Such relations can be direct as in the first_relation in Fig. 4, that is in the simple relation *Entity-i meets Entity-j*, In this relation no incident may be caused unless *Entity-I* is already classified as an intruder. In the second case an incident is determined because of the indirect relation as in the second example in the figure where *Entity-i throws object-k* and *Entity-j picks up Object-k*, where an entity throws an object, which will cause an incident, and then the indirect relation is established when the second entity picks up the thrown object. Which causes the second and indirect relation to be determined as *Entity-j* is considered part of the incident as well. Obviously, such entities, although they are classified as normal, appear to have relations to entities determined as incident related because throwing an object will automatically cause a verified alarm leading to an incident. Consequently, entities that appear to have any kind of relation to an entity involved in an incident must be seen as part of the incident i.e. their Event status must be switched from *Normal* to *Incident* and the crisis management staff must start acting accordingly. This means that the Handbook tells the operator to focus on the new incident related entity as well, which also is indicated in the operational picture that will show its status as *Incident*.

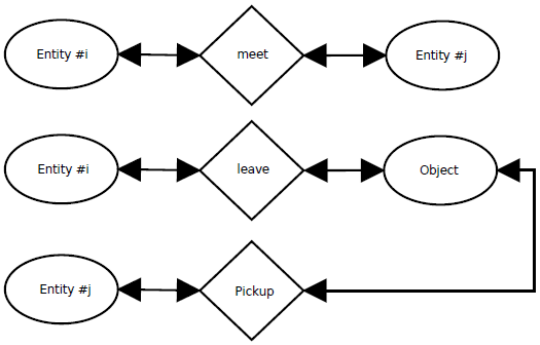

Figure 4. Examples of direct (above) and indirect entity relations (below).

VII. RELATED WORKS

Relations to other work in this context concerns several different aspects. Basically, they are concerned with Slow Intelligence, various approaches to recommendations systems but also to surveillance systems for protection of critical infrastructures. Of importance to note is also that this is the first attempt on the design of security and surveillance systems using Slow Intelligence for determination of entities involved in incidents at critical infrastructure facilities.

The Slow Intelligence approach was first proposed by Shi-Kuo Chang [1]. The visual specification of component-based Slow Intelligence Systems is described in [4]. This work introduces the visual description of super-components by Petri nets or other UML diagrams. It provides the foundation of the present work. Component-based Slow Intelligence Systems has been applied to many areas, including social influence analysis, topic and trend detection, high dimensional feature selection, image analysis, swimming activity recognition, and most recently pet care systems and energy control systems. In [5] the notion of an abstract machine for computation cycle was introduced. Our current approach is based upon it.

Related surveillance system for various approaches can be found in Goodall [6] where gathering of user requirements for a visualization system with capabilities for intrusion detection analysis is discussed. Shan, Wang, Li, and Chen [7] present a comprehensive design for decision support systems applied to emergency response. Hansson et al. [8] demonstrates the intensions to determine the general context for security systems as a foundation for user and system requirements. In [9] is a project called RESCUE discussed. In this work are a number of aspects that relates to this work discussed e.g. the focus on needs to assess situations for improved awareness. Pozzobon et al. [10] discuss primarily user requirements of surveillance systems

with special emphasis on security in ports. A further relationship to this work concerns basically user requirements needed for determination of user oriented capabilities which plays an important role in systems development. However, this is more in focus to the work discussed in [3].

Content based recommendation systems are generally based on descriptions of various items that may be of interest to a user with a particular profile see e.g. Pazzani and Billsus [11]. However, here the situation differs in that the recommendations can be seen as the result of observed events detected by the sensor system. Consequently, such recommendation systems will be event driven and the user has no other choice than to react to the events and given recommendations. Example of an application of this latter approach is proposed by Laliwala [12] in which an event-driven service-oriented agricultural recommendation system is proposed. Another similar example is given by Kim et al. [13].

VIII. DISCUSSION AND CONCLUSIONS

In this work a surveillance system for protection of critical infrastructures is proposed. The main focus concerns capabilities to identify entities involved in abnormal behavior that eventually will cause alarms and turn the status of the system status into *Incident*. Further aspects that have been subject to attention here are also determination of incidental events and relations between entities involved in such incidents. The approach taken has been to carry out the determination of such information based on an approach to Slow Intelligence; the outcome is a SIS controller that will use various search criteria to a situation-based recommendation Handbook and the maintenance of an operational picture system. Complementary to this, the required information structures are also described and discussed. Finally, to demonstrate the capabilities discussed a simple scenario is carried out.

The approach taken in this work contributes to Slow Intelligence research in the sense the scenarios describe realistic computation cycles for the SIS controller. Hence, further research must focus on analysis of the computation cycles to determine the properties of SIS controllers such as termination conditions, existence of endless loops and so on.

REFERENCES

1. S.-K. Chang, "A General Framework for Slow Intelligence Systems", International Journal of Software Engineering and Knowledge Engineering, Volume 20, Number 1, February 2010, 1-16.
2. S.-K. Chang, E. Jungert, A Self-Organizing Approach to Mission Initialization and Control in Emergency Management, Proceedings of the International Conference on Distributed Multimedia Systems, San Francisco, September 6-8, 2007, pp 51-56.
3. E. Jungert, N. Hallberg & N. Wadströmer, A system design for surveillance systems protecting critical infrastructures, Journal of Visual Languages and Computing, December 2014, Vol 25(6), pp 650-657.
4. S.-K. Chang, Y. Wang and Y. Sun, "Visual Specification of Component-based Slow Intelligence Systems", Proceedings of 2011 International Conference on Software Engineering and Knowledge Engineering, Miami, USA, July 7-9, 2011, 1-8.
5. S. K. Chang, W. H. Chen, B. Kao, L. Kuang, and Y. Z. Wang, "The design of pet care systems based upon Slow Intelligence principles," Int'l Journal of Software Engineering and Knowledge Engineering, 2014.
6. R. Goodall, "User requirements and design of a visualization for intrusion detection analysis", Proc. 2005 Workshop on Information Assurance and Security, pp. 394-401, June 2005.
7. S. Shan, L. Wang, L. Li, and Y. Chen, "An emergency response decision support system framework for application in e-government", Information Technology and Management, vol. 13, 2012, pp. 411-427.
8. M. Hansson, R. Granlund, N. Hallberg, F. Lantz, and E. Jungert, "A reference context module for development of security systems", Proc. of the Int. conf. on Distributed Multimedia Systems, Aug. 2011, pp. 64-69.
9. S. Mehrotra, C. Butts, D. Kalashnikov, N. Venk-atasubramanian, R. Rao, G. Chockalingam, R. Eguchi, B. Adams and C. Huyck, "Project Rescue: Challenges in Responding to the Unexpected", in *SPIE*, Vol. 5304, Jan 2004, pp. 179-192.
10. A. Pozzobon, G. Sciutto, and V. Recagno, "Security in ports: The user requirements for surveillance systems", In The of a Wireless Sensor Network Application from End-User Requirements", Proc. Of the 2010 6th int. Conf. on Mobile Ad hoc and Sensor Networks (MSN) Dec. 20-22, 2010, pp 168-175.Springer Int. Series in Eng. And Comp. Sci., Vol. 488, 1999, pp 18-26.
11. M. J. Pazzani, D. Billsus, Content-Based Recommendation Systems, Ed(s) P. Brusilovsky, A. Kobsa, W. Nejdl, The Adaptive Web - Methods and Strategies of Web Personalization, Springer Verlag, vol. 4321, 2007, pp 325 - 341.
12. Z- Laliwala, "Semantic and Rule Based Event-driven Service-Oriented Agricultural Recommendation System", 26th IEEE Int. conf. on Distrib. Comp. Systems Workshops (ICDCS), July 4-7, 2006.
13. J. K. Kim, H. K. Kim, Y. H. Cho, "A user-oriented contents recommendation in peer-to-peer architecture", Expert Systems with Applications, Jan 2008, Vol. 34(1), pp 300-312.