

Real-time abnormal situation detection based on particle advection in crowded scenes

Yunyoung Nam · Sangjin Hong

Received: 22 August 2013 / Accepted: 8 April 2014
© Springer-Verlag Berlin Heidelberg 2014

Abstract This paper presents a real-time abnormal situation detection method in crowded scenes based on the crowd motion characteristics including the particle energy and the motion directions. The particle energy is determined by computation of optical flow derived from two adjacent frames. The particle energy is modified by multiplying the foreground to background ratio. The motion directions are measured by mutual information of the direction histograms of two neighboring motion vector fields. Mutual information is used to measure the similarity between two direction histograms derived from three adjacent frames. The direction probability distribution for each frame can be directly estimated from the direction histogram by dividing the entries by the total number of the vectors. A metric for all the video frames is computed using normalized mutual information to detect the abnormal situation. Both the modified particle energy and mutual information of direction histograms contribute to the detection of the abnormal events. Furthermore, the dynamic abnormality is measured to detect the dynamical movement associated with severe change in the motion

state according to the spatio-temporal characteristics. In experiments, we will show that the proposed method detects the abnormal situations effectively.

Keywords Video scene analysis · Crowd behaviors · Abnormal event detection · Optical flow

1 Introduction

With the rapidly growing of Internet and storage capacity, IP-based video monitoring systems have become popular applications. As network video technology has improved, the cost of installing a surveillance system has dropped significantly, leading to an exponential increase in the use of security cameras. A typical video surveillance system employs a number of networked cameras and with the development of new technology. However, many of these cameras can have disjointed views [22, 23]. Human monitoring of surveillance cameras can be both expensive and ineffective due to the fact that each camera provides a huge amount of information [25], and even formally trained operations can lose concentration and miss important details after only ten minutes. Consequently, users are turning to intelligent software for automated video surveillance [29].

The ability to detect unusual or abnormal events is one of the most important issues to be addressed in the development of intelligent surveillance systems [9, 34, 35]. In particular, abnormal situation detection in crowded scenes is a challenging problem due to the potential complexity of a situation [9, 34, 35]. The properties of these unusual events may vary depending on environmental factors such as lighting (day/night), space (indoor/outdoor), and time of day (peak/off peak). The challenge problem is composed of

This research was supported by the Soonchunhyang University Research Fund and also supported by the International Collaborative R&D Program of the Ministry of Knowledge Economy (MKE), the Korean government, as a result of Development of Security Threat Control System with Multi-Sensor Integration and Image Analysis Project, 2010-TD-300802-002.

Y. Nam (✉)
Department of Computer Science and Engineering,
Soonchunhyang University, Asan 336-745, Korea
e-mail: ynam@sch.ac.kr

S. Hong
Department of Electrical and Computer Engineering, Stony
Brook University-SUNY, Stony Brook, NY, USA

several sub-challenges such as low-level processing, optical flow analysis and recognition based on classifying methodology. We considered a variety of motion characteristics including direction and magnitude. Due to the complexity of the crowded scenes, the direction histogram peaks are needed to identify significant local peaks, but are difficult to obtain. Various motion speeds can be detected using the standard approach of differences between frames, and although it not easily attained in real-world scenarios, especially in crowds, the perceived motion speed can be measured by the local level of movement or the local area size.

Crowd density is an important aspect of the foreground image and is related to image potential energy model. Based on the results of foreground segmentation, we can estimate crowd density on each flow respectively using the texture feature [18] or the moment feature [27]. Since these estimations are not satisfactory because of their inaccurate results, there are other approaches [14, 17, 16] in which crowd density can be obtained with more precision. In this paper, crowd density is estimated by the foreground to background ratio in each frame. In general, real world, actions usually occur in crowded scenes which have cluttered and dynamic environments. In these scenes, object segmentation results are often unreliable. Consequently, it is not always possible to track individual objects, especially in high-density crowds. In this paper, we will present real-time abnormal situation detection methods based on features of motion vectors and trajectories of moving objects in crowded scenes. We concentrate on monitoring emergency situations in crowds by classifying patterns of normal motion flow to identify unusual or emergency events. In this paper, abnormal situations are detected using particle advection, energy, and the normalized mutual information (NMI). We have compared our proposed method based on energy and NMI with the existing methods [19, 32] to evaluate the accuracy of abnormal events detection.

The remainder of this paper has six sections. In Sect. 3, we describe an overview of the application model and the process flow of our proposed methods. In Sect. 4, we address the problem of identifying the optical flow that contains velocity and direction. Section 5 explains grid size determination algorithm and presents abnormal situation detection methods in real time. Section 6 shows qualitative and quantitative experimental results. Section 2 reviews related work on video analysis of crowded scene. Finally, we conclude this paper in Sect. 7.

2 Related work

To classify abnormal events in crowds, normal behaviors are modeled and deviations from those models are

considered abnormal. In [2], HMM and principal component analysis are used. The authors combine Hidden Markov Models, spectral clustering, and principal component analysis of optical flow vectors for detecting crowd emergency scenarios. However, their experiments were carried out on simulated data.

Lagrangian particle dynamics [1] is used for the detection of flow instabilities and is an efficient methodology only for the segmentation of high-density crowd flows such as marathons or political events. However, since all particles with the same source or destination must be grouped together, it may be difficult to extract all of the semantically important dominant motions in a given video.

In [11], optical flow is used to detect when abnormal events occur without necessarily pointing the precise region of interest into the frames. The approach does not need a huge amount of data to enable learning pattern frequency but it is necessary to carefully define, in advance, an appropriate threshold and the regions of interest for every scenario. This approach works on uni-directional areas such as elevators. Also, the authors only considered the variance of motion characteristics such as direction variance and motion magnitude variance, but ignored the total value of the motion magnitude in a crowd event. The author also included direction histogram peaks in their system, but in most crowd scenes, the direction histogram peaks are hard to obtain.

Social force model [19] is used for the detection of abnormal behaviors in crowds. The method consists in matching a grid of particles with the frame and moving them along the underlying flow field. Then, the social force is computed between moving particles to extract interaction forces, to finally determine the ongoing behavior of the crowd through the change of interaction forces in time. Chaotic invariants [32] of Lagrangian particle trajectories are used to model abnormal patterns in crowded scenes.

In [31], events are modeled by grouping low-level motion features into topics using hierarchical Bayesian models. This method processes simple local motion features and ignores global context. Thus, it is well suited for modeling behavior correlations between stationary and moving objects but cannot model complex behaviors that occur in a big area of the scene. Also, a hierarchical pLSA [13] has been proposed to incorporate semantic scene representation to model global and local behavior patterns. However, these models do not incorporate timestamps associated with the occurrence of activities.

3 Application model and problem description

Our goal is to automatically recognize abnormal events in crowded scenes. Figure 1 shows examples of crowded and

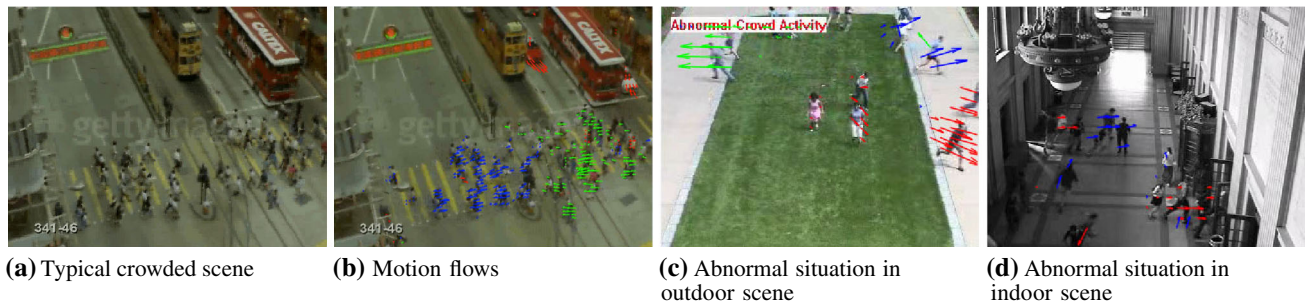


Fig. 1 Illustration of crowded objects motion analysis method

complicated scenes, such as a crossroad, a school, and shopping malls. Typical public scenes are crowded, creating difficulties for segmentation or tracking. In such scenes, it is often not easy to track individual objects because of frequent occlusions among objects, and because many different types of events often happen simultaneously. In this paper, we propose an intelligent video surveillance system to: (i) discover motion vectors and crowd flows in crowded scenes as shown in Fig. 1b–d; (ii) detect abnormal events as shown in Fig. 1c, d.

Our approach utilizes motion flows of the features based on optical flow in a scene. Generally, most traditional approaches on anomaly crowd event in visual surveillance systems analyze motion patterns following four steps: (i) preprocessing and feature extraction including background modeling [33] and optical flow computation [5], (ii) detecting and tracking of the motion flows present in the scene [20], (iii) extraction of motion patterns from the tracks, and eventually (iv) detection of events using motion patterns information. The change detection module starts with the adaptive mixture of Gaussians algorithm to extract the foreground and to exclude the noise [36]. We apply a median filter to the feature set and then use a mean filter to smooth the motion field and to remove the noise features.

The proposed methods analyze motion flows to detect the abnormal situation in crowded scenes. The methods are to measure motion features including speed and direction of motion flows based on a spatio-temporal approach. The speed is directly related to the length of the vectors. For each frame, the speed of every vector is calculated by its length and the number of tracking frames of the feature point associate with the vector. Also, the average vector of the optical flow vectors in a crowded scene is obtained by the direction and the magnitude of all the vectors in the frame.

Figure 2 illustrates the process flow of our proposed methods. A typical scenario for the abnormal behavior determination is as follows. First, our method extracts features from image sequences and adjusts feature values for effective measurement. Second, energy and direction histograms are calculated by motion vectors in a given image sequence. In the feature vector energy calculation step, the

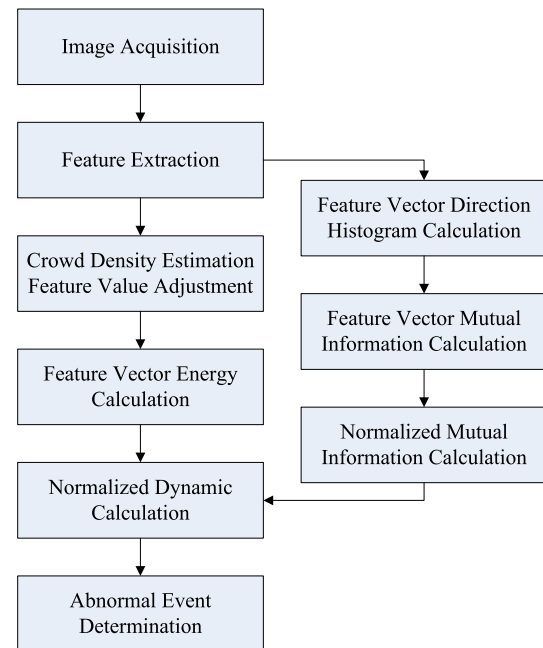


Fig. 2 Illustration of the general process flow for abnormal behavior detection

features are extracted by grid-based particle advection. In this paper, the features are a grid of points. We assume that the feature number of people is relatively stable in adjacent frames. Third, the mutual information and normalized mutual information are calculated by the direction histogram. In order to get a desirable output, it is necessary to choose the appropriate threshold carefully. Unfortunately, it is difficult to determine this threshold, and the false alarm rate is usually high. We propose the grid size determination algorithm to compute and optimize the thresholds. Finally, abnormal behavior is detected based on optimized thresholds and results are displayed on the screen.

4 Motion vector and data characterization

In this section, we address the problem of identifying the optical flow that contains velocity and direction.

4.1 Motion vector

A motion vector is represented by a line segment with a definite direction, connecting an initial point p with a terminal point q . When coordinates of a initial point are $p_k^{ij} = (x_k^{ij}, y_k^{ij})$, motion vector M_k^{ij} is obtained from tracking features of a series of images which are two adjacent frames $i, i-1$. M_k^{ij} is composed of a initial position p_k^{ij} , magnitude d_k^{ij} , and direction a_k^{ij} as follows:

$$M_k^{ij} = (p_k^{ij}, d_k^{ij}, a_k^{ij}), \quad (1)$$

where i is a frame number, j is a block number, and k is a motion vector number. Magnitude d_k^{ij} is computed by

$$d_k^{ij} = \sqrt{(x_k^{ij} - x_k^{i-1j})^2 + (y_k^{ij} - y_k^{i-1j})^2}. \quad (2)$$

In addition, velocity v_k^{ij} is calculated by

$$v_k^{ij} = \frac{\sum_{k=1}^m d_k^{ij}}{m}, \quad (3)$$

where m is the number of vectors in block j of frame i .

Orientation a_k^{ij} of M_k^{ij} is determined by the coordinates of the initial points which are two adjacent frames. The principal crowd directions are calculated considering the density percentage of feature points in each direction. If this percentage is bigger than a threshold, we assume there is a crowd in that direction.

For computing efficiently, the maximum number of features is limited and the camera image is divided into $m * n$ blocks. m and n values should be determined on viewing distance between a camera and a object. In this paper, the particle energy is measured by multiplying the foreground I_f to background ratio r_i of i th image as follows:

$$r_i = \frac{I_f}{I_{\text{width}} \times I_{\text{height}}}, \quad (4)$$

where I_{width} and I_{height} are the width and the height of a background image, respectively. This ratio is a rough estimate of the crowd density. The accurate crowd density information is not necessary for abnormal events detection.

4.2 Direction histogram of motion vector

We constructed a direction histogram from the motion vectors in a frame. In this paper, we divide 2π into 36 bins for the direction histogram. The calculated histogram indicates the direction distribution and direction tendency of the frame. For example, 2π is divided into eight bins, the Cartesian plane is divided into eight parts where each part is a direction between the angles $[\alpha, \alpha + 45]$ and $\alpha \in \{0, 45, 90, 135, 180, 225, 270, 315\}$ using the following equation:

$$A_k^{ij} = e, \text{ if } \left(\frac{2\pi}{N} \cdot (e-1)\right) < \frac{a_k^{ij} + 2\pi}{\frac{2\pi}{h}} \leq \left(\frac{2\pi}{N} \cdot e\right), \quad (5)$$

$$1 \leq e \leq N,$$

where h is the number of direction histogram bins and N is the number of quantization levels.

However, the direction of moving vectors can vary with the density percentage of feature points in each direction. Thus, depending on the direction distribution, each dominant directional window w_i should be adjusted to classify the principal and minor crowd directions. After analyzing the direction histogram for each frame based on dominant directional window selection method, the opposite direction of motion vector can be detected by direction probability distribution. For each direction, the size of w_i is determined by threshold a_{Eth}^i , which is calculated by

$$a_{Eth}^i = \frac{1}{N(b)} \times nb(M_k^i), \quad (6)$$

where $nb(M_k^i)$ is the total number of vectors. To find w_i in the histogram, x-intercept x_a is obtained by intersection with the graph and $y = a_{Eth}^i$ using the following equation:

$$x_a = \begin{cases} 0, & \text{if } p(0) \geq a_{Eth}^i \\ j, & \text{if } p(j) \cdot p(j+1) \leq 0 \text{ and } p(j) \geq a_{Eth}^i \\ j+1, & \text{if } p(j) \cdot p(j+1) \leq 0 \text{ and } p(j) \leq a_{Eth}^i \\ b-1, & \text{if } p(b-1) \geq a_{Eth}^i \end{cases} \quad (7)$$

The following set of X represents the bin number b in the histogram, when the number of vectors is above a_{Eth}^i .

$$X \in \{x_1, x_2, \dots, x_a, \dots, x_n\}, 0 \leq a < N(b), a \in N \quad (8)$$

Consequently, w_i is calculated by

$$w_i = \begin{cases} [x_a, x_{a+1}], & \text{if } a \text{ is an odd number and} \\ x_a \neq 0 \text{ and } x_n \neq N(b) \\ [x_{n-1}, x_a], & \text{if } x_a = 0 \text{ and } x_n = N(b) \end{cases} \quad (9)$$

4.3 Magnitude of motion vector

Figure 3 shows the simulation result of magnitude distribution of type 1–5 vectors, when ten moving objects move from a left region to a right region. Generally, static features move less than a minimum magnitude. By contrast, noise features have magnitudes that exceed the threshold. Noise features are the isolated features that have a big angle and magnitude difference with their near neighbors due to tracking calculation errors. In the simulations, we set the minimum motion magnitude to $\epsilon (= 1.0e^{-6})$ pixel per frame and the maximum to $\frac{\sqrt{(I_{\text{width}})^2 + (I_{\text{height}})^2}}{2}$ pixels per frame. The minimum and maximum values were taken experimentally. As objects move faster, the more features of motion vectors are lost. Ideally, the $4 \times 4 \times 3$ motion

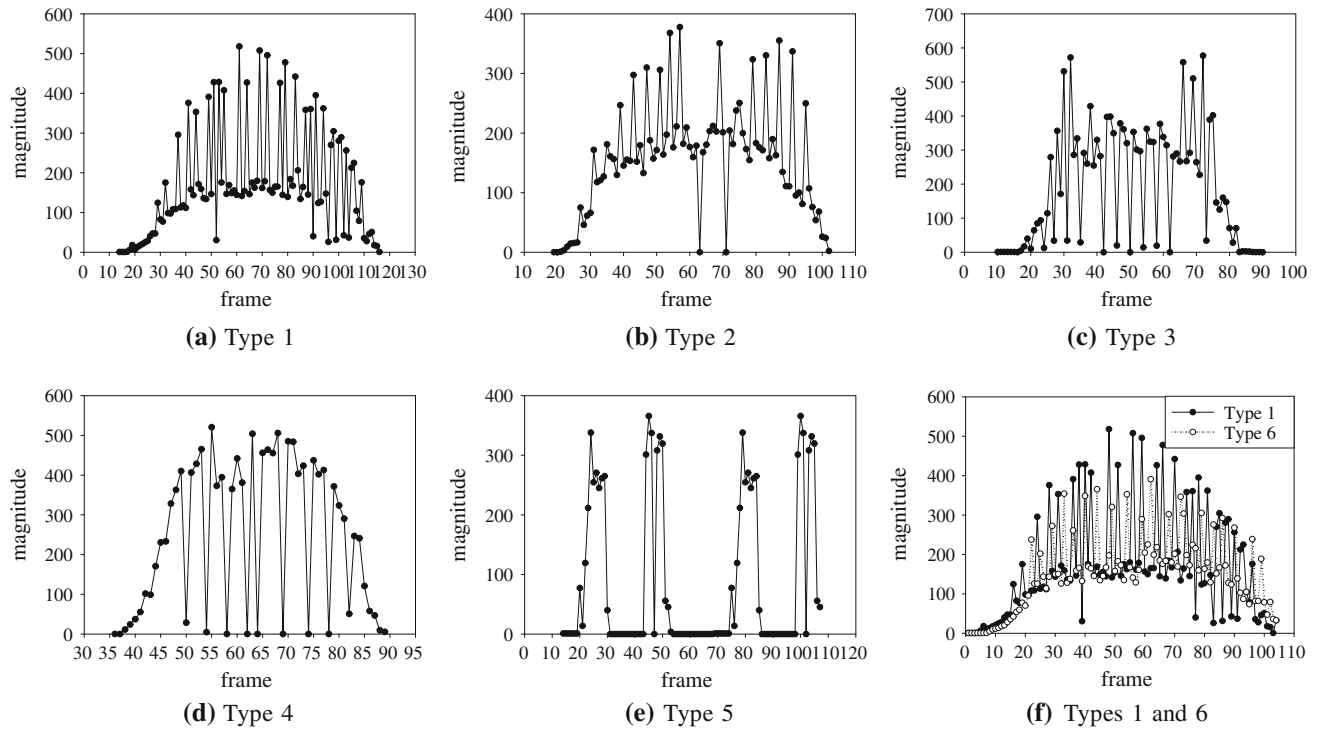


Fig. 3 Magnitude distribution of type 1–5 vectors. Ten objects move from a left region to a right region at speeds of 67 (type 1), 80 (type 2), 100 (type 3), 133 (type 4), 200 (type 5) pixels for a second. Motion

features of type 6 are obtained when one object moves from a right region to a left region while ten objects moves from a left region to a right region

vectors can be obtained from a object that is of size 15×15 pixels and moves at 67 pixels per second when a grid of points is 5×5 . However, motion features of the type 4 were not obtained in frames 50, 54, 58, 62, 64, 69, 74, 78, and 82.

Even though it is necessary to measure the average number of moving motion vectors according to the direction histogram, a problem arises when all vectors do not move in one direction, which means it can lose the feature value of each individual motion vector that moves in the opposite direction. The number of moving motion vectors in each direction is calculated by

$$nb(A_k^{ij}, t) = \sum_{i=1}^t \sum_{k=1}^m nb(a_k^{ij}), \quad t \geq 3, \quad (10)$$

where t is selection time. Also, the direction probability distribution for each frame can be directly estimated from the direction histogram by dividing the entries by the total number of the vectors as follows:

$$nb'(A_k^{ij}, t) = \frac{nb(A_k^{ij}, t)}{nb(M_k^{ij}, t)}, \quad 0 < A_k^{ij} \leq N, \quad (11)$$

where $nb(M_k^{ij}, t)$ is the total number of vectors for time t .

Figure 3 shows the comparison of motion features of type 6 with that of type 1. Motion features of type 6 are

obtained when one object moves from a right region to a left region while ten objects moves from a left region to a right region. The total number of motion features is reduced by that of an opposite moving object. Thus, magnitude distribution should be computed by considering correlations between the magnitude and the direction of vectors. The improved magnitude and velocity of vectors is calculated by

$$d_{k,A_k^{ij}}^{ij} = \sqrt{(x_k^{ij} - x_k^{i-1,j})^2 + (y_k^{ij} - y_k^{i-1,j})^2}, \quad (12)$$

$$\text{if } A_k^{ij} \ni a_k^{ij}$$

$$v_{k,A_k^{ij}}^{ij} = \frac{\sum_{i=1}^n \sum_{k=1}^m d_{k,A_k^{ij}}^{ij}}{n \times m}, \quad \text{if } A_k^{ij} = e \quad (13)$$

$$V_{k,A_k^{ij}}^{ij} = [v_{k,A_k^{ij}}^{ij} - \rho, v_{k,A_k^{ij}}^{ij} + \rho], \quad \text{if } A_k^{ij} = e \quad (14)$$

The calculation of direction and magnitude variances is not sufficient to detect abnormal situation. We build a direction histogram in which each column indicates the number of vectors in a given angle. The result is a histogram that indicates the direction tendencies, and the number of peaks in this histogram represents the different directions.

4.4 Flux analysis based on vector direction depending on grid size

Figure 4 shows a normalized direction histogram and magnitude of vectors in an image of crosswalk when the grid size are 2×2 , 5×5 , 10×10 , and 20×20 . In the figure, the normalized value of the direction histogram Z is calculated by

$$Z_{i,t} = nb'(A^i, t) \times 100, \quad (15)$$

where A^i is derived from Eq. 5. As mentioned in Sect. 4.2, as the grid size is smaller, the size of each w_i is larger, as shown in Fig. 4a. When each grid size is 2×2 , 5×5 , 10×10 , and 20×20 , the values of a_{th} is 11790.42, 1871.22, 451.08, and 114.17. When the grid size is 2×2 , w_i is in $[320^\circ, 10^\circ]$ and $[140^\circ, 190^\circ]$. When the grid size is 20×20 , w_i is in $[340^\circ, 0^\circ]$, $90^\circ, 260^\circ$, and $[160^\circ, 180^\circ]$. In other words, the sizes of w_i are 12, 9, 7, 8, when each grid size is 2×2 , 5×5 , 10×10 , and 20×20 . As the grid size is larger, the number of motion vectors in a crowded area is reduced. On the other hand, the number of motion vectors in a slack area is not changed. For example, the graph of 20×20 grid shows that additional w_i is in 90° and 260° . The motion vectors at 90° and 260° are captured by two moving cars on the road and several pedestrian in the top right corner, respectively.

Depending on a grid of points, the motion vectors are obtained from particle advection features of each frame. If the number of features does not change among the video sequence and the feature number for a object is relatively stable in adjacent frames, these features can be refined using a mask which represents the crowd region. As shown in Fig. 4b, the graph of vectors in 20×20 grid shows a wide range of magnitude. The variances of normalized

magnitude are 1.14, 2.31, 6.18, and 34.15 in 2×2 , 5×5 , 10×10 , and 20×20 grids. As the grid size is smaller, the variance of the magnitude is smaller.

5 Real-time abnormal event detection

5.1 Preprocessing step

In a preprocessing step, a grid of points is first generated as the feature set. Adaptive Gaussian mixture modeling (GMM) method [33] is applied to extracting the foreground of the video scene. The motion vectors are captured using optical flow technique, and the features are further refined by removing these features with large variation in magnitude and direction of their motion vectors to their neighboring motion vectors.

In our approach, both background removal and foreground segmentation employ optical flow, but optical flow is not exact, which will influence the performance of our framework. The other factor is occlusions in crowd scene. When a number of objects gather together, there will be lots of occlusions in the crowd. This makes it very difficult to estimate abnormal situation, because the texture of foreground becomes chaotic and the edge is hard to find. The noise features often come from calculation errors and are removed after the optical flow method is applied to the refined feature set. In this paper, we applied a 3×3 median filter to the feature set and then used a 3×3 mean filter to smooth the motion field.

Motion smoother is used to smooth out the abrupt motions. The motion analysis method needs to distinguish the noise features from calculation errors. To achieve the goal, we make the following assumptions depending on

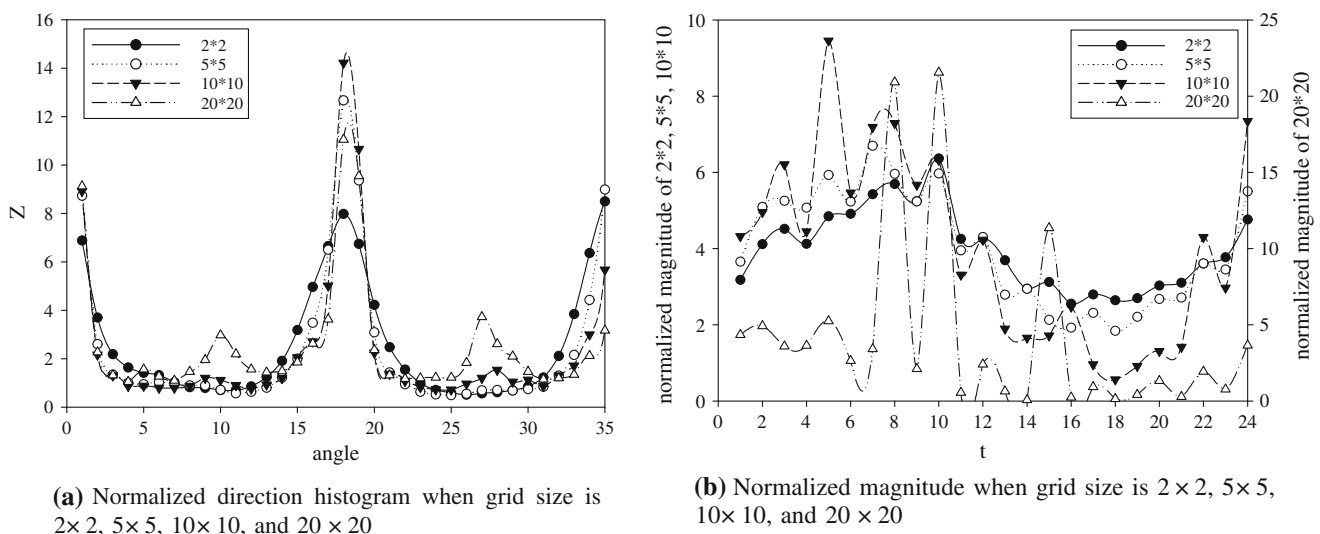


Fig. 4 Normalized direction histogram and magnitude of vectors in an image of crosswalk when grid sizes are 2×2 , 5×5 , 10×10 , and 20×20 (image resolution: 240×180)

assumptions about the nature of motions. In particular, we assume that normal motion is usually smooth with slow variations from frame to frame. On the other hand, noise in the motion involves rapid motion variations over time. In other words, the high-frequency components in the motion vector variations over time are considered to be effects of error motion and therefore need to be removed. Removing the high-frequency components can be done by applying low-pass filters on the motion vectors. We chose to use a moving average (MV) filter. The smoothed motion vectors are obtained by convolving the original motion vectors with the MV filter. Depending on the support of the filter, we only used to store motion vectors for a few frames at any time. For example, we need to store motion vectors for ten frames in VGA resolution image when $N = 10$. Therefore, real-time implementation is achievable because it needs to deal with a few frames.

In order to compute the optical flow between successive video frames in real time, the algorithm of Lucas and Kanade [15] for feature tracking is used. A pyramidal implementation of this algorithm is used to deal with larger feature displacements by avoiding local minima in a coarse to fine approach [6]. This method has proven to allow fast and reliable computation of optical flow information [4, 3]. In the field of optical flow estimation, real-time implementations have been presented, however, were limited to basic variational methods only [7, 8]. In our paper, we combine both variational image resolutions and the speed of numerical multi-grid strategies.

5.2 Grid size determination

The time complexity of methods should be considered to support the real-time application, which is applied numerous times and which, consequently, necessitates a trade-off between accuracy and speed. The complexity of the proposed algorithm is dependent on the one of the environment, which is the feature number. One of the disadvantages of these methods is their very high runtime complexity. Most feature detection methods are time-consuming when the feature number is large which makes the real-time application impossible. Thus, it is necessary to select the appropriate grid size and the regions of interest. Depending on the grid size variant, its parameters, and the image material, this can lead to motion vector fields with a very high precision, but it can also lead to inconsistent fields. If the size of a frame is $I_{\text{width}} \times I_{\text{height}}$, the computational cost for one motion vector field is $O\left(\frac{I_{\text{width}} \times I_{\text{height}}}{k^2}\right)$ (k is the grid size) if no special precautions are taken to reduce them.

In spite of the computational complexity raised by the synthetic vision technique, we demonstrate the ability of our proposed approach to address complex interaction

situations between numerous moving objects. The image sequences and their complexities have a great influence on the optical flow estimation. These complexities are object motion complexity, camera motion complexity, and scene complexity. In addition, the complexity of scenes from an image sequence may vary during the processing. This complexity in an image sequence means textures, shadows, distance of the objects from the camera, the number of the objects in a scene, size of the objects in a scene, velocity of the moving objects or the camera, position of objects toward each other, type of the movement of the objects or the camera.

The grid size determination algorithm is summarized in Algorithm 1. In the initialization step, grid size $G_w^p \times G_h^p$ of a point is determined by width I_{width} and height I_{height} of an image I as well as the ratio of the foreground to the background for frames F_t during training phase as summarized in Algorithm 2. ρ_g is a system parameter related to computational time for making the real-time application possible. The parameter ρ_g is calculated by

$$\rho_g = \frac{T_e - T_s}{N(F_t)} \cdot fps, \quad (16)$$

where T_e and T_s are an end time and a start time in frames F_t in grid g of a point to calculate the actual processing times of a program. Then, the methods called $get_grid_h(G_h^p_temp)$ and $get_grid_w(G_w^p_temp)$ find the grid size in the simulation table. The training algorithm consists in defining the thresholds for abnormal situation detection.

Algorithm 1: The grid size determination algorithm

```

input :  $I, I_{\text{height}}, I_{\text{width}}, G_w^p, G_h^p, F_t, fps, \rho_g$ 
output:  $G_w^p, G_h^p$ 
 $f_b = 0$  /*foreground size*/
 $begin = clock()$ 
for  $i = 1$  to  $n(F_t)$  do
    for  $j = 0$  to  $n(I_{\text{width}})$  do
        for  $k = 0$  to  $n(I_{\text{height}})$  do
            if  $is\_element(I, j, k)$  then
                 $f_b^i = f_b^i + 1$ 
                /*run the training algorithm*/
                 $run\_training()$ 
            end
        end
    end
     $f_r^i = \frac{f_b^i}{I_{\text{width}} \times I_{\text{height}}}$  /*ratio of foreground of background*/
     $f_r^{sum} = f_r^i$ 
     $f_r = \frac{f_r^{sum}}{n(F_t)}$ 
end
/*get processing time*/
 $end = clock()$ 
 $msec\_temp = 1000.0 * (end - begin)$ 
 $msec = msec\_temp / CLOCKS\_PER\_SEC$ 
 $\rho_g = get\_rho(msec, fps, n(F_t), n(I_{\text{width}}), n(I_{\text{height}}))$ 
/*get grid size*/
 $G_w^p\_temp = \lfloor f_r * I_{\text{width}} * \rho_g \rfloor$ 
 $G_w^p = get\_grid\_w(G_w^p\_temp)$ 
 $G_h^p\_temp = \lfloor f_r * I_{\text{height}} * \rho_g \rfloor$ 
 $G_h^p = get\_grid\_h(G_h^p\_temp)$ 

```

Algorithm 2: The training algorithm for normal pattern analysis

input : $I, m_t[]$ /* the initial point */ , $m_g[]$ /* the terminal point */ ,
 $I_{\text{height}}, I_{\text{width}}, F, G_w^p, G_h^p, F_t, B$ /* Block */ , $\epsilon = 1.0e - 6$ /* the
minimum value of the motion magnitude */ , d_{max} /* the maximum
values of the motion magnitude */

output: $E_{\text{th}}, A_{\text{th}}$

$E_t = 0$
 $E_{\text{sum}} = 0$
 $f_b = 0$ /*foreground size*/

for $i = 1$ **to** $n(F_t)$ **do**
 for $j = 1$ **to** $n(U_j)$ **do**
 $m_p[j].x = m_t[j].x - m_g[j].x$
 $m_p[j].y = m_t[j].y - m_g[j].y$
 $\text{medianfilter}(m_p[], 3)$
 $v[j] = m_p[j]$
 $a = \text{atan2}(v[j].y, v[j].x)$
 $d = \sqrt{(v[j].x)^2 + (v[j].y)^2}$
 if $d > \epsilon$ **and** $d \leq d_{\text{max}}$ **then**
 $E_t += d^2$
 $h_d^i[\frac{(a*2\pi)}{(\frac{2\pi}{n(B)})}] +=$
 $E_{\text{sum}} +=$
 end
 end
 for $j = 0$ **to** $n(I_{\text{width}})$ **do**
 for $k = 0$ **to** $n(I_{\text{height}})$ **do**
 if $\text{is_element}(I, j, k)$ **then**
 $f_b +=$
 end
 end
 end
 $f_r = \frac{f_b}{I_{\text{width}} * I_{\text{height}}}$ /*ratio of foreground of background*/
 $E_i = f_r^i * E_t$
 $E_{\text{th}} = \max(E_{\text{th}}, E_i)$
 for $j = 1$ **to** $n(B)$ **do**
 $h_{\text{sum}} += h_d^i[j]$
 end
 for $j = 1$ **to** $n(B)$ **do**
 if $\frac{h_{\text{sum}}^i}{n(B)} \leq h_d^i[j]$ **then**
 $\text{add}(A_{\text{th}}, j)$
 end
 end
end

5.3 Energy-based abnormal situation detection

To recognize abnormal situation in crowded scenes, several methods are developed based on the crowd motion characteristics including the particle energy and the motion directions. The motion variation is derived from the particle energy of two adjacent frames.

The particle energy is modified by each direction of vectors. The directional crowd energy $E_{A_k^{ij}}^i$ of each frame is as follows:

$$E_{A_k^{ij}}^i = r_i \sum_{j=1}^n (d_{k,A_k^{ij}}^{ij})^2, \quad (17)$$

where r_i is the ratio of foreground area to background area, i is a frame number, j is a block number, and k is a motion vector number.

The threshold magnitude $E_{\text{th}, A_k^{ij}}$ of each direction is determined by the maximum value of $\forall E_{A_k^{ij}}^i$ as follows:

$$E_{\text{th}, A_k^{ij}} = \arg \max_{A_k^{ij}} E_{A_k^{ij}}^i. \quad (18)$$

5.4 Dispersion-based abnormal situation detection

Ground truth flow clusters are defined by simple rules with regard to flow direction and location. People in crowds generally seek certain goals and destinations in the environment. Thus, it is reasonable to consider each pedestrian to have a desired direction and velocity. The direction threshold of each divided direction is determined by the maximum value of $\forall nb'(A_k^{ij})$ as follows:

$$A_{\text{th}}^{A_k^{ij}} = \arg \max_{A_k^{ij}} nb'(A_k^{ij}). \quad (19)$$

We also focus on the distribution of the particle in an image. A probability distribution of motion vectors can be estimated by counting the number of times each motion vector value occurs in the image and dividing those numbers by the total number of occurrences. An image consisting of almost a single intensity will have a low entropy value; it contains very little information [26]. A high entropy value will be yielded by an image with more or less equal quantities of many different intensities, which is an image containing a lot of information. We tried to measure dispersion of a probability distribution with a single sharp peak corresponds to a low entropy value, whereas a dispersed distribution yields a high entropy value. In this paper, entropy [12] is defined as follows:

$$H(X) = \sum_{i=1}^n N'(x_i) \log \left(\frac{1}{N'(x_i)} \right) \quad (20)$$

where X is a discrete random variable with possible values $\{x_1, \dots, x_n\}$ and N' is a probability distribution. Similarly, the joint entropy is defined as

$$H(X, Y) = \sum_{i=1}^n \sum_{j=1}^m N'(x_i, y_j) \log \left(\frac{1}{N'(x_i, y_j)} \right), \quad (21)$$

where Y is a discrete random variable with possible values $\{y_1, \dots, y_n\}$.

However, the crowd limits individual movement and the actual motion of pedestrian would differ from the desired velocity. Furthermore, individuals tend to approach their desired velocity based on the personal desire force.

Mutual information measures the mutual dependence of two random variables. Mutual information method is applied to measuring the similarity between two direction histograms derived from three adjacent frames. The definition of mutual information is based on the entropy,

$$I(X, Y) = H(X) + H(Y) - H(X, Y), \quad (22)$$

where $H(X)$ and $H(Y)$ are the Shannon entropy [24] of X and Y , respectively, computed on the probability distribution of the X and Y values.

This form contains the term $H(X, Y)$, which means that maximizing mutual information is related to minimizing joint entropy. High mutual information indicates a large reduction in uncertainty; low mutual information indicates a small reduction; and zero mutual information between two random variables means the variables are independent. Both the modified particle energy and mutual information of direction histograms contribute to the detection of the abnormal events.

In our approach, a direction histogram is built from the motion vectors in a certain frame. The calculated histogram indicates the direction distribution and direction tendency of the frame. In addition, the joint direction histogram is computed using two successive vector fields similarly. The two vector fields derived from three continuous frames. The direction probability distribution for each frame can be directly estimated from the direction histogram by dividing the entries by the total number of the vectors. Furthermore, entropy is used to describe the dispersion of direction probability distribution. A metric for all the video frames is computed using NMI to detect the abnormal event.

Normalized mutual information is expressed by the normalized variant of mutual information to the joint entropy. NMI is defined as follows,

$$\text{NMI}(X, Y) = 1 - \frac{H(X, Y)}{\max(H(X), H(Y))} \quad (23)$$

NMI has a fixed lower bound of 0 and upper bound of 1. It takes the value of 1 when the two clusterings are identical and 0 when the two clusterings are independent.

Consequently, the proposed method not only extracts two sets of feature vectors which can be used to compute the probability of direction distribution, but calculates the mutual information based on the entropies of two probability distributions and the joint entropy.

5.5 Dynamics-based abnormal situation detection

There are various measurements for system disorder such as entropy and standard deviation. Entropy is an important concept in physics and information theory, which is also widely used as a quantitative measurement for uncertainty or unpredictability. Standard deviation is an easy to compute statistical feature describing the variance or diversity of a group of data. Practically, we use NMI and standard deviation to measure the motion disorder, because it leads to a better overall performance while costing much less computing compared with other measurements.

We focus on detecting abnormality in a dense crowded environment and observing their dynamical movement according to their spatio-temporal characteristics. Such tracking system used the concept from dynamical systems called Lagrangian particle dynamics, and idea is taken from computational fluid dynamics (CFD) analysis.

The crowd dynamics is to measure the sudden changes of the motion in a video sequence. The calculation of energy variance is not sufficient for abnormal event detection because it only considers the magnitudes of motion vectors while the directions of motion vectors have been ignored. Thus, we include normalized mutual information of direction histograms in our method. Normalized dynamics D' are calculated by

$$D' = \gamma \cdot E_{A_k}^{ij} \cdot \text{NMI}(X, Y) + (1 - \gamma) \cdot E_{A_k}^{ij} \cdot \sigma(\text{NMI}(X, Y)), \quad (24)$$

where $\gamma \in [0, 1]$ is a fusing parameter, σ denotes the standard deviation. The dynamic abnormality is very important in crowd abnormality detection due to the fact that abnormality is often associated with severely change in the motion state, such as gathering, scattering and chaos situations. In experiments, we will show that our method can detect the abnormal events effectively.

6 Experiments

To evaluate the performance of the proposed approach for real-time abnormal situation detection, we conducted experiments on four different datasets such as UCF [10], UMN [21], prison riot, and open-street CCTV dataset. The test sets are available at [21]. All experiments were run on a computer with 4GB RAM and a 3.2GHz CPU.

Some of the standard measures we used in experiments were misdetection rate, false alarm rate, and receiver operating characteristic (ROC) curves [30] for evaluating abnormal event detection algorithms. To have a more detailed comparison of the different approaches, ROC curves are plotted, which take into consideration both detection rate and false alarm rate for multiple threshold values. This procedure is repeated for multiple thresholds to determine an ROC curve. Finally, the performance is validated by plotting the ROC curves obtained over all possible values of the threshold.

6.1 UFC dataset

In the experiments, we have analyzed the velocity and direction of moving objects in a camera. Video clips were retrieved from the publicly available UCF crowd dataset.

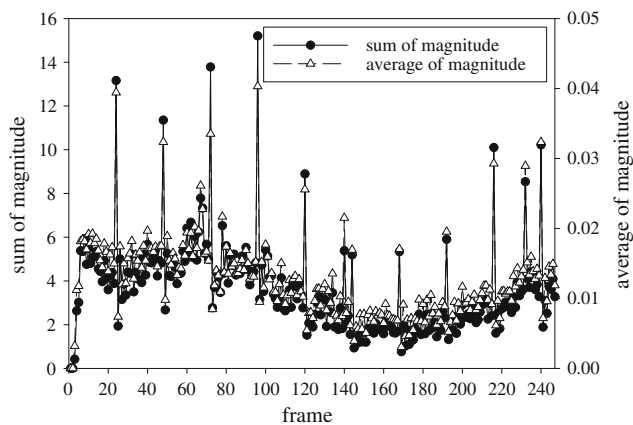


Fig. 5 Speed variance of motion vectors with time t

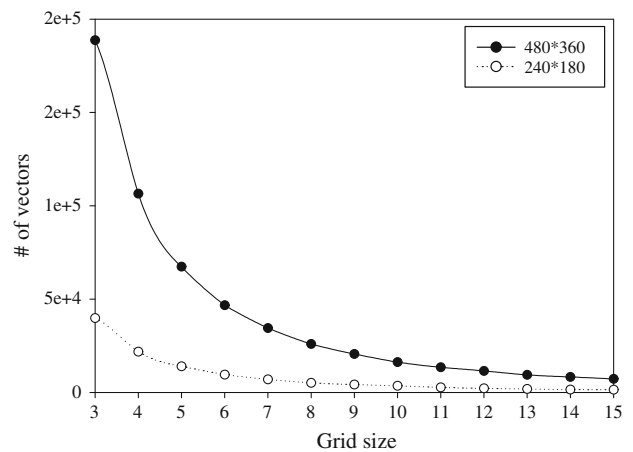
Figure 5 shows the speed variance of motion vectors from frame 1 to frame 247. In the figure, graph shows 12 peaks in frame 24, 48, 72,..., 240, because the video stream obtained from each camera is encoded at a frame rate of 25 fps using standard encoding algorithms such as H.263, MPEG-1 or MPEG-4. To measure the speed of vectors efficiently, these noise errors are removed by simple filters according to encoding rate.

Figure 6 shows the number of vectors variation and the computational time when image resolutions are 240×180 and 480×360 . The test videos were encoded at 25 frames per second. The duration of the test videos is approximately 10 s. To make the real-time application possible, image sequences should be processed over a frame rate of 25 fps. As shown in Fig. 6a, as the grid size increases, the number of vectors has dropped significantly. As shown in Fig. 6b, the grid size should be over 9×9 and 4×4 for real-time process, when image resolutions are 480×360 and 240×180 , respectively.

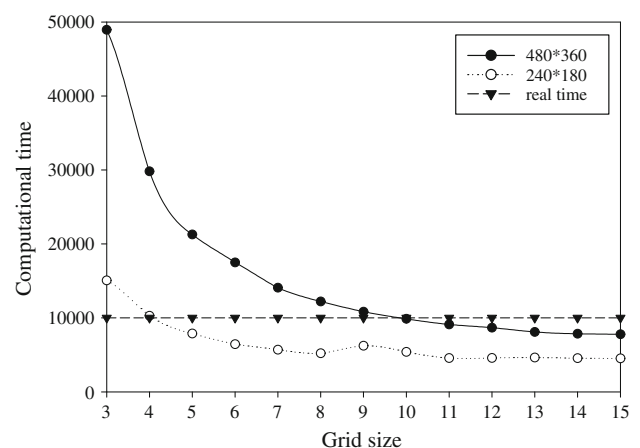
In order to compare various grid sizes, the camera image is divided into 2×2 , 5×5 , 10×10 , 20×20 blocks when maximum number of features is 10^3 . As shown in Fig. 7, the analyzed images contain too many motion vectors when block size is 2×2 . Also, the analyzed images contain little motion vectors when block sizes are 10×10 and 20×20 . In this case, the appropriate block is 5×5 for real-time application.

6.2 UMN dataset

The UMN dataset consists of eleven video sequences acquired in three different crowded scenarios including both indoor and outdoor scenes. In the dataset, pedestrians initially walk randomly and exhibit escape panic by running in different directions in the end. Figure 8 shows the results of the proposed method obtained on the UMN



(a) Number of vectors with respect to the grid size



(b) Computational time depending on the grid size

Fig. 6 The number of vectors variation and computational time when image resolutions are 240×180 and 480×360

dataset. In the figure, the right images of each scene are the detection results of the normal and abnormal situation from the dataset.

Table 1 shows the experimental results for the grid size in terms of the area under the ROC curve (AUC). The AUCs of methods based on energy are 0.9949, 0.9948, 0.9939 when the grid size is 7, 5, 10 in scene 1, scene 2, and scene 3, respectively. When the crowd density is low, the method of the small grid size achieves better performance than that of the large grid size. Scene 1 and scene 2 have high-density crowds. In these case, the small grid size arises false alarms. Consequently, the optimal grid size is determined by the crowd density. In this paper, we achieved better performance based on the grid size determination algorithm.

Figure 9 shows the performance of the proposed method on three different scenes of the UMN dataset. Table 2 compares the performance of our proposed methods in

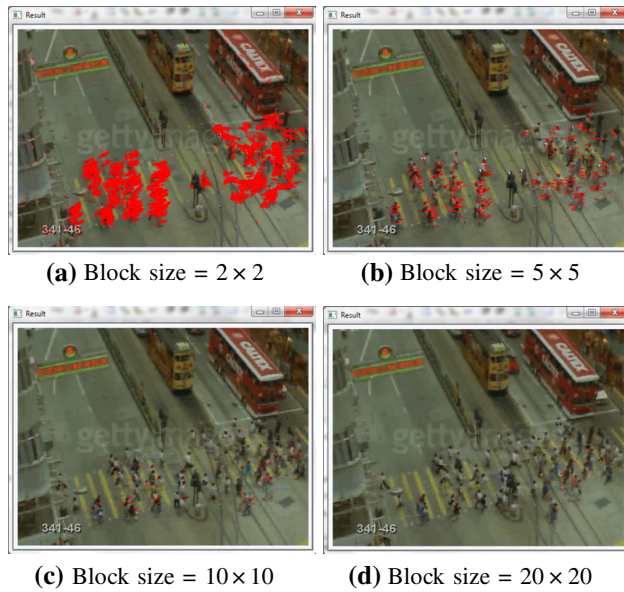


Fig. 7 Illustration of motion vectors depending on block size

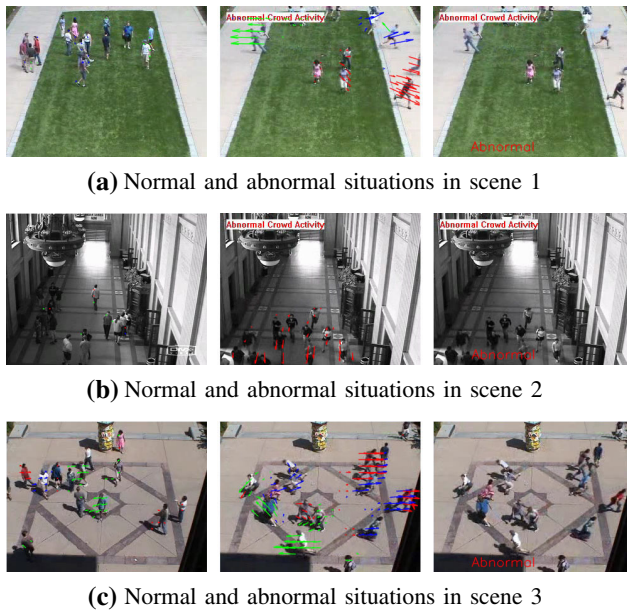


Fig. 8 Results of the proposed scheme on other sequences of the UMN dataset

Table 1 Experimental results of grid size

Grid	Scene		
	Scene 1	Scene 2	Scene 3
5	0.7952	0.9948	0.3497
7	0.9949	0.9897	0.7148
9	0.9748	0.9875	0.9181
10	0.9738	0.9696	0.9939
11	0.9233	0.8619	0.9904
12	0.8934	0.8556	0.9841

Bold values are the highest AUCs corresponding scenes

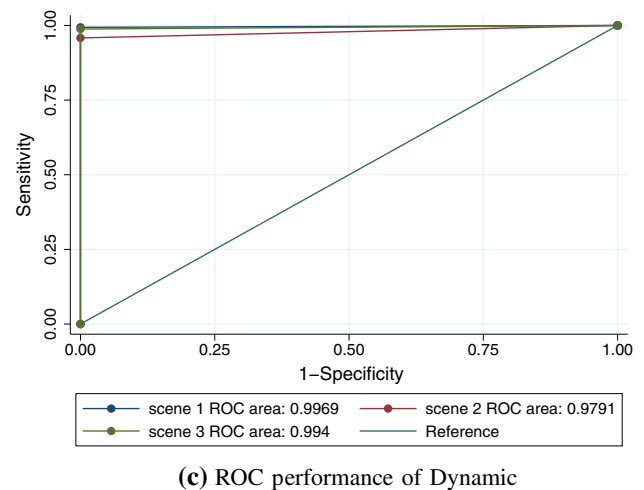
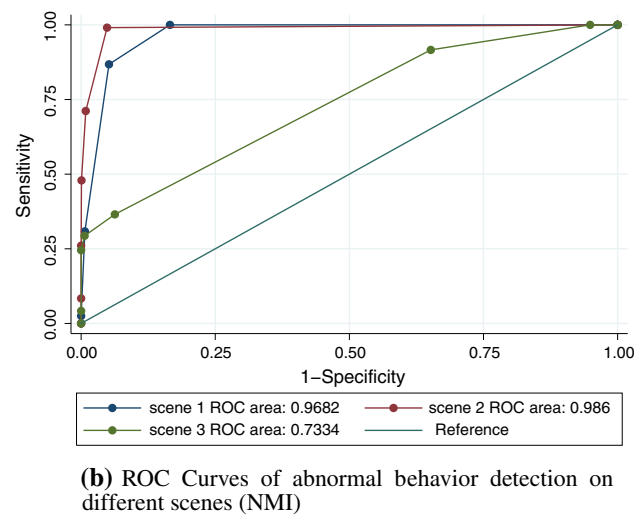
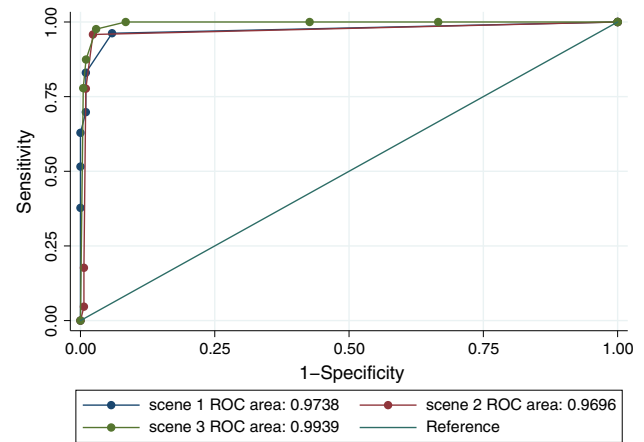


Fig. 9 ROC performance on UMN dataset

terms of the area under the ROC curve (AUC). In case of UMN, the AUCs of methods based on energy, NMI, and dynamics are 0.9791, 0.8958, 0.9901, respectively.

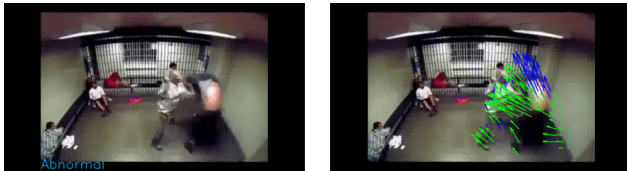
Table 3 provides the quantitative comparisons to the state-of-the-art methods. The AUC of our method is from 0.9791 to 0.9969, which outperforms [19] and is comparable to [32]. However, our method is a more general solution because we did not use any feature extraction methods. Most feature detection methods are time-consuming when the feature number is large which makes the

Table 2 Comparison of experimental results when the grid size is 10

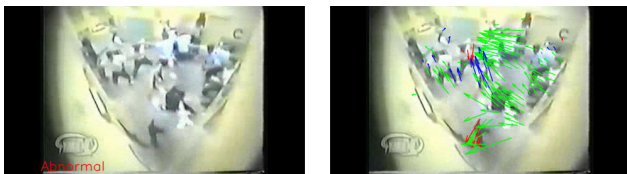
Dataset	Method	Area under ROC
UMN	Energy	0.9791
	NMI	0.8958
	Dynamics	0.9901

Table 3 Comparison of our proposed method with the stat-of-the-art methods in the UMN dataset when the grid size is 10

Method	Area under ROC
Chaotic invariants [32]	0.99
Social force [19]	0.96
Optical flow [19]	0.84
NN	0.93
Proposed method in scene 1	0.9969
Proposed method in scene 2	0.9791
Proposed method in scene 3	0.994
Proposed method in all scenes	0.9901



(a) Abnormal situation in scene 1



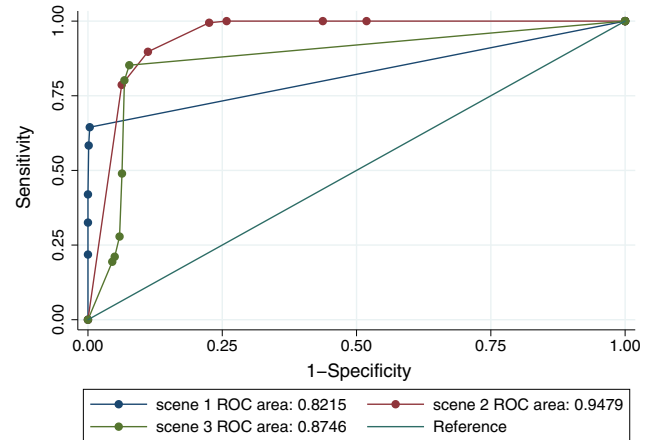
(b) Abnormal situation in scene 2



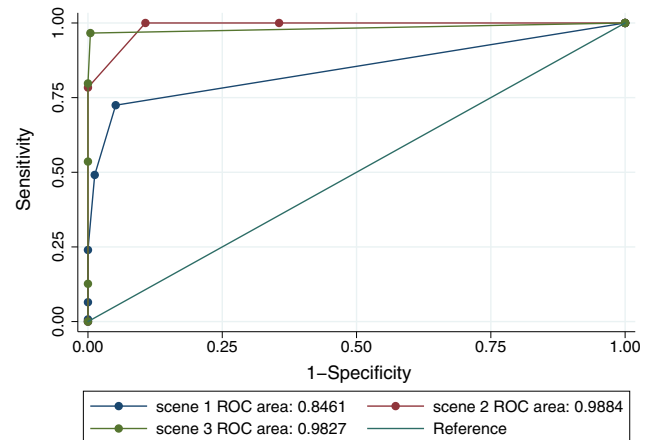
(c) Abnormal situation in scene 3

Fig. 10 Results of the proposed scheme on other sequences of the prison dataset

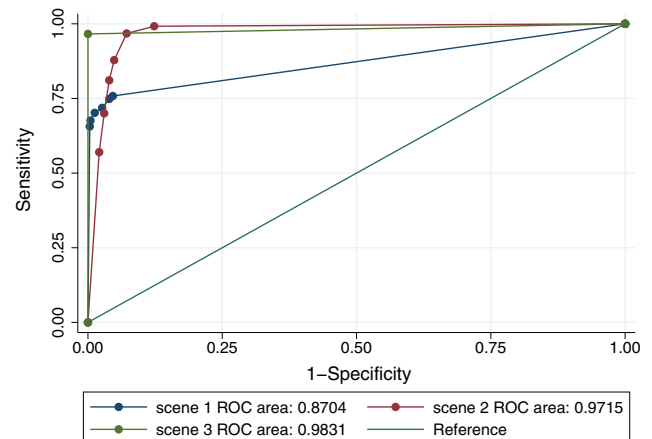
real-time application impossible. In addition, nearest neighbor (NN) method can also be used in high-dimensional space by comparing the distances between the



(a) ROC Curves of abnormal behavior detection on different scenes (Energy)



(b) ROC Curves of abnormal behavior detection on different scenes (NMI)



(c) ROC performance of Energy and NMI

Fig. 11 ROC performance on prison dataset

Table 4 Comparison of experimental results when the grid size is 10

Dataset	Method	Area under ROC
Prison	Energy	0.8813
	NMI	0.9391
	Dynamics	0.9417

testing sample and each training samples. The AUC of NN is 0.93, which is lower than ours. This demonstrates high accuracy of our method over NN method.

6.3 Prison riot dataset

In order to evaluate the proposed method on real applications, we collected a real video set from websites. The collected video dataset comprises of three sequences that represent riots in prisons that are captured with different angles, resolutions, background, and include abnormalities like fist fighting, and clashing. All the collected sequences start with a normal situation, which is then followed by a sequence of abnormal behavioral frames. Figure 10 shows the experimental results force obtained on the frames from this dataset. Figure 11 illustrates the performance of the proposed method on some frames of the different sequences in this dataset.

Table 4 compares the performance of our proposed methods in terms of AUC. In the case of a prison riot dataset, the AUCs of methods based on energy, NMI, and dynamics are 0.8813, 0.9391, 0.9417, respectively.

7 Conclusion

In this paper, we have presented a real-time abnormal situation detection method in crowded scenes using the crowd motion characteristics including the particle energy and the motion directions. The motion variation is derived from the particle energy of two adjacent frames, which is measured by mutual information of the direction histograms of two neighboring motion vector fields.

The direction histogram is built from the motion vectors in frames. The direction probability distribution for each frame is directly estimated from the direction histogram by dividing the entries by the total number of the vectors. Furthermore, entropy is used to describe the dispersion of direction probability distribution. A metric for all the video frames is computed using NMI to detect abnormal situations.

The proposed method not only extracts two sets of feature vectors which are used to compute the probability of direction distribution, but calculates the mutual information based on the entropies of two probability distributions and the joint entropy. In experiments, we expect that

our method can detect an abnormal situation effectively. In the future, we plan to improve the threshold computation by automating the construction of scenario models. Furthermore, we will expand application domains, to include complex domains such as animals and swarming insects.

Acknowledgments They would like to thank Dianne Greco for previewing our manuscript.

References

1. Ali, S., Shah, M.: A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–6 (2007). doi:[10.1109/CVPR.2007.382977](https://doi.org/10.1109/CVPR.2007.382977)
2. Andrade, E.L., Blunsden, S., Fisher, R.B.: Hidden markov models for optical flow analysis in crowds. In: ICPR '06, Proceedings of the 18th International Conference on Pattern Recognition, vol. 01, pp. 460–463. IEEE Computer Society, Washington, DC (2006). doi:[10.1109/ICPR.2006.621](https://doi.org/10.1109/ICPR.2006.621)
3. Baker, S., Matthews, I.: Lucas-kanade 20 years on: a unifying framework. *Int. J. Comput. Vis.* **56**(3), 221–255 (2004)
4. Barron, J., Fleet, D., Beauchemin, S., Burkitt, T.A.: Performance of optical flow techniques. In: Proceedings CVPR '92, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 236–242 (1992). doi:[10.1109/CVPR.1992.223269](https://doi.org/10.1109/CVPR.1992.223269)
5. Black, M.J., Anandan, P.: A framework for the robust estimation of optical flow. In: Proceedings of Fourth International Conference on Computer Vision, pp. 231–236. IEEE press, New York (1993)
6. Bouguet, J.Y.: Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corp.* **5** (2001)
7. Bruhn, A., Weickert, J., Feddern, C., Kohlberger, T., Schnorr, C.: Variational optical flow computation in real time. *IEEE Trans. Image Process.* **14**(5), 608–615 (2005a)
8. Bruhn, A., Weickert, J., Kohlberger, T., Schnörr, C.: Discontinuity-preserving computation of variational optic flow in real-time. In: Scale Space and PDE Methods in Computer Vision, pp. 279–290. Springer, Berlin (2005b)
9. Duong, T., Bui, H., Phung, D., Venkatesh, S.: Activity recognition and abnormality detection with the switching hidden semi-markov model. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR, vol. 1, pp. 838–845 (2005). doi:[10.1109/CVPR.2005.61](https://doi.org/10.1109/CVPR.2005.61)
10. Hu, M., Ali, S., Shah, M.: Detecting global motion patterns in complex videos. In: ICPR 2008, 19th International Conference on Pattern Recognition, pp. 1–5 (2008). doi:[10.1109/ICPR.2008.4760950](https://doi.org/10.1109/ICPR.2008.4760950)
11. Ihaddadene, N., Djeraba, C.: Real-time crowd motion analysis. In: 19th International Conference on Pattern Recognition (ICPR 2008), pp. 1–4, IEEE, Tampa (2008). doi:[10.1109/ICPR.2008.4761041](https://doi.org/10.1109/ICPR.2008.4761041)
12. Jaynes, E.T.: Information theory and statistical mechanics. *Phys. Rev.* **106**(4), 620 (1957)
13. Li, J., Gong, S., Xiang, T.: Global behaviour inference using probabilistic latent semantic analysis. In: BMVC'08, p. 10
14. Li, W., Wu, X., Zhao, H.A.: New techniques of foreground detection, segmentation and density estimation for crowded objects motion analysis. *JIP* **19**, 190–200 (2011)
15. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the

- 7th International Joint Conference on Artificial Intelligence, vol. 2, pp. 674–679. Morgan Kaufmann Publishers Inc., San Francisco (1981). <http://dl.acm.org/citation.cfm?id=1623264.1623280>
16. Ma, R., Li, L., Huang, W., Tian, Q.: On pixel count based crowd density estimation for visual surveillance. In: IEEE Conference on Cybernetics and Intelligent Systems, vol. 1, pp. 170–173 (2004). doi:[10.1109/ICCIS.2004.1460406](https://doi.org/10.1109/ICCIS.2004.1460406)
 17. Marana, A., Velastin, S., Costa, L., Lotufo, R.: Estimation of crowd density using image processing. In: IEE Colloquium on Image Processing for Security Applications (Digest No.: 1997/074), pp. 11/1–11/8 (1997). doi:[10.1049/ic:19970387](https://doi.org/10.1049/ic:19970387)
 18. Marana, A.N., Cavenaghi, M.A., Ulson, R.S., Drumond, F.L.: Real-time crowd density estimation using images. *Adv. Vis. Comput.* **3804**, 355–362 (2005)
 19. Mehran, R., Oyama, A., Shah, M.: Abnormal crowd behavior detection using social force model. In: CVPR 2009, IEEE Conference on Computer Vision and Pattern Recognition, pp. 935–942 (2009). doi:[10.1109/CVPR.2009.5206641](https://doi.org/10.1109/CVPR.2009.5206641)
 20. Nam, Y.: Crowd flux analysis and abnormal event detection in unstructured and structured scenes. *Multimed. Tools Appl.* pp. 1–29 (2013a)
 21. Nam, Y.: Dr. yunyoung nam-google sites (2013b). <https://sites.google.com/site/yynams/crowd-activity>
 22. Nam, Y., Rho, S., Park, J.: Intelligent video surveillance system: 3-tier context-aware surveillance system with metadata. *Multimed. Tools Appl.* **57**, 315–334 (2012). doi:[10.1007/s11042-010-0677-x](https://doi.org/10.1007/s11042-010-0677-x)
 23. Nam, Y., Rho, S., Park, J.: Inference topology of distributed camera networks with multiple cameras. *Multimed. Tools Appl.* **67**(1), 289–309 (2013). doi:[10.1007/s11042-012-0997-0](https://doi.org/10.1007/s11042-012-0997-0)
 24. Papoulis, A., Pillai, S.U.: Probability, random variables, and stochastic processes. Tata McGraw-Hill Education, New Delhi (2002)
 25. Pavón, J., Gómez-Sanz, J., Fernández-Caballero, A., Valencia-Jiménez, J.J.: Development of intelligent multisensor surveillance systems with agents. *Robot Auton. Syst.* **55**(12), 892–903 (2007). doi:[10.1016/j.robot.2007.07.009](https://doi.org/10.1016/j.robot.2007.07.009)
 26. Pluim, J., Maintz, J., Viergever, M.: Mutual-information-based registration of medical images: a survey. *IEEE Trans. Med. Imaging* **22**(8), 986–1004 (2003). doi:[10.1109/TMI.2003.815867](https://doi.org/10.1109/TMI.2003.815867)
 27. Rahmalan, H., Nixon, M., Carter, J.: On crowd density estimation for surveillance. In: The Institution of Engineering and Technology Conference on Crime and Security, pp. 540–545 (2006)
 28. UMN: Umn: Unusual crowd activity dataset of university of minnesota (2013). <http://mha.cs.umn.edu/Movies/Crowd-Activity-All.avi>
 29. Valera, M., Velastin, S.: Intelligent distributed surveillance systems: a review. In: IEE Proceedings on Vision, Image and Signal Processing, vol. **152**(2), pp. 192–204 (2005). doi:[10.1049/ip-vis:20041147](https://doi.org/10.1049/ip-vis:20041147)
 30. Van-Trees, H.: Detection, Estimation, and Modulation Theory. Part I. Wiley, New York (2001)
 31. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**, 539–555 (2009). doi:[10.1109/TPAMI.2008.87](https://doi.org/10.1109/TPAMI.2008.87), <http://dl.acm.org/citation.cfm?id=1512152.1512378>
 32. Wu, S., Moore, B., Shah, M.: Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2054–2060 (2010). doi:[10.1109/CVPR.2010.5539882](https://doi.org/10.1109/CVPR.2010.5539882)
 33. Yang, H., Nam, Y., Cho, W.D., Choi, Y.J.: Adaptive background modeling for effective ghost removal and robust left object detection. In: 2nd International Conference on Information Technology Convergence and Services (ITCS), pp. 1–6 (2010). doi:[10.1109/ITCS.2010.5581283](https://doi.org/10.1109/ITCS.2010.5581283)
 34. Zhang, D., Gatica-Perez, D., Bengio, S., McCowan, I.: Semi-supervised adapted hmms for unusual event detection. In: CVPR 2005. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 611–618 (2005). doi:[10.1109/CVPR.2005.316](https://doi.org/10.1109/CVPR.2005.316)
 35. Zhong, H., Shi, J., Visontai, M.: Detecting unusual activity in video. In: CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. II-819–II-826 (2004). doi:[10.1109/CVPR.2004.1315249](https://doi.org/10.1109/CVPR.2004.1315249)
 36. Zivkovic, Z.: Improved adaptive Gaussian mixture model for background subtraction. In: 17th International Conference on (ICPR'04) Proceedings of the Pattern Recognition, vol. 2, pp. 28–31. IEEE Computer Society, Washington, DC (2004). doi:[10.1109/ICPR.2004.479](https://doi.org/10.1109/ICPR.2004.479)



Yunyoung Nam received the B.S., M.S., and Ph.D. degrees in computer engineering from Ajou University, Korea in 2001, 2003, and 2007 respectively. He was a Senior Researcher in the Center of Excellence in Ubiquitous System (CUS) from 2007 to 2010. He was a Research Professor in Ajou University from 2010 to 2011. He also spent time as a Visiting Scholar at Center of Excellence for Wireless and Information Technology (CEWIT), Stony Brook University, New York. He was a Postdoctoral Fellow at Worcester Polytechnic Institute, Massachusetts from 2013 to 2014. He is currently an assistant professor in the Department of Computer Science and Engineering at Soonchunhyang University. His research interests include multimedia database, ubiquitous computing, image processing, pattern recognition, context-awareness, conflict resolution, wearable computing, intelligent video surveillance, cloud computing, and biomedical signal processing.



Sangjin Hong Sangjin Hong received the B.S and M.S degrees in Electrical Engineering & Computer Science from the University of California, Berkeley. He received his Ph.D in Electrical Engineering & Computer Science from the University of Michigan, Ann Arbor. He is currently with the department of Electrical and Computer Engineering at Stony Brook University - SUNY. Before joining SUNY, he has worked at Ford Aerospace Corp.

Computer Systems Division as a systems engineer. He also worked at Samsung Electronics in Korea as a technical consultant. His current research interests are in the areas of low power system design of multimedia wireless communications and digital signal processing systems, reconfigurable Systems on Chip design and optimization, and VLSI signal processing. Prof. Hong served on numerous Technical Program Committees for IEEE conferences. He is currently serving as an editorial board member in EURASIP Journal of Advances in Signal Processing, Journal of Signal Processing Systems, and Scientific World Journal. Prof. Hong is a Senior Member of IEEE. He is also a member of Eta Kappa Nu and Tau Beta Pi Honor societies.