International Journal of Innovative
Computing, Information and Control
Volume **8**, Number **6**, June **2012**

ICIC International ©2012 ISSN 1349-4198
pp. 4219–4234

A COLLABORATIVE MULTI-AGENT FRAMEWORK FOR ABNORMAL ACTIVITY DETECTION IN CROWDED AREAS

Naveed Ejaz[1], Umar Manzoor[2], Samia Nefti[2] and Sung Wook Baik[1,*]

[1]College of Electronics and Information Engineering
Sejong University
No. 98, Gunja-Dong, Gwangjin-Gu, Seoul 143-747, Korea
naveed@sju.ac.kr; *Corresponding author: sbaik@sejong.ac.kr

[2]Department of Computer Science
The University of Salford
Salford, Greater Manchester M5 4WT, United Kingdom
umarmanzoor@gmail.com; s.nefti-meziani@salford.ac.uk

Received March 2011; revised July 2011

Abstract. *It is common to use Close Circuit Television (CCTV) cameras for the purpose of monitoring in urban areas. Because of laborious nature of the task, the human operators tend to lose attention level, thus causing a possibility of missing important events. The intelligent video surveillance systems can help human operators in performing automatic analysis of video feed for suspicious events. Most of the existing systems require segmenting individuals from the scenes for interpreting their actions. However, segmentation is usually not possible in high density crowded scenes. Moreover, there is a lack of work on automated generation of a collaborative view in a multi-camera network of CCTV cameras. In this paper, we propose an agent based framework for the detection of suspicious activities in crowded scenes in a distributed multi-camera CCTV network environment. The proposed scheme does not require segmentation of individuals from the scene and is thus insensitive to the crowd density. The use of Multi-Agent paradigm has incorporated decentralization, autonomy, fault tolerance and flexibility. We have evaluated our framework on our own generated dataset, a web dataset and a standard dataset from University of Minnesota. The results are promising and show the potential of our framework to work in real environments.*

Keywords: Crowd behavior analysis, Agent based video surveillance, Collaborative multi-agent framework, Abnormal activity detection

1. **Introduction.** The security threats in the world have changed their shape in the last decade or so. The primary security concern these days is from the terrorists attacking civilian and military targets round the world. Keeping these security threats in mind, the public and private organizations are striving to increase their defense capabilities against terrorist attacks [1]. For this purpose, the Closed Circuit Television (CCTV) cameras are frequently installed in urban areas these days. Traditionally, the CCTV feed is monitored by human operators observing multiple screens simultaneously for suspicious events. However, the published figures indicate that the ability of even a well motivated operator to concentrate on a monitor screen drops by 90% after 20 minutes [2]. For this reason, the human operators tend to miss important events in the CCTV feed.

One of the methods to cope with this problem is to employ intelligent video surveillance systems (video analytics) which are capable of automatic analyzing the CCTV feed to detect abnormal activities [3,4]. Most of these systems employ techniques that involve segmenting individuals from the scene and then interpreting their actions [5]. In such systems, each individual is separately tracked in the scene and his actions are matched with the pre-defined list of illegal activities. However, it may not be possible to segment out individuals from a high density crowd. Moreover, the analysis of crowded scenes poses additional challenges of occlusion, emergent behaviors and self-organizing activities [6].

This situation demands a dedicated framework for detection of abnormal activity in crowded scenes. The specialized techniques for analyzing crowd must not track people at individual level. Rather, a global preferential behavior of the crowd must be taken into consideration [6]. The basic assumption in such methodologies is that an abnormal activity by individual(s) in a crowded environment affects the overall dynamics of crowd [1].

Most of the intelligent surveillance systems designed for crowd behavior detection focus on automated analysis of a single camera feed, thus not giving the collaborative view of the activities [7]. Generally, multi-camera CCTV networks are installed at public places (universities, railway stations, airports, public places, etc.) in which Local Area Networks (LANs) are used to transmit video streams for recording and display purposes. The almost ubiquitous multi-camera CCTV networks demand efficient techniques for the generation of a collaborative view of all cameras in the network that can serve as a helping tool for the CCTV camera operator. In this context, we propose an agent based framework for the detection of suspicious activities in a distributed multi-camera CCTV network environment. Our framework is inspired by the agent based system for activity monitoring on network proposed in [8]. Multi-agent based architectures are ideal for network based application because of their distributed nature and flexibility [8]. The use of agent based architecture for the distributed system of multiple cameras results in a flexible, self recovering, fault tolerant, and decentralized system. To best of our knowledge, there is no study available which addresses the generation of a collaborative view for activity detection in a network of CCTV cameras.

We evaluated our framework on our own generated dataset, a standard dataset from University of Minnesota and a web dataset. Moreover, we compare our results with two other techniques in the literature based on 'Recall' and 'Precision' metrics. The results are promising and show that our technique generates less false positives.

The main contributions of our paper include

- Design of a novel agent based framework for collaborative detection of suspicious activities in a distributed multi-camera CCTV network environment,
- Consideration of common abnormal events in crowded scenes in a way that does not require segmenting individuals in the scene,
- Comparison of our framework with some of the other techniques in the literature based on a standard dataset.

The remainder of this paper is organized as follows. In Section 2, we present a brief overview of work done in crowd behavior monitoring. Section 3 justifies the use of agent based paradigm to model our distributed system. Section 4 introduces agent based architecture, each agent's responsibility and the coordination mechanism. In Section 5, we critically evaluate and compare the performance of the proposed system on different datasets. Finally, the conclusions are drawn in Section 6.

2. Previous Work.

Generally, most of the techniques presented for the automated recognition of crowd behavior assume that the individuals can be separated out from the crowd and can be tracked to predict the overall behavior of the crowd [6]. The specific to crowd systems are mostly concerned with the estimation of crowd density [11]. Crowd density estimation techniques can be classified into counting and empirical methods. The principle approach that is common in all counting based methods is to segment humans using features (face, gait, etc.) and count the number of persons in the image. The empirical methods generally classify crowd density on subjective scales instead of directly estimating crowd density. For instance, Marana et al. in [14] computes the Gray level dependence matrix (GLDM) using the texture features like contrast, homogeneity, entropy and energy.

Another perspective from which crowd behavior has been determined in literature is the detection of stationary objects in the crowded environments [12,15]. For instance, Chen et al. [13] proposed a three stage method to obtain wide field of view surveillance. However, detection of stationary objects is not sufficient to find suspicious activity in the crowd because of the occlusion of the stationary object.

Zhan et al. in [3] reviews the latest techniques used for crowd analysis and describes the two main approaches used in solving the problem of understanding crowd behaviors. The first approach, as stated earlier, assumes that crowd is a collection of individuals. This approach is generally not feasible in complex crowded scenes because of difficulties in segmentation, occlusion and tracking. The second approach treats the crowd as a global entity [16]. Isard et al. in [17] developed a Bayesian Multiple Blob Tracker that tracks individuals in each frame and computes crowd velocity information from each individual's information. However, this approach fails in dense crowds and is also affected by the distance of crowd from camera. Most of the global techniques for detection of crowd behavior generally use optical flow to extract crowd related features. Bouchafa et al. [30] proposed motion estimation and a motionless detection method in crowded scenes. In this scheme, optical flow has been employed for motion estimation whereas a motion detection based method is used for motionless detection.

A method for modeling crowd events in a controlled environment has been described in [19]. However, the technique is computationally expensive and restricted to constrained environments only. Mehran et al. in [6] modified the use of optical flow in the form of social force model to detect abnormal behaviors in crowded scenes. A grid of particles is placed over the image as simulation of individuals and their interaction forces are estimated. Ali et al. in [20] proposed a technique based on coherent structures from fluid dynamics and particle advection. The work presented in [15] gives an overview of motion-based methods to detect three scenarios in context of public transport surveillance. The work was based on a worldwide survey [21] which was conducted to know the major hazardous situations in public transport surveillance. The results of this survey identified over-crowding, unexpected directions of motion and stationary objects as three critical situations. Saxena et al. in [11] proposed a multiple-frame feature point detection and tracking, and modeled the crowd events for specific scenarios. Musa and Watada in [17] reviewed some important schemes for video tracking systems.

The study of existing literature reveals that there is lack of visual surveillance systems which takes complete CCTV network into consideration. Our system provides following unique features which distinguish it from other similar applications:

- Collaboration between individual cameras
- Fully autonomous (no human intervention)
- Distributed and decentralized design
- Fault tolerance
- Flexibility, scalability, manageability and adaptability

3. **Motivation for Using Agent Based Architecture.** In this section we provide justification for using agent based architecture in our framework. The proposed system is a distributed system composed up of individual nodes (IP cameras) collaborating with each other to track abnormal activities in crowded scenes. There are a variety of architectures

available for distributed systems including client/server, peer to peer and layered architectures, etc. However, the agent based architecture provides certain critical features that suites our application.

An agent is a program which autonomously acts on some environment on behalf of the user to achieve its design objectives [9]. When two or more agents coordinate their actions/data with each other in order to achieve a common goal(s), such a system is called Multi Agent System (MAS). Owing to its distributed nature and flexibility, MAS is particularly suitable for network based application [28]. Nwana [9] classified the agents into seven types based on their attributes. One of them is 'Mobile Agent' which has the capability of moving from one node to another node of the network autonomously with its state and data intact. A mobile agent uses the saved data and state to resume execution at new location. The unique characteristics (distributed nature, flexibility) of agent paradigm make it a perfect choice to implement real world application especially distributed applications [10].

We propose the use of a multi-agent system for the automatic detection of abnormal activity in an area monitored by a network of CCTV cameras. In our framework, a Controller Agent is responsible for initializing and monitoring the whole system. Controller Agent creates Video Surveillance Agent for each node, i.e., CCTV camera. Video Surveillance Agent creates Motion Vector Computation Agent which is responsible for detecting abnormal activity at one particular node. The Controller Agent waits for the response from the Video Surveillance Agents and alert the operator in case of any violation.

Mobile agents are appropriate for detection of abnormal activity on network because mobility allows autonomy needed to move on networks, allowing agents to autonomously monitor and report illegal activities on a node. Moreover, the use of mobile agents has made the system fully autonomous. The agent based architecture is chosen because of its tremendous capabilities like flexibility, self recovering, fault tolerance, and decentralization [10].

4. **Framework.** The proposed framework is based on multi-agent paradigm. We assume multiple-stationary cameras with stationary background for each camera. Further, it is assumed that background frame (a frame with no activity) for each camera is provided. These assumptions are quite reasonable considering the practical CCTV camera networks. The architecture of the system is based on layered architecture [22] whereby the layers are designed in a way that failure of one layer does not affect the overall performance of the system. The architecture is shown in Figure 1. There are three agents naming Controller Agent (CA), Video Surveillance Agent (VSA) and Motion Vector Computational Agent (MVCA).

4.1. **Controller agent (CA).** The responsibility of Controller Agent (CA) is to initialize and monitor the whole system. When the system starts up CA is created and system configuration XML is passed as argument which includes number of CCTVs in the network, CCTV ID, CCTV machine names, CCTV machine IP Addresses, Siblings Information (Neighboring CCTV IDs), etc. Sample XML is shown in Figure 2.

After creation CA performs the following tasks:

(1) CA loads the system configuration XML file and stores the information in vector set model C_T.
$$C_T = \{CCTV_1,\ CCTV_2,\ \ldots,\ CCTV_m\} \tag{1}$$
(2) CA makes a configuration map of the CCTV network for coordination/collaboration.
(3) CA creates and initializes N number of Video Surveillance Agents (VSA) where N depends on the number of CCTVs in the network. CA uses VSA to CCTV ratio of

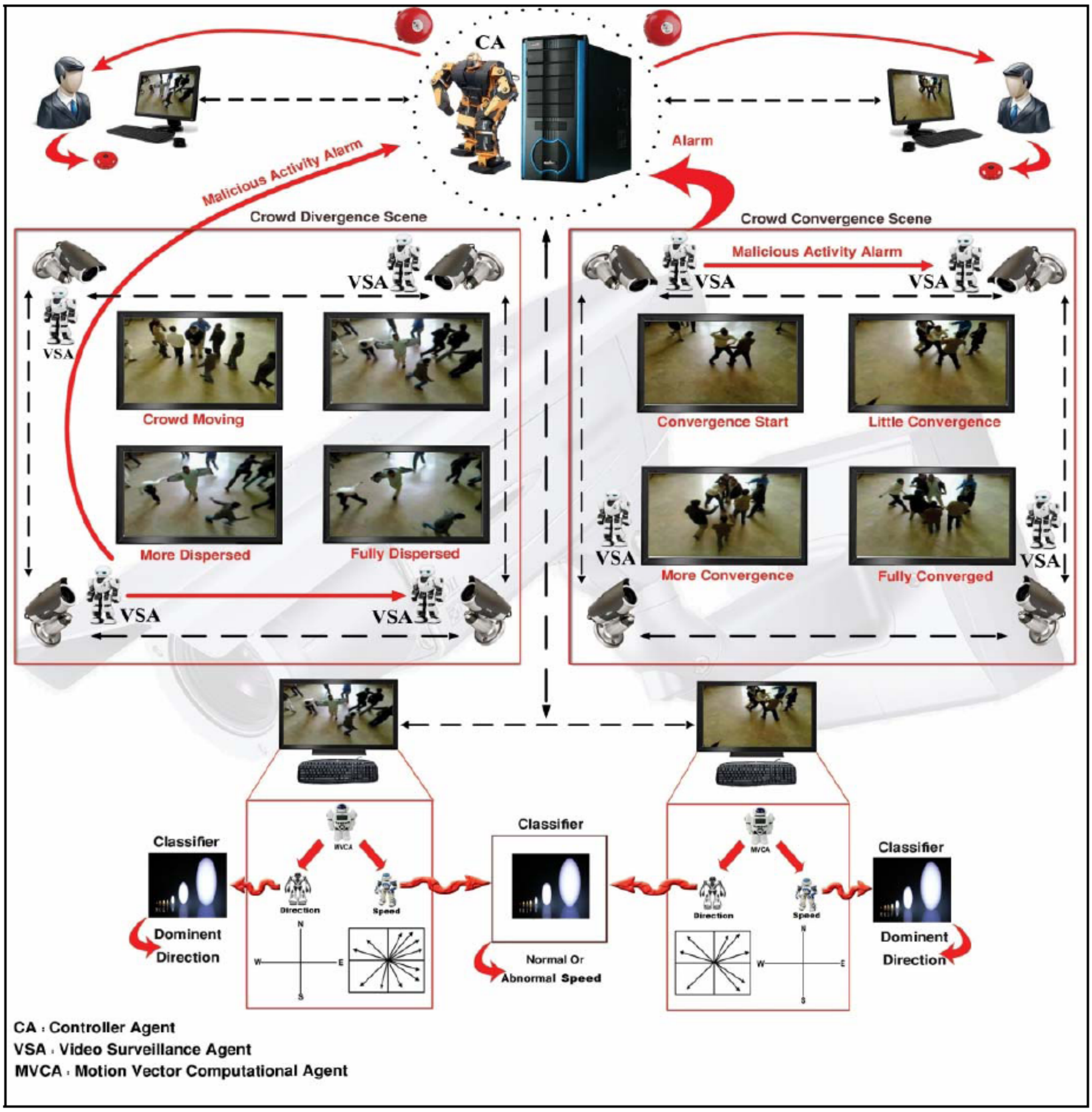


FIGURE 1. System architecture

1:1 (i.e., one VSA is responsible to monitor one CCTV video feed). After initialization each VSA moves to destination server to monitor video feed of the assigned CCTV.

(4) Once VSA reaches its destination server, CA transfers the compiled code of Motion Vector Computation Agent (MVCA) to VSA.

After initialization CA waits for the response from VSAs and alerts the operator in case of any violation. Under normal circumstances, CA displays all the CCTV video feeds on the operator screen. If any abnormal activity is reported by VSA(s), CA alerts the CCTV operator and changes the view on the operator screen so that operator can focus on the suspicious activity as shown in Figure 3(b). Suspicious activity reported to CA contains violation details, direction, magnitude (speed), time, etc. CA, in turn, passes this information to Operator View Module for operator view selection as well as updates its database with violation details. If the crowd is diverging in the suspicious

```
Sample Network XML File
<Cameras>
    <Camera 1>
          <ID>B-01</ID>
          <Configuration>
             <IP>172.168.4.10</IP>
             <MachineName>PC-1</MachineName>
          </Configuration>
          <Neighbors>
                <ID>B-02</ID>
                <ID>B-03</ID>
                   ...
                   ...
          </Neighbors>
    </Camera 1>
    </Camera 2>
       ...
       ...
    </Camera 2>
    ...
    ...
</Cameras>
```

FIGURE 2. Sample system configuration XML

(a) Original frames

(b) Key frames

FIGURE 3. Key frame extraction

activity, CA shows the neighboring CCTV feeds on the operator screen. Meanwhile, if another suspicious activity is reported, CA alerts the operator and changes the view.

4.2. **Video surveillance agent (VSA).** After creation and initialization, Video Surveillance Agent (VSA) moves to the assigned destination and performs the following steps:

(1) VSA creates Motion Vector Computation Agent (MVCA) and initializes it with video frame rate.
(2) VSA waits for the updates from MVCA. In case of any suspicious activity, it sends the violation details to CA and informs the neighboring VSA about the activity.

4.3. **Motion vector computational agent (MVCA).** Motion Vector Computation Agent (MVCA) plays very important role as it classifies the video feed as normal or abnormal. MVCA is responsible to extract the frames of the CCTV video feed and store them in frames vector set F.

$$F = \{F_1,\ F_2,\ \ldots,\ F_n | F_i \in C_T\} \tag{2}$$

C_T is the total frames in the CCTV video. MVCA classification algorithm consists of five steps: (I) Pre-processing; (II) Optical Flow Calculation; (III) Feature Extraction; (IV) Classification of Crowd Behavior and (V) Classification of Disturbed Crowd.

4.3.1. *Pre-processing.* Generally the size of frame vector set F contains redundant (similar/repeated frames) which can be removed to reduce the number of frames for processing. To reduce the size of frame vector F, we have used two approaches:

- We sample the frames by selecting only the alternate frame for processing. The size of the vector set model F is reduced to half after this step.
- The size of frame set is further reduced by extracting Key frames after sampling. For this purpose, we discard all those frames where there is no inter-frame movement. Figure 3 shows an example of this key frame extraction step. The set of key frames 'K_F' is then used for further processing.

$$K_F = \{KF_1,\ KF_2,\ \ldots,\ KF_n | KF_i \in C_T\} \tag{3}$$

After extraction of key frames, the background is removed from each key frame to separate the crowd pixels from the background [23]. This is done by simply subtracting the background frame from each key frame. This step removes clutter and reduces the number of pixels to process. After the subtraction of background image from the input frame, a grayscale image is obtained which consists of foreground pixels and some noise.

4.3.2. *Optical flow calculation.* The next step is to determine the crowd behavior by finding the dominant direction and speed of crowd. The optical flow has been used for this purpose. Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image sequence [25]. Algorithms for estimating optical flow compute a motion vector field by finding the motion of every pixel in the image [23]. Lucas Kanade Algorithm [26] has been used for the computation of optical flow. This is a differential method that calculates the optical flow between two frames taken at times t and $t + \delta t$. The method is based on local Taylor series approximations of the image. Lucas Kanade algorithm produces a sparse set of flow vectors and is robust in the presence of noise.

4.3.3. *Feature extraction.* The purpose of this step is to extract features from the optical flow of frames. The start and end points of velocity vectors, obtained from Lucas Kanade algorithm, are used to calculate the magnitude and direction θ. To reduce the computational cost, the velocity vectors are aggregated on discrete directions with an interval of 5. For each vector, the magnitude of x and y components are also calculated. After computing the required features for inter-frame movement, the global features are obtained by summing up the magnitudes of all those vectors whose directions are same.

4.3.4. *Classification of crowd behavior.* The purpose of this phase is to classify the behavior of crowd based on the motion vectors and the identified features. The behavior of the crowd is determined by the relative speed and direction. The motion vectors are used to derive a polar plot of accumulative optical flow of the scene. In order to classify the dominant direction of crowd movement, the polar graph has been divided into eight discrete regions. The different combinations of these regions can be used to classify the direction of crowd movement. Table 1 shows these eight regions and the range of angles included in each region.

Table 2 shows combinations of these regions and the direction label associated with these combinations. If any combination of regions contains percentage of vectors greater

Table 1. Identified regions in the polar graph

Region Number	**Range of Angles (In degrees)**
1	(340-20]
2	(20-70]
3	(70-110]
4	(110-160]
5	(160-200]
6	(200-250]
7	(250-290]
8	(290-340]

Table 2. Direction classification

Direction	**Regions of Max. Vectors**
East	R8, R1, R2
North	R2, R3, R4
West	R4, R5, R6
South	R6, R7, R8
North East	R1, R2, R3
North West	R3, R4, R5
South West	R5, R6, R7
South East	R7, R8, R1
Disturbed Crowd	Any other Combination

Table 3. Specific direction classification

Direction	**Regions of Max. Vectors**
Right Up	R1, R2
Up Right	R2, R3
Up Left	R3, R4
Left Up	R4, R5
Left Down	R5, R6
Down Left	R6, R7
Down Right	R7, R8
Right Down	R8, R1
Disturbed Crowd	Any other Combination

than a threshold (70% of the total vectors), the dominant direction is classified in the corresponding direction. Once the motion is classified, it can be made more specific by checking regions in combination of two as shown in Table 3.

4.3.5. *Classification of disturbed crowd.* If the optical flow vectors are not dominant in any region then this implies that the crowd is not moving in any specific direction (i.e., the crowd is either converging or diverging from a point). To quantify this behavior of crowd, four types of vectors are defined as shown in Figure 4.

The image is divided into 4 quadrants similar to the Cartesian coordinate system. All these types of vectors are labeled as 'Convergent Vector' or 'Divergent Vector' based on the image quadrant. Table 4 shows the interpretation of each vector corresponding to each quadrant.

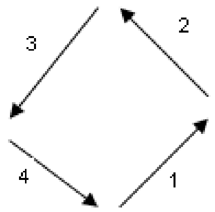


Figure 4. Type of vectors

Table 4. Classification of converging and diverging vectors

Vector Type	Quadrant	Label
1	1	Divergent
	2	Divergent
	3	Convergent
	4	Divergent
2	1	Divergent
	2	Divergent
	3	Convergent
	4	Divergent
3	1	Convergent
	2	Divergent
	3	Divergent
	4	Divergent
4	1	Divergent
	2	Divergent
	3	Convergent
	4	Divergent

Table 5. Different speed of crowd

Crowd Speed	Description
Blocked	Crowd is barely moving
Slow	Crowd is moving at a Slow Speed
Normal	Crowd is moving at a Walking Speed
Brisk	Crowd is moving briskly
Fast	Crowd is running

4.3.6. *Classification of speed.* The complete behavior of crowd cannot be determined without finding the speed of the crowd. For the computation of average speed of the crowd, the magnitudes of motion vectors of all iterations are summed up. Using this average speed, the average speed of entire scene is calculated. Five distinct speed labels are listed in Table 5. We use training based k-nearest neighbor algorithm (KNN) for classification of speed. If MVCA classification algorithm classifies the video feed as suspicious, it reports the event to the VSA for alerting the operator.

5. **Experiments and Results.** In order to test the framework we use three different crowd data sets. Firstly, we have developed our own data set of 50 videos having different behaviors. In our data set the scenes are relatively simple with no background clutter and noise. This data set contains videos of normal walking, fast moving, scattering and

converging crowded scenes. Secondly, we evaluate the proposed technique on various videos taken from web which contain real world crowd movements [29]. Lastly, we analyze our framework on publically available dataset of University of Minnesota [27]. This dataset contains videos of different scenarios of escape events. In each video, the starting part comprises of normal behavior which is followed by the sequences of the abnormal behavior. The ground truth annotative data is also available along with the videos. The training data used for this experiment contains a total number of 20 videos with varying speeds. For comparison with other techniques, we use a part of our system in which only a single camera is used. We compare our framework with scheme of Saxena et al. [11] and Bouchafa et al. [30].

Figure 5 shows the sequence of images for normal, fast, scattering and converging crowds respectively (from left to right). The corresponding optical flow polar graphs are shown in Figures 5(b), 5(d), 5(f) and 5(h). Table 6 shows the distribution of optical flow vectors in all eight regions and the classification of direction and speed as per the framework defined in Section 3. The decision of scattering and converging scenes is made by calculating the percentage of converging and diverging vectors as shown in Table 7.

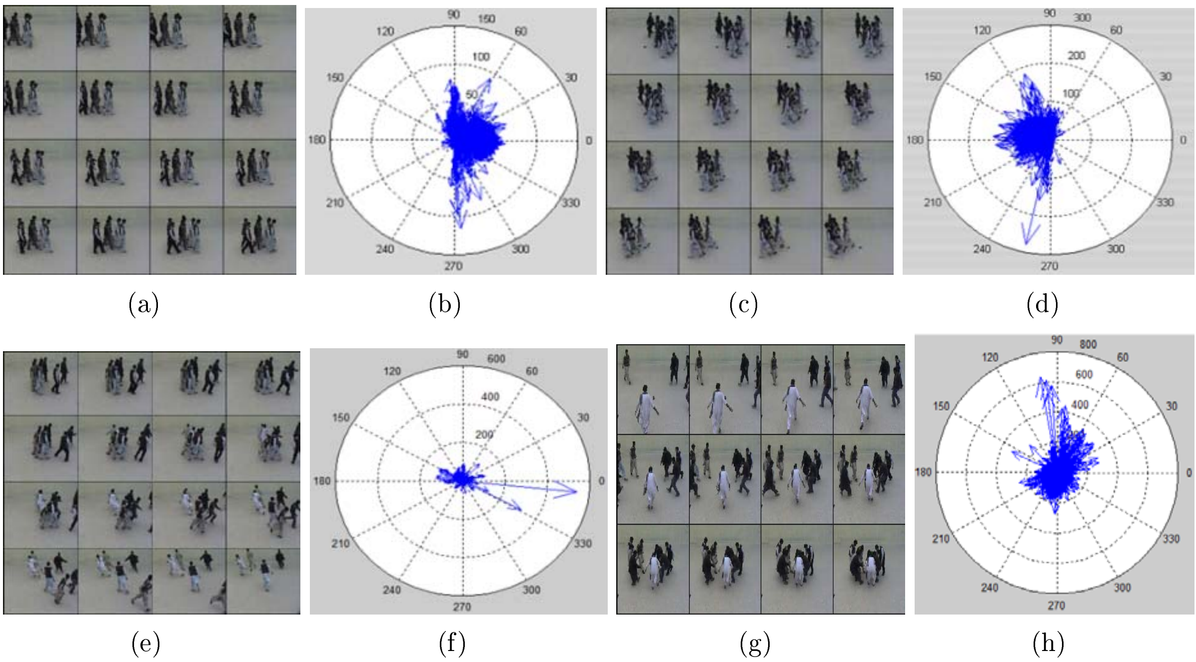


FIGURE 5. Normal, fast, scattering and converging scenes with corresponding polar graphs

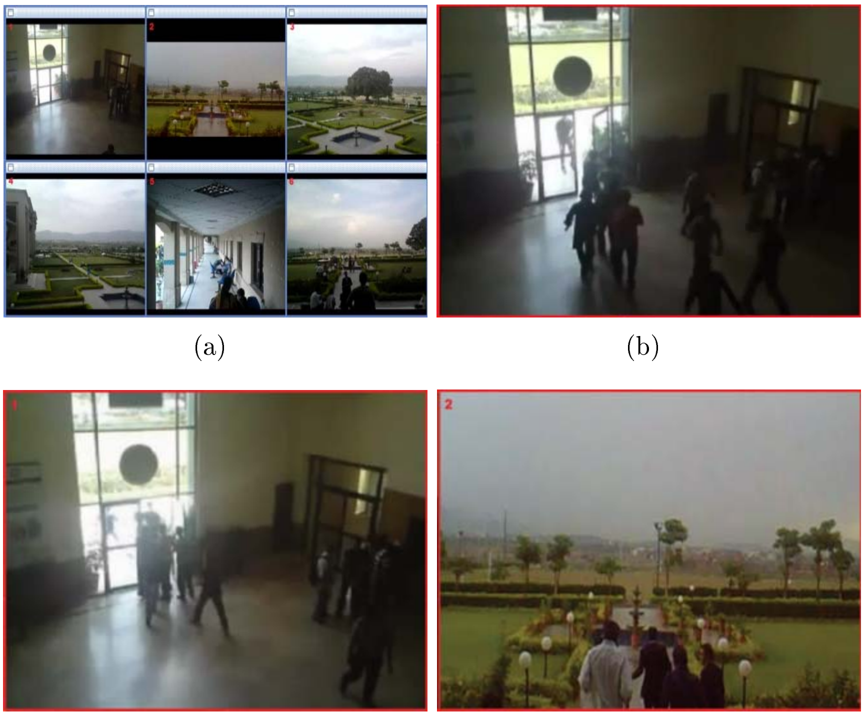
Figure 6 shows the result of collaboration using multi-agent framework. Figure 6(a) shows the operator view of the CCTV feed when everything is normal. There are 9 CCTV cameras in the scene. The malicious activity starts in the first CCTV camera feed in which a group of people start moving with a fast speed. This abnormal activity is reported by Video Surveillance Agent to Operator View Module via Controller Agent. The operator is alerted by diverting her attention on this particular camera feed by showing feed from only this camera and also buzzing the alarm as shown is Figure 6(b). From the dominant direction of crowd it is determined that the crowd is moving in the direction of CCTV Camera 2 so the operator's view is changed to display feed from both Camera 1 and Camera 2 as shown in Figures 6(c) and 6(d).

Table 6. Normal walk scene vectors clustered in regions

Region No.	**Percentage of Vectors**			
	Normal Scene	**Fast Scene**	**Scattering**	**Converging**
1	51.66	5.37	17.93	14.72
2	12.61	5.21	16.15	10.72
3	6.61	11.42	10.38	15.54
4	3.01	15.07	6.72	12.19
5	2.00	27.15	11.37	10.07
6	1.15	27.41	11.76	9.90
7	13.86	5.47	14.06	12.74
8	9.60	2.89	11.64	14.13
Average Speed	*6.9*	*10.16*	*8.82*	*14.47*
Speed Classification	*Normal*	*Fast*	*Fast*	*Fast*
Direction	*East*	*West*	*Scattering*	*Converging*

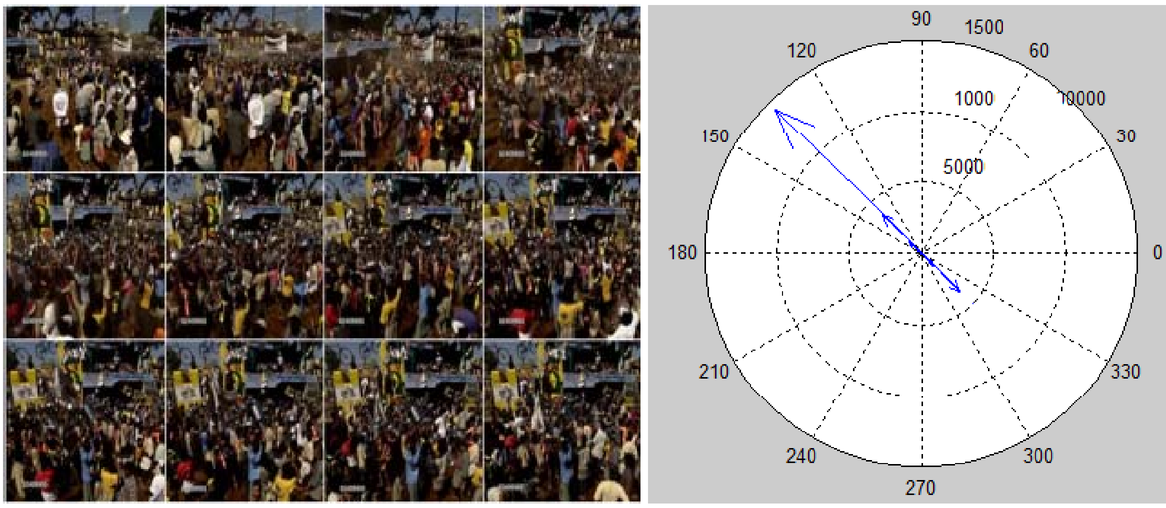
Table 7. Converging and diverging vectors

Method	**Scattering Scene**	**Converging Scene**
Total Vectors	38551	52398
Total Diverging Vectors (% age)	82.37	30.42
Total Converging Vectors (% age)	17.63	69.58
Decision	*Diverging*	*Converging*



(a) (b) (c) (d)

Figure 6. Operator's views in case of abnormal activity



(a) Web data scene 1 (b) Optical flow

Figure 7. A sample scene from web data set and corresponding optical flow

Figure 8. Different scenes in UMN dataset

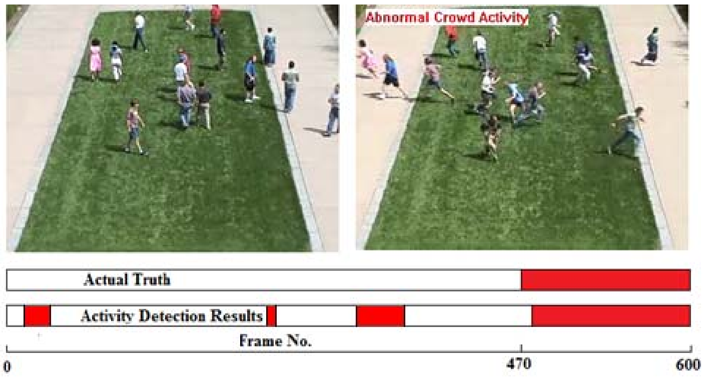
Figure 7(a) shows the sample frame of a public procession from web data set. The crowd is moving in one direction with normal speed so it is classified as normal motion. Figure 7(b) shows the corresponding optical flow polar graph.

Figure 8 shows a representative frame for each of the scenes in UMN data set.

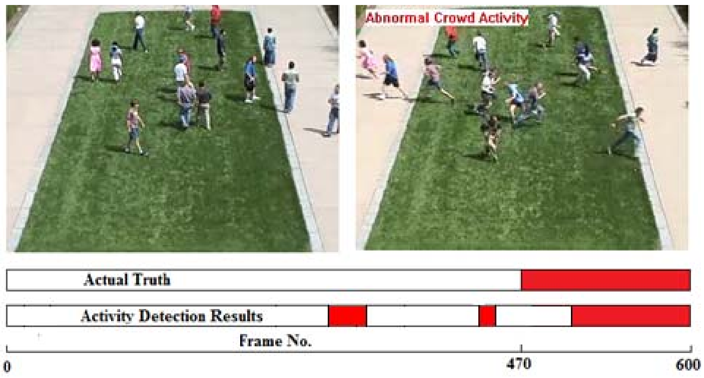
Figure 9(a) illustrates the first scene of UMN data set by showing sample normal and abnormal frames. Figure 9(b) shows the qualitative results of activity recognition for the technique of Saxena at al. [9]. As compared to ground truth, the technique yields three false positives. Figures 9(c) and 9(d) show the activity recognition diagram for the scheme of Bouchafa et al. and our technique respectively. Our technique results in less number of false positive and therefore yields results closer to the ground truth annotations.

We used metrics ‘Recall’ and ‘Precision’ to quantitative compare our method with rest of the techniques. In our context, Recall measures the probability that a detected abnormal event is actually relevant. On the other hand, Precision measures the probability that an actual abnormal event is detected. The advantage of using Recall and Precision is that they quantify the quality of visual surveillance in an objective fashion. It is worth mentioning that Recall and Precision are complementary metrics and a successful scheme

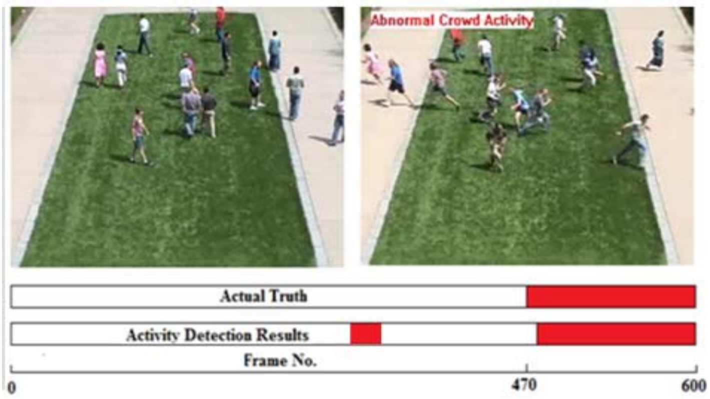
(a)



(b)



(c)



(d)

FIGURE 9. Visual results of various techniques on first video of UMN dataset

must have a high value for both of these parameters. Recall and Precision are defined as:

$$\text{Recall} = \frac{t_p}{t_p + f_n} \tag{4}$$

$$\text{Precision} = \frac{t_p}{t_p + f_p} \tag{5}$$

t_p, f_p and f_n are the number of true positives, false positives and false negatives respectively. Table 8 shows the average value of 'Recall' and 'Precision' for all the techniques under consideration for the three types of data sets. It is quite evident that our framework produces better results in all types of data sets showing its superiority. A higher value of Recall and Precision indicate that the system yield less number of false positives and false negatives. In other words, the results are closer to the ground truth annotations made by humans.

Table 8. Comparison of different techniques (R = Recall, P = Precision)

	Saxena et al.		**Bouchafa et al.**		**Our Technique**	
	R	P	R	P	R	P
Our Data Set	0.86	0.87	0.88	0.89	0.92	0.94
UMN Data Set	0.79	0.8	0.8	0.82	0.85	0.84
Web Data Set	0.76	0.75	0.78	0.76	0.81	0.79

Lastly, Table 9 illustrates the average time taken by each operation of the system. For these results the simulations have been carried out using general purpose computers (Pentium D 3.4 GHz equipped with 2 GB RAM). However, the more sophisticated hardware is expected to significantly reduce the computational time. It is clearly evident that the computation of optical flow is the bottleneck in the performance of our system. A more efficient scheme must be used for optical flow computation to make system workable in real time.

Table 9. Average time taken by different operations

Operation Description	**Average Time (s)**
CA creation and Initialization	2.4
VSA agent creation and initialization	2.8
MVCA agent creation and initialization	2.6
Computation of Optical Flow (10 second scene)	58.2
Detection of Malicious Activity (10 second scene)	4
Report a malicious activity on the network	1.3
Generating a collaborative view	7.65
Agents movement on the network	0.6

6. **Conclusions.** In this paper, we have presented an agent based framework for the detection of suspicious activities in crowded scenes in a distributed multi-camera CCTV network environment. We have demonstrated the ability of framework to detect the crowd behavior and generate collaborative view for the operator. We also performed a quantitative comparison of our technique with other techniques in the literature. The proposed approach results in effective surveillance along with providing a fault tolerant, dependable and flexible system. However, the effectiveness of our framework can be further improved by employing an efficient implementation of optical flow. Moreover, the algorithms can

be implemented in hardware to provide real time capabilities. In future we intend to combine the feed of multiple cameras to generate a 3D view in collaboration phase.

Acknowledgment. This research is supported by Seoul Research and Business Development (R & BD) Program (JP090972M0214832).

REFERENCES

[1] E. L. Andrade, S. Blunsden and R. B. Fisher, Modeling crowd scenes for event detection, *Proc. of the 18th Int. Conf. on Pattern Recognition*, Hong Kong, pp.175-178, 2006.

[2] C. Gomersall, *Challenges and Developments in Intelligent Video Surveillance*, http://www.sour cesecurity.com/news/articles/co3223ga.1852.html.

[3] B. Zhan, D. N. Monekosso, P. Remagnino, S. Velastin and L.-Q. Xu, Crowd analysis: A survey, *Machine Vision and Applications*, vol.19, no.5, pp.345-357, 2008.

[4] M. Saitou, A. Kojima, T. Kitahashi and K. Fukunaga, Dynamic recognition of human actions and objects using dual hierarchal models, *International Journal of Innovative Computing, Information and Control*, vol.3, no.6(A), pp.1359-1368, 2007.

[5] D. Helbing, P. Molnár, I. J. Farkas and K. Bolay, Self-organizing pedestrian movement, *Environment and Planning B: Planning and Design*, vol.28, no.3, pp.361-383, 2001.

[6] R. Mehran, A. Oyama and M. Shah, Abnormal crowd behavior detection using social force model, *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Miami, pp.935-942, 2009.

[7] Z. B. Musa and J. Watada, Multi-camera tracking system for human motions in different areas and situations, *International Journal of Innovative Computing, Information and Control*, vol.4, no.5, pp.1213-1222, 2008.

[8] U. Manzoor and S. Nefti, An agent based system for activity monitoring on network – ABSAMN, *Expert Systems with Applications*, vol.36, no.8, pp.10987-10994, 2009.

[9] H. S. Nwana, Software agents: An overview, *Knowledge Engineering Review*, vol.11, no.3, pp.205-244, 1996.

[10] R. Wang, X. Wang and M. J. Kim, Motivated learning agent model for distributed collaborative systems, *Expert Systems with Applications*, vol.38, no.2, pp.1079-1088, 2011.

[11] S. Saxena, F. Bremond, M. Thonnat and R. Ma, Crowd behavior recognition for video surveillance, *Proc. of the 10th Int. Conference on Advanced Concepts for Intelligent Vision Systems*, pp.970-981, 2008.

[12] T. T. Zin, P. Tin, H. Hama, S. Nakajima and T. Toriu, Effective multiple stochastic background modeling for stationary objects detection in complex environments, *ICIC Express Letters*, vol.5, no.10, pp.3767-3772, 2011.

[13] W.-C. Hu, D.-Y. Huang and W.-H. Chen, Adaptive wide field-of-view surveillance based on an IP camera on a rotational platform for automatic detection of abandoned and removed objects, *ICIC Express Letters, Part B: Applications*, vol.1, no.1, pp.45-50, 2010.

[14] A. N. Marana, S. A. Velastin, L. F. Costa and R. A. Lotufo, Automatic estimation of crowd density using texture, *Safety Science*, vol.28, no.3, pp.165-175, 1998.

[15] S. A. Velastin, B. A. Boghossian and M. A. V. Silva, A motion-based image processing system for detecting potentially dangerous situations in underground railway stations, *Transportation Research Part C: Emerging Technologies*, vol.14, no.2, pp.96-113, 2006.

[16] A. B. Chan and N. Vasconcelos, Mixtures of dynamic textures, *Proc. of the 10th IEEE Int. Conf. on Computer Vision*, Washington DC, USA, pp.641-647, 2005.

[17] M. Isard and J. MacCormick, BraMBLe: A bayesian multiple-blob tracker, *Proc. of Int. Conf. on Computer Vision*, Canada, pp.34-41, 2001.

[18] Z. Musa and J. Watada, Video tracking system: A survey, *ICIC Express Letters*, vol.2, no.1, pp.65-72, 2008.

[19] P. Reisman, O. Mano, S. Avidan and A. Shashua, Crowd detection in video sequences, *Proc. of IEEE Intelligent Vehicles Symposium*, Italy, pp.66-71, 2004.

[20] S. Ali and M. Shah, A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis, *IEEE Conf. on Computer Vision and Pattern Recognition*, pp.1-6, 2007.

[21] A. Langlais, User needs analysis, *CROMATICA TR-1016 (CEC Framework IV Telematics Applications Programme)*, 1996.

[22] G. Weiss, *Multi-Agent Systems a Modern Approach to Distributed Artificial Intelligence*, The MIT Press Cambridge, Massachusetts, USA, 1999.
[23] C. Stauffer and W. E. L. Grimson, Adaptive background mixture models for real-time tracking, *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, HI, USA, pp.246-252, 1999.
[24] C.-S. Liu, N.-C. Yang, C.-M. Kuo and W.-L. Wei, Motion vector re-estimation for video transcoding with arbitrary video downsizing, *International Journal of Innovative Computing, Information and Control*, vol.6, no.6, pp.2657-2670, 2010.
[25] E. L. Andrade and R. B. Fisher, Simulation of crowd problems for computer vision, *Proc. of the 1st Int. Workshop on Crowd Simulation*, Switzerland, pp.71-80, 2005.
[26] B. D. Lucas and T. Kanade, An iterative image registration technique with an application to stereo vision, *Proc. of Imaging Understanding Workshop*, pp.121-130, 1981.
[27] *Unusual Crowd Activity Dataset of University of Minnesota*, http://mha.cs.umn.edu/movies/crowdactivity-all.avi.
[28] H. Chen, Z. Ji, M. Hu, Z. Fu and R. Jiang, Design and performance evaluation of a multi-agent-based dynamic lifetime security scheme for AODV routing protocol, *Journal of Network and Computer Applications*, vol.30, no.1, pp.145-166, 2007.
[29] *Crowd Data Set from Computer Vision Lab, University of Central Florida*, http://server.cs.ucf.edu/~vision/.
[30] S. Bouchafa, D. Aubert and S. Bouza, Crowd motion estimation and motionless detection in subway corridors by image processing, *IEEE Conf. on Intelligent Transportation System*, pp.332-337, 1998.