Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Detecting interaction/complexity within crowd movements using braid entropy*

Murat AKPULAT‡[1], Murat EKİNCİ[2]

[1]*Kelkit Aydın Doğan Vocational School, Gümüşhane University, Gümüşhane 29100, Turkey*

[2]*Department of Computer Engineering, Karadeniz Technical University, Trabzon 61080, Turkey*

E-mail: muratakpulat@gumushane.edu.tr; mekinci@ktu.edu.tr

Received May 20, 2018; Revision accepted Sept. 7, 2018; Crosschecked June 11, 2019

Abstract: The segmentation of moving and non-moving regions in an image within the field of crowd analysis is a crucial process in terms of understanding crowd behavior. In many studies, similar movements were segmented according to the location, adjacency to each other, direction, and average speed. However, these segments may not in turn indicate the same types of behavior in each region. The purpose of this study is to better understand crowd behavior by locally measuring the degree of interaction/complexity within the segment. For this purpose, the flow of motion in the image is primarily represented as a series of trajectories. The image is divided into hexagonal cells and the finite time braid entropy (FTBE) values are calculated according to the different projection angles of each cell. These values depend on the complexity of the spiral structure that the trajectories generated throughout the movement and show the degree of interaction among pedestrians. In this study, behaviors of different complexities determined in segments are pictured as similar movements on the whole. This study has been tested on 49 different video sequences from the UCF and CUHK databases.

Key words: Crowd behavior; Motion segmentation; Motion entropy; Crowd scene analysis; Complexity detection; Braid entropy

https://doi.org/10.1631/FITEE.1800313 **CLC number:** TP391.4

1 Introduction

Crowd behavior analysis has become increasingly important in recent years as a field that researchers in computer vision follow closely. Effective alternative solutions have emerged from this interest. Within this framework, existing studies cover the topics such as segmentation of stable and moving regions, understanding, defining, and classifying the way a crowd creates a motion pattern, and tracking a pedestrian or groups in crowds via video sequences and determining abnormal behaviors. The crowd was divided into moving and non-moving regions in many studies. Objects close to each other moving similarly are clustered in the moving areas, with the final goal of defining and classifying cluster behaviors.

The segmentation of crowd movement has been ongoing for over 10 years. There are literature surveys evaluating the existing methods on the subject (Hu et al., 2004; Yilmaz et al., 2006; Zhan et al., 2008; Junior et al., 2010; Thida et al., 2013; Zitouni et al., 2016). Li et al. (2015) handled the segmentation problem of crowd movements by dividing them into three groups: flow field-based segmentation, similarity-based clustering, and probability model-based clustering. Generally, edge-, graph-, and watershed-based segmentation methods used in the studies are classified as flow field-based

‡ Corresponding author

* Project supported by the Gümüşhane University Scientific Research Projects Coordination Department (No. 15.B0311.02.01)

ORCID: Murat AKPULAT, http://orcid.org/0000-0001-8469-0034

segmentation. Successful clustering studies in high-density crowd movements have been achieved. However, evaluation of long-range and overlapping motion patterns has not been successful.

Zhou et al. (2012a) proposed the coherent neighbor invariance method, in which each motion vector represents a movement in the crowd and it can dynamically create a segment movement based on the proximity-similarity relationship with its k neighbor vectors. Similarly, Hu et al. (2008b) created clusters using the directions of motion vectors and their proximity to each other in motion fields. In another study (Hu et al., 2008a), clustering was based on the dominant motion information in the cluster.

Ali and Shah (2007) transformed the motion fields obtained by the optical flow method into trajectories via the particle advection method, and then employed the finite time Lyapunov exponent (FTLE) method for segmentation to determine the boundaries of the motion fields. Mehran et al. (2010) proposed a dynamic clustering system by applying fluid dynamics methods to trajectories with the developed streakline framework. Wang et al. (2014) produced a streakline framework and combined it with a high-accuracy variational model to cluster crowd movements.

Gao et al. (2017) segmented the moving areas in a crowd according to the streak flow and crowd collectiveness. For the streak flow, the use of motion dynamics in the crowd benefits from three structural features for crowd collectiveness. Lin et al. (2016) tried to identify the semantic regions and recurrent activity movements in crowded movements, which meant that the activity was identifiable with the semantic groups formed from the motion vectors obtained from the optical flow. Chen et al. (2017) tried to group movements in the crowd according to their flow dynamics using the anchor-based manifold ranking (AMR) method. de Almeida et al. (2017) attempted to identify local and general changes in crowd behaviors especially during abnormal events with a method dependent on the histogram values of the motion vectors.

In the studies which are grouped as similarity-based clustering, the motion vectors obtained were transformed into trajectories or tracklets, and clustering operations were performed according to the similarity relations between them. This produces successful results in structured-unstructured crowd scenes. In addition, effective solutions have been presented for problems such as overlapping, scene clutter, and tracking errors.

Cheriyadat and Radke (2008) proposed clustering algorithms based on distance measure. The longest common subsequence (LCSS) values between trajectories were created using motion vectors obtained by the optical flow to detect the dominant motion. Zhao and Medioni (2011) tried to clarify local geometric structures and motion patterns with tensor voting as an unsupervised learning model based on the developed tracklets. Jodoin et al. (2013) highlighted the dominant movement with meta-tracking by first calculating the orientation distribution function (ODF) value of each motion vector. This value maintains the motion vectors according to direction. Then it creates motion trajectories that it expresses as meta-tracks. Finally, the hierarchical clustering method forms the dominant patterns of motion by clustering the meta-tracks that most closely resemble each other.

Fan et al. (2018) aimed to segment coherent and similar relationships in a crowd. In addition, the method could be applied in cases of different scenes and scale scenarios. Wu et al. (2017) aimed to define the pattern of movement of a crowd with the scale and rotation invariant curl and divergence of motion trajectories (CDT) descriptor. In this way, they identified five types of behaviors (lane, clockwise arch, counterclockwise arch, bottleneck, and fountainhead) exhibited by the crowd. Shao et al. (2017) detected and classified groups in the crowd according to collectiveness, stability, uniformity, and conflict, representing the intra-group stability and inter-group conflict of the motion vectors. Fradi et al. (2017) aimed at detecting abnormal and violent incidents and the classification of crowd behavior, and developed a method in which the crowd was represented as a graph and each node in the graph represented a tracklet.

The last category of segmentation studies is probability model-based clustering, including extensive use of the Gaussian mixture model (GMM), random field topic (RFT), and latent Dirichlet allocation (LDA) methods. Through these methods, a number of studies have measured the probabilistic estimates of when, where, and how crowd events occur.

Yang et al. (2009) performed clustering through

diffusion map distribution, which is created by the direction and location values of motion vectors without requiring object detection or tracking operations. Similarly, Saleemi et al. (2010) produced results based on the probabilistic distribution of the position, direction, and magnitude of motion vectors obtained via the optical flow. Zhou et al. (2011, 2012b) sought to solve the problem of detecting semantic areas in unstructured spaces using the RFT method.

Rao et al. (2016) used the new probabilistic-based method and the lengths of the optical flow vectors, and tried to detect such phenomena in the crowd as walking, running, merging, and splitting.

In addition, some studies have modeled the forces in crowd movements with the intention of defining the distribution of the crowd in motion to determine suitable emergency evacuation routes (He et al., 2013). Chen et al. (2009) studied the interaction among pedestrians using a grid-based model. Later, Yuan et al. (2017) introduced a new force occurring among pedestrians at the moment of an opposite direction of flow and subsequently tested models in a simulation environment.

When viewed from a broad perspective, the aim of all these studies was to create segments of crowd movements with a similar acceleration and similar direction, assuming that these segments exhibited a similar pattern in each region. However, the present study aims to closely investigate the behavior of each segment and reveal the behavioral differences within the cluster.

Zhao et al. (2015) aimed to measure the abnormal behavior of the crowd with their proposed entropy model. For this purpose, they used the probability distribution of motion vectors obtained from the optical flow. Similarly, Ali (2013) examined the complexity of crowd movements, and classified different crowd images as irregular and regular crowd behavior movements as low, middle, and high levels according to entropy values calculated by the braid entropy theory. However, comparing the entropy value of crowd movements in different places may not always produce accurate results due to the characteristics of the methods used in the study. In Ali (2013), the motion vectors were obtained by the optical flow method. When acquiring motion knowledge in the optical flow method, the distance of the camera from the moving objects and the angles of perspective are important. For example, a moving object with a camera positioned close to it can possibly result in a high entropy value, but this may not actually reflect complexity. In the same way, the entropy value of the optical flow motion information obtained with a camera looking at the crowd from a distance can be low, which may not indicate that the crowd flows smoothly. Therefore, instead of comparing the entropy values of different locations, the goal was to regionally measure, assess, and compare the entropy values of the same image and to determine how the values change over time.

In many studies, a segment was assigned assuming that the movements close to each other demonstrate the same kind of flow behavior in the same direction, at similar speeds, and to the same locations. However, the crowd movement in these segments may not show the same behavior in each region. To understand the behavior of a crowd, it is important to determine the degree of interaction and complexity in some regions.

The purpose of this study is to better understand crowd behavior by measuring the degree of interaction and complexity among the pedestrians in a crowd. In general, the contributions of this study are as follows: (1) The degree of interaction or complexity is locally determined within a high-density moving group, which proceeds in the same direction as shown in the resulting complexity map, although this was expressed as a cluster in many other studies; (2) When the whole image is taken into consideration, complex-smooth behaviors in moving areas can be classified and a general understanding of the flow of the movement can be reached.

2 Material and methods

The steps in the study are presented in three sections, as shown in the algorithm overview in Fig. 1.

First, motion vectors obtained by the optical

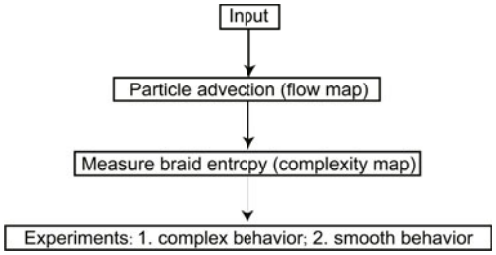

Fig. 1 Steps of the study

flow method are transformed into trajectories via particle advection. The tracks of moving objects in the video sequences are represented as trajectories. A flow map is generated as the input to the system.

The image is then divided into hexagonal cells. A braid is formed with trajectories corresponding to each cell. To create the braid, trajectories are tracked over time according to an angle value. The braid is defined as a spiral pattern occurring when three or more trajectories changed locations with each other over time. Trajectories are given symbolic counterparts for the swaps they made with each other.

The complexity of the braid is calculated depending on its symbolic counterparts. In the next step, the interaction/complexity flow value in each cell of the image, separated by uniform hexagons, is calculated using finite time braid entropy (FTBE). In this way, a complexity map is created to show the crowd behavior.

Eventually, the method is interpreted on the basis of the quantitative and qualitative results obtained by testing a total of 49 different video sequences from the UCF (Ali and Shah, 2007) and CUHK (Zhou et al., 2013) databases.

3 Calculations

3.1 Particle advection

Optical flow is a method used to obtain motion information in an image (Horn and Schunck, 1981; Lucas and Kanade, 1981; Barron et al., 1994). The method is based on the temporal derivation of the x and y axes, depending on the intensity value between consecutive frame pairs of the video sequences. Thus, the motion of the image at the pixel level is obtained as its size and direction. The equations are written according to the amount of change in the axes with the precondition that the light source does not change with time (Eq. (1)), while the general formula of the optical flow method is obtained by Eq. (2).

$$\frac{\mathrm{d}I}{\mathrm{d}t}=1\Rightarrow I(x,y,t)=I(x+\mathrm{d}x,y+\mathrm{d}y,t+\mathrm{d}t), \quad (1)$$

$$Ixu+Iyv=-It. \quad (2)$$

The motion vectors (u,v) in Eq. (2) were transformed into motion trajectories by the particle advection method (Bouguet, 2001; Brox et al., 2004; Ali and Shah, 2007; Mehran et al., 2010). The method was realized by the movement of imaginary particles located on each pixel in the image starting from the very first frame according to the optical flow data obtained throughout the video sequence. Thus, the three-dimensional (3D) motion data (u,v,t) obtained by the optical flow was converted into two-dimensional (2D) observable motion data using Eqs. (3) and (4):

$$x_{i(t+1)}=x_{i(t)}+u[x_{i(t)},y_{i(t)}], \quad (3)$$

$$y_{i(t+1)}=y_{i(t)}+v[x_{i(t)},y_{i(t)}]. \quad (4)$$

Here, $x_{i(t)}$ and $y_{i(t)}$ denote the x and y locations at time t of the i^{th} particle, respectively. $x_{i(t+1)}$ and $y_{i(t+1)}$ denote locations at time $t+1$. To obtain the trajectories, we needed to add the same number of particles ($m\times n=N$) as the number of pixels used in the image up to the end of the video sequence:

$$\mathbf{tjr}_1=\{(x_{11},y_{11}),(x_{12},y_{12}),\ldots,(x_{1T},y_{1T})\}, \quad (5)$$

$$\mathbf{tjr}_2=\{(x_{21},y_{21}),(x_{22},y_{22}),\ldots,(x_{2T},y_{2T})\}, \quad (6)$$

$$\vdots$$

$$\mathbf{tjr}_N=\{(x_{N1},y_{N1}),(x_{N2},y_{N2}),\ldots,(x_{NT},y_{NT})\}, \quad (7)$$

where T denotes the total frame number (Fig. 2a) and $(\mathbf{tjr}_1,\mathbf{tjr}_2,\ldots,\mathbf{tjr}_N)$ indicate trajectories. The flow map **FlowMap** (Eq. (8)), representing the motion flow (shown as blue tracks in Fig. 2b), was used as an input value for the system. Since static fields, noise, and trajectories are not long enough and are not capable of representing the image, particles that moved shorter than the average distance were not included in the calculations in the next stages.

$$\mathbf{FlowMap}_{\mathrm{step}}=\{\mathbf{tjr}_1,\mathbf{tjr}_2,\ldots,\mathbf{tjr}_N\}. \quad (8)$$

In addition, the optical flow method requirement provides more successful results when the speed

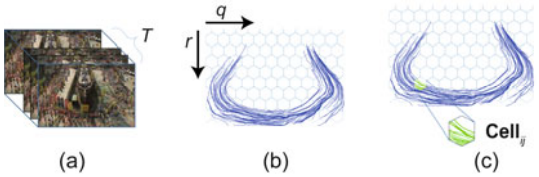

Fig. 2 Dividing the image into a hexagonal grid: (a) totally T frames used as the input; (b) flow map shown as blue lines (q is the number of cells/column and r is the number of cells/row of the hexagonal grid); (c) trajectory fragments corresponding to Cell$_{ij}$

References to color refer to the online version of this figure

of movement is less than one pixel and the movement intensity is higher. In other words, the motion vectors obtained from the high-density flows represent the true motion more successfully.

3.2 Hexagonal grid

As mentioned above, the original value of this work is not in computing the entropy value of a whole frame of motion and comparing it with other different images or subsequent frames, but to handle it by dividing the existing frame into equal parts. At this point, it then becomes possible to separate a motion into semantic regions, to compare the entropy value of each segment to neighboring segments, and to detect how those entropy values change.

For this purpose, the frame was divided into a hexagonal grid (Fig. 2b) and the overall address values of the $q \times r$ cells were determined (Eq. (9)). Each trajectory in the flow map (Eq. (8)) was divided according to its corresponding parts in the cells. For example, the trajectory fragments corresponding to $\mathbf{Cell}_{ij}$ in Fig. 2c are shown in green lines. After this step, the processes took place according to the trajectory parts corresponding to each cell. Algorithm 1 illustrates the division of the trajectories into cells and the calculation with FTBE entropy.

$$\mathbf{Cell}_{ij} = \{(1,1),(1,2),\ldots,(q,r)\}. \tag{9}$$

Algorithm 1 Assigning trajectory parts into cells

Input: Trajectory **FlowMap** in Eq. (8)
Output: Each **Cell** has trajectory parts by position

```
for i = 1 to q (number of columns) do
    for j = 1 to r (number of rows) do
        for k = 1 to N (number of trajectories) do
            if any part of tjr_k in Cell_ij then
                Add part of tjr_k into Cell_ij
            end if
        end for
        Entropy_Cell_ij = FTBE(Cell_ij)
    end for
end for
```

3.3 Measurement of braid entropy

Topological entropy can be defined as a measure of complexity in a dynamic system. By measuring the value of the braid topological entropy, the degree of complexity of the braid can be calculated (Thiffeault, 2010; Budišić and Thiffeault, 2015). In this study, the braid entropy theorem was used to calculate the complexity of crowd flow motion. The "braid" is derived from the braiding of hair. According to the theory, there is a mathematical representation of the helical movements that go into hair braiding. Grouped hair strands form a spiral pattern within a certain order. The displacement of a hair strand by others is mathematically represented in the spiral pattern. Eventually, a spiral braid is expressed by the mathematical representation of consecutive relocations of these strands. The degree of complication in the structure can be determined by starting off with the mathematical representation of the spiral pattern. Braid theory calculates the complexity (entropy value) of this spiral pattern (Allshouse and Thiffeault, 2012; Thiffeault and Budisic, 2014).

The adaptation of this theory to the crowded flow irregularity is as follows: Each trajectory obtained by the particle advection method corresponds to strands in the braid theory. The trajectories create a spiral structure throughout the crowd flow. Therefore, the problem of calculating the complexity of the crowd turns into a problem of calculating the complexity of this spiral pattern throughout the crowd flow. The physical braid can be defined as a pattern, in which three or more trajectories (strands) are intertwined (Fig. 3a). By taking the projection of the trajectory with respect to the x axis (or any axis) and time, a geometric representation of the braid is created in a process called mapping (Fig. 3b).

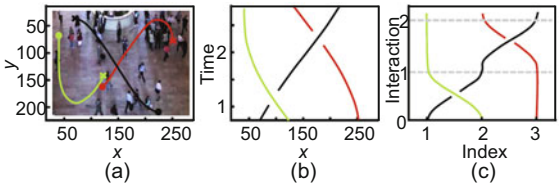

Fig. 3 Three sample trajectories: (a) trajectory starts with the cross point and finishes with dot points in the physical braid; (b) trajectories projected onto the plane containing the x axis and time; (c) the standard braid diagram corresponding to (b) (Allshouse and Thiffeault, 2012)

3.4 Sequence of the generator

In the projection process, the trajectories were sorted by their location relative to the selected axis. The place indices that occurred in the sorting result were determined as the start identity document (ID) of the trajectories. The trajectory displacement

which moves from the front or to the rear of each other is called a "cross." For example, in the algebraic representation in Fig. 3c, the projection of three trajectories with respect to the x axis is taken, and index values are assigned as one to three. In the interval between t and $t+1$, one cross operation is performed. Three values are obtained in each cross process: cross time, cross direction (clockwise or counterclockwise), and cross location (trajectories between which the procedure occurs).

For example, in the spiral pattern shown in Fig. 3c, at time $t=1$, the intersection between the first and second trajectories in clockwise σ_1, and at time $t=2$, the intersection between the second and third trajectories in counterclockwise σ_2^{-1} are symbolically expressed as $\sigma_1\sigma_2^{-1}$. Although the symbol having the smaller index from the trajectories in the cross process determines the number of the symbols, the movement clockwise or counterclockwise determines the sign ($^{-}_{+}1$) of the symbol. At the end of the process, all crossing operations are sorted by time ($\sigma_1\sigma_2^{-1}$), and this sequence is called "generator" (Moussafir, 2006; Allshouse, 2010).

3.5 Length of the loop

Up to this point, the trajectories have been represented by the algebraic notation and the sequence of the generator has been calculated. Following that, these values are used to calculate entropy. This is done with the concept of a "loop" (Fig. 4). The red dots represent trajectories. The trajectories form a loop through the process of displacing each other from the front or back. If the length of the loop in time increases exponentially, this indicates that the crowd shows a complex behavior. Otherwise, the behavior of the crowd is in a straight line. In other words, the increasing number of cycles indicates that the complexity of the motion of the crowd flow has greatly increased. Some features of the braid are illustrated by the vertical green lines shown on the initial coordinate plane (Fig. 4), where the number of intersections that occur above or below the particle with μ_i, and the number of intersections between particles with v_i, are expressed.

The Dynnikov coordinate system expresses the behavior of loops by taking the difference of μ_i and v_i values in the initial coordinate plane. The summary definitions a_i and b_i show the behavior of neighboring

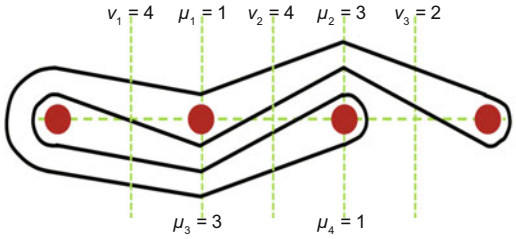

Fig. 4 A sample topological loop drawn for four punctures

References to color refer to the online version of this figure. The μ_i's and v_i's are presented in the plot and the corresponding Dynnikov coordinates are $\boldsymbol{a}=[1,-1]$ and $\boldsymbol{b}=[0,1]$ (Allshouse, 2010; Allshouse and Thiffeault, 2012)

trajectories in the loop:

$$\begin{cases} a_i = \frac{1}{2}\left(\mu_{2i} - \mu_{2i-1}\right), \\ b_i = \frac{1}{2}\left(v_i - v_{i+1}\right), \end{cases} \tag{10}$$

where $i=1,2,\ldots,n-2$.

These values are signed integers and are combined in the Dynnikov coordinate system to represent a loop:

$$\boldsymbol{u} = (a_i, a_{i+1}, \ldots,\ a_{n-2}, b_i, b_{i+1}, \ldots, b_{n-2}). \tag{11}$$

The length of the loop was calculated using these values. Eq. (12) was used for this operation (Moussafir, 2006). The length of the loop is a proportional value, which depends on the intersection of all the trajectories in the cycle with each other (horizontal green line in Fig. 4). Here, a scaler value L_{q} depends on the growth rate of the loop over time.

$$\begin{cases} L_{\mathrm{q}} = |a_1| + |a_{n-2}| + \sum\limits_{i=1}^{n-3} |a_{i+1} - a_i| + \sum\limits_{i=0}^{n-1} |b_i|, \\ b_0 = -\max\limits_{1\le i\le n-2} \left(|a_i| + \max\left(b_i, 0\right) + \sum\limits_{j=1}^{i-1} b_j \right), \\ b_{n-1} = -b_0 - \sum\limits_{i=1}^{n-2} b_i. \end{cases} \tag{12}$$

For a system with n trajectories, if $(n-1)$ loops occurred between neighboring trajectories and $(n-2)\cdot(n-1)$ loops occurred above and below with non-adjacent trajectories and $(n-1)^2$ cycles occurred in total, the generator sequence was then applied. Finally, the time complexity of the algorithm, which included each hexagon, was calculated as $O(n^2)$.

3.6 Finite time braid entropy computation

Finally, as seen in Fig. 5, the FTBE value was calculated by Eq. (13). Trajectories are evaluated within the time interval I ($I = [t_0, t_0 + T]$).

$$\textbf{Entropy} = \frac{1}{T} \log L_{\mathrm{q}}, \tag{13}$$

where T is the time interval between the first cross and the last cross in each cell. FTBE is the complexity value that depends on the cross processing at time interval T.

If the growth rate of L_{q} is low, the loop represents a simple movement. On the other hand, if there is an exponential growth, this indicates a complex movement.

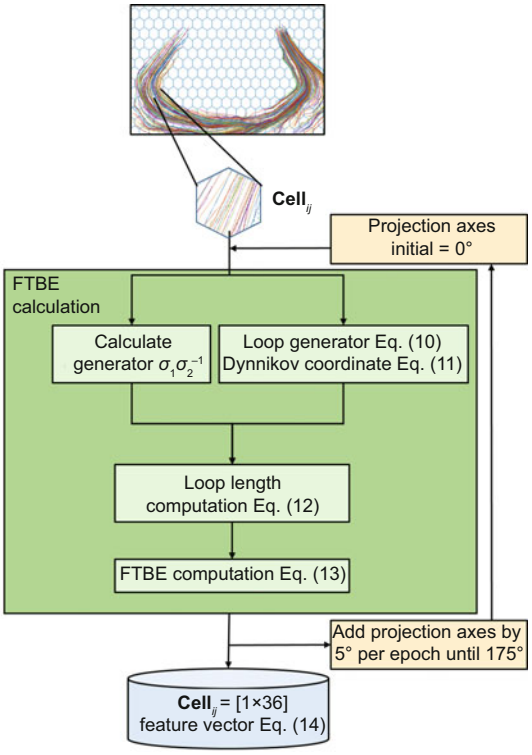

Fig. 5 Calculating the feature vector for each cell using braid entropy

3.7 Projection axes

Trajectories obtained from the movements in the crowd flow should be arranged in the braid theory method. As a prerequisite of the method, the direction of the strands forming the braid should always be forward. When the trajectories shown in Fig. 2b are projected according to any angle, the desired properties of the braid can be obtained. For this study, the projection process was applied to the trajectory groups within each hexagon at an interval of 5° (Fig. 6a). In fact, the interactions among the trajectories were detected from 36 different perspectives. In addition, the cross operations performed throughout the motion produced unique values for the respective projection angles.

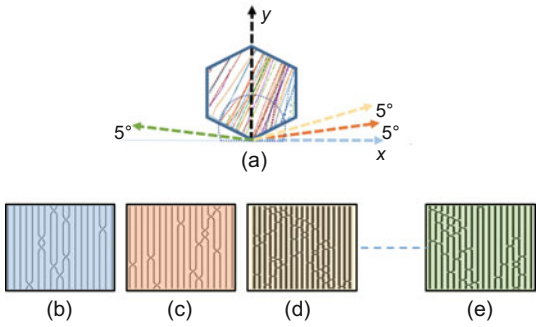

Fig. 6 Projection process differences of 5° with a trajectory corresponding to any hexagon (a), according to the x axis (b), 5° axis (c), 10° axis (d), and a 175° projection process and algebraic braid (e)

References to color refer to the online version of this figure

Moreover, for the step time of each hexagonal cell, 36 different exact braid entropy values were calculated:

$$\textbf{Entropy}_{\textbf{Cell}_{ij}}^{\mathrm{step}} = [b_{i1}, b_{i2}, \ldots, b_{i36}]\,. \tag{14}$$

These values were determined from 36 different angles of the hexagon and the results expressed the complexity of the crowd behavior. The cross operators obtained by the projection on the axes at different angles are shown in Figs. 6b–6e, in accordance with their colors.

At this stage, the fact should again be emphasized that images of crowd flow movements recorded with a single projection can cause a problem in properly detecting the interactions between the pedestrians. In this study, considering the projection problem, two basic steps were used to determine the interactions in the crowd. First, the image was divided into equal regions, so that the focus was on only the behavior in the relevant region of movement. It is expected that the movement in the relevant region will always show the same behavior, especially in a structured crowd flow. A change in this area is believed to indicate some information that could be attributed

to the interaction of the people in the crowd. Second, the interaction was calculated with 36 different angles using the braid entropy method. In this way, it was possible to calculate, independently from the projection and the behaviors in this region alone.

3.8 Complexity map

Finally, the braid entropy feature vector of size 1×36 calculated for $\mathbf{Cell}_{ij}$ in Eq. (14) was calculated for all the cells in the grid, and ComplexityMap was generated at the corresponding step. The complexity map was calculated for all existing step values:

$$\text{ComplexityMap}_{\text{step}} = \left\{\mathbf{Entropy}_{\mathbf{Cell}_{ij}}\right\}, \quad (15)$$

where step $= 1, 2, \ldots, L/K$ with L the total number of frames in the scenario and K the number of frames in one step, and $(i, j) = \{(1, 1), (1, 2), \ldots, (q, r)\}$.

4 Results and discussion

The complexity map we created with this study revealed that, unlike other studies, the flowing motion of a crowd going in the same direction does not show the same behavior in each region. In other words, there were regions with more or fewer complexities in the interactions of the flow of the crowd moving in the same direction. When these regions are detected and viewed as a whole, the behavior of the crowd can be better understood in terms of complexity.

For example, looking at the literature studies in Figs. 7c and 7d, it was supposed that a marathon crowd continued to move in the same pattern in each region. In fact, when the images are examined, it appears that there is a relatively complex movement in which the interaction increases among individuals at the rotation points of motion. These studies in the literature ignored knowledge of complexity in crowd segments since they treated the behavior as a whole.

Similarly, in the studies shown in Figs. 7e and 7f, dots are used to show the movement of the individuals, depending on only the movement information between two frames. However, it was concluded that the 2-frame approach does not contain sufficient information about the complexity of the movement because it shows the movement as a whole and the behavior as the same types.

Fig. 7 shows the regional complexity results. Cells (regions) are colored from green to red (small to

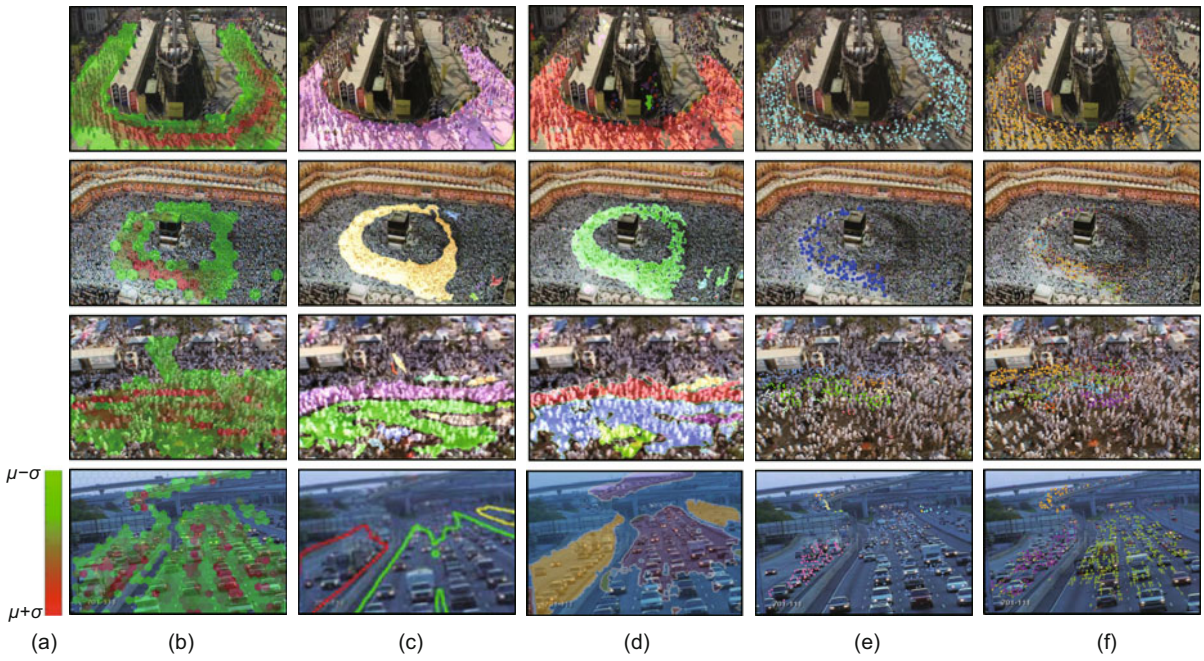

Fig. 7 The present study results compared with those of studies from the literature: (a) color scale; (b) our results; (c) Ali and Shah (2007); (d) Mehran et al. (2010); (e) Zhou et al. (2012a); (f) Zhou et al. (2013)

References to color refer to the online version of this figure

large) according to the average of values in Eq. (14) calculated for each cell. Although the cases where the average value of the cell is low indicate that the movement is stable, flat, and relatively low in density, a high average value shows a relatively high entropy. When the results are examined, it can be seen that the interaction among people increases at the turning points of the crowd, and that the movement is more complex than that in the other regions. The lowest value (shown in green) was determined by subtracting the average value of all cells from the standard deviation, while the maximum value (shown in red) was determined as the sum of these values. Intermediate values are shown according to the corresponding color tone in the scale (Fig. 7a).

In light of these results, we should also mention that there is a difference between braid entropy and the regional density, although there is a correct proportion. The value measured with braid entropy is the value of the complexity of the trajectories due to their "cross" operations during the flow. On the other hand, the density can be thought of as the number of trajectories in a certain region.

In this case, it is highly probable that the entropy values are high in regions where the trajectory density is high because it is likely that there will be an interaction between people in a high-intensity movement. However, entropy values will still be high if there is an interaction between people in areas with a low intensity. Moreover, in cases where there are no interactions between people in high-density regions, the entropy value will be low (if the trajectory goes through the movement without crossing).

In this study, a total of 49 different scenarios were tested, including 19 scenarios from the UCF crowd segmentation dataset (Ali and Shah, 2007) and 30 scenarios from the CUHK crowd dataset (Zhou et al., 2013). The images in UCF were in sizes ranging from 350×230 to 720×480 pixels, and each image in CUHK was 1000×670 pixels in size. Images in both databases have frame numbers ranging from 50 to 800. The results obtained when testing the marathon image in the UCF database are shown in Fig. 8. For this image, the database

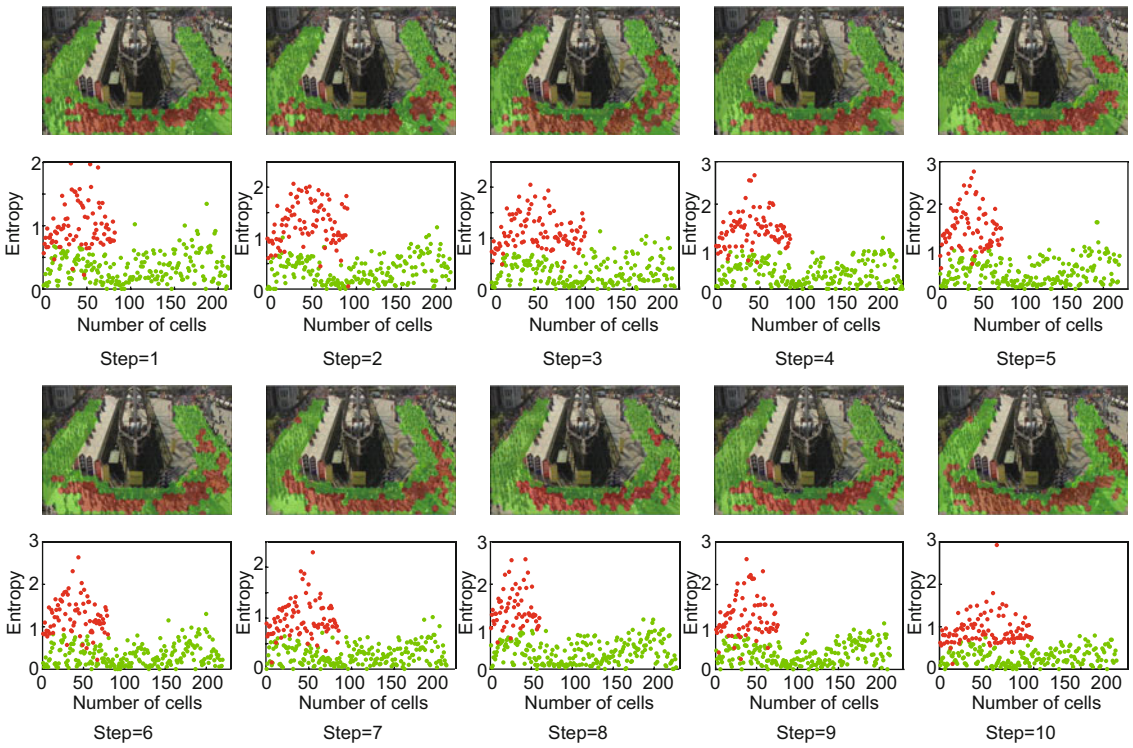

Fig. 8 Results of testing the marathon image in the UCF database

References to color refer to the online version of this figure. For each step moment, calculations were made using 50 frames and the quantitative results are shown on the coordinate plane under the image. The two classes of the Gaussian mixture model (GMM) are colored red (complex) and green (smooth). Each point in the plane represents a cell and is the average of the feature vector values in Eq. (14)

contains $L = 500$ frames ($K = 50$ frames, $h = 20$ pixels). The results from step 1 to step 10 (step $= 1, 2, \ldots, 10$) are shown in Fig. 8. For the step time, the braid entropy value was calculated for each cell by FTBE and a 1×36 feature vector was generated as described in Eq. (14). Finally, the complexity map (Eq. (15)), composed of feature vectors of all cells, could be separated into two regions with GMM clustering: those where the motion complexity or interaction was great and those with regular flow motion. The cells shown in Fig. 8 and colored in red indicate that the complexity or interaction was intense. The green regions indicate a stable, flat, and less complex (smooth) movement. The quantitative values of these regions are shown in Fig. 8 on the coordinate plane under each figure. Each point in the coordinate plane represents a cell. The x axis shows the number of cells having movement inside and joining the processes, while the y axis indicates the mean value of the feature vector of the cell.

In Fig. 9a, all the steps are presented in a single graph as a summary. The red graph shows the average values of the regions where the complexity is greater than the two divided regions, while the green graph shows the regions with a relatively low complexity. This can give an idea as to how the values of the complex and non-complex regions change throughout the movement.

Here, there were two parameters that need to be decided: K (number of frames processed in each step) and h (radius of hexagonal cells). For different K and h values, the results, depending on the average of all the steps of each region, are shown in Figs. 9b and 9c. According to these graphs, the values of $K = 50$ frames and $h = 20$ pixels are acceptable.

In addition, a summary graph was created for 10 different scenarios from the database. The results are shown in Fig. 10 with the results of the complex and non-complex regions of the scene images, where $K = 50$ and $h = 20$. However, the number of steps varied depending on the total number of frames for each view.

As mentioned in the previous sections, the motion of the flow of a crowd is represented by trajectories. When the results in Fig. 10 and the database are examined, such as bottleneck or fountainhead behavior and arch behavior, they form a complex pattern, since the trajectories make an entangled pattern with each other, and the entropy value in these regions seems to be higher than the regions in which a smooth-line behavior is observed.

The results for the 49 different scenarios used in the study are shown in Fig. 11. To present these results in a summary, the average value of the two regions in each video sequence was used. On the x axis, the names of the images used are given, and on the y axis, the average complexity values of the regions are given.

5 Conclusions and future work

We have examined the interactions among people in crowd movements. The motion information in the video sequences was represented by trajectories.

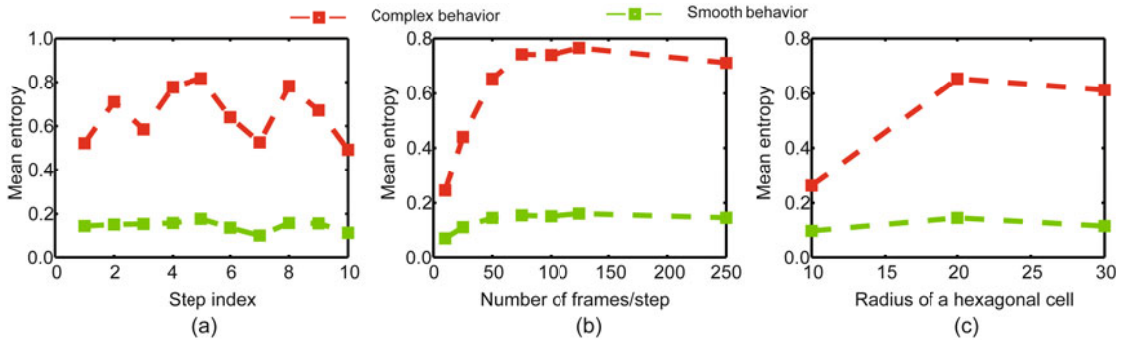

Fig. 9 Results obtained for different frame rates and hexagonal sizes in summary graphs: (a) mean entropy values of the complex and smooth areas in all steps in Fig. 8; (b) average entropy values of the complex and smooth regions changed according to the number of frames (K) used at each step of Fig. 8; (c) average entropy values of the complex and smooth regions changed according to the radius of the cell used in each image

References to color refer to the online version of this figure

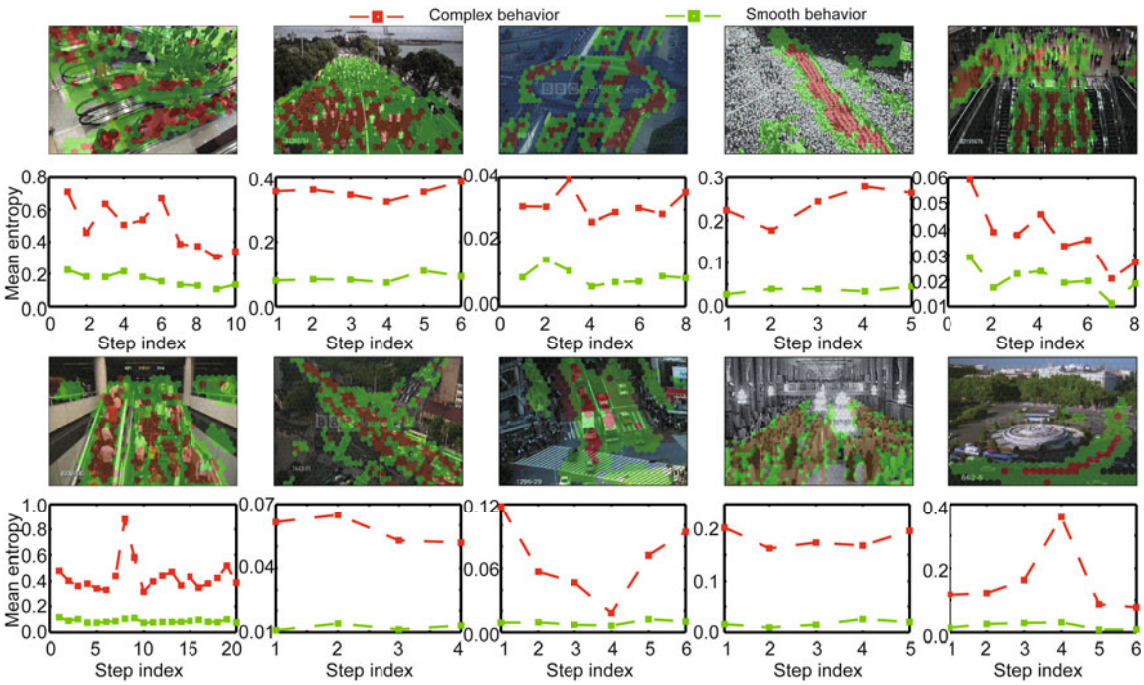

Fig. 10 Summary graph calculated for the marathon scenario in Fig. 9a as shown in different stations from the database

References to color refer to the online version of this figure. Red and green lines represent the complex and smooth behaviors, respectively

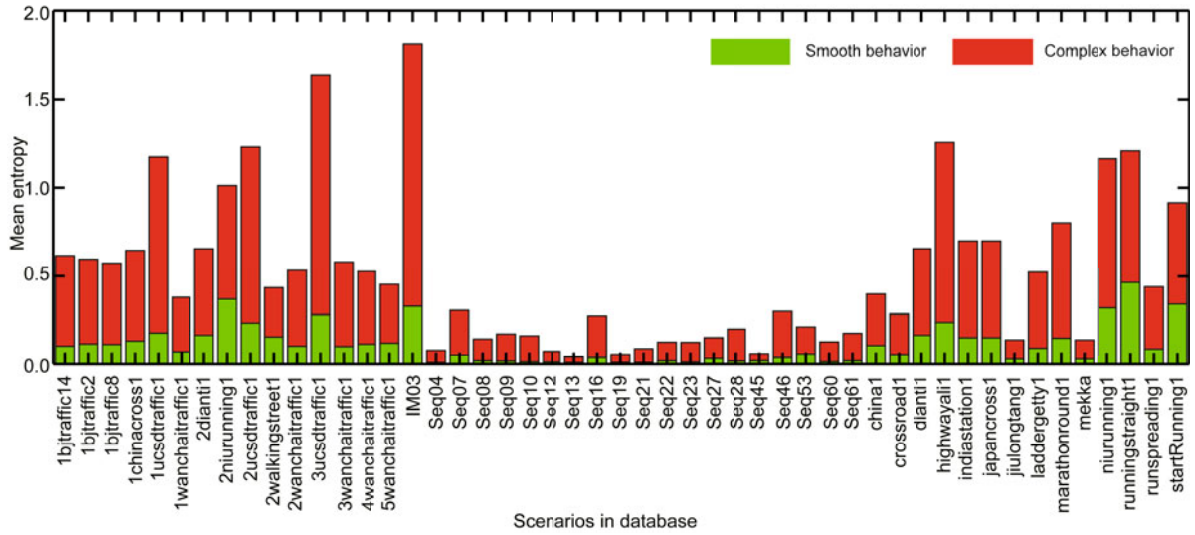

Fig. 11 Results calculated for 49 different scenarios. References to color refer to the online version of this figure

The spiral pattern formed by the trajectories was investigated locally and the degree of complexity in the region was determined using the FTBE value calculated from 36 different directions at an interval of 5°. Ultimately, by interpreting the behavior of the crowd using the generated complexity map, we identified the regions in which the movement was proceeding smoothly and those in which the interactions among the individuals increased. We also demonstrated that braid theory can be a useful method for interpreting trajectories.

In this study, the degree of interaction and complexity were determined locally within a high-density moving group proceeding in the same direction and were shown as the resulting complexity map. This is expressed as a cluster in many studies. When the whole image was taken into consideration, the complex and smooth behaviors in the moving areas can be classified and a general understanding of the flow of the movement can be reached.

In future work, we plan to examine unstructured movements, evaluate the anomalies in results for these environments, and test the feature vectors obtained for each region using learning models. We are also considering a study analyzing the ways in which the braid entropy value is changed by different behavior types such as the bottleneck, fountainhead, and arch.

Compliance with ethics guidelines

Murat AKPULAT and Murat EKİNCİ declare that they have no conflict of interest.

References

Ali S, 2013. Measuring flow complexity in videos. Proc IEEE Int Conf on Computer Vision, p.1097-1104. https://doi.org/10.1109/ICCV.2013.140

Ali S, Shah M, 2007. A Lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.1-6. https://doi.org/10.1109/CVPR.2007.382977

Allshouse MR, 2010. Finding Lagrangian Structures via the Application of Braid Theory. Woods Hole Oceanographic Institute, Massachusetts.

Allshouse MR, Thiffeault JL, 2012. Detecting coherent structures using braids. *Phys D*, 241(2):95-105. https://doi.org/10.1016/j.physd.2011.10.002

Barron JL, Fleet DJ, Beauchemin SS, 1994. Performance of optical flow techniques. *Int J Comput Vis*, 12(1):43-77. https://doi.org/10.1007/BF01420984

Bouguet JY, 2001. Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm. *Intel Co*, 5(1-10):4.

Brox T, Bruhn A, Papenberg N, et al., 2004. High accuracy optical flow estimation based on a theory for warping. European Conf on Computer Vision, p.25-36. https://doi.org/10.1007/978-3-540-24673-2_3

Budišić M, Thiffeault JL, 2015. Finite-time braiding exponents. *Chaos*, 25(8):087407. https://doi.org/10.1063/1.4927438

Chen M, Bärwolff G, Schwandt H, 2009. A derived grid-based model for simulation of pedestrian flow. *J Zhejiang Univ-Sci A*, 10(2):209-220. https://doi.org/10.1631/jzus.A0820049

Chen ML, Wang Q, Li XL, 2017. Anchor-based group detection in crowd scenes. IEEE Int Conf on Acoustics, Speech and Signal Processing, p.1378-1382. https://doi.org/10.1109/ICASSP.2017.7952382

Cheriyadat AM, Radke RJ, 2008. Detecting dominant motions in dense crowds. *IEEE J Sel Top Signal Process*, 2(4):568-581. https://doi.org/10.1109/JSTSP.2008.2001306

de Almeida IR, Cassol VJ, Badler NI, et al., 2017. Detection of global and local motion changes in human crowds. *IEEE Trans Circ Syst Video Technol*, 27(3):603-612. https://doi.org/10.1109/TCSVT.2016.2596199

Fan ZY, Jiang J, Weng SQ, et al., 2018. Adaptive crowd segmentation based on coherent motion detection. *J Signal Process Syst*, 90(12):1651-1666. https://doi.org/10.1007/s11265-017-1309-8

Fradi H, Luvison B, Pham QC, 2017. Crowd behavior analysis using local mid-level visual descriptors. *IEEE Trans Circ Syst Video Technol*, 27(3):589-602. https://doi.org/10.1109/TCSVT.2016.2615443

Gao ML, Wang YT, Jiang J, et al., 2017. Crowd motion segmentation via streak flow and collectiveness. Chinese Automation Congress, p.4067-4070. https://doi.org/10.1109/CAC.2017.8243492

He GQ, Yang Y, Chen ZH, et al., 2013. A review of behavior mechanisms and crowd evacuation animation in emergency exercises. *J Zhejiang Univ-Sci C (Comput & Electron)*, 14(7):477-485. https://doi.org/10.1631/jzus.CIDE1301

Horn BKP, Schunck BG, 1981. Determining optical flow. *Artif Intell*, 17(1-3):185-203. https://doi.org/10.1016/0004-3702(81)90024-2

Hu M, Ali S, Shah M, 2008a. Detecting global motion patterns in complex videos. Proc 19th Int Conf on Pattern Recognition, p.1-5. https://doi.org/10.1109/ICPR.2008.4760950

Hu M, Ali S, Shah M, 2008b. Learning motion patterns in crowded scenes using motion flow field. Proc 19th Int Conf on Pattern Recognition, p.1-5. https://doi.org/10.1109/ICPR.2008.4761183

Hu WM, Tan TN, Wang L, et al., 2004. A survey on visual surveillance of object motion and behaviors. *IEEE Trans Syst Man Cybern Part C*, 34(3):334-352. https://doi.org/10.1109/TSMCC.2004.829274

Jodoin PM, Benezeth Y, Wang Y, 2013. Meta-tracking for video scene understanding. Proc 10th IEEE Int Conf on Advanced Video and Signal Based Surveillance, p.1-6. https://doi.org/10.1109/AVSS.2013.6636607

Junior JCSJ, Musse SR, Jung CR, 2010. Crowd analysis using computer vision techniques. *IEEE Signal Process Mag*, 27(5):66-77. https://doi.org/10.1109/MSP.2010.937394

Li T, Chang H, Wang M, et al., 2015. Crowded scene analysis: a survey. *IEEE Trans Circ Syst Video Technol*, 25(3):367-386. https://doi.org/10.1109/TCSVT.2014.2358029

Lin WY, Mi Y, Wang WY, et al., 2016. A diffusion and clustering-based approach for finding coherent motions and understanding crowd scenes. *IEEE Trans Image Process*, 25(4):1674-1687. https://doi.org/10.1109/TIP.2016.2531281

Lucas BD, Kanade T, 1981. An iterative image registration technique with an application to stereo vision. Proc 7th Int Joint Conf on Artificial Intelligence, p.674-679.

Mehran R, Moore BE, Shah M, 2010. A streakline representation of flow in crowded scenes. Proc 11th European Conf on Computer Vision, p.439-452. https://doi.org/10.1007/978-3-642-15558-1_32

Moussafir JO, 2006. On computing the entropy of braids. *Funct Anal Other Math*, 1(1):37-46. https://doi.org/10.1007/s11853-007-0004-x

Rao AS, Gubbi J, Marusic S, et al., 2016. Crowd event detection on optical flow manifolds. *IEEE Trans Cybern*, 46(7):1524-1537. https://doi.org/10.1109/TCYB.2015.2451136

Saleemi I, Hartung L, Shah M, 2010. Scene understanding by statistical modeling of motion patterns. Proc IEEE Computer Society Conf on Computer Vision and Pattern Recognition, p.2069-2076. https://doi.org/10.1109/CVPR.2010.5539884

Shao J, Loy CC, Wang XG, 2017. Learning scene-independent group descriptors for crowd understanding. *IEEE Trans Circ Syst Video Technol*, 27(6):1290-1303. https://doi.org/10.1109/TCSVT.2016.2539878

Thida M, Yong YL, Climent-Pérez P, et al., 2013. A literature review on video analytics of crowded scenes. In: Atrey PK, Kankanhalli MS, Cavallaro A (Eds.), Intelligent Multimedia Surveillance: Current Trends and Research. Springer Berlin, p.17-36. https://doi.org/10.1007/978-3-642-41512-8_2

Thiffeault JL, 2010. Braids of entangled particle trajectories. *Chaos*, 20(1):017516. https://doi.org/10.1063/1.3262494

Thiffeault JL, Budisic M, 2014. Braidlab: a software package for braids and loops. https://arxiv.org/abs/1410.0849v2

Wang XF, Yang XM, He XH, et al., 2014. A high accuracy flow segmentation method in crowded scenes based on streakline. *Optik*, 125(3):924-929. https://doi.org/10.1016/j.ijleo.2013.07.166

Wu S, Yang H, Zheng SB, et al., 2017. Crowd behavior analysis via curl and divergence of motion trajectories. *Int J Comput Vis*, 123(3):499-519. https://doi.org/10.1007/s11263-017-1005-y

Yang Y, Liu JE, Shah M, 2009. Video scene understanding using multi-scale analysis. Proc IEEE 12th Int Conf on Computer Vision, p.1669-1676. https://doi.org/10.1109/ICCV.2009.5459376

Yilmaz A, Javed O, Shah M, 2006. Object tracking: a survey. *ACM Comput Surv*, 38(4):13. https://doi.org/10.1145/1177352.1177355

Yuan ZL, Jia HF, Liao MJ, et al., 2017. Simulation model of self-organizing pedestrian movement considering following behavior. *Front Inform Technol Electron Eng*, 18(8): 1142-1150. https://doi.org/10.1631/FITEE.1601592

Zhan BB, Monekosso DN, Remagnino P, et al., 2008. Crowd analysis: a survey. *Mach Vis Appl*, 19(5-6):345-357. https://doi.org/10.1007/s00138-008-0132-4

Zhao XM, Medioni G, 2011. Robust unsupervised motion pattern inference from video and applications. Proc Int Conf on Computer Vision, p.715-722. https://doi.org/10.1109/ICCV.2011.6126308

Zhao Y, Yuan MQ, Su GF, et al., 2015. Crowd macro state detection using entropy model. *Phys A*, 431:84-93. https://doi.org/10.1016/j.physa.2015.02.068

Zhou BL, Wang XG, Tang XO, 2011. Random field topic model for semantic region analysis in crowded scenes from tracklets. Proc IEEE Computer Society Conf on Computer Vision and Pattern Recognition, p.3441-3448. https://doi.org/10.1109/CVPR.2011.5995459

Zhou BL, Tang XO, Wang XG, 2012a. Coherent filtering: detecting coherent motions from crowd clutters. European Conf on Computer Vision, p.857-871. https://doi.org/10.1007/978-3-642-33709-3_61

Zhou BL, Wang XG, Tang XO, 2012b. Understanding collective crowd behaviors: learning a mixture model of dynamic pedestrian-agents. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.2871-2878.

Zhou BL, Tang XO, Wang XG, 2013. Measuring crowd collectiveness. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.3049-3056. https://doi.org/10.1109/CVPR.2013.392

Zitouni MS, Bhaskar H, Dias J, et al., 2016. Advances and trends in visual crowd analysis: a systematic survey and evaluation of crowd modelling techniques. *Neurocomputing*, 186:139-159. https://doi.org/10.1016/j.neucom.2015.12.070