Noname manuscript No.
(will be inserted by the editor)

Bio-inspired relevant interaction modelling in Cognitive crowd management

Simone Chiappino · Pietro Morerio · Lucio Marcenaro · Carlo S. Regazzoni

Received: date / Accepted: date

Abstract Cognitive algorithms, integrated in intelligent systems, represent an important innovation in designing interactive smart environments. More in details, Cognitive Systems have important applications in anomaly detection and management in advanced video surveillance. These algorithms mainly address the problem of modelling interactions and behaviours among the main entities in a scene.

A bio-inspired structure is here proposed, which is able to encode and synthesize signals, not only for the description of single entities behaviours, but also for modelling cause-effect relationships between user actions and changes in environment configurations. Such models are stored within a memory (Autobiographical Memory) during a learning phase. Here the system operates an effective knowledge transfer from a human operator towards an automatic systems called Cognitive Surveillance Node (CSN), which is part of a complex cognitive JDL-based and bio-inspired architecture. After such a knowledge-transfer phase, learned representations can be used, at different levels, either to support human decisions, by detecting anomalous interaction models and thus compensating for human shortcomings, or, in an automatic decision scenario, to identify anomalous patterns and choose the best strategy to preserve stability of the entire system. Results are presented in a video surveillance scenario, where the CSN can observe two interacting entities consisting in a simulated crowd and a human operator. These can interact within a visual 3D simulator, where crowd behaviour is modelled by means of Social Forces. The way anomalies are detected and consequently handled is demonstrated,

Department of Naval, Electric, Electronic and Telecommunications Engineering, University of Genoa
Via Opera Pia 11A, 16100, Genoa, Italy
Tel.: +39-010-3532212
Fax: +39-010-3532134
e-mail: (see http://www.isip40.it)

on synthetic and also on real video sequences, in both the user-support and automatic modes.

Keywords Cognitive systems · Bio-inspired learning · Anomalous interactions · Crowd monitoring · Self Organizing Map

1 Introduction

Several works have been devoted in the last decade to link traditional computer vision tasks to high-level context aware functionalities such as scene understanding, behaviour analysis, interaction classification or recognition of possible threats or dangerous situations (Remagnino et al, 2007), (Trivedi et al, 2000), (Lipton et al, 2003), (Trivedi et al, 2007).

Among the several disciplines which are involved in the design of next generation security and safety systems, cognitive sciences represent one of the most promising in terms of capability of provoking improvements with respect to state of the art. As a matter of fact, several recent studies have proposed the application of smart functionalities to camera and sensor networks in order to move from object recognition paradigm to event/situation recognition (Espina and Velastin, 2005). Such a trend change has substantial implications for what concerns the processing of signals, as it will be shown throughout this work. The application of bio-inspired models to safety and security tasks represents a relevant added value. In fact, such models enhance the capability not only of detecting the presence of an intruder in a forbidden area or recognizing the trajectory of an object in an urban scenario (e.g. a baggage in a station or a car on the road) but also of interpreting the behaviour of the entity, or properly selecting events of interest with respect to normal situations, or even to automatically take decisions and perform actions on the environment.

The application of neurobiological sciences to the field of cognitive radar and cognitive radios lately led to the rise of a new broad discipline which was formalized in some works by S. Haykin (Haykin, 2011), (Haykin, 2012b), (Haykin, 2012a) under the name of *Cognitive Dynamic Systems*, These works eventually gather and synthesize some of the main intuition of the last decades in this field. A working definition of *Cognitive Dynamic Systems* is given:

> Cognitive dynamic systems build up rules of behaviour over time through learning from continuous experiential interactions with the environment, and thereby deal with environmental uncertainties.

The underlying hidden assumption behind the formalization of this discipline is that animal and human brains are the best cognitive systems on the market and are thus to be emulated.

In this work, the features of a cognitive architecture, motivated by the work of Damasio (Damasio, 2000) and based on the Joint Directors of Laboratories model (JDL) (Hall and Llinas, 1997), are described. Damasio's theories describe cognitive entities as complex systems capable of learning based on the

experience of *interactions* between themselves and the external world. The application of the proposed framework to crowd analysis is presented. A novel fashion for signals to be organized and processed is also proposed. Such a fashion is implicitly accounted for in previous works (Dore et al, 2011), (Dore et al, 2010a), (Chiappino et al, 2012) and (Chiappino et al, 2013d) and motivated by the fact that it traces intelligent biological patterns.

In a video surveillance scenario, the proposed Cognitive Node (CN) can be applied to the crowd analysis domain in order to identify patterns that deviate from expected behaviour: an abnormal behaviour is defined as any kind of deviation from central tendencies defined as *normality condition*. The CN operating mode is made up of *learning* and *detection* phases. During the learning period the CN stores the observed interactions between human operator actions and the resulting crowd state changes. It is important to note that the human actions acquired are devoted to avoid abnormal situation, e.g. overcrowding or abnormal flow directions. The automatic system is able to effectively learn representations of *normal* user-environment relationships for *standard crowd behaviour maintenance* through the aforementioned data structure and architecture. After such a knowledge acquisition phase, learned representations can be used at two different levels: first, to support human decisions by detecting anomalous *crowd-operator* interactions and compensating for human shortcomings; secondly, in an automatic decision scenario, to autonomously identify anomalous *crowd-environment* configurations and choose the best strategy to preserve stability of the entire system (i.e. a proper security level in the monitored area) by putting in action effective countermeasures.

Many video analysis algorithms have been developed in order to identify crowd behaviours. For instance, in (Mehran et al, 2009) a method for crowd behaviour analysis based on social forces and optical flow is proposed. More recently, in (Solmaz et al, 2012) the authors present an innovative method based on people flow estimation. A new abstract viscous fluid field is proposed in (Su et al, 2012) for detecting crowd events. The main contribution of this paper is to propose and develop an innovative cognitive video surveillance system, which is able to detect anomalies by learning behavioural models from observations of crowd evolution and consequent human operator (re)actions. The system acquires the crowding states, by video analysis techniques, and it receives from the user his countermeasures, in order to maintain stability and to avoid abnormal situations. This knowledge (i.e. models of normal interactions) is transferred from a human operator to the system, providing it with crowding dynamic models augmented by user actions. A simulated crowd monitoring environment have been used for training and testing.

The issue of modelling and simulating crowds will not be discussed in details for the sake of brevity, although it represents a central matter in applying the theory which will be presented. A comprehensive traction of such interconnected fields is given in (Chiappino et al, 2013a). We here point out just a few concepts. First, the use of a simulator is necessary in order to gather enough data for training and testing, as video sequences of the desired kind are not available for training. A simple CN application on real video sequences

is presented in order to show the capabilities of the system, which is however trained with simulated data. Secondly, it is unrealistic to track every single person in a high density crowded scene, especially if a single camera is available: the visual information gathered by the sensor is simply not enough to accomplish such a task. This remark has led to consider global approaches to crowd monitoring such as in (Moore et al, 2011) and (Morerio et al, 2012). At last, the model employed in simulation combines technical and social aspects following the current trend in literature. As shown for instance in (Mehran et al, 2009), (Pellegrini et al, 2009), (Luber et al, 2010) and (Mazzon et al, 2013), a social force model describing interactions among the individual members of a group of people has been proposed to detect abnormal events in crowd videos. Here people are treated as interacting particles subject to internal and external physical forces which determine their motion and global behaviour. Such a point of view is also widely employed in this work.

The remaining of this work is organised as follows. Sections 2 and 3 present the proposed bio-inspired models for cognition and knowledge representation respectively. The applications of such models to crowd monitoring are presented in section 4. Section 5 describes the proposed approach for anomaly detections, while results are given in section 6. Conclusions are drawn in section 7.

2 A bio-inspired cognitive model for Cognitive Surveillance Systems

The proposed approach to Intelligent Video Surveillance (IVS) has been implemented according to a bio-inspired model of human reasoning and consciousness grounded on the work of the neuro-physiologist A. Damasio (Damasio, 2000).

As already mentioned, Damasio's theories describe cognitive entities as complex systems capable of incremental learning based on the experience of relationships between themselves and the external world. Two specific brain devices can be defined to formalize the aforementioned concept: Damasio names them *proto-self* and *core-self*. Such devices are specifically devoted to monitor and manage the internal status of an entity (proto-self) and the external world (core-self). Thus, crucial aspects in modelling a cognitive entity following Damasio's model are first of all the capability of accessing entities' internal status and secondly the analysis of the surrounding environment. Relevant information comes from the relationships between the two. This approach can be mapped into a sensing framework by dividing the sensors into endo-sensors (or proto-sensors) and eso-sensors (or core-sensors) as they monitor, respectively, the internal or external state of the interacting entities.

Applying these concepts to the video analysis domain, interacting entities can be represented either by a guard monitoring a smart environment or by a subject driving an intelligent vehicle as well as a guard and an intruder interacting in some monitored area, while considering a crowd management

scenario, eso-sensors can monitor the crowd, while endo-sensors can provide information about system parameters, as it will be clearer in the following.

Referring to the sample proposed framework, four main blocks have been identified as representative of a cognitive-based sensing architecture as the control centre, the CN, the (intelligent) sensing nodes and the mobile terminal and/or actuators. The tasks which can be accomplished by each block are shown in Fig. 1, establishing a preliminary bridge between the concepts introduced by Damasio and the effective implementation of the system.

The core of the proposed architecture is the already mentioned CN, which can be described as a module that is able to receive data from sensors of all kinds, to process them, defining different *configurations* as *interactions* between proto and core states. Such a bio-inspired knowledge representation permits to asses potentially dangerous or anomalous events and situations and possibly to interact with the environment itself.

2.1 Cognitive Cycle for single and multiple entities representation

Within the proposed scheme, the representation of each entity has to be structured in a multi-level hierarchical way. As a whole, the closed processing loop realized by the CN in case of a given interaction between an observed object and the system can be represented by means of the so-called Cognitive Cycle (CC - see Fig. 1) which is based on four fundamental logical blocks:

- Sensing: the system has to continuously acquire knowledge about interacting objects and their own internal status.
- Analysis: the collected raw knowledge is processed in order to obtain a precise and concise representation of occurring events and causal interactions.
- Decision: the precise information provided by the analysis phase is processed and a decision strategy is selected according to the goal of the system.
- Action: the system fulfils the configuration computed during the decision phase as a direct action over the environment or as a message provided to some actuator.

The proposed model for cognition has many analogies with the one adopted by Haykin in its formalization of *Cognitive Dynamic Systems* (Haykin, 2012b) and referred to as the *Fuster's Paradigm*: Joaquin Fuster proposes in fact the concept of *cognit* and an abstract model for cognition, based on five fundamental building blocks, namely *perception*, *memory*, *attention*, *intelligence* and *language* (Fuster, 2005). Perception represents the information gain block, and corresponds to the sensing block of the CC; similarly, intelligence matches the analysis logical block and also, according to the Fuster's paradigm, includes the decision-making stage. Memory is associated, within the CC, to a learning phase which is continuous and basically involves all the stages of the cognitive cycle: this will be explained more in details in sections 3.3 and 4. The attention

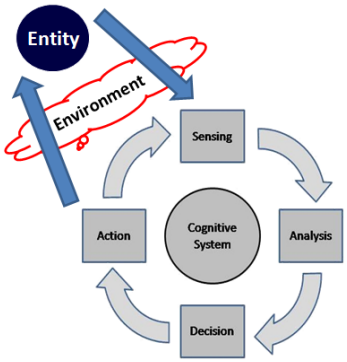


Fig. 1 Cognitive Cycle (single object representation)

block is meant to optimize the information flow within the Dynamic Cognitive System: this aspect goes beyond the purposes of this work. Eventually, the language block is intended to provide efficient communication on a person to person basis but it is not considered here (and not even by Haykin in his works).

The CC, by *experiencing interactions* between the CN and the external object, provides different configurations also called *cause-effect relationships*. Starting from these relations it is possible to define object representations based on theirs dispositional capabilities, i.e. the objects can be disposed (or not) to change in some way. More formally, an observed object x is disposed to D in different *C-cases* (i.e. situations), where D defines the dispositional propriety of x by a set of C configurations (i.e. cause-effect relationships), called *dispositional statements* (Bird, 2012).

A set of dispositional proprieties gives a *dispositional embodied description* of an object, and it includes reactions it generates in the cognitive system, i.e. possible actions that the system can plan and perform when a situation involving that object is observed or predicted. According to this statement, it is possible to refer to the representation model depicted in Fig. 1 as to an Embodied Cognitive Cycle (ECC). The cognitive cycle can be seen as a way of representing generic observed objects within the CN by means of a multi-level representation involving both the bottom-up analysis chain and the top-down decision chain (see Fig. 2). With respect to security and safety domains, in which the ECC is here applied, the above mentioned dispositional proprieties are associated to a precise objective: to maintain stability of the equilibrium between the object and the environment (i.e. maintenance of the proper level of security and/or safety). Anomaly is a deviation from the normality and it can be considered as a violation of a certain dispositional propriety.

As a consequence, each entity is provided with a 'security/safety oriented ECC (S/S-ECC)' which is representative of the entity itself within the CN. Moreover, the mapping of the S/S-ECC onto the CN chain shown in Fig. 2 can be viewed as the result of the interaction between two entities, each one

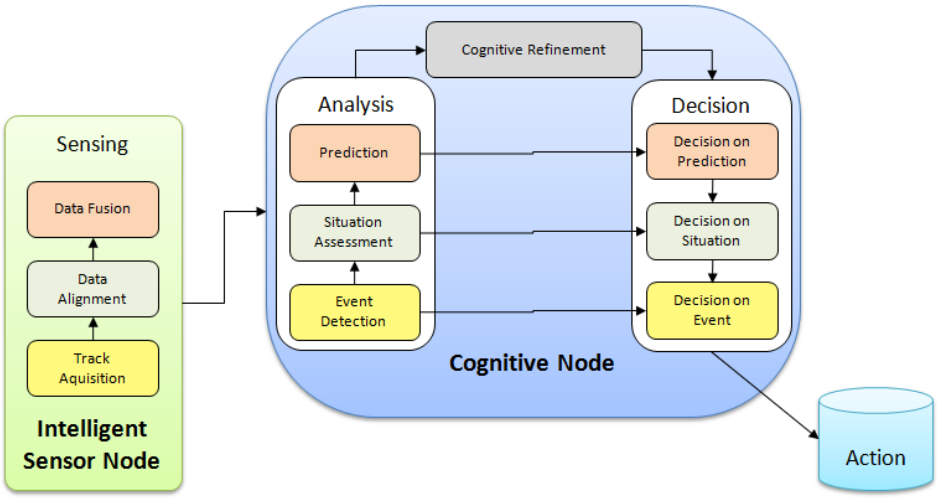


Fig. 2 Cognitive Node-based object representation: Bottom-up analysis and top-down decision chain.

described as a cognitive cycle too. In particular, if the external object (eso) and the internal autonomous system (endo) are represented as a couple of Interacting Virtual Cognitive Cycles (IVCC), the IVCCs can be matched with the CN structure (i.e. the bottom-up and the top-down chains) by associating parts of the knowledge related with the different ECC phases to the multilevel structure processing parts of the CN (Fig. 2).

More in detail, the representation model of the ECC (top left corner of Fig. 3) is centered on the cognitive system that can be considered by itself as a cognitive entity. Therefore, it is possible to map the proposed representation as in the top right corner of Fig. 3, where two IVCCs, the one representing the entity (or object - $IVCC_O$) and the other representing the cognitive system ($IVCC_S$), interact in a given environment. In this model, the sensing and action blocks of the $IVCC_S$ correspond to the sensing and action blocks of the ECC (see bottom right corner of the figure). However, in the $IVCC_S$, such blocks assume a parallel virtual representation of the physical sensing and action observed corresponding respectively to the Intelligent Sensing Node and the Actuator blocks in the general framework.

The analysis phase of the $IVCC_S$ ($Analysis - IVCC_S$) can be considered as including a virtual representation of the four stages characterizing the state of the interacting object. Sensing phase can be mapped in the event detection sub-block of the $An-IVCC_S$ (ED_{System}) as well as the object event detection (ED_{Object}). Similarly, the system situation assessment sub-block (SA_{System}) includes a virtual representation of the object situation assessment (SA_{Object}). Finally, as shown in the bottom left corner of Fig. 3, the prediction, decision and action parts of the object can be considered as knowledge that allows the cognitive system to predict the future behaviour of the interacting object

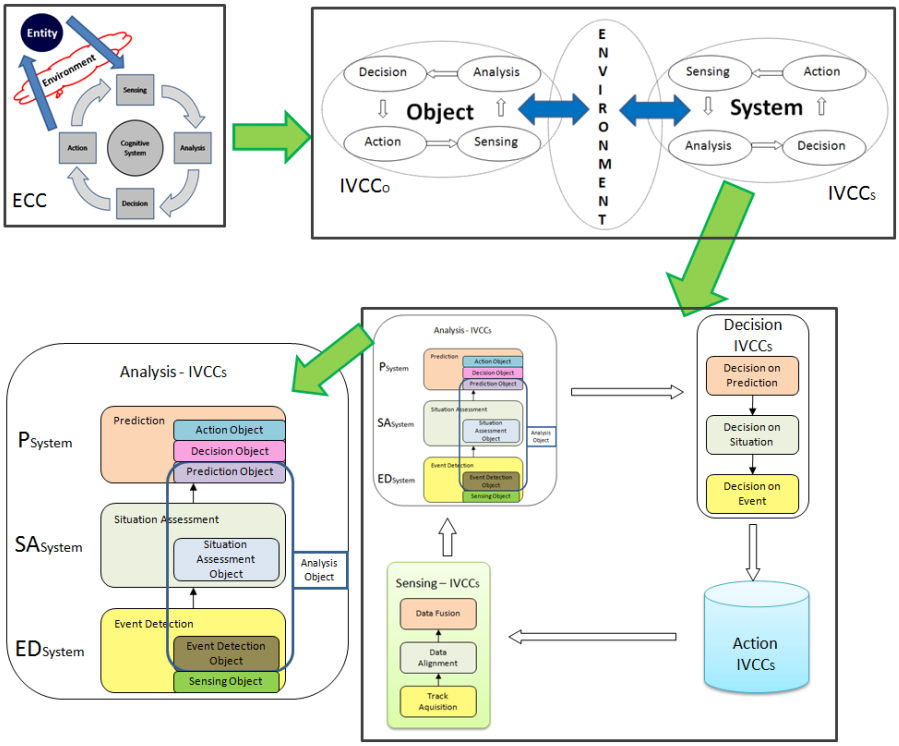


Fig. 3 Embodied Cognitive Cycle, Interactive Virtual Cognitive Cycles and Cognitive Node matching representation

itself (the interacting objects are here the system and the one observed external object). Prediction, decision and action can be included in the prediction sub-block of the system (P_{System}).

The proposed interpretation of the matching among the embodied cognitive model, the interactive virtual cycles representing the entities acting in the environment (including the system) and the CN, allows considering the CN as a universal machine for processing ECCs with respect to a large variety of application domains. In general, each ECC starts with ISN (intelligent Sensor Node) data, including an interacting entity (eso-sensor) and a system reflexive observation (endo-sensor). The observed data are considered from two different perspectives (the object's and the system's) by creating a description of the current state of the entities using knowledge learned in previous experiences. Such process happens at event detection and situation assessment sub-blocks. Then, a prediction of future actions taken by the $IVCC_O$, contextualized with the self-prediction of future planned actions of the system, occur at the prediction sub-block. The use of the knowledge of the $IVCC_O$ ends at this stage. Finally, the $IVCC_S$ is completed by adjusting plans of

the system in the representation of its decision and action phases that are, as stated above, a parallel virtualization of the ECC. In addition, it is relevant to briefly point out that a similar decomposition can be adopted in the case when two interactive entities are observed. The description of the interacting subjects can be modelled observing that the two entities can form a single meta-entity to which is associated a meta cognitive cycle interacting with the autonomous system. The meta-entity (ME) can simply be considered as a composition of the two cognitive cycles associated to the initial entity couple. The advantage of the proposed representation, involving the description of an Embodied Cognitive Cycle by means of an IVCC couple is that the same mechanism used to represent the interaction of a ME with the autonomous system can be also used to represent the interaction between two observed entities forming an observed meta-entity, as proposed in (Dore and Regazzoni, 2009). Dynamic Bayesian Networks (DBNs) can be used to represent cognitive cycles and IVCCs based on a structure called Autobiographical Memory (AM) (Dore et al, 2010b) (Dore et al, 2010b). DBNs provide a tool for describing embodied objects within the CN in a way that can allow incremental learning from experience (Murphy, 2002). Section 3 is devoted to the demonstration of such a claim.

2.2 The Cognitive Node

The general architecture of the Cognitive Node is depicted in Fig. 4. Intelligent sensors are able to acquire raw data from physical sensors and to generate feature vectors corresponding to the entities to be observed by the CN. Acquired feature vectors must be fused spatially and temporally in the first stages of the node, if they are coming from different intelligent sensors.

As briefly introduced in the previous section, the CN is internally subdivided into two main parts, namely the analysis and the decision blocks. These two stages are linked together by the cognitive refinement block, whose role will be shortly explained. Analysis blocks are responsible for organizing sensors information and finding interesting or notable configurations of the observed entities at different levels. Those levels can communicate directly with the human operator through network interfaces in the upper part of Fig. 4. This is basically what can be done by a standard signal processing system being able to alert a supervisor whenever a specific anomalous interaction behaviour is detected. A prediction module is able to use the stored experience of the node through the internal AM for estimating a possible evolution of the observed environment. All the processed data and predictions generated by the analysis steps are used as input of the *cognitive refinement* block. This module can be seen as a surrogate of the human operator: during the configuration of the system it is able to learn how to distinguish between different levels of potentially dangerous situations. This process can be done by manually labelling different zones of the observed data or by implementing a specific algorithm for the particular cognitive application. In the on-line phase, the CN works

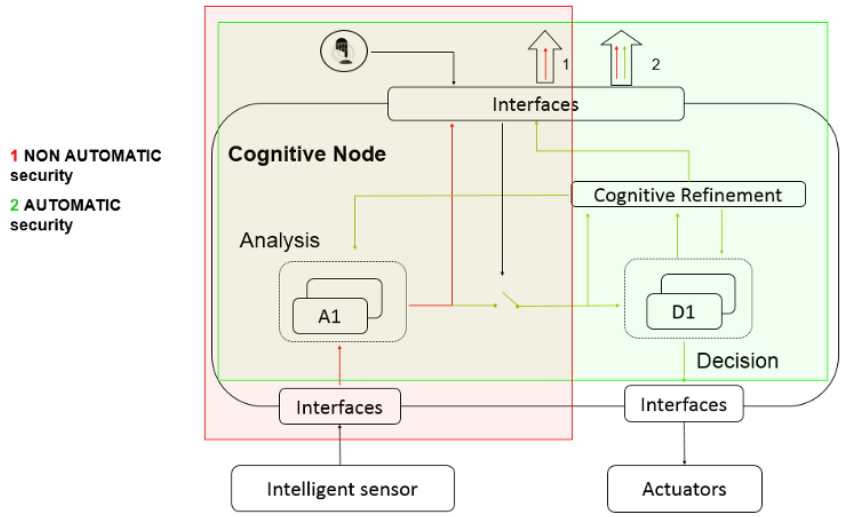


Fig. 4 Cognitive Node Architecture

in two different ways: for operator support and in automatic mode. In both cases the cognitive refinement module is able to detect if a predicted condition starts to drift away from the standard observed interaction, thus getting the overall system closer to a warning situation. Specifically, in the human support case, the switch, depicted in Fig. 4, is opened. The CN, by means of the cognitive refinement block, can detect anomalies as possible discrepancies from standard operator-crowd interactions. During the automatic mode, the switch is closed and the information contained into the cognitive refinement is employed to identify specific crowd-environment situations. The communication link towards the operator permits a direct warning about anomalous situations relative to crowd normal behaviours. Decision modules of the CN are responsible for selecting the best actions to be automatically performed by the system for avoiding a dangerous situation. Those actions can be performed on the fully cooperative parts of the observed system; all the decisions taken by the CN are made with the precise intent of maintaining the environment in a controllable, alarm-free state. If all the actions of the node are unable to keep the system in a standard state and the measured warning level continues to increase, the node itself can decide to stop the cognitive cycle and to give command of the controllable parts of the system back to the human operator, who is always given the possibility to decide on his own and completely bypass the automatic system or to be acknowledged of each single action that the CN is processing (Interfaces, Fig. 4).

As a final remark, we would like to point out that, as well as the proposed perception-action cycle for crowd monitoring, robot control mechanisms also are often motivated by biology. However, there are some conceptual differ-

ences between the two approaches. Robot control strategies, such as Reinforcement Learning, allow for optimizing actions by evaluating their rewards. The presented mechanism, based on Damasio's concept of Autobiographical Self, during an off-line phase, acquires and mathematically models interaction information by observations of two entities operator and crowd (i.e. proto and core). During the on-line phase, the cognitive system uses the previously stored knowledge for actively interacting with the external world. In the case of operator-crowd, a prediction mechanism drives the system actions, selecting the possible countermeasure according to learned rules during the training. The proposed algorithm is a general framework for acquiring and building up the rule sets in different context.

3 Information extraction and probabilistic model for knowledge representation

Interactions between two entities can be described in terms of mathematical relationships. Such a mathematical description must obviously rest on a feature extraction phase, which is addressed to get relevant information about the entities.

This section is devoted to the analysis of the main features that allow to design a probabilistic model able to learn interactions. After information is extracted, Dynamic Bayesian Networks (DBNs) can be used to represent cognitive cycles and IVCCs (Dore et al, 2010b), as already mentioned in section 2.1, based on the AM algorithm, thus providing a tool for describing embodied objects within the CN in a way that can allow incremental learning from experience. It was already pointed out that also interactions between the operator and the system can be represented as an IVCC. In that case, the operator-system interaction can be differently used as an internal reference for the CN as the operator can be seen as a teaching entity addressing most effective actions towards the goal of maintaining security/safety levels during the learning phase. This learning phase represents an effective knowledge transfer from human operator towards an automatic system.

A proposed framework for information extraction is composed of two main blocks: Data Fusion (DF) and Event Detection (ED). DF involves source separation and feature extraction: these two phases permit to recognize the same entities monitored by different heterogeneous sensors. The ED block extracts information related to changing in the signals acquired by sensors. Events will be eventually defined, in order to develop a specific probabilistic models able to describe different kinds of the relationships permitting to detect anomalous interactions.

3.1 Data fusion

Many different approaches can be used for designing architectures embedded on system, which are able to collect heterogeneous environmental information.

According to the functionalities provided by the systems, data fusion mechanisms should be considered as logical tasks which can be subdivided in a multi-modal architecture. An interesting method of the data fusion model is the JDL model (Hall and Llinas, 1997).

The JDL model includes five levels of processing, that represent the description of increasing level of abstraction (Dore et al, 2009). In our description, information on two distinct entities are fused and aligned at different levels.

The data fusion module is able to receive data from intelligent sensors on the field, and to fuse them from a temporal and spatial point of view. If one considers a set of S intelligent sensors, each $k \in S$ sends to the CN a vector of features $\mathbf{x}(k, t_k) = \{x_1, x_2, \ldots, x_{N_k}\}$ where $k = \{1, 2, \ldots, S\}$ at time instant t_k. Intelligent sensors send feature vectors asynchronously to the CN, that must be able to register them temporally and spatially before sending data to upper level processing modules.

From a temporal point of view, the data fusion module collects and stores into an internal buffer all newest measurements $\mathbf{x}_{k,t_k^*}$ from intelligent sensors $k = \{1, 2, \ldots, S\}$. The time instant t_k^* represents the last time when the feature vectors are acquired from each sensor that are received. Data acquisition time can vary from sensor to sensor.

As soon as a new feature vector is acquired from sensor k, the data fusion module can compute an extended feature vector by combining all measurements from all considered intelligent sensors $\varphi(\hat{t}) = f(\mathbf{x}_{1,t_1^*}, \mathbf{x}_{2,t_2^*}, \ldots, \mathbf{x}_{S,t_S^*})$, where $\hat{t} \geq \{t_1^*, t_2^*, \ldots, t_S^*\}$.

Thus the generation rate of the data fusion module can be estimated by considering the minimum time interval between two sequential measurements of the higher frequency sensor. If $\Delta t_k^n = (t_k^n - t_k^{n-1})$ is the time interval between arrival times of feature vectors $\mathbf{x}(k, t^n)$ and $\mathbf{x}(k, t^{n-1})$ for sensor k, the actual data rate of the fusion block can be estimated by computing $min_k(\Delta t_k^n)$.

The analytic expression of the fusion function $\varphi(\hat{t})$, depends on the physical relationship between measured quantities and cannot be studied with a generic approach. In the following scenarios, feature vectors are mainly generated by (real, but possibly simulated) video analytics algorithms that are able to process images acquired from video-surveillance cameras and extract scene descriptors (i.e., trajectories of moving objects, crowd densities within a certain environment, human activity related features, etc.).

In any case one can suppose that the fused feature vectors produced as output of this module have the following form:

$$\mathbf{x}(t) = \{\mathbf{x}_C, \mathbf{x}_P\} = \{\mathbf{x}_{C_1}, \mathbf{x}_{C_2}, \ldots, \mathbf{x}_{C_n}, \mathbf{x}_{P_1}, \mathbf{x}_{P_2}, \ldots, \mathbf{x}_{P_m}\}, \tag{1}$$

where n and m represent feature numbers of the core and proto state vectors, $\mathbf{x}_C$ and $\mathbf{x}_P$ respectively (i.e. the dimensionality of the vectors). Equation 1 expresses a general form for the global feature vector that is the result of the data fusion module. Vector $\mathbf{x}_C$ identifies features related to so-called *core* objects, i.e., entities that are detected within the considered environment but

that are not part of the internal state of the system itself. Vector $\mathbf{x}_P$ identifies *proto* object features that are specific for entities that can be completely controlled by the CN.

3.2 Event detection

The data fusion phase permits to obtain a high dimensionality core and proto *multi-dimensional space*, where each point represents a state vector of features at a specific time instant: $\mathbf{x}_P(t)$ and $\mathbf{x}_C(t)$. Using this representation it is possible to interpret the changes of state vectors as movements, *trajectories*, in a multi-dimensional space. Furthermore, as the dynamic evolution of one entity depends on the other entity, relationships between such trajectories describe interactions.

A Self Organizing Map (Kohonen, 1990) (SOM) unsupervised classifier is employed in this work at two different logic levels: first, to detect events in term of relevant state changing, secondly to represent complex relationships between the entities in a low-dimensional space. The latter logic level will be discussed in detail through the next sections. The SOM operates a dimensionality reduction, by mapping the multidimensional proto or core state vectors ($\mathbf{x}_P(t)$ and $\mathbf{x}_C(t)$) onto a lower M-dimensional space, where M is the dimension of the Kohonen's neuron layer (from here on we consider $M = 2$ without loss of generality). Input vectors are clustered according to their similarities, after the neural network is trained.

The choice of SOMs to perform feature reduction and clustering is motivated by their capabilities to reproduce in a plausible mathematical way the global behaviour of the winner-takes-all and lateral inhibition mechanism shown by distributed bio-inspired decision mechanisms. Besides, a SOM allows for the establishment of a representation of reality based on *analogies*: similar (though not necessarily identical!) input vectors are likely to be mapped by the Kohonen's map to the same neuron (in a non-injective way). Similarity is, in our case, measured by simple euclidean distance between vectors; however more complicated metrics can be employed to this end.

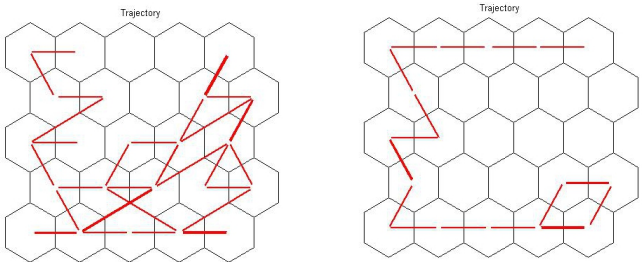
More in details, neural networks such as SOM, Neural Gas (NG) (Martinetz and Schulten, 1991) and Growing Neural Gas (GNG) algorithms (Fritzke, 1995) are inspired by Hebbians theory and permit the adaptation of neurons during the learning process. The Neural Gas represents a very interesting and powerful tool for vector quantization and data compression techniques. NG derives from SOM and it improves the input data topology preservation through an adaptive method based on neighbourhood relationships learning between the weight vector (associated to neuronal unit) and each external stimuli (associated to input vector). In this paper we have supposed that the global environment is divided in different rooms, each one controlled by cameras. A camera-embedded people counter is able to provide an estimation of number of people. The considered state vectors $\mathbf{x}_C$ are multidimensional and we are interested in reducing it to a 2-D space. However in other applications, where

it is highly desirable to conserve the topology, we have explored the possibility of automatically determining the set of regions to monitor according to environmental topology. In this case the input information can be the people trajectories and an we use the Instantaneous Topological Map (ITM) for learning structured input data manifold (Chiappino et al, 2013b). SOMs present a fixed number of neuronal units, while for GNG the number of neurons is automatically decided during the training phase. The study of the dimension of the reduced space is very important for us, because it is correlated to definition of the events. Fixing the dimension of the SOM layer it is possible to maintain limited the total number of possible events. A common learning problem, in designing models, is to acquire all possible configurations, i.e. all possible events. To this end, in this stage of our study, a fixed number of neurons is better than a self-adaptable topology. The Growing Hierarchical SOMs (GH-SOMs) represent another interesting tools (Rauber et al, 2002). They can increase the number of neurons and layers by means of distance measurements between neuronal weights and input data. These mechanisms of adapting layer sizes permit accuracy on original data reconstructions. On the other hand, we are interested in studying the optimum number of units for balancing the learning efficiency, the knowledge representation and the prediction capabilities of the AM. These facts will become clearer in section 4.1. A technique for the definition of contextual knowledge was prosed in (Marchesotti et al, 2005). By using a single 2-D SOM, an event classifications was obtained by fusing of the heterogeneous vectors, shown in Equation 1. But in this case the relationships between the entities are "fused" in the neurons. According to Damasios theory, by means of different SOMs, for separately mapping core and proto vectors, it is possible to detect relevant transitions between SOM neurons, i.e. the *events*. Such distinct core and proto events are basic units of the AM, which represents a bio-inspired fusing method for modelling the dependences between two entities (Chiappino et al, 2013c).

The clustering process, applied to internal and external data, allows one to obtain a mapping of proto and core vectors $\mathbf{x}_P(t)$ and $\mathbf{x}_C(t)$ in 2-D vectors, corresponding to the positions of the neurons in the SOM-map, that we call respectively proto *Super-states* Sx_P and core *Super-states* Sx_C. Each cluster of Super-states, deriving from the SOM classifiers, is then associated with a semantic label related to the contextual situation:

$$\begin{aligned} Sx_P^i &\mapsto l_P^i, \quad i = 1, \ldots, N_P \\ Sx_C^j &\mapsto l_C^j, \quad j = 1, \ldots, N_C \end{aligned} \tag{2}$$

where the notation Sx_P^i and Sx_C^j indicates that the Super-state belongs, respectively, to the i-th proto label and to the j-th core label; N_P and N_C are, respectively, the maximum number of the proto and core Super-states labels. Then, the result of this process is the building of a 2D map divided in regions labelled with a meaningful identifier related to the ongoing situation. It must be noted that changes of state vectors $\mathbf{x}_P(t)$ and $\mathbf{x}_C(t)$ do not necessary imply a change of Super-state labels $Sx_P^i \mapsto l_P^i$ and $Sx_C^j \mapsto l_C^j$. This means that



(a) (b)

Fig. 5 Examples of temporal proximity *trajectories* among fired neurons in 2-D SOM-map (5×5) for different core state vector sequences. The trajectories are non-linear and discontinuous.

there are some modifications which are irrelevant from the point of view of the chosen semantic representation of the situation. On the other hand, when the Super-state labels Sx_P^i and Sx_C^j change in subsequent time instants, a contextual situation modification, i.e. an *event* occurs. It is then possible to define proto (ϵ_P) and core (ϵ_C) events. Actually, even *null* events (i.e. null changes in Super states) can be defined. In fact, these could be relevant while considering very slowly changing systems and dynamics or whenever lengthy immobility could become relevant.

A Kohonen's layer consists of a 2-D matrix of neurons, identified by an hexagonal location. The network is constructed based on competitive learning: all the output neurons that win the competition are subsequently activated by input state vectors. Two SOM-nodes are considered as near if they are consecutively active at two successive time instants. It is possible to connect all fired neurons describing a temporal proximity *trajectory* among neurons. Not necessarily different input state sequences describe different trajectories in the Super-state space. By sequentially analysing the dynamic evolution of Super-states, proto and core events can be detected and visualized by trajectories into 2-D SOM-map.

The output of the SOM algorithm is in fact a list of clusters (or zones) within the Kohonen's layer, that describe a trajectory. Two trajectories for two different core state sequences are presented in Fig. 5. The ED module also considers dynamical aspects of the evolution of clustered features: transition probabilities between different Super-states (i.e. zones) are computed, in such a way that the outcome of the training process can be ideally compared to a DBN. Instead of considering sequences of Super-states to describe the evolution of each entity, it is possible to consider proto and core *event* series, which can be modelled by two Event based DBNs (E-DBNs) (Patnaik et al, 2009) as explained in the next section.

3.3 Autobiographical memory

According to Damasio's theory, the sequences of proto (internal) and core (external) events can be organized into two kinds of triplets in order to account for interactions between entities: $\{\epsilon_P^-, \epsilon_C, \epsilon_P^+\}$ (passive interaction) and $\{\epsilon_C^-, \epsilon_P, \epsilon_C^+\}$, (active interaction), to represent the causal relationships, in terms of initial situation (first event), cause (second event) and consequent effect on the examined entity (third event) [1].

The resulting information becomes an approximation of what Damasio himself calls the *Autobiographical Memory* where these triplets, which encode possible interactions between entities, are memorized. The basic idea behind the algorithms is to estimate the frequency of occurrence of the effects caused by a certain external event in order to derive two probability distributions:

$$p(\epsilon_P^+|\epsilon_C, \epsilon_P^-), \tag{3}$$

$$p(\epsilon_C^+|\epsilon_P, \epsilon_C^-), \tag{4}$$

representing the causality of observed events in the interaction. The sequence of events is represented by a statistical graphical model in order to introduce a mathematical description of the proposed interaction model. This choice is motivated by the fact that the interaction pattern is composed by a temporal sequence of interdependent events and then it can be seen as a stochastic process. Therefore, an approach for modelling sequences of events that relies on a probabilistic model results particularly appropriate.

The interaction patterns are modelled by a Coupled Event based DBN (CE-DBN) in order to have a representation able to statistically encode the relevant data variability. The proposed CE-DBN graph, shown in Fig. 6, aims at describing interactions taking into account the neuro-physiologically motivated model of the Autobiographical Model. The conditional probability densities (CPD) $p(\epsilon_t^P|\epsilon_{t-1}^P)$ and $p(\epsilon_t^C|\epsilon_{t-1}^C)$ encode the motion pattern of the objects in the environment regardless the presence of other objects. Note that each triplet can be seen as one dispositional statement (configuration) with an associated conditional probability, Equations 3 and 4. The AM provides a dispositional description, a set of dispositional proprieties, for proto and core entities.

The dispositional proprieties describe a precise objective: to maintain stability of the equilibrium between the object and the environment (i.e. maintenance of the proper level of security and/or safety). Anomaly can be seen as a deviation from the normality and it can be considered as a violation of a certain dispositional propriety. The interactions between the two objects are

[1] An active interaction (represented by a triplet) is defined when an internal modification (proto event) is the cause of external environmental change, i.e. the third event in the triplet is a core event. Vice versa the passive triplet is defined when an external environmental change (core event) provokes an internal modification, i.e. the third event in the triplet is a proto event.

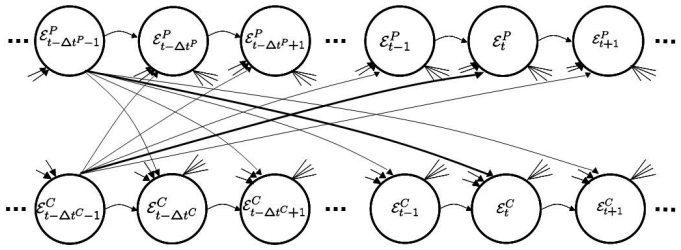


Fig. 6 Coupled Event based Dynamic Bayesian Network model representing interactions within an AM

described by the CPDs:

$$p(\epsilon_t^P|\epsilon_{t-\Delta t^C}^C), \tag{5}$$

$$p(\epsilon_t^C|\epsilon_{t-\Delta t^P}^P). \tag{6}$$

In particular, Equation 5 describes the probability that the events ϵ^C, occurred at time $t - \Delta t^C$ and performed by the object associated to the core context, has caused the event ϵ^P in the proto context. Reversed interpretation in terms of causal events should be given to $p(\epsilon_t^C|\epsilon_{t-\Delta t^P}^P)$.

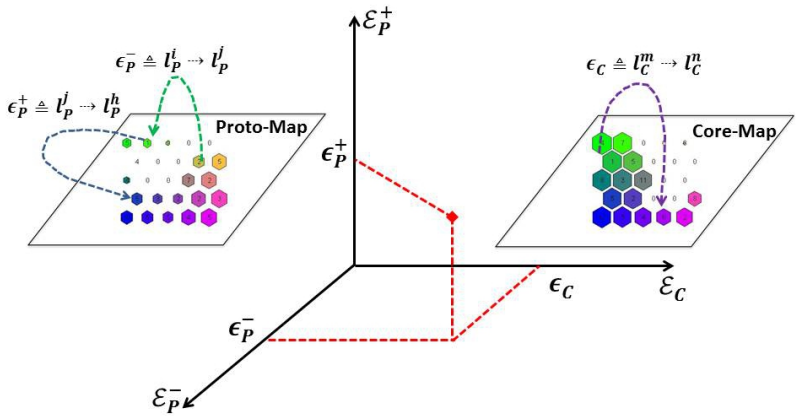


Fig. 7 Graphical representation of the mapping into AM 3-D space of passive triplet $\{\epsilon_P^-, \epsilon_C, \epsilon_P^+\}$. The symbols $l_{P/C}^x$ represent the contextual SOM-label associated to each cluster. In this example the proto or core events are represented by: $l_{P/C}^x \dashrightarrow l_{P/C}^y$, where $x \neq y$. The transitions into Proto and Core-Map are dashed for representing the non-linearity and discontinuity of the trajectories.

Considering the definition of the core consciousness, the causal relationships between the two entities are encoded in two conditional probability densities (CPDs):

$$p(\epsilon_t^P|\epsilon_{t-\Delta t^C}^C, \epsilon_{t-\Delta t^P}^P) \tag{7}$$

$$p(\epsilon_t^C|\epsilon_{t-\Delta t^P}^P, \epsilon_{t-\Delta t^C}^C) \tag{8}$$

As a matter of fact, the probability densities in Equations 7-8 consider both the interaction (i.e. Eq. 5 or Eq. 6) and the initial situation (i.e. $\epsilon^P_{t-\Delta t^P}$ or $\epsilon^C_{t-\Delta t^C}$).

4 Autobiographical Memory domain applications: Surveillance and Crowd Management scenarios

In the previous section a probabilistic model based on CE-DBN was sketched in order to describe multiple entity interactions. The knowledge thus represented inside the proposed CN can be employed in many different domains: surveillance scenarios and crowd analysis-management are just two limited examples. Generally, in surveillance scenarios the goal of the system is the analysis of interactions and recognition of specific behaviour between two or more people (external entities). On the other hand, in the crowd analysis domain, the focus of the system can be shifted toward the analysis and classifications interactions that occur between the crowd and a human operator who is in charge of maintaining a proper security level within the monitored area (for this purpose, the crowd can be seen as a unique macro-entity). The two entities can be represented as a couple of $IVCC$s, as proposed in section 2.2, namely an $IVCC_O$ and an $IVCC_s$ respectively.

In this section two aspects will be discussed, namely the probabilistic model learning phase and the detection phase for surveillance and crowd scenarios. During the (off-line) learning phase the CN observes both entities, i.e. the human operator and the crowd, storing their interactions within the AM. As for the (on-line) detection phase, it will be shown how different definitions of the probabilistic model are needed.

The system is designed to support a human operator in crowd management during the on-line phase. This task is accomplished by recognizing specific operator-crowd abnormal interactions. Typically, in people flow redirection problems, an abnormal interaction can be detected whenever the user puts in action wrong countermeasures to avoid the panic or overcrowding situations. In this case the CN ought to drive the operator to choose correct actions by either simply signalling the anomaly or by suggesting actions to be performed based on its learned experience.

4.1 Learning phase: interaction representations

During an off-line phase, the CN is able to learn and store into the AM a set of triplets (i.e. interactions) for different *situations*: $\{\epsilon^-_P, \epsilon_C, \epsilon^+_P\}$ (passive triplet) and $\{\epsilon^-_C, \epsilon_P, \epsilon^+_C\}$ (active triplet). The crowd configurations are captured by *core sensors*, while the operator actions are mapped into *proto sensors*. Each triplet represents a point of a 3-D space. In Fig. 7 an example of 3-D space mapping of a passive triplet is depicted. This representation allows to sketch the set of triplets stored into an AM. We point out that the ordering of the

events along the $\mathcal{E}_P^-$, $\mathcal{E}_C$ and $\mathcal{E}_P^+$ axes is not relevant as what is really significant is only the number of occurrences of a certain triplet. However, each generic triplet of events can be associated to an *influence model*, i.e. a specific AM can model the dynamic evolutions of interactions for a specific context. It is possible to define a *switching variable* θ as influence parameter (Pan et al, 2012).

Each triplet is associated to a probability, derived from an estimate of two conditional probability densities: $p(\epsilon_P^+|\epsilon_C, \epsilon_P^-, \theta)$ and $p(\epsilon_C^+|\epsilon_P, \epsilon_C^-, \theta)$ (cfr. 7 and 8), which are proportional to the number of votes that the particular triplet received, i.e. the number of occurrences observed during the AM training phase that represents a specific interaction (i.e. an influence model). Fig. 8 shows an example of conditional relationship for a passive triplet: ϵ_P^+ given the two previous events ϵ_C ϵ_P^- and the interaction model θ.

A temporal histogram is associated to each AM element (i.e. to each triplet), in order to store the temporal information related to events of the triplets. For example, taking into consideration a passive triplet $\{\epsilon_P^-, \epsilon_C, \epsilon_P^+\}$,with given events, the histogram permits to evaluate the probability that a specific proto event ϵ_P^+ takes place τ_{CP^+} seconds after the core event ϵ_C. The histogram bin dimension must be selected by performing a trade off between the precision of the temporal prediction that it is required by the application and the number of training samples available.

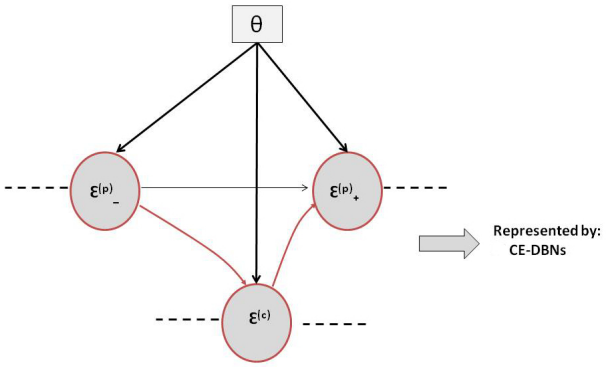


Fig. 8 Example of CE-DBNs for passive triplet, e.g. $\{\epsilon_P^-, \epsilon_C, \epsilon_P^+\}$, with a parameter θ tied across proto-core-proto transitions.

4.2 Detection phase: surveillance scenarios

After a learning phase, the CN, by using the AM, has the capability of recognizing the interactions while they take place, in an on-line timing. In (Dore et al, 2010b) the exploitation of an AM for the detection of different kinds of interactions between two people was proposed. For this reason, a cumulative measure

is computed exploiting the information encoded in the proposed Coupled E-DBN model. To accomplish this task, for each interaction $i : i = 1, \ldots, N_I$, where N_I is the number of considered interactions, a set of couple of trajectories (core and proto) are used to train the model (θ_i), originating a trajectory into a 3-D space (as shown in Fig. 7). To detect the type of cause-effect relationship between entities, for each triplet $(\epsilon_t^{P,C}, \epsilon_{t-\Delta t^{C,P}}^{C,P}, \epsilon_{t-\Delta t^{P,C}}^{P,C})$ the following measure is computed:

$$l_t^i = l_{t-\Delta t^{C,P}}^i + p(\epsilon_t^{P,C}, \epsilon_{t-\Delta t^{C,P}}^{C,P}, \epsilon_{t-\Delta t^{P,C}}^{P,C} | \theta_i), \tag{9}$$

where $l_{t-\Delta t^{C,P}}^i$ is the measure computed at the time in which the previous event has been observed and with $p(\epsilon_t^{P,C}, \epsilon_{t-\Delta t^{C,P}}^{C,P}, \epsilon_{t-\Delta t^{P,C}}^{P,C} | \theta_i)$ the probability that the observed triplet belongs to the i-th interaction model is indicated. For each triplet of events, the best matching influence model is chosen as $i^* = \arg\max_i l_i$ with $i = 1, \ldots, N_I$. The high level of criticality of the detection phase entails that, if mismatching between the observed data and learned knowledge is detected, the system can call the attention of the operator. In this case the learning phase starts up again.

4.3 Detection phase: Crowd management scenarios

In human-to-human interactions, at each state change of one entity typically corresponds a state change of the other. In this case it is possible to affirm that the entities have the same (or at least a similar) dynamic. On the contrary, in crowd scenarios, the dynamics of the entities are extremely different, namely the crowd changes its status more frequently than the operator. Generally the number of people, in a room or in a zone, can change without any operator actions. In all the cases in which the dynamics between entities show significant differences, the AM can be considered as a *sparse* collection of triplets. In order to design a robust classification algorithm for abnormal interaction recognitions, an approach to encode a statistical sparse model using the Self Organizing Map is needed. The following section is dedicated to this scope.

5 Proposed approach for abnormal interaction detection in crowd monitoring domain

The proposed cognitive video surveillance system has two main purposes. The first and most important one is to detect the interaction anomalies between operator and crowd. The second is to substitute or to help the user during the crowd management, recognizing anomaly interactions with crowd. The presented cognitive system accomplishes both these goals by learning a specific behavioural model for operator-crowd interactions, in which the crowd is correctly controlled by a user. This model describe normal conditions of crowding management. CN can detect anomalous operator-crowd interactions

as deviation from normality situation. In automatic operating mode, the system substitutes the operator and interacts directly with the crowd. When crowd reaction patterns are not conform to expected behaviour an anomalous configuration (i.e. interaction) is detected. The method used for interaction modelling and above mentioned anomalous detections is here presented. An interaction behaviour cannot be completely represented by a triplet alone: a set of triplets must be analysed in order to individuate a model. A common learning problem can be formalized as follows: the generic sequence of triplets $Tr_j = \{\epsilon^-_{P,C}, \epsilon_{C,P}, \epsilon^+_{P,C}\}$, $j = 1, \ldots, N_T$, where N_T is the number of triplets in that specific sequence, can belong to different observed models θ_i, $i = 1, \ldots, N_I$ (N_I is the number of operator-crowd interaction models). Fig. 9 shows triplet *encoding* by means of a mapping function $f(.)$. For sparse collected data, i.e. sparse triplets, the *mapping function* defined as $f(\epsilon^-_{P,C}, \epsilon_{C,P}, \epsilon^+_{P,C}) = p(\epsilon^+_{P,C} | \epsilon_{C,P}, \epsilon^-_{P,C}, \theta_i)$ is not potentially useful in order to distinguish triplet associations with specific kinds of operator-crowd interaction models.

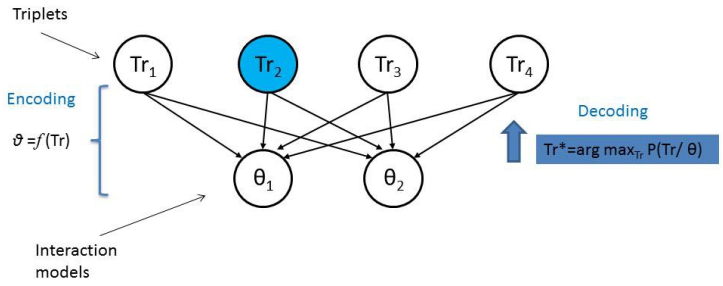


Fig. 9 Model learning problem: triplet recovering from model. Tr_j represents j^{th} generic triplet, θ_i is i^{th} interaction model.

A different transform function, $\hat{f}(\epsilon^-_{P,C}, \epsilon_{C,P}, \epsilon^+_{P,C})$, is defined for triplet mapping into 2-D space to decrease miss-classification errors. A specific dimensionality reduction method can be employed to encode the AM. In this way, it is possible to obtain a probabilistic model for *rare-interaction* detections, in order to describe high-complexity relationships between entities by means of simpler formulas (Rish and Grabarnik, 2009). The mapping function $\hat{f}(.)$ must meet the following requirements: maximum information preservation of the operator-crowd interactions and correct reconstruction of original data optimizing the classification accuracy.

5.1 Dimensional reduction and preservation of the information

A large number of methods have been addressed for dimensional reduction: they are typically classified in linear and non-linear methods. This section

addresses a fundamental issue rising in reduction problems: interaction information contained in primary data must be preserved. Two well-known feature reduction techniques, namely Principal Component Analysis (PCA) (linear method) (Shlens, 2005) and Self Organizing Map (non-linear method) are compared.

In Table 1, a comparison between PCA and SOM is presented, where binary formats are varied for output vectors encoding. The binary format is expressed by $[wl, fl]$, where wl represents word-length and fl is the fraction-length. In particular Table 1 presents the error measures, calculated as average Euclidean distance, between original data D and reconstructed data $\hat{D}$. It is possible to note that, by increasing the number of bits, the SOM behaves better than PCA.

Table 1 SOM and PCA comparison

Binary Format	SOM-map	PCA err	SOM err
$[3, 1]$	8×8	0.1857	1.4430
$[5, 1]$	32×32	0.1954	0.0938
$[5, 2]$	32×32	0.0846	0.0938
$[6, 2]$	64×64	0.0803	$2.8175 \cdot 10^{-7}$

5.2 Self Organizing Map: classification evaluation

Taking into account a SOM layer formed by K neurons, its dimensions are adapted in order to find the best matching couple (l, w) such that $l \times w = K$. The number of core (or proto) Super-states is then K and the total number of possible core (or proto) events is K^2, taking null-events as relevant as explained in (Chiappino et al, 2013e). The parameter K must be tuned: in fact, by decreasing the SOM-map size, many different input state vectors can fall into the same cluster: this fact generates a rougher classification but ensures that only relevant events are likely to be selected. On the other hand, by employing high-dimensional Kohonen's layers, the classification is improved, whereas the total number of irrelevant events increases.

The dimension of the layer is a relevant parameter in our system. A small layer allows the system to summarize its knowledge in a few concepts, which is positive, although classification of situations may result too rough in some cases. On the other hand, very large layers result in a very sparsely populated Superstate space, meaning that the system would need massive training in order to observe, and later recognize, any possible situation. At the moment such a parameter was empirically tuned.

We define a data set D as follows:

$$D = \{D(t) : t \in 0, \ldots, T\}, \tag{10}$$

where $D(t) \in \mathbb{R}^N$ is a vector $D(t) = [d_1(t), \ldots, d_N(t)]'$ in which each component $d_i(t)$ will represent, in our application (section 6), the number of

Table 2 Classification evaluation for different SOM-layer

SOM-map	$H(D)$	$H(D\|C)$	$I_M(D,C)$
2×2	6.3750	0.7249	5.6501
4×4	6.3750	0.1291	6.2459
5×5	6.3750	0.0384	6.3366
7×7	6.3750	0	6.3750

people in the i^{th} room at instant t. The clustering process performed by the SOM is defined by means of a transformation function $f_n(D) : D \rightarrow S$ with $S = \{S_k(t) : t \in 0, \ldots, T\}$, $k = 1, \ldots, K$ is the index of the neuron and T maximum training time. The vectors $S_k(t) \in \mathbb{R}^M$, with $M < N$ ($M = 2$ in our case), represent the coordinates into the SOM Map of the neurons fired at the time t. Each element of the data set can be determined as: $D(t) = C_k(t) + n_k(t)$, where $C_k(t) \in \mathbb{R}^N$ is the vector of weights for the k^{th} neuron which is associated with $S_k(t)$. $n_k(t)$ can be considered as a Gaussian noise $\mathcal{N}_k(0, \Sigma_k)$. The covariance matrix Σ_k is computed in each k^{th} SOM node considering all the training vectors which have activated the k^{th} neuron. It is possible to define a conditional probability density function $p(D|C_k)$ as follows:

$$p(D|C_k) = [p(C_k)\, w_k(t)]^{-1} exp - \{\sum_{t=0}^{T} \Omega(t)' \Sigma_k^{-1} \Omega\}, \tag{11}$$

where $p(C_k)$ is the probability neuron activation and it is computed as the number of samples in the node over the total number of training samples. $\Omega(t) = D(t) - C_k+$ and $w_k = [(2\pi)^{N|2}(det(\Sigma_k))^{0.5}]$.

A possible criterion to evaluate a SOM, given a data set D, relies on *Average Mutual Information* (AMI) $I_M(D, C)$, (Finn, 1993), defined by Equation 12:

$$I_M(D, C) = H(D) - H(D|C), \tag{12}$$

where $H(D)$ is the data set entropy, while the conditional entropy of the normal multivariate distribution of $p(D|C) = \sum_{k=1}^{K} p(D|C_k)$ is defined as

$$H(D|C) := 0.5 \ln[(2\pi e)^N |\Sigma|], \tag{13}$$

where Σ is the covariance matrix of normal multivariate p.d.f. $p(D|C)$. To investigate the capabilities of the Self Organizing Maps we set up a test: an artificial data set D for training was constructed consisting of 143 vectors, with sampling time equal to $4[s]$, provided by our crowding simulator. Each vector is formed by $N = 6$ components and contains the number of people in each room.

Table 2 lists entropies for different size of the SOM layer. Over 7×7 the quality of the classification have not significant improvements from AMI point of view. However, we are not concerned with an extremely precise description of the core state space (we do not want to maximize $I_M(D, C)$ at all costs). We certainly need a sufficient amount of information to be preserved, but at the

same time, as explained in this subsection, we need our system to be capable of synthesizing knowledge by establishing analogies.

A representation of reality based on analogies is necessary in order to deal with situations never seen during the training phase.

The SOM can be used for dividing a set of training data D into different multivariate time series $\{D_k\}_{k=1}^{K}$ where $D_k = \{D_{1,k}, \cdots, D_{n,k}\}$ associated to the $k-th$ neuron, such as $D_k \cup D_j = \varnothing$ with $k \neq j$ and $\bigcup_{k=1}^{K} D_k = D$. These sub-sequences of vectors can be modelled by local *Vector Auto Regressive* (VAR) models (Pfaff, 2008). The number of generated VAR models correspond to the number of neurons of the SOM. The local matching measurement between the sequence of input data and the output of the local VAR models specifies how much of the output variation has been represented by the SOM. Considering a multivariate time series D_k, an auto regressive model of order p, denoted as VAR(p), describes $i-th$ vector $D_{i,k}$ as linear combination of the previous state vectors:

$$D_{i,k} = \Phi_0 + \Phi_1 D_{i-1,k} + \Phi_2 D_{i-2,k} + \cdots + \Phi_p D_{i-p,k} + \epsilon_i, \tag{14}$$

where $\Phi_0, \cdots, \Phi_p$ are $(N \times N)$ parameter matrices and ϵ_i represents a $(N \times 1)$ white noise. By the multivariate time series D_k we have modelled a VAR(2) as $D_{i,k} = \hat{\Phi}_0 + \hat{\Phi}_1 D_{i-1,k} + \hat{\Phi}_2 D_{i-2,k} + \epsilon_i$, where $\hat{\Phi}_0$, $\hat{\Phi}_1$ and $\hat{\Phi}_2$ are estimated coefficient matrices which have stored in each SOM node. Each VAR model has been used as linear predictor filter. A dataset D^c, different from D, has been used for the classification phase. Also in this case the SOM divides the data into different multivariate time series $\{D_k^c\}_{k=1}^{K}$ where $D_k^c = \left\{D_{1,k}^c, \cdots, D_{n,k}^c\right\}$ associated to the $k-th$ neuron, such as $D_k^c \cup D_j^c = \varnothing$ with $k \neq j$ and $\bigcup_{k=1}^{K} D_k^c = D^c$. We have compared one period ahead forecast sequences $\hat{D}_k$ obtained by VAR(2) model built over different SOM layer sizes with D_k^c. Fig. 10 shows an example of curve trends for predicted vector sequences by VAR(2) model built over different SOM layer sizes. A comparison between simulated data and the predictor filter outputs is provided by *FIT* measurement:

$$FIT = \frac{\sum_{i=1}^{n} \|D_{i,k}^c - \hat{D}_{i,k}\|}{\sum_{i=1}^{n} \|D_{i,k}^c - \overline{D}_k^c\|}, \tag{15}$$

where $D_{i,k}^c \in D_k^c$, $\hat{D}_{i,k}$ is the output of the $k-th$ VAR(2) model and $\overline{D}_k^c = E\{D_k^c\}$. The averages of the *FIT* between one period ahead forecast obtained by VAR(2) models and simulated data (formed by 140 vectors) show that a SOM with small layers are able to build analogies between the stored data into the same neurons during the training phase and the classification phase.

After a training phase of the chosen SOM, a mapping function $\hat{f}(Tr_j)$ to project each triplet into 2-D SOM-map can be defined. The output of this function is a list of zones, i.e. trajectories (which can be actually discontinuous), within the SOM-map. Dynamic aspects of the evolution of clustered

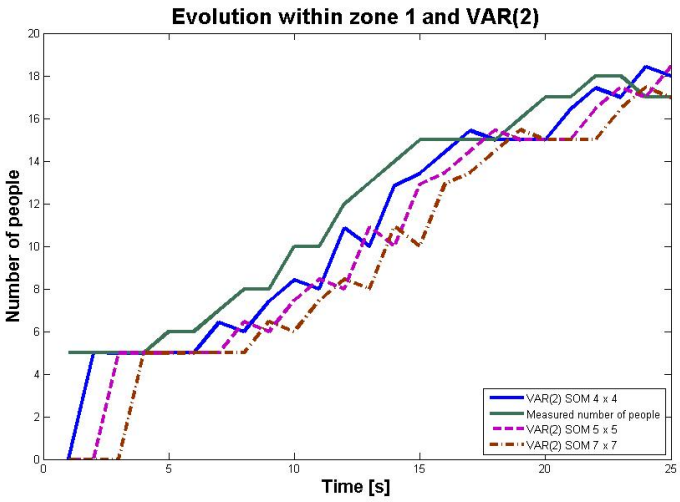


Fig. 10 Example of graphical comparison between VAR(2) models and simulated data which represent the number of the people within the zone 1. The averages of the matching between VAR(2) model outputs and 140 simulated vectors (expressed in percentages) are the following: SOM 4×4 fit: 67.14%; SOM 5×5 fit: 53.6%; SOM 7×7 fit: 40.18%.

triplets are also considered: transition probabilities between different zones are computed, in such a way that the outcome of the training process can be ideally compared to an Hidden Markov Model (HMM) (Oliver and Pentland, 2000).

6 Results

The simulated monitored environment is shown in Fig. 11. The configuration of doors, walls and rooms is however customizable and a wide range of scenarios can be set for tests. A crowd enters a room of the simulator and is given the motivation of moving toward the exit of the building. *Births* of new characters occur during the simulation, modelled by a Poisson distribution (we hypothesize a fixed average incoming rate: data coming from different simulations are thus comparable). The human operator must act on door configuration in order to direct crowd flows, thus preventing overcrowding

The use of a graphical engine (freely available at `http://www.horde3d.org/`) has been introduced in order to make the simulation realistic in the AM (section 4) training phase. Here a human operator acts on doors configuration in order to prevent room overcrowding, based on the visual output, which need to be as realistic as possible. Namely, the simulator has to output realistic data both from the behavioural point of view, in order to effectively interact with the human operator, and from the visual point of view, in order to grant an effective interface by truly depicting reality. Reactions of an operator faced

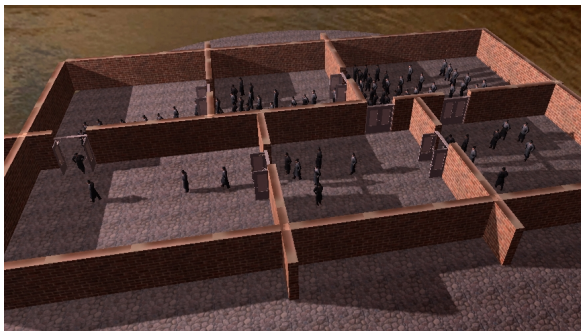


Fig. 11 The simulated monitored environment.

with an unrealistic visual output could be extremely different and strongly depend on rendering quality. For this reason, characters are also animated to simulate walk motion (at first glance a crowded environment with still people could look less populated than it really is).

Crowd behaviour within the simulator is modeled based on Social Forces, which were mentioned in section 1. This model assimilates each character on the scene to a particle subject to 2D forces, and treats it consequently from a strictly physical point of view. Its motion equations are derived from Newtons law $\mathbf{F} = m\mathbf{a}$. The forces a character is driven by are substantially of three kinds (Luber et al, 2010). An attractive motivational force $\mathbf{F}_{mot}$ pulls characters toward some scheduled destination, while repulsive physical forces $\mathbf{F}_{phy}$ and interaction forces $\mathbf{F}_{int}$ prevent from collision into physical objects and take into account interactions within characters. An additional linear drag (viscous resistance) $\mathbf{F}_{res}$ takes into account the fact that no character actually persists in its state of constant speed but tends to stop its motion as motivation runs out. This force is in fact accounted for and included in $\mathbf{F}_{mot}$. Such forces are sketched in Fig. 12. Chaotic fluctuations are included, according to the modified social force model proposed in (Soh et al, 2004). These fluctuation account for random individual variations in behaviour and lead to crowd motion self organization.

The three forces are estimated at each time instant for each character, whose position is then updated according to the motion equation and normalized according to the current fps rate supported by the graphical engine (which strongly depends on the number of characters to be handled). As already mentioned, people incoming rate is modelled as a Poisson distribution. Their *death* occurs as they get to their final scheduled destination. A human operator interacts with the crowd by opening doors to let it flow, while trying to minimize the time a door remains open. Although somehow simplified with respect to more complex works, such as (Luber et al, 2010) (where additional assumptions on trajectories' regularity are made), the developed model results in a good overall output, where people behave correctly and realistically.

The simulator also includes (simulated) sensors. These try to reproduce (processed) sensor data coming from different cameras looking at different

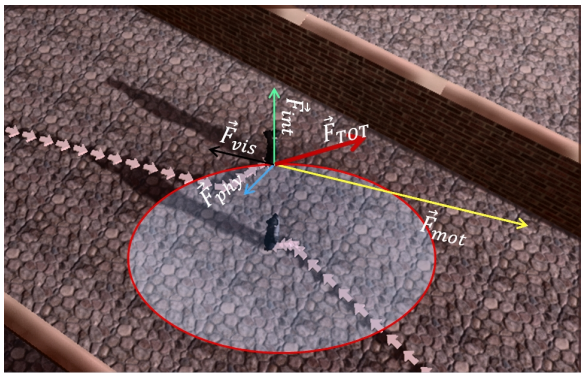


Fig. 12 Vectorial sum of forces $\mathbf{F}_{TOT}$ and influence range of characters.

subsets (rooms) of the monitored scene. A virtual people estimation algorithm outputs the number of people by simply adding some noise to the mere number of people framed by the virtual camera, thus trying to mimic the output of real image processing algorithms, such as (Kilambi et al, 2008). The state vector of the system (which corresponds to the external object, *eso*) is (cfr. equation 10)

$$\mathbf{X}_{Cr}(t) = \{x_{Cr_1}(t), x_{Cr_2}(t), \ldots, x_{Cr_N}(t)\}\,, \tag{16}$$

with $N = 6$ in our case (six cameras, one for each room). $x_{Cr_n}(t)$ is the number of people in room n. The people flow starts in a *hall room*, that corresponds to x_{Cr_1}. A 7×7 2D SOM is then trained in order to cluster the state vector space. The SOM Super States (better say, their variations) define events. The internal (endo) state of the system (namely, the doors' configuration) is simply modelled by a binary vector storing the state of each door (true if open, false if closed). Variations of such a vector define proto events.

An AM is then trained by a human operator who opens virtual gates in order to let the crowd stream outside in case high occupancy states are reached and, at the same time, to minimize the time gates remain open.

In our case, four kinds of simulated scenarios for different crowd behaviours (see Table 3), have been taken into consideration, in order to memorize the interactions between a human operator (proto-self) and the crowd (core-self) as formalized in 2. For instance, the first crowd behaviour, identified by $1d$, has $\mu = \sigma^2 = 1$ for the Poisson probability mass function, weak interaction force, and a relatively short interaction range.

After mapping the AM into a 2-D space, by means of a SOM, the operator's reactions to different crowd fluctuations, stored and encoded by $\hat{f}$, can be used on-line to choose an optimal strategy, i.e. to emulate the actions of a human operator, by predicting not only his behaviour but also crowd's reaction to it.

A reference model θ_i for operator-crowd interactions is then designed (refer to Fig. 9). We define a sequence of passive triplets (related to $i = 1d$ crowd behaviour, Table 3) as:

Table 3 Different crowd behaviours in simulated scenarios

ID	Num. persons/second	Interaction range	Interaction Forces
1d	1	1 m	Low
2d	2	2 m	Low
1p	1	1 m	High
2p	2	2 m	High

$$\{Tr_k\} = (Tr_1, Tr_2, \dots, Tr_k, \dots, Tr_K), \tag{17}$$

where $Tr = \{\epsilon_P^-, \epsilon_C, \epsilon_P^+\}$. The mapping function $\hat{f}(Tr_k) = S_k$ defines a corresponding sequence of Super states into the SOM-map as follows: $\{S_k\} = (S_1, S_2, \dots, S_k, \dots, S_K)$. In the simulation, the maximum time between two subsequent Super states $(\dots, S_k, S_{k+1}, \dots)$ is taken as $8[s]$. After such a time lapse, a new interaction (Super state) is considered. The k^{th} Super state probability is a vector P whose elements are defined as: $P_k = P(S_k)$; it corresponds to the number of occurrences of S_k over $\{S_k\}$ with $k = 1, \dots, K$. The elements of the transition probability matrix M, are defined as: $M(S_k, S_{k+1}) = P_{k+1|k} = P(S_{k+1}|S_k)$.

A test sequence of passive triplets $\{Tr_k^{ID}\}$ (one for each crowd behaviour listed in Table 3) is simulated and processed by $S_k^{ID} = \hat{f}(Tr_k^{ID})$ in order to generate $\{S_k^{ID}\}$ with $k = 1, \dots, K$. A weighted average of transition probabilities between subsequent Super states $(\dots, S_k^{ID}, S_{k+1}^{ID}, \dots)$ is computed as follows:

$$P_i^{ID}(t) = \frac{1}{W} \sum_{k=1}^{W} P_k^{ID} P_{k+1|k}^{ID}, \tag{18}$$

where $P_k^{ID} = P(S_k^{ID}|\theta_i)$ and $P_{k+1|k}^{ID} = M(S_k^{ID}, S_{k+1}^{ID}|\theta_i)$, while W, called *moving evaluation windows*, defines the number of test sequence triplets considered at each step t. We define the probability to reach $k+1^{th}$ Super state starting from the k^{th}, as follows: $P_{k \mapsto k+1}^{ID} = P_k^{ID} P_{k+1|k}^{ID}$. The recognition of the interaction model is performed by taking the maximum probability: $(i^*, t) = \arg\max_i P_i^{ID}(t)$ with $i = 1, \dots, N_I$ and $t = 1, \dots, T$. The couple (i^*, t) defines the kind of recognized operator-crowd interaction θ_i and also the maximum time $W \cdot 8 + t \cdot 8[s]$ in which the detection is performed.

Different average of transition probabilities curves, with $W = 2, 5, 10, 15$ and $T = 10$ steps, are evaluated. An example with $W = 10$ is shown in Fig. 13. The four interaction behaviours (red curve) are compared with the reference model (blue curve). Using only a few triplets (i.e. lower W values, e.g. $W = 2$ and $W = 5$) for each time step, some behaviour models result confused. The separation distance between the curves increases when the evaluation window values increase, e.g. with $W = 10$ and $W = 15$.

The Mean Square Error (MSE) is computed, in order to evaluate the distances between the observed interaction behaviour curves and the reference behaviour model. The minimum MSE provides a similarity measure

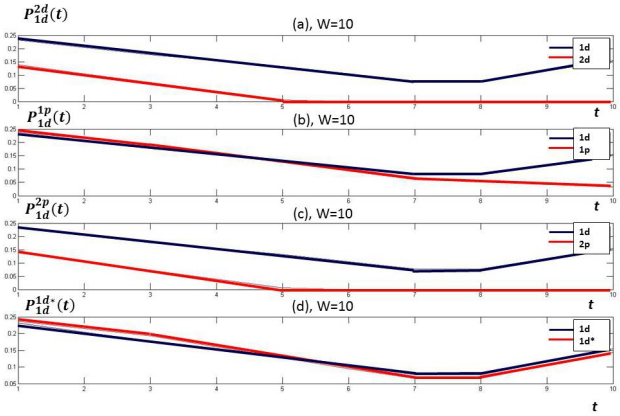


Fig. 13 Classification examples of interaction behaviour using evaluation window W=10.

between interaction behaviours. At each time step $t = 1, \ldots, T$ as follows: $MSE(t) = \frac{1}{W}\sum_{k=1}^{W}(P^*_{k\mapsto k+1} - P_{k\mapsto k+1})^2$, where $P_{k\mapsto k+1}$ and $P^*_{k\mapsto k+1}$ correspond to probability values over $\{S_k\}$ and $\{S^*_k\}$, i.e. reference and observed sequences. The anomalous interactions between an operator and the monitored crowd could emerge after a normal behaviour, e.g., a careless user does not open some doors. In this situation the CN, working in its on-line modality, is able to recognize anomalous crowd management. Fig. 14 shows the normal behaviour, in the specific case of $ID = 1d$ (blue curve) and compares it the with observed operator-crowd interactions (red curve). Using an evaluation window $W = 10$, two processes start to drift away at $t = 6.4[s]$. In a corresponding manner MSE starts to grow up. The rule of detection is $\nabla MSE(t) > 0$ for $t \in [t_{min}, t_{max}]$. The system detects operator-crowd anomalous interactions when the curve gradient is positive for an *evaluation period* $t_{ep} = t_{max} - t_{min}$. In the on-line modality the CN when an anomalous interaction has been recognized, the system alerts the operator sending a message. Such a message can contain the last detected abnormal passive triplet, e.g. first user action (proto event), evolution of the crowd (core event) and consequent operator action (proto event). In the case shown in Fig. 14, the anomalous situation is due to wrong consequent user action, i.e. the operator does not open some doors and the number of people increase.

6.1 Example of application on real video sequences

In order to give consistence to the proposed cognitive video surveillance system, an experiment has been conducted on a available video sequences from the "Performance Evaluation of Tracking and Surveillance" (PETS) workshop dataset (Ferryman and Crowley, 2009) for single camera (S3 High level, Time $14 - 16$, View_0001, sequence length is 223 frames, frame rate is $\sim 7[fps]$).

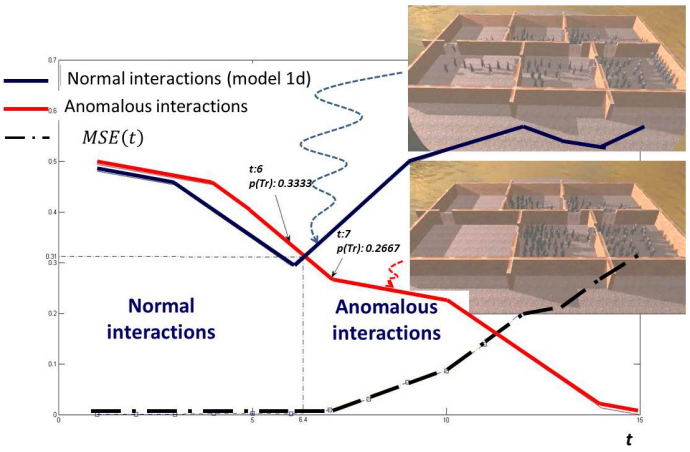


Fig. 14 Detection of anomalous operator-crowd interactions. The system detects an anomalous interaction when the operator does not open two doors and the number of people increase. This incorrect crowding management situation is shown in the figure and compared with the correct situation.

The main target of this experiment is to demonstrate how the system is able to recognize interaction between an operator and the crowd behaviour in a video sequence. For this purpose the real environment has been partitioned in three zones, which are supposed to be monitored by cameras, as shown in Fig. 15 (a). In the simulated environment, the zones correspond to three rooms, Fig. 15 (b). In the sequence, two crowd behaviours corresponding to different people flows have been individuated. The fist flow direction when the people go across the scene from zone 1 to zone 3 (i.e. from frame_0000 to frame_0107), while the second flow when the people move from zone 3 to zone 1 (i.e. from frame_0108 to frame_0222). By using the simulator these two different people behaviours have been reproduced: for the first flow the people enter the scene in zone 1 and head out in zone 3, while for the second the people enter in zone 3 and leave in zone 1 (second flow). In the simulator a human operator can manage the crowd flow, from a room to another, by acting on doors, $d1$ $d2$ $d3$ $d4$. The user opens the door when the people are near to it. In order to reproduce the same interaction using the real video sequences, it has been supposed to have the same configuration of the doors. A people counting algorithm (Morerio et al, 2012) provides an estimate of the total number of people in the zones present in video sequences, Fig. 15 (a). In this virtual environment a people counting module is simulated in order to count the people into a sub-area of the room (dashed circular areas, Fig. 15 (b)).

The test is composed by two parts: learning and detection (on-line). During the learning phase, the cognitive system has learned two probabilistic models from the simulation, i.e. two AMs, in order to describe two crowd behaviours. The rules used to memorize such two models are specified as follows: if the

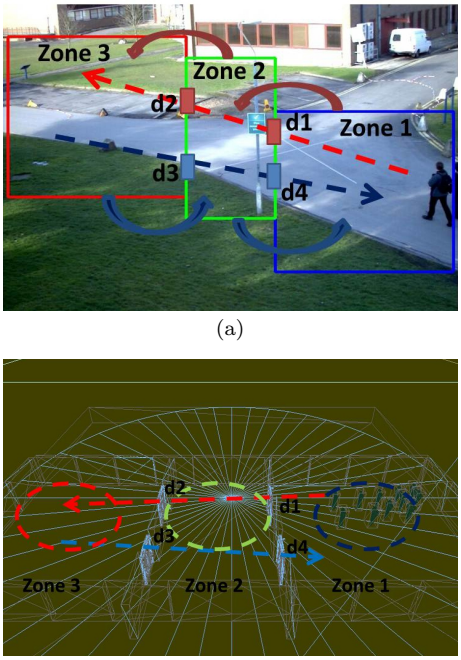


Fig. 15 Example of real environment (a) and simulated scenario (b) used for the test, the virtual rooms correspond to the zones. The red dashed line corresponds to people flow direction from zone 1 to zone 3; the blue dashed line describes people movement from zone 3 to zone 1. Dashed circular areas qualitatively correspond to the parts of the rooms monitored by cameras equipped with people counter module. $d1 - d4$ are the doors.

operator sees the people moving from zone 1 to zone 3 must open only $d1$ $d2$ according to the people flow; the user has to act on $d3$ $d4$ if people flow is in opposite direction. Four scenes for the two people flows have been simulated, each scene is $60[s]$ long. The simulated people counters provide number of people in each zone per second. During the second part the system works on real video sequences. The CN recognizes the observed situations according to the memorized knowledges. During autonomous phase, the CN, to the end to interact directly with the crowd, must discriminate different crowd-environment configurations. Fig. 16 presents four sample frames about different crowd behaviours: in case (a) the people flow moves leading red arrow (i.e. from zone 1 to zone 3), in case (b) the opposed people movement direction is presented (i.e. from zone 3 to zone 1). In cases (c) and (d) the groups of the people have different movement directions, namely people splitting and merging. In these last two cases, the system does not find any correspondence between observed scene and stored interaction models. In particular, the CN can consider the scene (c) as anomalous crowd-environment interaction compared with (a) situation. The same consideration can be done for (d) and (b) situations. When anomalous crowd-environment interactions are detected, the cognitive system

sends an alarm message in order to inform the human operator. After this

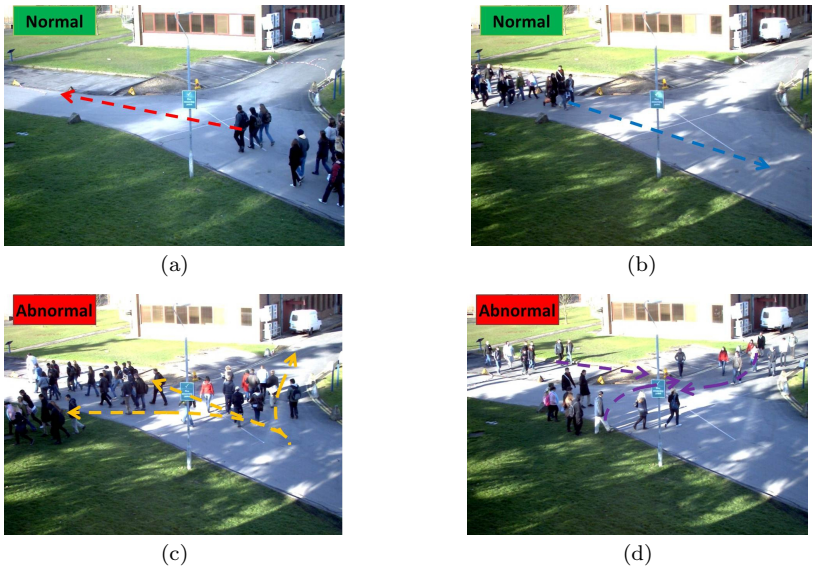


Fig. 16 Sample frames for four different crowd-environment interactions. Different people flows are presented: two opposite directions of movement (a) (b), people splitting (c), people merging (d). (a) and (b) represent normal behaviours, while (c) and (d) represent two abnormal behaviours.

phase, the CN is able to predict most likely future actions and when it will be performed. During the operator support phase, the cognitive system individuates anomalies in terms of differences between predicted actions and user actions.

The SOM-map dimensions produce different results in terms of knowledge representation for crowd-environment interactions. In particular, small Kohonen's layers increase SOM capability of creating analogies between different input data. This effect becomes much relevant when the input data is corrupted by noise. A test has been conducted employing two people counters, namely $PC1$ and $PC2$, characterised by different accuracies, i.e. $a_{PC1} = 80\%$, $a_{PC2} = 60\%$. The experiment can be divided in two parts. Firstly, we have manually built the ground truth for the video sequence. We use this information in order to generate the sequences of the super-states. Through three different SOMs, i.e. SOM 16, SOM 25 and SOM 32, the original data have been mapped into SOM-spaces. Secondly, by using of people counter [28], it is possible to obtain the number of people ($PC1$) with estimated accuracy of 80% ($a_{PC1} = 80\%$). Tuning a people counter parameter another set of number of people ($PC2$), with less accuracy, has been acquired ($a_{PC2} = 60\%$). We have manually corrupted the parameter for decreasing the accuracy of people number estimation. The data provided by $PC1$ and $PC2$ are classified by us-

ing the three different SOMs so that six different sequences of fired neurons are obtained. The events (i.e. super states transitions), which correspond to passages across the zones (i.e. Zone 1 $\mapsto$ Zone 2, Zone 2 $\mapsto$ Zone 3 and Zone 3 $\mapsto$ Zone 2, Zone 2 $\mapsto$ Zone 1), are compared with the events generated from ground truth. When the system recognizes the same events it is possible to affirm that a specific SOM is able to provide an efficient crowd-environment interaction representations. Vice versa a failure will be detected. Failures are due to the poor capability of larger SOM layers of finding analogies between input data: similar inputs may be mapped to different neurons (see Subsection 5.2). In table 4, the performances (in people flow detections) of three SOMs (16, 25 and 36 neurons respectively) are shown. The interesting result is that a 16-neuron SOM is able to detect three zone passages also in the presence of corrupted data input.

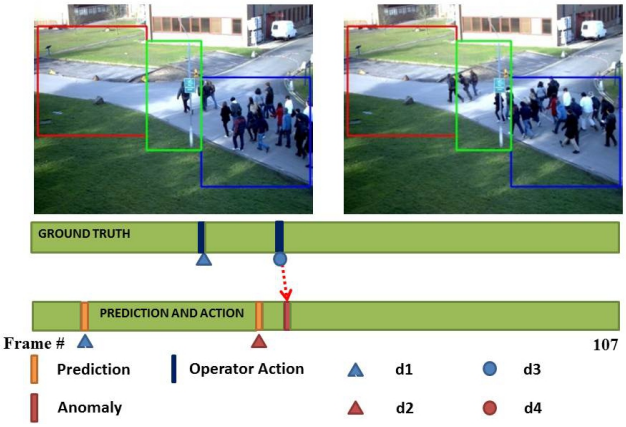
Table 4 People flows detections using different people counters and SOMs. Accuracy of the precision: $a_{PC1} = 80\%$, $a_{PC2} = 60\%$. Success = 1, Failure = 0.

	SOM 16		**SOM 25**		**SOM 36**	
Direction	**PC 1**	**PC 2**	**PC 1**	**PC 2**	**PC 1**	**PC 2**
Zone 1 $\mapsto$ Zone 2	1	1	1	1	1	0
Zone 2 $\mapsto$ Zone 3	1	0	1	0	0	0
Zone 3 $\mapsto$ Zone 2	1	1	1	1	1	0
Zone 2 $\mapsto$ Zone 1	1	1	1	0	1	0

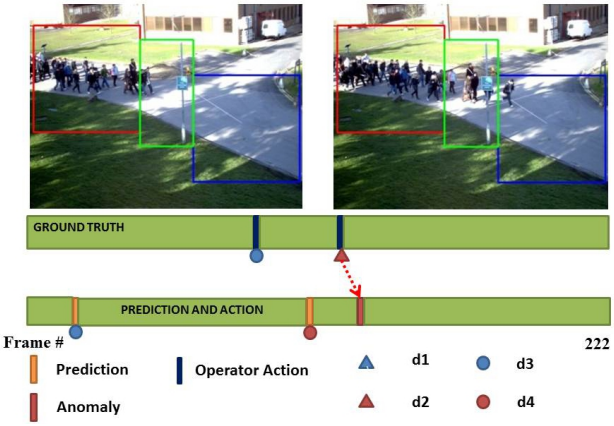
For the test, a SOM 25 and PC1 have been employed. In Fig. 17 the cognitive system predictions and detections of normal and abnormal interactions between an operator and the crowd are shown. In the figure, the operator actions and corresponding video frames are represented (blue operator action rectangle) in the ground truth bar. The prediction (yellow prediction rectangle) and action bar represents the cognitive system actions. The anomaly is represented by a red anomaly rectangle. Considering the case (a), the system predicts the first zone crossing (i.e. from zone 1 to zone 2) as *to open* $d1$ (specified by blue triangle). In this case, the operator action finds a correspondence with the predicted action (i.e. $d1$). During the second zone crossing (i.e. from zone 2 to zone 3) the system detects an anomalous operator-crowd interaction behaviour: the user opens an uncorrected door (i.e. $d3$ indicated by blue circle). The case (b) presents the same analysis for a different people flow direction.

7 Conclusion

An automatic systems called Cognitive Surveillance Node (CSN), which is part of a complex cognitive JDL-based and bio-inspired architecture was presented in this work. Also, a bio-inspired structure was proposed, for encoding and synthesizing signals for modelling cause-effect relationships between entities. In particular, the case where one of such entities is a human operator was studied.



(a)



(b)

Fig. 17 The qualitative results of the normal and anomalous operator-crowd interaction detection, during the operator support phase. The ground truth bar represents the operator actions in corresponding with video frames. Prediction and action bar represents the cognitive system actions.

Interaction models are stored within an AM during a learning phase. Knowledge is thus transferred from a human operator towards the CSN. Learned representations can be used, at different levels, either to support human decisions by detecting anomalous interaction models and thus compensating for human shortcomings, or, in an automatic decision scenario, to identify anomalous patterns and choose the best strategy to preserve stability of the entire system.

Results are shown in a simulated visual 3D environment in the context of crowd monitoring. The simulated crowd is modelled according to the Social Forces Model. The results show two possible applications of the CSN for

crowd surveillance applications: first, the system can support the operator for crowd management and people flow redirection by detecting drift from some learned interaction models; secondly, to work in automatic mode and thus autonomously detecting anomalies in crowd behaviour. Furthermore, it has been shown how user-crowd interaction knowledge, learned from the simulator and modelled as proposed is useful in order to detect anomalies on real video sequences.

Future developments of this work will include a detailed study on the impact of other self organizing maps, e.g. GNG or GH-SOM on the performances of our system. In particular, we are interested to design a cognitive control-based architecture that is able to switch among various contextual knowledge representation levels provided by different AMs.

References

Bird A (2012) Routledge Companion to Philosophy of Language. Routledge Philosophy Companions, Routledge

Chiappino S, Morerio P, Marcenaro L, Fuiano E, Repetto G, Regazzoni C (2012) A multi-sensor cognitive approach for active security monitoring of abnormal overcrowding situations in critical infrastructure. 15th International Conference on Information Fusion

Chiappino S, Marcenaro L, Morerio P, Regazzoni C (2013a) Event based switched dynamic bayesian networks for autonomous cognitive crowd monitoring. In: Augmented Vision and Reality, Augmented Vision and Reality, Springer Berlin Heidelberg, pp 1–30

Chiappino S, Marcenaro L, Regazzoni C (2013b) Selective attention automatic focus for cognitive crowd monitoring. In: Proc. of IEEE Int. Conference on Advanced Video and Signal based Surveillance (AVSS), Krakow, Poland

Chiappino S, Morerio P, Marcenaro L, Regazzoni C (2013c) Run length encoded dynamic bayesian networks for probabilistic interaction modeling. In: 21th European Signal Processing Conference (EUSIPCO 2013)

Chiappino S, Morerio P, Marcenaro L, Regazzoni CS (2013d) A bio-inspired knowledge representation method for anomaly detection in cognitive video surveillance systems. In: Information Fusion (FUSION), 2013 16th International Conference on, pp 242–249

Chiappino S, Morerio P, Marcenaro L, Regazzoni CS (2013e) Event definition for stability preservation in bio-inspired cognitive crowd monitoring. In: Digital Signal Processing (DSP), 2013 18th International Conference on, pp 1–6, DOI 10.1109/ICDSP.2013.6622802

Damasio A (2000) The Feeling of What Happens: Body and Emotion in the Making of Consciousness. Harvest Books, URL `http://www.amazon.co.uk/exec/obidos/ASIN/0156010755/citeulike-21`

Dore A, Regazzoni CS (2009) Bayesian bio-ispired model for learning interactive trajectories. In: Proc. of the IEEE International Conference on Advanced Video and Signal based Surveillance, AVSS 2009, Genoa, Italy

Dore A, Pinasco M, Regazzoni C (2009) Multi-modal data fusion techniques and applications. In: Aghajan H, Cavallaro A (eds) Multi-camera networks: Concepts and Applications, Elsevier, pp 213–237

Dore A, Cattoni A, Regazzoni C (2010a) Interaction modeling and prediction in smart spaces: A bio-inspired approach based on autobiographical memory. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on 40(6):1191 –1205, DOI 10.1109/TSMCA.2010.2052600

Dore A, Cattoni A, Regazzoni C (2010b) Interaction modeling and prediction in smart spaces: a bio-inspired approach based on autobiographical memory. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on

Dore A, Pinasco M, Ciardelli L, Regazzoni CS (2011) A bio-inspired system model for interactive surveillance applications. JAISE 3(2):147–163

Espina MV, Velastin SA (2005) Intelligent distributed surveillance systems: A review. IEE Proceedings - Vision, Image and Signal Processing 152(2):192–204

Ferryman J, Crowley JL (eds) (2009) Eleventh IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, PETS 2009, URL http://www.cvg.rdg.ac.uk/PETS2009

Finn JT (1993) Use of the average mutual information index in evaluating classification error and consistency. International journal of geographical information systems 7(4):349–366, DOI 10.1080/02693799308901966, URL http://www.tandfonline.com/doi/abs/10.1080/02693799308901966, http://www.tandfonline.com/doi/pdf/10.1080/02693799308901966

Fritzke B (1995) A growing neural gas network learns topologies. In: Advances in Neural Information Processing Systems 7, MIT Press, pp 625–632

Fuster J (2005) Cortex and Mind: Unifying Cognition. Oxford University Press, USA, URL http://books.google.it/books?id=3R-9dcFw95wC

Hall DL, Llinas J (1997) An introduction to multisensor data fusion. Proceedings of the IEEE 85:6–23, URL http://dx.doi.org/10.1109/5.554205

Haykin S (2011) Cognitive dynamic systems: An integrative field that will be a hallmark of the 21st century. In: IEEE ICCI*CC, IEEE, p 2

Haykin S (2012a) Cognitive Dynamic Systems: Perception-Action Cycle, Radar and Radio. Cambridge University Press, URL http://books.google.it/books?id=GMDdQEVm74UC

Haykin S (2012b) Cognitive dynamic systems: Radar, control, and radio. Proceedings of the IEEE 100(7):2095–2103

Kilambi P, Ribnick E, Joshi AJ, Masoud O, Papanikolopoulos N (2008) Estimating pedestrian counts in groups. Computer Vision and Image Understanding 110(1):43–59

Kohonen T (1990) The self-organizing map. Proceedings of the IEEE 78(9):1464 –1480, DOI 10.1109/5.58325

Lipton A, Heartwell C, Haering N, Madden D (2003) Automated video protection, monitoring & detection. IEEE Aerospace and Electronic Systems Magazine 18(5):3–18

Luber M, Stork JA, Tipaldi GD, Arras KO (2010) People tracking with human motion predictions from social forces. In: Proc. of the Int. Conf. on Robotics & Automation (ICRA), Anchorage, USA

Marchesotti L, Piva S, Regazzoni C (2005) Structured context-analysis techniques in biologically inspired ambient-intelligence systems. IEEE Transactions on Systems, Man and Cybernetics, Part A 35(1):106–120

Martinetz TM, Schulten KJ (1991) A "neural gas" network learns topologies. In: Kohonen T, Mäkisara K, Simula O, Kangas J (eds) Proceedings of the International Conference on Artificial Neural Networks 1991 (Espoo, Finland), Amsterdam; New York: North-Holland, pp 397–402

Mazzon R, Poiesi F, Cavallaro A (2013) Detection and tracking of groups in crowd. In: Proc. of IEEE Int. Conference on Advanced Video and Signal based Surveillance (AVSS), Krakow, Poland

Mehran R, Oyama A, Shah M (2009) Abnormal crowd behavior detection using social force model. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pp 935 –942, DOI 10.1109/CVPR.2009.5206641

Moore BE, Ali S, Mehran R, Shah M (2011) Visual crowd surveillance through a hydrodynamics lens. Commun ACM 54(12):64–73, DOI 10.1145/2043174.2043192, URL http://doi.acm.org/10.1145/2043174.2043192

Morerio P, Marcenaro L, Regazzoni CS (2012) People count estimation in small crowds. In: Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on, pp 476 –480, DOI 10.1109/AVSS.2012.88

Murphy K (2002) Dynamic bayesian networks: Representation, inference and learning. PhD thesis, UC Berkeley, Computer Science Division

Oliver N, Pentland A (2000) Graphical models for driver behavior recognition in a smartcar. In: Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE, pp 7 –12, DOI 10.1109/IVS.2000.898310

Pan W, Dong W, Cebrin M, Kim T, Fowler JH, Pentland A (2012) Modeling dynamical influence in human interaction: Using data to make better inferences about influence within social systems. IEEE Signal Process Mag pp 77–86

Patnaik D, Laxman S, Ramakrishnan N (2009) Discovering excitatory networks from discrete event streams with applications to neuronal spike train analysis. In: Proceedings of the 2009 Ninth IEEE International Conference on Data Mining, IEEE Computer Society, Washington, DC, USA, ICDM '09, pp 407–416, DOI 10.1109/ICDM.2009.73, URL http://dx.doi.org/10.1109/ICDM.2009.73

Pellegrini S, Ess A, Schindler K, van Gool L (2009) You'll never walk alone: Modeling social behavior for multi-target tracking. In: International Conference on Computer Vision

Pfaff B (2008) Var, svar and svec models: Implementation within r package vars. Journal of Statistical Software 27(4):1–32, URL http://www.jstatsoft.org/v27/i04

Rauber A, Merkl D, Dittenbach M (2002) The growing hierarchical self-organizing map: Exploratory analysis of high-dimensional data. IEEE Transactions on Neural Networks 13:1331–1341

Remagnino P, Velastin SA, Foresti GL, Trivedi M (2007) Novel concepts and challenges for the next generation of video surveillance systems. Mach Vision Applications 18(3):135–137

Rish I, Grabarnik G (2009) Sparse signal recovery with exponential-family noise. In: Proceedings of the 47th annual Allerton conference on Communication, control, and computing, IEEE Press, Piscataway, NJ, USA, Allerton'09, pp 60–66, URL http://dl.acm.org/citation.cfm?id=1793974.1793985

Shlens J (2005) A tutorial on principal component analysis. In: Systems Neurobiology Laboratory, Salk Institute for Biological Studies

Soh C, Raveendran P, Taha Z (2004) Automatic generation of self-organized virtual crowd using chaotic perturbation. In: TENCON 2004. 2004 IEEE Region 10 Conference, vol B, pp 124 – 127 Vol. 2, DOI 10.1109/TENCON.2004.1414547

Solmaz B, Moore BE, Shah M (2012) Identifying behaviors in crowd scenes using stability analysis for dynamical systems. Pattern Analysis and Machine Intelligence, IEEE Transactions on 34(10):2064 –2070, DOI 10.1109/TPAMI.2012.123

Su H, Yang H, Zheng S, Fan Y, Wei S (2012) Crowd event perception based on spatio-temporal viscous fluid field. In: Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on, pp 458 –463, DOI 10.1109/AVSS.2012.32

Trivedi M, Huang K, Mikic I (2000) Intelligent environments and active camera networks. In: Proceedings of the IEEE International Conference on System, Man and Cybernetics, pp 804–809

Trivedi MM, Gandhi T, McCall J (2007) Looking-in and looking-out of a vehicle: Computer-vision-based enhanced vehicle safety. Intelligent Transportation Systems, IEEE Transactions on 8(1):108 –120, DOI 10.1109/TITS.2006.889442