Distributed Framework for Composite Event Recognition in a Calibrated Pan-Tilt Camera Network

Ayesha Choudhary
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi, India
ayesha@cse.iitd.ernet.in

Santanu Chaudhury
Department of Electrical Engineering
Indian Institute of Technology, Delhi, India
santanuc@ee.iitd.ernet.in

Subhashis Banerjee
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi, India
suban@cse.iitd.ernet.in

ABSTRACT

In this paper, we propose a real-time distributed framework for composite event recognition in a calibrated pan-tilt camera network. A composite event comprises of events that occur simultaneously or sequentially at different locations across time. Distributed composite event recognition requires distributed multi-camera multi-object tracking and distributed multi-camera event recognition. We apply belief propagation to reach a consensus on the global identities of the objects in the pan-tilt camera network and to arrive at a consensus on the event recognized by multiple cameras simultaneously observing it. We propose a hidden Markov model based approach for composite event recognition. We also propose a novel probabilistic Latent Semantic Analysis based algorithm for pair-wise interaction recognition and present an application of our distributed composite event recognition framework, where the events are interactions between pairs of objects.

1. INTRODUCTION

In this paper, we propose a real-time distributed framework for composite event recognition in a calibrated pan-tilt camera network. Composite events comprise of events that occur simultaneously at different locations in space as well as events that are related in both space and time. Therefore, to be able to recognize a composite event, we need to detect and track multiple objects through space and time and recognize all the events (associated with these objects) that eventually comprise a composite event. In our framework, we incorporate single camera event analysis in each camera. We propose the use of multi-layered belief propagation for arriving at a consensus on the global identity of an object among cameras that simultaneously track the object as well as among those that track the object across space and time. For distributed recognition of an event by multiple cameras simultaneously viewing the event, firstly, single camera event recognition is carried out in each camera and then, we propose the use of belief propagation to reach a consensus on the recognized event.

Recognizing a composite event is a challenging problem because of the variability and uncertainty associated with both the composite event as well as the events that comprise it. The variability exists mainly because each instance of the same event will have a similar but not the same pattern and can occur for different lengths in time. As the time scale of the events vary, the time scale for the composite events shall also be variable. Moreover, the same event will, in general, be a part of more than one composite event, as it is the sequence of these events that uniquely define a composite event. In such a system, the state of the system is unknown and dynamic and the system can remain in a state for a variable duration of time for different instances of the same event. Therefore, rule-based event recognition systems are not well-suited for composite event recognition. Finite state machines can provide the state sequence which defines the composite event but require the knowledge about the state of the system. Hidden Markov models (HMMs) [11] are best suited for recognition of a temporal sequence of observations when the state of the system is dynamic, unknown, and variable. Therefore, we propose a HMM based approach for recognition of composite events in the pan-tilt camera network, based on the recognition of events, in each of the sub-networks across space and time, comprising the composite event.

A pan-tilt camera network is best suited for surveillance of wide areas. As a small number of pan-tilt cameras can cover a wide area, such networks reduce the cost and complexity associated with a static camera network that would require a large number of cameras to cover the area. Although pan-tilt camera networks are cost effective, the complexity in a pan-tilt camera network arises due to the dynamic nature of the cameras. A pan-tilt camera network consists of many smaller sub-networks, where cameras in a sub-network observe a common region. As the cameras pan and/or tilt, these sub-networks change across time leading to the change in the topology of the complete network.

In such a scenario, distributed processing becomes a necessity, specially if the information has to be processed in real-time. This is mainly because, different sub-networks observe different events simultaneously, making it difficult for a central server to simultaneously recognize the events that occur in the various sub-networks. Distributed processing on the other hand, requires processing of information at each camera in the network, independent of the others. This

ICVGIP '10, December 12-15, 2010, Chennai, India

may lead to inconsistencies in the estimation, as each camera processes the information it gathers and observes based on it's view. We use our distributed calibration algorithm from [5] because it is also necessary to calibrate the camera network in a distributed manner. Distributed event recognition also requires a single camera system for object detection, tracking and event recognition along with a method to reach consistent and accurate decisions about the events that occur in the area under observation.

Distributed processing is also advantageous as it is robust against failures and addition or removal of cameras from the network does not impact the complete system. Distributed processing makes the system modular as processing is done at each camera and scalable as the communication between the cameras takes place among a much smaller set of cameras compared to the total number of cameras in the network.

We present an application of our framework, where an event implies interaction between a pair of objects in the scene. Composite event is therefore, composed of various different interactions of an object with another static or moving object in the scene. To this end, we develop a novel probabilistic Latent Semantic Analysis [7] based pair-wise interaction recognition algorithm and apply our pLSA based unusual event detection framework in [4], to be able to recognize the pair-wise interactions as well as detect unusual interactions. Our framework is capable of detecting unusual events as well as unusual composite events.

In the following sections, we discuss the related work and the details of our framework.

2. RELATED WORK

Research on multi-camera networks is mainly focused on static camera networks and centralized processing, but multi-camera tracking in static as well as active camera networks has also become an important area of research [6], [2]. Recently, in [14], concepts from multi-player learning in games is used for keeping the complete scene in view as well as acquiring the targets at a desired resolution and Kalman consensus filter is used for consensus among neighboring cameras for multi-target tracking in a self-configuring active camera network. Authors in [10] present a hierarchical framework to manage static as well as pan-tilt-zoom cameras for visual surveillance of a parking lot. They use static cameras to detect an object and then the active cameras for tracking the suspicious objects at a higher resolution. Authors in [12], propose a multi-agent architecture for the understanding of scene dynamics by merging the information streamed from multiple cameras. In [1], the authors describe an architecture for a multi-camera, multi-resolution surveillance system. Most of the multi-camera distributed systems consists of both static and active cameras and follow the master-slave configuration, where the active cameras are used for zooming in on the target of interest. Authors in [9], present a cooperative distributed system for real-time multi-target tracking. In this, a group of active cameras cooperatively track multiple objects and has a manager assigned to each such group for inter-group as well as intra-group communication to manage the dynamic topology and intra-group cooperative tracking of multiple objects. In [8], the authors present a nearly real-time surveillance system for multi-target tracking by multiple pan-tilt-zoom cameras. A master-slave relationship between the cameras is decided on the fly and each camera does multi-target tracking, such that targets are tracked in the complete scene as often as possible.

Active camera network systems such as, [14, 9, 8] only focus on multi-target tracking, Authors in [13] also focus on consensus based activity recognition in an active camera network, where they consider that a set of cameras simultaneously view an object performing some action and reach a consensus among these cameras on the recognized activity. However, they do not address recognition of events or activities that occur across sub-networks in a pan-tilt camera network. Our framework recognizes composite events that occurs across space and time in the area under observation of a calibrated pan-tilt camera network. It is flexible and modular and along with recognizing events that constitute a composite event it also performs the task of multi-target tracking.

In the next sections, we give an overview of our framework and discuss the composite event recognition architecture.

3. OVERVIEW

We assume that the pan-tilt camera network consists of $N \geq 3$ cameras and that the camera network is completely calibrated. We use our algorithm [5] for distributed calibration of the pan-tilt camera network. We assume that each camera has a unique number, $i \in \{1, 2, \ldots, N\}$ associated with it and also has a processing unit attached with it. We also assume that there exists an underlying communication network such that each camera can send a message to all other cameras.

A sub-network in the camera network is composed of all cameras which view a common region simultaneously. Each sub-network corresponds to a graph $G = (V, E)$ such that $V = \{\text{set of all cameras in the sub-network}\}$ and an edge e_{ij} exists between nodes $C_i \in V$ and $C_j \in V$ if they view a common region. Therefore, each sub-network forms a complete graph. In a pan-tilt camera network, at each time instance, many such graphs exist in the system. Moreover, if a camera pans and/ or tilts, it becomes a part of a new graph and each camera can belong to only one graph at any point in time.

Since the camera network is completely calibrated, each camera has *a priori* information of all the other cameras with which it can have an overlap in some pan-tilt position of both the cameras. Two cameras are therefore, said to be *neighbors* if they can have sufficient overlap in some pan-tilt position. Therefore, whenever a camera pans and/or tilts, it sends a message to all its neighbors about it's new pan-tilt position. Each neighbor of that camera then evaluates the set of cameras with which it currently has sufficient overlap. In this manner, each camera finds the nodes of the graphs to which it belongs.

We also assume that the cameras sweep in discrete pan-tilt steps at a fixed zoom while they are not tracking any object. As soon as a camera detects an object, it becomes static and begins to track the object. The camera also sends a message to its neighbors about the $3D$ position and the global identity of the object. All it's neighbors that are not observing or tracking any object at that point in time, pan and/or tilt to bring this object in their view and form a graph as described above. While the cameras track objects in their view, they also collect information required for recognizing the interaction (or event) associated with the objects.

When an object is about to get out of the view of a cam-

era, the camera on the basis of direction of motion as well as the predicted position of the object, pans and/or tilts to the next discrete pan-tilt position, if possible. It also sends a message about its new pan-tilt position, the $3D$ position and global identity of the object and another graph is formed. However, before the camera(s) in a graph pan-tilt, they exchange information and use belief propagation to reach a consensus on the global identities of these object(s). Moreover, each camera implements the event recognition module using the data it has collected till that point in time and uses belief propagation to reach a consensus on the recognized event.

If the cameras remain static for a time period greater than a fixed time period, say T_1, then at the end of T_1, the cameras implement event recognition and belief propagation for reaching the consensus on the global identities of the objects in their view as well as belief propagation for consensus on events recognized during that time period. This is carried out at the end of every T_1 time step, starting from the time the new graph is formed. The time periods are concatenated if there is no change in information in the consecutive time periods. At the end of each time period T_1, after a consensus has been reached in each graph, from each camera in the network, a token collects the information regarding the number of graphs in the network, which camera belongs to which graph, the objects observed in each graph on the basis of their global identities and the events recognized in each graph. The token then imparts this information collected from each camera to every other camera in the network. When an object exits from the area under observation of the pan-tilt camera network, each camera implements the HMM based composite event recognition module for recognizing the composite event associated with that object.

Composite event recognition therefore, involves (a) distributed multi-object tracking in a multi-camera network; (b) distributed multi-camera event recognition and (c) composite event recognition using HMMs. In the following sections, we discuss our composite event recognition architecture and then each of the above mentioned parts of our framework.

4. COMPOSITE EVENT RECOGNITION ARCHITECTURE

We propose a three layered architecture as shown in the Figure 1. The first layer shown in Figure 1(a) consists of the processing carried out in each camera of the pan-tilt camera network. It consists of modules for object detection, assignment of the global identity when an object is first detected in the scene, object tracking and event recognition. The next level, shown in Figure 1(b), is at the level of a graph and consists of two modules, one for belief propagation to arrive at a consensus on the global identity of the objects observed during a time period and the other, for consensus on the event recognized by all the cameras in the graph during the same time period. Each camera in a graph implements these modules after exchanging information on the global identity it assigns to the objects in it's view and the event it has recognized during that time period. This time period is either equal to T_1 or is less than T_1 during which the graph was static. Figure 1(c), depicts the top-most layer of the architecture, which consists of the HMM-based module for composite event recognition, based on the information collected

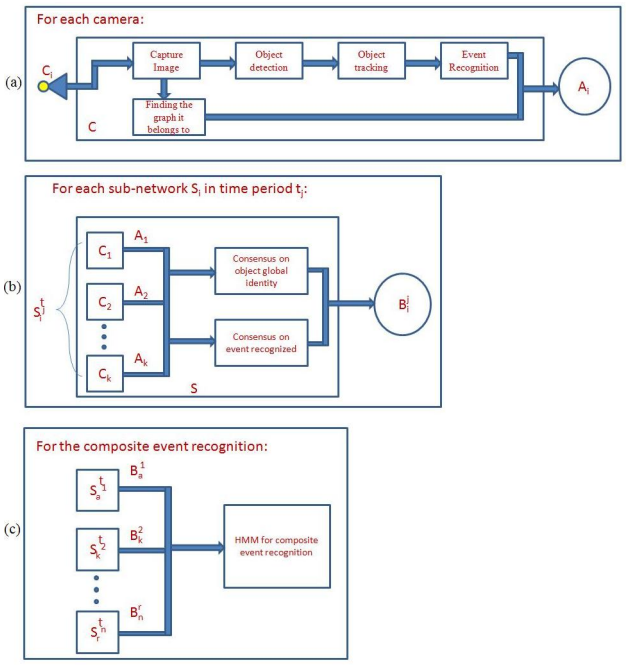

Figure 1: The three-layered architecture for distributed composite event recognition in a pan-tilt camera network.

in each graph across space and time. The token that collects the information at the end of a fixed time period, imparts this information to each camera in the network. Each camera then uses this information with respect to an object or a pair of objects and implements the HMM-based module for recognizing composite events associated with the object(s).

In Section 5, we discuss distributed multi-camera multi-object tracking which constitutes the single camera object tracking module and the module for multi-layered belief propagation within and across graphs for reaching a consensus on the global identities of the objects in the network. In Section 6, we detail distributed multi-camera pair-wise interaction recognition which includes the modules on pair-wise interaction (event) recognition in a single camera and the module for belief propagation based consensus on event recognition within a graph. We discuss our HMM-based composite event recognition module in Section 7.

5. DISTRIBUTED MULTI-CAMERA MULTI-OBJECT TRACKING

We use our single camera component-based clustering framework from [3] for tracking objects in each camera. We cluster in the components defining object properties such as $3D$ position of objects, size of objects, color correlogram of objects, etc., using our incremental clustering algorithm [3] in each of the components to be able to track the object in real-time. The tracks of each object is found by composition of clusters from each of the component spaces. Since the $3D$ position is unique to an object, that is only one object can be at a particular $3D$ position at any point in time, we use $3D$ position as one of the components for tracking the object. As the camera network is calibrated, we calculate the $3D$

position of an object from its $2D$ position assuming that the object moves on a ground plane.

Assume that the camera matrix is given by P and the $3D$ position on the ground plane by $\hat{X}$ and its $2D$ position in the image by $\hat{x}$. Then, $P = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \end{bmatrix}$, $\hat{X} = \begin{bmatrix} X & 0 & Z & 1 \end{bmatrix}'$ and $\hat{x} = \begin{bmatrix} x & y & 1 \end{bmatrix}'$ and,

$$\begin{aligned} \hat{x} &= P\hat{X} && (1)\\ &= \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \end{bmatrix} \begin{bmatrix} X & 0 & Z & 1 \end{bmatrix}' && (2)\\ &= \begin{bmatrix} p_1 & p_3 & p_4 \end{bmatrix} \begin{bmatrix} X & Z & 1 \end{bmatrix}' \end{aligned}$$

Therefore, if $\tilde{P} = \begin{bmatrix} p_1 & p_3 & p_4 \end{bmatrix}$, we get the $3D$ position of the object on the ground plane as

$$\tilde{X} = \tilde{P}^{-1}\tilde{x} \quad (3)$$

where, $\tilde{X} = \begin{bmatrix} X & Z & 1 \end{bmatrix}'$ and $\tilde{x} = \begin{bmatrix} x & y & 1 \end{bmatrix}'$

We use the $3D$ position to reach a consensus on the global identity of the object within and across graphs as discussed in the next section.

5.1 Consensus on global identity of objects

Let N_h be the total number of objects in the pan-tilt camera network, and N_G be the total number of cameras in a graph G.

Let $Y = \{y_1, y_2, \ldots, y_{N_G}\}$ be the set of observations of all the cameras in graph, G.

Assume that each camera C_i in G receives a message from it's neighbors containing $3D$ position X_i and the global identity O_i of n objects, $1 \leq i \leq n$. When an object O enters a cameras view, it first checks whether it has received any message from its neighbors or not. Moreover, it computes the $3D$ position, X of object O. Then, if C_i has received a message from its neighbor, it calculates,

$$d_j = e^{-d(X,X_j)/\sigma} \qquad \forall \quad 1 \leq j \leq n \quad (4)$$

where, $d(X, X_j)$ is the Euclidean distance between X and X_j, $1 \leq j \leq n$, and σ is the normalizing constant.

Let h_{O_k} be the hypothesis that the object's global identity is O_k. Then, the belief of the i^{th} camera is,

$$P(y(i)|h_{O_k}) = max_{1 \leq j \leq n} d_j \quad (5)$$

In case, the camera does not receive any message from any of it's neighbors, we set $P(y(i)|h_{O_k}) = \frac{1}{2}$ because there is still a possibility that the object was previously present in the network. The prior probability for the j^{th} hypothesis is then defined on the basis of the belief of the camera. That is,

$$P(h_{O_k}) = \begin{cases} P(h_{O_k}|Y_{previous\_graph}) & \text{if} P(y(i)|h_{O_k}) > \frac{1}{2} \\ \frac{1}{N_h+1} & \text{otherwise} \end{cases} \quad (6)$$

Within a graph, the most probable hypothesis is the MAP estimate of the belief of the graph about the object's identity.

$$P(h_{O_i}|Y_{current\_graph}) = \arg\max_k P(h_{O_k}) \prod_{m=1}^{N_G} P(y(m)|h_{O_k}) \quad (7)$$

Thus, the identity of the object is set to be the one for which the probability $P(h_{O_k}|Y_{current\_graph})$ is maximum. Therefore, if the object was already in the system, the belief will be stronger and it shall get its correct identity while if it is a new object, it shall get recognized as a new object.

6. DISTRIBUTED PAIR-WISE INTERACTION RECOGNITION IN A MULTI-CAMERA NETWORK

Each camera in a graph recognizes the interactions independently using our interaction recognition algorithm described in Section 6.1. Because of the uncertainty in measurements there is a need to reach a consensus on the interaction that takes place while the objects are in a graph. We use belief propagation to arrive at this consensus and use the probabilities calculated in the pLSA based recognition algorithm as the likelihood of the interaction hypothesis as explained in detail in Section 6.2.

6.1 Pair-wise interaction recognition using pLSA

We use pLSA [7] for recognizing the interactions among a pair of objects. As the objects are tracked in real-time, the feature vector for interaction recognition is formed. The features that we use for interaction recognition are the changes in relative speed, relative direction, relative distance and relative angle among a pair of objects in the scene. Other features can also be used depending on the type of interactions that are to be recognized. Each component of the feature vector is treated as a word and in the learning phase, the word-document matrix is a concatenation of these feature vectors. The words are the number of times that the change in the component of the feature vector occurred. That is, if the word in the pLSA recognition system is relative distance increasing or relative distance decreasing or relative distance constant and so on for each of these relative features, then, these quantities are increased by 1 if they are true for the current frame. Finally, they are normalized so that they are independent of the size of the document. A document is the clip captured by the camera during which the feature vector is formed. A large amount of data is used to learn the distributions in pLSA for the usual interactions. Each hidden class represents a single interaction type. We use the pLSA based framework in [4] for unusual interaction recognition. In this case also, during the test phase, we add a new class, called the *unusual* class to the set of hidden classes. In case labeled data is available, we still carry out the learning considering it as unlabeled data. At the end of the learning process, we label the hidden topic with the label associated with the data clustered into that topic.

Let w denote the words, which are the components of the interaction feature vector, d denote the document and h_a denote the hidden class which represents an interaction (usual and unusual class). During the test phase, the term-document matrix comprises of the test feature vector and the Expectation-Maximization algorithm is used to compute the probability $P(h_a|d)$ of the test document belonging to the hidden class h_a. Therefore, the document is classified to either the unusual class or to one of the usual classes. We label a document as unusual if it belongs to the *unusual* class or if its log-likelihood of belonging to the usual class to which it is classified is below a predefined threshold.

Interaction recognition in our system is carried out by each camera of a graph to which an object belongs at the end of each fixed time interval. Since each camera in the

graph computes the pLSA probabilities, it is not necessary that each camera gets the same result. This is also because the feature vector formed is dependent on features computed from the individual camera's view and have a certain amount of uncertainty associated with them. Therefore, to be able to say which interaction took place within a certain time interval, a consensus among the cameras of the graph has to be reached. We again use belief propagation to come to a consensus. We also believe that the interactions in each time period are independent of interactions in the next or previous time periods. We now discuss how consensus on interaction recognition using belief propagation is reached.The hypothesis considered for belief propagation are all the valid interactions as well as the unusual one.

6.2 Consensus on recognized pair-wise interactions

Let $z(i)$ be the observation of the i^{th} camera C_i in graph G_k,
h_a be the a^{th} interaction hypothesis,
N_k be the total number of hypothesis, and
N_G be the total number of cameras in G.

We use the probabilities $P(h_a|d)$ calculated in each camera during pLSA test phase as the likelihood for the a^{th} interaction hypothesis h_a.

Let $Z_G = \{z(1), z(2), \ldots, z(N_G)\}$, be the set of all observations among the cameras of G.

Each camera sends the probabilities $P(h_a|d) \quad \forall \quad i = \{1, 2, \ldots, N_k\}$ to all other cameras in G. Then, the likelihood of the a^{th} hypothesis is,

$$P(z(i)_{G_k}|h_a) = P(h_a|d) \qquad \forall \quad h_a \tag{8}$$

We assume that each interaction is equally likely. Therefore, the prior probability for the a^{th} hypothesis is

$$P(h_a) = 1/N_k \tag{9}$$

Then, for each interaction hypothesis h_a, the *a posteriori* probability is,

$$P(h_a|Z_G) = \alpha_G P(h_a) \prod_{i=1}^{N_G} P(z(i)|h_a) \tag{10}$$

The above equation assumes independence among the observations $P(z(i)|h_a)$, which is true as each camera independently observes and applies pLSA to recognize the interaction. Here, α_{VG} is the normalizing constant. The hypothesis with the MAP estimate is taken to be the consensus on the interaction that took place during that time period.

$$P(h_{MAP}|Z_G) = \arg\max_i P(h_i|Z_G) \tag{11}$$

Therefore, the interaction among a pair of objects is recognized as hypothesis h_{MAP}.

7. HMM BASED COMPOSITE EVENT RECOGNITION

The information collected by a token from each camera is imparted to every other camera, therefore, HMM based composite event recognition is carried out at each camera in the network. Since we are interested in pair-wise object recognition, we define two types of states in each HMM formed for modeling a particular composite event. Moreover, in this case, a composite event is recognized with respect to a particular pair of objects. Therefore, from the data collected, we use the information regarding each pair of objects. Thus, one of the two types of states of the HMM denotes that there are two mutually exclusive graphs, each observing one object of interest in a pair. The other type of state represents that both the objects are in the same graph and such a state exists based on the total number of interactions that are considered for pLSA based recognition. The steps followed for HMM based composite event recognition are:

1. Define $E = e_1, e_2, \ldots, e_N$ to be the set of N composite event classes for modeling.
2. Collect a large labeled set of training data for each of the composite event classes.
3. Solve the estimation problem, based on the training data for each class, to obtain a model λ_i for each class e_i, $1 \leq i \leq N$ that best represents the composite event.
4. During the recognition phase, for the unknown composite event e, evaluate $P(e|\lambda_i) \quad 1 \leq i \leq N$. Then, the composite event belongs to the class e_j, if
$$P(e|\lambda_j) = max_{1 \leq i \leq N} P(e|\lambda_i) \tag{12}$$
and,
$$P(e|\lambda_j) > threshold \tag{13}$$
5. If $P(e|\lambda_j) < threshold$, label the composite event as an *unusual* composite event.

In case, the composite event is recognized as a usual event, we use the Viterbi algorithm to get the state sequence that best explains the composite event. Moreover, we also label a composite event as an unusual composite event, if any of the events comprising it are labeled as unusual by the pLSA based event recognition module.

8. RESULTS

We use 3 SONY EVI-D70 cameras for our experiment in the outdoor scene. In this case, the interactions between the objects are defined as (a) single; (b) pick up object; (c) drop object; and (d) independent. Independent implies that there is no interaction between the two objects, while single object implies that there is only one object in the scene. Pick-up and drop define picking up an object and leaving an object respectively. There are two HMMs that are built *a priori*.

In the first HMM, objects move around the camera network and the pair-wise interactions are either *single* or *independent*. The other HMM represents that it is common for people to drop bags and move around the area under observation of the network, but nobody else interacts with the bag once it is dropped.

Figure 2 shows a usual composite event occurring in the area observed by the three cameras. Two cameras, C_1 and C_2, view a common region forming the sub-network S_1, while the third camera C_3 views another region and is the only camera in the sub-network S_2. In the views of C_1 and C_2 an object O_1 is tracked and the cameras pan to keep the object in its view. The interaction *drop object* is recognized in the panned view of both the cameras using the pLSA algorithm for interaction recognition in each camera

independently. Therefore, a new sub-network S_3 is formed and a new object O_3 (the bag) is detected and tracked in this sub-network. Consensus on the interaction is reached among these two cameras. During the same time period, in camera C_3, an object O_2 is detected and tracked and the interaction is recognized as single. Objects move across these two sub-networks and their global identity is maintained, as is depicted by the color of their trajectories. O_2 is then tracked in S_3 till it leaves the network, while O_1 is tracked in S_2. In S_3, there is no interaction between O_2 and O_3 and therefore, the pair-wise interaction is recognized as *independent*. This composite event is recognized by one of the HMMs built *a priori* and is therefore, recognized as a usual composite event. Moreover, each of the events comprising this composite event is also usual.

In Figure 8, an unusual event is observed by the three cameras. The composite event is similar to the one described above, but in this case, after O_1 drops the bag, it enters S_2 where object O_2 is already present. The interaction between these two objects is recognized as *independent*. Moreover, when O_2 enters S_3, it interacts with O_3 as it picks up the bag and moves away with it. This interaction is recognized as *pick object* and the cameras pan to continue to track O_2 by panning and forming the sub-network S_1. The global identities of the objects are consistent across all sub-networks as can be seen by the color of the trajectories. The composite event of one person dropping the bag and another picking it up and moving away with it is not recognized by any of the pre-built HMMs. Therefore, it is detected as an unusual composite event.

9. CONCLUSION

In this paper, we have proposed a real-time, distributed algorithm for composite event recognition in a pan-tilt camera network. We have shown that multi-layered belief propagation can be applied to reach a consensus on the global identities of the objects, as they move within the area under observation of the pan-tilt camera network. We have also shown that belief propagation can also be used to reach a consensus on the event recognized by each of the cameras that simultaneously view the event. We have also proposed an HMM based approach for composite event recognition in a pan-tilt camera network. We have proposed a pLSA based pair-wise interaction recognition algorithm and presented an application of our framework, where the events are the pair-wise interactions and the composite event comprises of these pair-wise interactions that occur across space and time, in the area under observation of the pan-tilt camera network.

10. REFERENCES

[1] N. Bellotto, E. Sommerlade, B. Benfold, C. Bibby, I. Reid, D. Roth, C. Fernandez, L. V. Gool, and J. Gonzalez. A distributed camera system for multi-resolution surveillance. *In Proceedings of International Conference on Distributed Smart Cameras(ICDSC)*, 2009.

[2] J. Black and T. Ellis. Multi-camera image tracking. *Image and Vision Computing*, 24(11):1256–1267, 2006.

[3] A. Choudhary, S. Chaudhury, and S. Banerjee. A framework for analysis of surveillance videos. *In Proceedings of ICVGIP*, 2008.

[4] A. Choudhary, M. Pal, S. Banerjee, and S. Chaudhury. Unusual activity analysis using video epitomes and plsa. *In Proceedings of ICVGIP*, 2008.

[5] A. Choudhary, G. Sharma, S. Chaudhury, and S. Banerjee. Distributed calirbration of a pan-tilt camera network using multi-layered belief proapagation. *In Proceedings of IEEE Workshop on Camera Networks in conjunction with CVPR*, 2010.

[6] R. Collins, A. Lipton, H. Fujiyoshi, and T. Kanade. Algorithms for cooperative multisensor surveillance. *Proceedings of the IEEE*, 89(10):1456–1477, 2001.

[7] T. Hofmann. Probabilistic latent semantic analysis. *In Proceedings of Uncertainty in Artificial Intelligence (UAI)*, 1999.

[8] C.-M. Huang, Y.-T. Lin, and L.-C. Fu. Effective visual surveillance with cooperation of multiple active cameras. *In Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2008.

[9] T. Matsuyama and N. Ukita. Real-time multitarget tracking by a cooperative distributed vision system. *Proceedings of the IEEE*, 90(7):1136–1150, 2002.

[10] C. Micheloni, G. L. Foresti, and L. Snidaro. A network of cooperative cameras for visual surveillance. *Visual Image and Signal Processing*, 15(2), 2005.

[11] L. Rabiner and B.-H. Juang. Fundamentals of speech recognition. *Prentice-Hall, Englewood Cliffs, NJ,*, 1993.

[12] P. Remagnino, A. I. Shihab, and G. A. Jones. Distributed intelligence for multi-camera visual surveillance. *Pattern Recognition*, 37(4):675–689, 2004.

[13] B. Song, A. T. Kamal, C. Soto, C. Ding, A. K. Roy-Chowdhury, and J. A. Farrell. Tracking and activity recognition through consensus in distributed camera networks. *In IEEE Transactions on Image Processing*, 2010.

[14] C. Soto, B. Song, and A. K. Roy-Chowdhury. Distributed multi-target tracking in a self-configuring camera network. *In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Figure 2: Two consecutive rows represents view from a single pan-tilt camera. The first two rows (camera C_1 and the last two rows (camera C_2) form a sub-network. Initially, they are part of the sub-network S_1 and then, when they pan-tilt to continue tracking the object in their view, they form the sub-network S_3. The middle two rows (camera C_3) forms another sub-network S_2. In this case, object O_1, represented by the red trajectory, is tracked initially in S_1 and then in S_3 and the interactions are recognized as *single* in S_1 and *drop object* in S_3, as the person drops the bag and moves away from it. This also leads to the detection of a new object O_3 (the bag). While O_1 moves around in S_1 and S_3, object O_2 represented by the green trajectory, moves around in S_2. O_1 and O_2 then move across the sub-networks and it is seen that the global identity is maintained. While in S_2, the interactions between O_2 and O_3 are recognized as *independent*, while the interaction in S_3 is recognized as *single*. Therefore, in this case, one person drops a bag and another person walks in the scene, ignoring the bag. This composite event is explained by one of the HMMs built *a priori*, and is therefore, detected as a usual composite event.

Figure 3: Two consecutive rows represents view from a single pan-tilt camera. The first two rows (camera C_1) and the last two rows (camera C_2) form a sub-network. Initially, they are part of the sub-network S_1 and then, when they pan-tilt to continue tracking the object in their view, they form the sub-network S_3. The middle two rows (camera C_3) forms another sub-network S_2. In this case, object O_1, represented by the red trajectory, is tracked initially in S_1 and then in S_3 and the interactions are recognized as *single* in S_1 and *drop object* in S_3, as the person drops the bag and moves away from it. This also leads to the detection of a new object O_3 (the bag). While O_1 moves around in S_1 and S_3, object O_2 represented by the green trajectory, moves around in S_2. O_1 then moves into S_2 while O_2 is still present and the interactions between them are recognized as *independent*. O_2 then moves across the sub-networks and enters S_3 and it is seen that the global identity is maintained as objects move across sub-networks. While in S_3, the interactions between O_2 and O_3 are recognized as *pick object*, and then O_2 is tracked by C_1 and C_2 by panning and forming S_1, while the interaction in S_2 is recognized as *single*. Therefore, in this case, one person drops a bag but another person picks it up and moves on. This composite event is not explained by any of the HMMs built *a priori*, and is therefore, detected as an unusual composite event.