Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.039479
Article

Visual Motion Segmentation in Crowd Videos Based on Spatial-Angular Stacked Sparse Autoencoders

Adel Hafeezallah[1], Ahlam Al-Dhamari[2,3,*] and Syed Abd Rahman Abu-Bakar[2]

[1]Department of Electrical Engineering, Taibah University, Madinah, Saudi Arabia
[2]Department of Electronic and Computer Engineering, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Johor Bahru, 81310, Malaysia
[3]Department of Computer Engineering, Hodeidah University, Hodeidah, Yemen
*Corresponding Author: Ahlam Al-Dhamari. Email: kmahlam@utm.my
Received: 31 January 2023; Accepted: 20 March 2023; Published: 26 May 2023

Abstract: Visual motion segmentation (VMS) is an important and key part of many intelligent crowd systems. It can be used to figure out the flow behavior through a crowd and to spot unusual life-threatening incidents like crowd stampedes and crashes, which pose a serious risk to public safety and have resulted in numerous fatalities over the past few decades. Trajectory clustering has become one of the most popular methods in VMS. However, complex data, such as a large number of samples and parameters, makes it difficult for trajectory clustering to work well with accurate motion segmentation results. This study introduces a spatial-angular stacked sparse autoencoder model (SA-SSAE) with *l2*-regularization and softmax, a powerful deep learning method for visual motion segmentation to cluster similar motion patterns that belong to the same cluster. The proposed model can extract meaningful high-level features using only spatial-angular features obtained from refined tracklets (a.k.a 'trajectories'). We adopt *l2*-regularization and sparsity regularization, which can learn sparse representations of features, to guarantee the sparsity of the autoencoders. We employ the softmax layer to map the data points into accurate cluster representations. One of the best advantages of the SA-SSAE framework is it can manage VMS even when individuals move around randomly. This framework helps cluster the motion patterns effectively with higher accuracy. We put forward a new dataset with its manual ground truth, including 21 crowd videos. Experiments conducted on two crowd benchmarks demonstrate that the proposed model can more accurately group trajectories than the traditional clustering approaches used in previous studies. The proposed SA-SSAE framework achieved a 0.11 improvement in accuracy and a 0.13 improvement in the F-measure compared with the best current method using the CUHK dataset.

Keywords: Visual motion segmentation; crowd behavior analysis; trajectory analysis; crowd dynamics; autoencoders; motion patterns

1 Introduction

Crowd behavior analysis (CBA) is one of the most significant and critical topics for ensuring that large events in public areas run smoothly, peacefully, and without casualties. Furthermore, CBA is a multidisciplinary topic that concentrates on a variety of domains, including biology, sociology, and computer vision. Among the important applications of CBA is visual motion segmentation (VMS), which provides a significant amount of information about crowd dynamics in both natural and human communities. Generally, one person's information can only convey a limited amount of local information about the scene. On the other hand, individuals are recognized as union members when crowd motion arises, and they share the same characteristics that are extremely significant for research across many fields. VMS aims to break down a visual image into cohesive subsets corresponding to rigidly and independently moving targets. The pedestrians within each set show collective behaviors and similar motion paths. VMS is an essential preprocessing for a variety of computer vision applications, and it has become a burgeoning study topic in the last few decades. When moving objects are semantically and independently categorized, we obtained motion segments that we can utilize in a variety of video-surveillance tasks, including motion analysis, video indexing, traffic monitoring, activity recognition, crowd counting, crowd tracking, abnormal event detection, disasters recognition, and semantic scene segmentation [1–4].

Even though VMS has progressed considerably [5–10] in the past few years, further improvements are still required to accomplish satisfactory performance. The major challenge in VMS is when the target is too small scaled. Because of the high occlusion in crowd images, state-of-the-art (SOTA) approaches use feature points, which then combined into one group using similar motions to avoid directly detecting pedestrians. Depending on how the objects move in the scene, both structured and unstructured crowd images can be formed. A structured crowd scene has consistent spatiotemporal motion patterns formed by objects that move in concert over the whole scene. Put another way, every spatial location in any structured crowd scene has an identical motion pattern, and the motion's direction does not change most of the time (Fig. 1a). An unstructured crowd scene, on the other hand, consists of non-uniform spatiotemporal motion patterns formed by irregularly moving objects whose movement direction continually changes and cannot be anticipated (Fig. 1b).

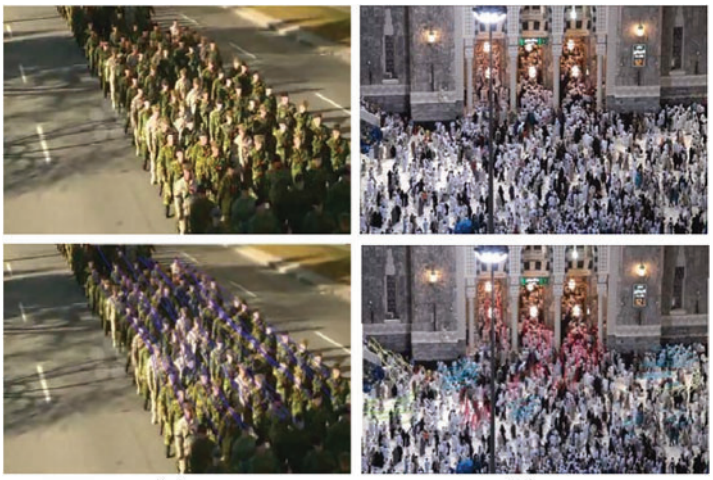


Figure 1: The motion patterns are exhibited as tracklets; every color represents a particular pattern. (a) Structured crowd (b) unstructured crowd

With the evolution of surveillance devices, massive amounts of human trajectory data have been captured, making it vitally difficult and crucial to extract valuable data. A human trajectory is a set of sequenced spatio-temporal data from a single person. Human trajectories provide insight into a variety of real-world applications. An effective way to analyze human trajectories is through trajectory clustering. Trajectory clustering approaches are classified into three groups based on the availability of labeled data: supervised, unsupervised, and semi-supervised. The learning of supervised models occurs before trajectory clustering. Labeled data is typically employed to train a function that creates clusters by mapping data to labels. Then, this function is utilized to predict the clusters of unlabeled data. The objective of unsupervised models is to cluster data without the aid of humans or labeled data. By analyzing unlabeled datasets, an inference function can be built, which can then be used to group data. The former two models are combined through semi-supervised modeling, which is modified using unlabeled data after learning it on labeled data [11]. However, traditional clustering methods would struggle to achieve satisfactory performance if the original data were not evenly distributed owing to high intravariance, as shown on the left side of Fig. 2.

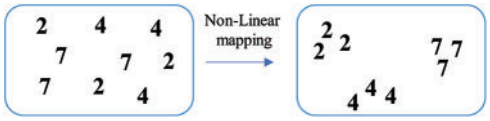


Figure 2: The distribution of the original data is on the left side. Because of the large intravariance, it is hard to separate the data properly. By performing a non-linear transformation function, the points are compacted in a new space with regard to the appropriate cluster centers, as shown on the right side

To address the earlier issue, we aim to map the spatial-angular feature space to a new feature space that is more suited for clustering tasks. The sparse-autoencoder network is a strong contender to solve the above issue. It uses iterative learning to learn both the encoder (EN) and the decoder (DE) to provide a non-linear transformation function. EN is actually the non-linear transformation function, and DE requires reconstructing precise data from the feature representation produced by the EN. Repeating this procedure ensures that the transformation function is reliable and can accurately represent the data. This paper proposes an effective model to cluster human trajectories in different crowd places based on spatial-angular stacked sparse autoencoders (SA-SSAE). When using high-dimensional large-scale datasets, conventional clustering techniques suffer from major performance concerns. For instance, dimensionality reduction techniques must be used prior to using the clustering algorithm to extract features from raw data [12]. Deep learning has always been at the heart of tackling these concerns [13], so we used stacked sparse autoencoders (SSAE) to extract features from the spatio-temporal trajectory data and to group the trajectories together based on their shared characteristics.

The contributions of our work are as follows:

- An efficient framework called SA-SSAE for VMS has been proposed. The SA-SSAE framework is indispensable for high-level crowd behavior analysis. The proposed framework's capacity to handle pedestrian flows that are randomly dispersed is one of its main attractions. To our knowledge, this is the first study that leverages stacked sparse autoencoders for visual motion segmentation in crowd videos using trajectory data employing the Chinese University of Hong Kong (CUHK) crowd benchmark.
- The generalized Kanade-Lucas-Tomasi key point tracker (gKLT) is applied to the input video sequences to extract the trajectories of motion patterns. Individual trajectory formation is a primary concern in the VMS. For studying and analyzing crowded environments, the accurate

extraction of individual trajectories over time is crucial. In this study, the generalized KLT tracker is applied to video sequences to construct individual trajectories.

- Dataset with ground-truth annotations is presented to validate our framework's performance. The Hajj, a significant pilgrimage to Mecca in Saudi Arabia, is one of Islam's five pillars. Every year, up to four million pilgrims carry out the Hajj rituals. As a result, it ranks as one of the most significant pedestrian issues worldwide. Therefore, we collected our Hajj dataset from real-world crowd scenes in Mecca. The dataset includes 21 crowd videos with different scenes and scenarios.
- Our framework is evaluated using two real-world datasets with various crowd densities. On both datasets, we found that our framework can construct high-quality clusters and outperform existing methods quantitatively.

The remainder of the paper is laid out as follows: Section 2 discusses related work. Section 3 presents the proposed spatial-angular stacked sparse autoencoder model for VMS. Comparative results are discussed in Section 4. Concluding remarks are presented in Section 5.

2 Related Work

The various methods for VMS are reviewed in this section. The VMS approaches can be categorized into three main subsets depending on the density of movement flows. Approaches addressing a maximum of five humans are described under the domain of low-level density (LLD). Approaches that address between five and fifteen humans are characterized as mid-level density (MLD). Likewise, approaches that target more than fifteen humans fall under the subset of high-level density (HLD) [14].

LLD Approaches: Nguyen et al. proposed a consensual approach for visual motion segmentation in dynamic views [15]. The model combined unsupervised techniques to address the label correspondence issue. Seyedhosseini et al. put forward a discriminative learning scheme, called CHM, for semantic segmentation. It benefits from contextual data at various resolutions in a hierarchy. The ability of CHM to optimize a posterior probability at various resolutions is its major feature. It effectively and greedily implements this optimization. CHM trains a number of classifiers at various resolutions and uses the gained findings to learn a classifier at the original resolution [16]. An approach for motion segmentation by employing optical flow orientations was proposed by Narayana et al. [17]. The over-segmentation of an image into depth-dependent units was addressed by the utilization of optical flow orientations. Their approach could automatically obtain the quantity of foreground motions. Meunier et al. proposed a CNN-based fully unsupervised approach for VMS from optical flow. Meunier et al. hypothesized that the optical flow input could be expressed as a set of parametric motion models, commonly quadratic or affine. The basic principle of their method is to utilize the Expectation-Maximization to rationally build a loss function and a training procedure for their ground-truth-free neural network for VMS [18]. Choudhury et al. proposed a method for VMS by combining the advantages of appearance-based and motion-based segmentation. They put forward supervising a network of image segmentation with the function of predicting areas that are likely to have simple motion patterns and so are likely to match with targets [19].

MLD Approaches: Mukherjee et al. presented a linear-time video segmentation approach [20] that uses a Gaussian mixture model (GMM) to cluster each video sequence. It also utilizes recursive filtering to re-obtain the parameters of the GMM. In addition to updating the variance iteratively and creating or removing clusters as needed, their hybrid approach can uniquely propagate Gaussian clusters through each subsequent frame. However, a distance threshold parameter is required as its

primary input. A cluster similarity criterion, which may be based on a user-defined distance metric, controls how new clusters are included and removed. A unified conditional random field (CRF model) was proposed by [21] for multiple-targets joint-tracking and segmentation in complex video sequences. The model utilizes low-level image features to associate each super pixel with a particular objective or to designate it as a background.

HLD Approaches: A statistical approach established using the Lagrangian-particle-dynamics [22] for flow segmentation and detection of flow instability was presented in [9]. The flow field created by the crowd motion is handled as an aperiodic dynamical system. To determine the Lagrangian coherent structures existent in the underlying flow, a Finite Time Lyapunov exponent field is built using the greatest eigenvalue of the tensor. In a normalized cuts framework, the Lagrangian coherent structures are used to identify the boundaries of the flow segments by dividing the flow into regions with markedly distinct dynamics. Establishing correspondences between flow segments over time allows for the detection of any alteration in the number of flow segments, called instability. From the point of vision, [6] carefully looked at groups' basic and universal aspects that are present in several crowd environments. These aspects are essential to comprehending congested environments and are driven by sociopsychological investigations. Moreover, a group discovery approach was put forward by learning the collective transition priors.

The VMS problem may alternatively be considered a trajectory clustering (TC) challenge. Determining a proper metric to calculate the similarity of trajectories with different attributes and a proper method to cluster the trajectories on the basis of their commonalities are the two primary issues in TC [23]. Most VMS approaches explored to date either have limited applicability to certain categories of crowd scenes or suffer from major performance concerns. Based on spatial-angular stacked sparse autoencoders, this study provides a model for clustering human trajectories in various crowd environments. The proposed framework is effective and robust for various crowd scene scenarios.

3 Proposed Spatial-Angular Stacked Sparse Autoencoder Model for VMS

The proposed intelligent VMS framework is described in detail in this section. The SSAE network is built based on spatial-angular motion information and is used to find clusters of similar instances in an unlabeled dataset. The framework is illustrated in Fig. 3. Five main steps are involved in constructing the proposed framework: *generating trajectories*, *refining the generated trajectories*, *obtaining spatial-angular motion information*, *applying spatial-angular stacked sparse autoencoders*, and *softmax layer*. SA-SSAE is proposed to automatically segment motion patterns, expressed as trajectories. The SA-SSAE framework employs the generalized KLT tracker (gKLT tracker [24]), which is proposed as an improvement to the standard KLT tracker [25]), owing to its computing efficiency and tracking accuracy. Every generated trajectory is made up of a set of 2D spatial coordinates $\{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\}$, where $L \in [1, k]$ is the trajectory length. The following subsection describes each step where trajectories are refined following Shao et al.'s approach. The motion point features that are made from the new trajectories are then used to make both spatial and angular features. This valuable motion information is then fed into the stack sparse autoencoder network to output motion segments.

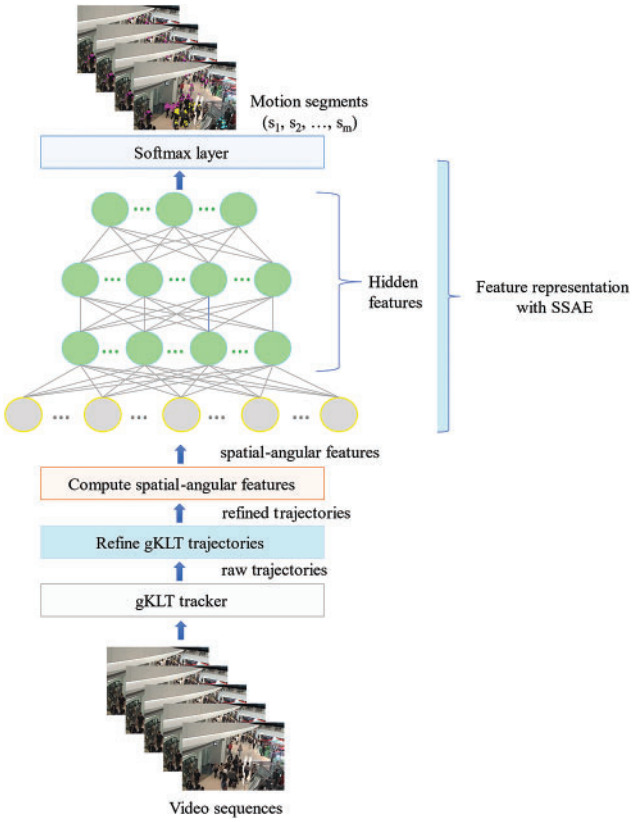


Figure 3: Flowchart of the SA-SSAE framework

3.1 Generating Trajectories

A major challenge in the VMS is the formation of individual trajectories. The precise extraction of individual trajectories over time is critical for investigating and examining crowded scenarios. Following state-of-the-art papers [6,26–28], we utilize the gKLT due to its efficacy in determining trajectories, particularly for small objects in crowd images. The gKLT was first proposed by [25]. First, gKLT obtains the feature points in the moving foreground with enough texture data to detect. The feature points are then tracked, and their velocities are calculated frame by frame based on their displacements. A collection of tracked feature points is thus acquired. The gKLT tracker commences with a group of sparse features in the present frame f_i and seeks to locate their positions in the subsequent frame f_{i+1} by matching a patch of an image surrounding a feature to its identical image patch in the subsequent frame. The assumption of brightness constancy indicates that the intensities of the patches in the subsequent image will not vary significantly. Utilizing a patch enables differentiation between surrounding points of equivalent intensity. The $\omega(x)$ window function, which is commonly a Gaussian function, is employed to highlight nearby pixels more than far-off ones. This explains why points nearer to the feature point are likelier to exhibit comparable motion than those further away. Calculating the error function is the next step in the matching:

$$\varepsilon(d) = \sum_{f(x_0)} [f_{i+1}(x+d) - f_i(x)]^2 \omega\left(\frac{x - x_0}{\eta_w}\right) \tag{1}$$

through the window function's support $\omega(x_0)$, which is positioned over the feature point x_0. The estimated displacement d for the feature point at x_0 in picture f_i is obtained by decreasing the weighted nonlinear least squares formula. The simulation results (Fig. 4) show that the gKLT algorithm effectively tracks feature points that indicate crowd movement.

Figure 4: gKLT algorithm simulation results

3.2 Refining the Generated Trajectories

Through the former processing, a collection of tracked feature points was obtained. Nevertheless, the points gKLT generated do not accurately describe the pedestrians because, in some cases, there could be numerous points inside the same part of a moving person. Moreover, the points can be located in the background or formed owing to changes in illumination, producing noisy, short, and static tracklets. In our framework, following Shao et al. [6], such tracklets are filtered out, which increase VMS's overall performance.

3.3 Obtaining Spatial-Angular Motion Information

The spatial locations of motion features are obtained from the newly refined trajectories. The spatial location features of a trajectory are critical because trajectories that are spaced far apart, even if they are identical in shape, often do not belong to the same cluster. The spatial location features for each trajectory t_i can be calculated using the following equation:

$$sp_{t_i}(v_i) = \frac{1}{k}\sum_{j=1}^{k}(v_i) \tag{2}$$

where $v_i = [x_i, y_i]$, and k is the trajectory length. The $sp_{t_i}(v_i)$ denotes the spatial location feature vector. Angular features describe the direction of the crowd's motion [28]. To compute the average feature, the average displacement $\overline{A}_{t_i}$ of a trajectory is obtained first using Eq. (3) below:

$$\overline{A}_{t_i} = \frac{1}{(n-1)}\sum_{s=2}^{n}\left(\left(x_{s_{t_i}} - x_{(s-1)_{t_i}}\right), \left(y_{s_{t_i}} - y_{(s-1)_{t_i}}\right)\right) \tag{3}$$

The average displacement vector $\overline{A}$ contains two components $\overline{u}$ and $\overline{v}$. Now, using Eq. (3), the average angular can be calculated based on the following equation [28]:

$$\Theta_{t_i} = \begin{cases} cos^{-1}\left(\frac{\overline{A}\cdot\hat{A}}{||\overline{A}||*||\hat{A}||}\right) * \frac{180}{\pi}, & \overline{v} > 0 \\ \left[2\pi - cos^{-1}\left(\frac{\overline{A}\cdot\hat{A}}{||\overline{A}||*||\hat{A}||}\right)\right] * \frac{180}{\pi}, & \overline{u} \neq 0, \overline{v} \leq 0 \\ 0, & \overline{u}, \overline{v} = 0 \end{cases} \tag{4}$$

$\hat{A}$ indicates the horizontal direction's unit vector. The value of $\Theta_{t_i} \in [0, 2\pi - 1]$ differs based on the values of the vectors $\overline{u}$ and $\overline{v}$. Furthermore, a Θ_{t_i} value of zero indicates the absence of motion.

3.4 Spatial-Angular Stacked Sparse Autoencoders

In our sparse-autoencoder with smoothed *l*2-regularization technique, *m* numbers of input spatial-angular training features $\{X_m\}_{m=1}^{M}$ are given such that $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\}$, where $x^{(i)} \in \mathbb{R}$, and $y^i \in \{1, 2, 3, \ldots, C\}$ are the labels. These training samples are fed into the proposed stacked sparse-autoencoders network. The encoder and decoder are two main components of the sparse-autoencoder training process. The encoder maps the input data into the hidden representation while the decoder reconstructs data from the hidden representation. The hidden encoder vector calculated from X_m is denoted by the letter h_m. $\hat{X}_m$ indicates the output layer decoder vector. Therefore, the following is the encoding procedure:

$$h_m = f_e(W_e X_m + b_e) \tag{5}$$

f_e shows the encoding function, W_e is the encoding weight parameter, and b_e represents the corresponding bias. The following is a description of the decoder process:

$$\hat{X}_m = f_d\,(W_d h_m + b_d) \tag{6}$$

f_d represents the decoding function. W_d and b_d are the decoding weight and bias, respectively.

The sparse-autoencoder model minimizes the reconstruction error to learn a meaningful hidden representation. Thus, to diminish the reconstruction error and resolve the parameters W_e, W_d, b_e, and b_d, the sparse-autoencoder's parameter settings are optimized as follows:

$$\phi = argmin\frac{1}{m}\sum_{k=1}^{m} L\left(X_k, \widehat{X}_k\right) \tag{7}$$

$L\left(X_k, \widehat{X}_k\right)$ represents the loss function, where $L\left(X_k, \widehat{X}_k\right) = \left|\left|X - \hat{X}\right|\right|$. Fig. 5 presents the SSAE architecture. SSAE in our framework is built by stacking *two* sparse-autoencoders into *m* hidden layers using an unsupervised learning method called "*layerwise learning*", which is subsequently fine-tuned utilizing a supervised method. By including a regularizer in the cost-function, it is feasible to promote an autoencoder's sparsity. This regularizer is based on a neuron's average output activation value, described as follows:

$$\rho_i = \frac{1}{k}\sum_{i=1}^{k} h_n^{(1)}\left(x_j\right) = \frac{1}{k}\sum_{i=1}^{k} f_e\left(w_n^{(1)T} x_i + b_n^{(1)}\right) \tag{8}$$

where k represents the overall quantity of the training patterns. x_i is the i^{th} training pattern, $w_n^{(1)T}$ is the n^{th} row of the weight matrix W^1, and $b_n^{(1)}$ is the n^{th} element of the bias vector $b^{(1)}$. The sparse-autoencoder uses back-propagation (bp) to lessen the cost-function, and it can be computed as follows:

$$J_{sparse} = \left[\frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\left|\left|h_{W,b}\left(x_i - \hat{x}_i\right)\right|\right|^2\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_1}\sum_{j=1}^{s_l+1}(W_{ji}^{(l)})^2 + \upsilon\sum_{j=1}^{n} S\left(a_j\right) \tag{9}$$

where the first expression of Eq. (9) presents the average sum-of-squares error, λ is a variable to control the relative weight of the regularization. Incorporating the *l*2-weight regularizer makes the solution "smoother" and improves its generalization ability. υ is the sparsity penalty term's weight. The λ and υ can be specified while training the sparse-autoencoder. $S(\cdot)$ denotes the sparsity regularizer that regulates the sparsity of the hidden layer output. S has a low value for every neuron "specializing"

in the hidden layer, it produces a high output for a few training samples. On account of this, a lower sparsity fraction promotes a high level of sparsity. a_j presents the average output of the j[th]-hidden-unit and can be obtained using Eq. (10) below:

$$a_j = \frac{1}{k}\sum_{i=1}^{k} u_j^{(i)} \tag{10}$$

where u is the j^{th}-hidden-unit-output of an i^{th}-training pattern. The parameters $l2$-weight regularize, and sparsity regularize are utilized to prevent overfitting. The stack sparse-autoencoders are linked to a softmax layer that predicts the probabilistic assignments of clusters. The softmax layer's mathematical model is as follows [29]:

$$O_\theta\left(x^i\right) = \begin{bmatrix} p(y^i = 1|x^i;\theta) \\ p(y^i = 2|x^i;\theta) \\ p(y^i = 3|x^i;\theta) \\ \cdot \\ \cdot \\ \cdot \\ p(y^i = r|x^i;\theta) \end{bmatrix} = \frac{1}{\sum_{i=1}^{r} e^{\theta_j^T x^i}} \begin{bmatrix} e^{\theta_1^T x^i} \\ e^{\theta_2^T x^i} \\ e^{\theta_3^T x^i} \\ \cdot \\ \cdot \\ \cdot \\ e^{\theta_r^T x^i} \end{bmatrix} \tag{11}$$

where $\{\theta_1, \theta_2, \ldots, \theta_r\}$ represents the model parameters and the expression $1/\sum_{i=1}^{r} e^{\theta_j^T x^i}$ normalizes the distribution to guarantee that the sum is equal to one.

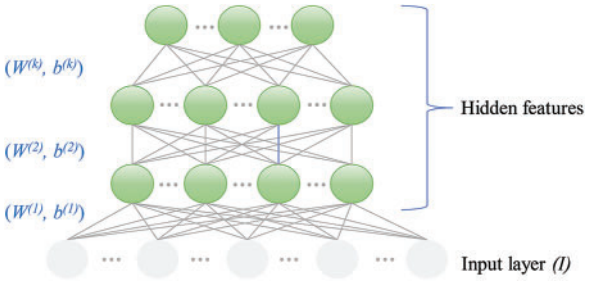


Figure 5: Stacked sparse autoencoder architecture

4 Experiments

In the dense crowd scenes, individuals are a gathering of many groups with similar motion characteristics. Groups are essential components of a crowd. A novel crowd segmentation framework based on spatial-angular stacked sparse autoencoders was presented in this paper to obtain fundamental crowd interactions for subsequent crowd behavior analysis. The proposed framework can be applied to various dense crowd scenes. In this section, we extensively evaluate the performance of the proposed framework for VMS on two crowd video datasets: the Hajj and the CUHK crowd datasets.

4.1 Motion Segmentation Benchmarks

Hajj Benchmark: The Hajj benchmark consists of crowd videos shot in various indoor and outdoor locations in Mecca. Islam's Hajj, a major pilgrimage to Mecca in Saudi Arabia, is one of the five pillars of Islam. The Hajj rituals are performed annually by up to four million pilgrims. Consequently, it is one

of the most significant global pedestrian issues. The benchmark has 21 real-world crowd videos with various scenarios and situations. It has several challenges, including occlusions, lighting, and various object scales. The Hajj benchmark videos contain six scenes, as shown in Fig. 6.

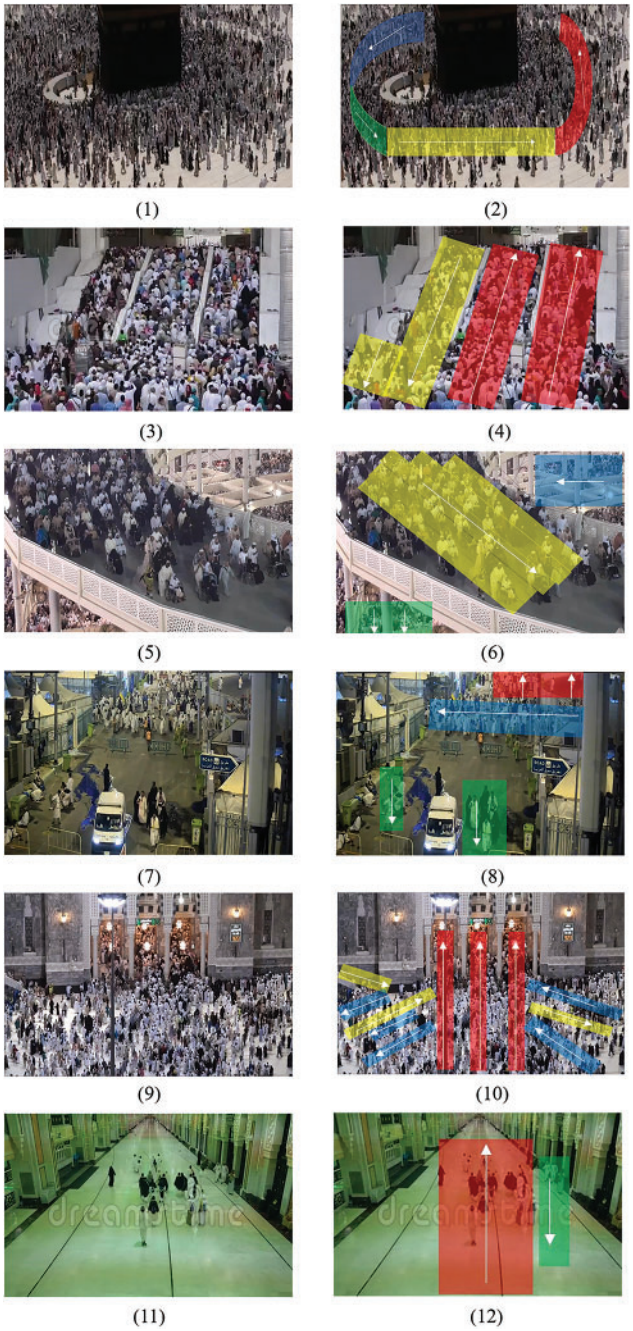


Figure 6: A few examples of Hajj benchmark frames. The original frames of crowded indoor and outdoor scenes are on the right, and the motion patterns are on the left

CUHK Benchmark: The CUHK is a set of crowd videos that includes 55 videos captured by Shao et al. [6], as well as former crowd benchmarks that have already been released from the following SOTA studies [9,24,30,31]. The CUHK collection has 300 annotated crowd clips with different crowd densities, and the benchmark has the groups for each clip's motion pattern segmentation. These 300 video clips are used to validate the proposed model following the SOTA approaches in [6,14,28]. Based on the crowd dynamics, these clips are grouped into structured and unstructured groups. Additionally, each scene is grouped into indoor and outdoor subcategories based on the location of the recording and the nature of its content. Fig. 7 shows examples of crowd images from the CUHK benchmark.

Figure 7: CUHK benchmark sample frames

4.2 Evaluation Methods and Results

VMS is evaluated as a clustering issue, and the performance evaluation can be achieved by utilizing widely well-known cluster assessment measures: Purity, Rand Index (RI), Normalized Mutual Information (NMI), accuracy, and F-measure [14,32,33]. A larger value denotes better clustering performance; all performance measurements lie within the range [0, 1]. The values of the training parameters for the sparse autoencoders are listed in Table 1. Purelin function and logistic sigmoid function, respectively, serve as the transfer functions for the encoder and decoder. A linear transfer function known as the Purelin function is defined as follows:

$$f(z) = z \tag{12}$$

The Logistic sigmoid function is given as:

$$f(z) = \frac{1}{1 + e^{-z}} \tag{13}$$

Moreover, scaled conjugate gradient descent (SCGD) was employed for the autoencoder training process to optimize the weights and bias by minimizing $J_{sparse}(W, b)$ in Eq. (9). The encoder attempts to encode the considerable input features into a smaller hidden representation with only the related information. The decoder turns the process back to reproduce the same set of features as the input. After training, weights will be assigned to every neuron in the hidden layer, enabling them to provide an

efficient stimulus against input visual information. The feature set produced from the first sparse AE's weights is utilized to train the second sparse AE. The first sparse AE's feature t-Distributed Stochastic Neighbor Embedding (t-SNE) plot is shown in Fig. 8a. The t-SNE demonstrates very clearly that the feature set exhibits good discriminatory behavior. The size of the hidden layers for the first sparse AE is 30, and the size of the hidden layers for the second sparse AE is 10, leading to a smaller representation of the features. A significant difference that may be seen in the t-SNE plots in Figs. 8a and 8b is that the features are compressed about the appropriate cluster centers in the new feature space. Fig. 9 presents some visual motion segmentation results of real-life crowd scenes from the CUHK and Hajj crowd benchmarks. Different colors stand for different motion segments. It is clear that the proposed SA-SSAE framework works very well compared to the ground truth segmentation groups.

Table 1: Parameters of SA-SSAE model

No	Parameter	Value
1	Max epochs	400
2	*l2* weight regularization	0.1
3	Sparsity regularization	2
4	Sparsity proportion	0.01
5	Scale data	False

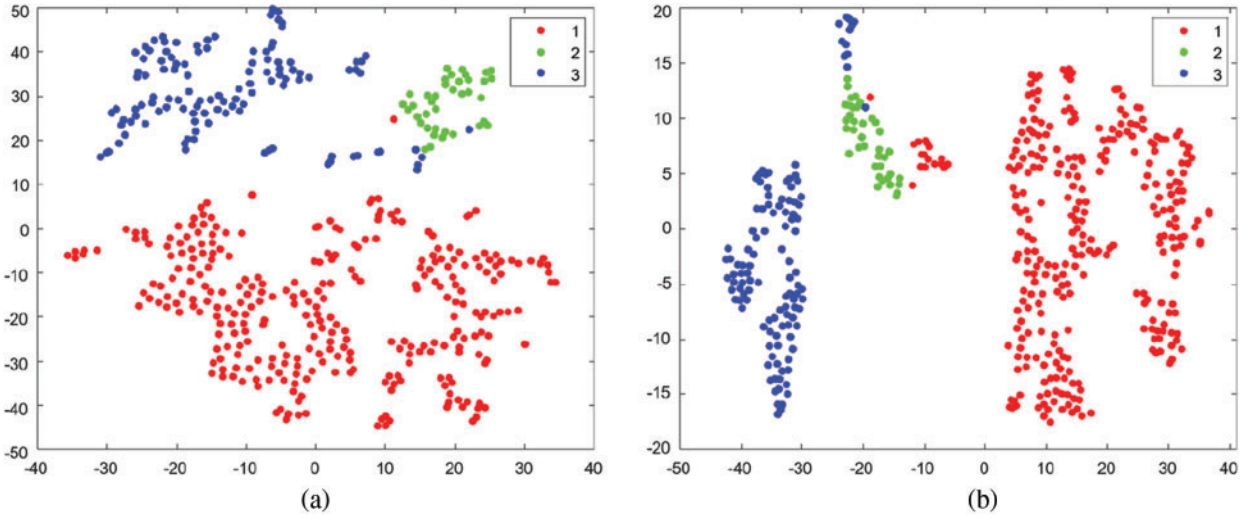


Figure 8: The t-SNE plot of features produced from (a) the first sparse AE, (b) the second sparse AE, using the "exact" method, which maximizes the Kullback-Leibler divergence (KLD) of distributions between the original data space and the embedded data space

Figure 9: Visual motion segmentation results based on the proposed framework

4.2.1 Results on Hajj Crowd Benchmark

Table 2 compares the results and shows the average performance for 21 crowd videos of the Hajj benchmark. It indicates that the proposed model surpasses the recent SADC approach for VMS by a large margin in terms of purity, NMI, and RI. Upon closer examination, we discovered that the NMI-performance-metric always produces a value of zero when one of the two grouping assignments (ground truth or clustering outcome) has only one cluster and the other clustering has multiple clusters. This is not permitted when calculating the NMI. This is because of an underlying mathematical problem with how mutual information is computed. However, such a debate is outside the purview of this study.

Table 2: Results based on Hajj benchmark

Method	Purity	NMI	RI
SADC [28]	0.85	0.21	0.74
SA-SSAE	**0.93**	**0.70**	**0.91**

4.2.2 Results on CUHK Crowd Benchmark

To demonstrate the efficacy of the proposed model, its performance results are compared to the following SOTA methods: MCC [24], CT [6], and SADC [28]. As shown in Table 3, the proposed SA-SSAE model outperforms all other approaches. As mentioned previously, the NMI will equal zero if one of the two clustering assignments has only one cluster and the other clustering has more than one.

With 25% of the overall videos in the CUHK benchmark having one group ground-truth, this explains why other approaches give smaller values for NMI.

Table 3: SA-SSAE CHUK

Method	Purity	NMI	RI
MCC [24]	0.69	0.43	0.70
CT [6]	0.76	0.41	0.73
SADC [28]	0.93	0.78	0.89
SA-SSAE [ours]	**0.96**	**0.84**	**0.95**

Table 4 presents a comparative assessment in terms of the Acc and F-measure using nine relevant approaches HC [34], CF [35], CT [10], CDC [26], MCC [24], AMR [36], HSIM [14], MPF-*l1* [33], and MPF-*l2* [33]. Additionally, Fig. 10 illustrates the relative improvement of the SA-SSAE model over the other approaches. The proposed model gets the highest Acc and F-measure values, which shows that it is better at getting motion pattern segments. It is well known that performance decreases when pedestrian distribution varies. All SOTA approaches [10,24,26,33–36] disregard the changes in the distribution of individuals, which only remains true if the motion flow is coherent across time. Such variations generate isolated areas, which in turn, lower overall performance. Furthermore, the SOTA approaches [10,24,26,33–36] cannot group structurally comparable pixels into significative segments. Discovering and segregating isolated pedestrian segments presents a highly complicated issue. Even though the HSIM approach is emphasized for its ability to handle randomly distributed pedestrian flows, our SA-SSAE model achieves better results regarding the F-measure. The relative improvement of the SA-SSAE framework compared to the HSIM approach is 0.36.

Table 4: Comparative assessment in terms of the Acc and F-measure employing CUHK benchmark

Method	Acc	F-measure
HC [34]	0.63	0.62
CF [35]	0.70	0.67
CT [10]	0.75	0.74
CDC [26]	0.67	0.67
MCC [24]	0.68	0.67
AMR [36]	0.78	0.76
HSIM [14]	-	0.58
MPF-*l1* [33]	0.83	0.80
MPF-*l2* [33]	0.80	0.79
SA-SSAE [ours]	**0.94**	**0.93**

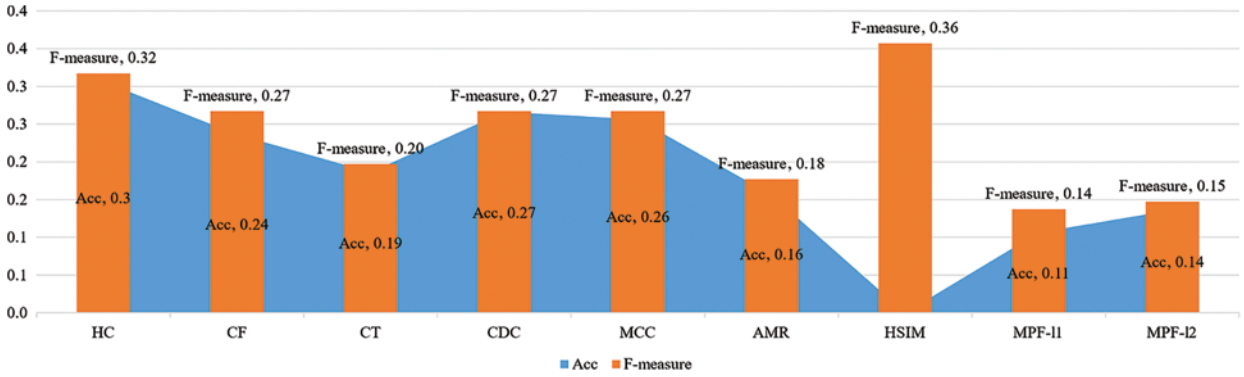
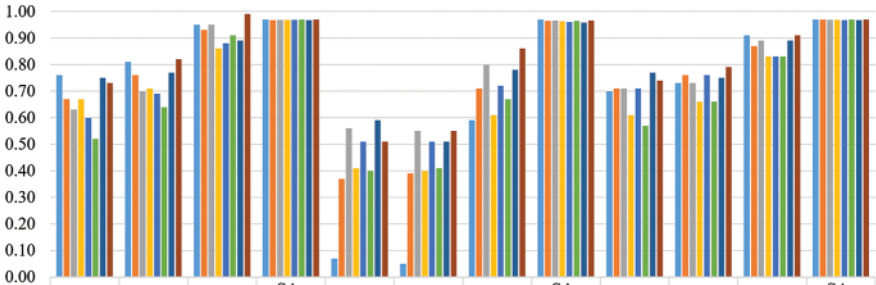


Figure 10: The relative improvements of the SA-SSAE model compared to HC, CF, CT, CDC, MCC, AMR, HSIM, MPF-*l1*, and MPF-*l2*

Fig. 11 depicts the comparison findings across several crowd scenario categories (mass movement, street, street-market, station, mall, public pathway, cross-walk, and escalator). The proposed SA-SSAE model performs the best among the other three approaches for all the crowd scenario categories. Moreover, the proposed model provides robust results for all evaluated metrics because it considers the trajectory history for several known frames, which is a significant factor in the model's training. The NMI results for the mass movement category for MCC and CT approaches are very low (close to zero). The SADC approach succeeds in increasing the results of NMI to 0.59. However, SA-SSAE outperforms SADC by 0.38, which proves our model's efficiency.



	Purity				NMI				RI			
	CM	CT	SADC	SA-SSAE	CM	CT	SADC	SA-SSAE	CM	CT	SADC	SA-SSAE
Massmovement	0.76	0.81	0.95	0.97	0.07	0.05	0.59	0.97	0.70	0.73	0.91	0.97
Street	0.67	0.76	0.93	0.97	0.37	0.39	0.71	0.96	0.71	0.76	0.87	0.97
Street-Market	0.63	0.70	0.95	0.97	0.56	0.55	0.80	0.97	0.71	0.73	0.89	0.97
Station	0.67	0.71	0.86	0.97	0.41	0.40	0.61	0.96	0.61	0.66	0.83	0.97
Mall	0.60	0.69	0.88	0.97	0.51	0.51	0.72	0.96	0.71	0.76	0.83	0.97
Public-walkway	0.52	0.64	0.91	0.97	0.40	0.41	0.67	0.96	0.57	0.66	0.83	0.97
Cross-walk	0.75	0.77	0.89	0.97	0.59	0.51	0.78	0.96	0.77	0.75	0.89	0.97
Escalator	0.73	0.82	0.99	0.97	0.51	0.55	0.86	0.97	0.74	0.79	0.91	0.97

Figure 11: Quantitative comparison according to the scene type (mass movement, street, street-market, station, mall, public-walkway, cross-walk, escalator)

Similarly, Figs. 12 and 13 compare the findings across several crowd scenario categories, structured or unstructured, as well as indoor or outdoor, respectively. SA-SSAE excels in all categories.

4.2.3 Sparsity Parameter Study

The optimal value of the sparsity proportion parameter for the SA-SSAE model was determined via comparative experiments using the CUHK benchmark. Table 5 illustrates that the sparsity proportion gives the best results for all performance metrics when its value is between 0.01 and 0.1. Thus, 0.01 was determined for the sparsity proportion parameter.

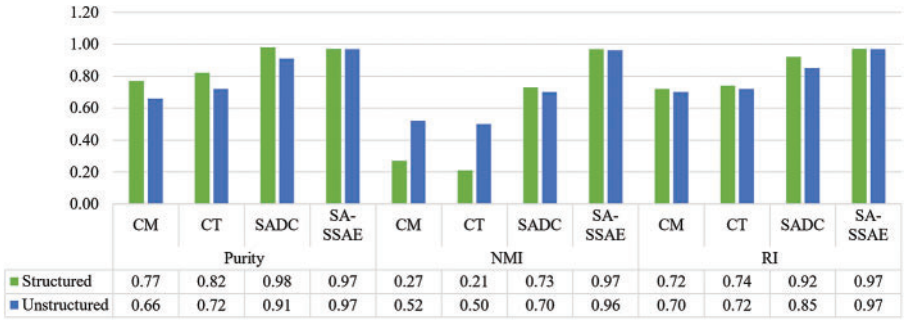


Figure 12: Quantitative comparison according to the crowd movement (structured or unstructured)

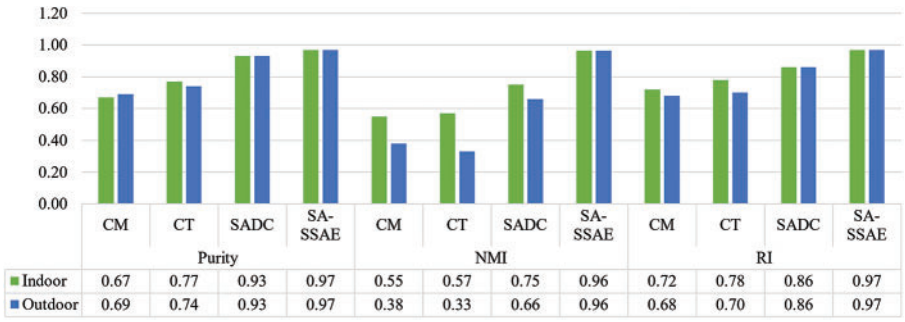


Figure 13: Quantitative comparison according to the crowd location (indoor or outdoor)

Table 5: Various values of sparsity portion parameter. The sparsity proportion value must be in the range [0, 1]

Sparsity Proportion	Purity	NMI	RI	Acc	F-measure
0.00	0.73	0.26	0.64	0.74	0.75
0.001	0.73	0.27	0.65	0.75	0.75
0.01	**0.96**	**0.84**	**0.95**	**0.94**	**0.94**
0.1	**0.96**	**0.84**	**0.95**	**0.94**	**0.94**
0.5	0.96	0.83	0.95	0.94	0.93
1	0.73	0.26	0.64	0.74	0.75

5 Conclusion

This study proposed a deep learning framework for VMS using spatial-angular stacked sparse autoencoders. The framework is meant to aid in proactive stampede avoidance to enhance people's

safety in crowd scenes. In this study, it is suggested to decompose the motion tracks in a given crowd scene into groups. Each group contains trajectories with similar behavioral characteristics. We prove that sparse stack autoencoders can be adopted effectively for VMS and provide better results than traditional clustering methods. According to experimental results, the proposed SA-SSAE model is superior to the comparative methods in terms of purity, NMI, RI, accuracy, and F-measure. Future research will focus on methods for integrating our VSM framework with outlier detection. Another crucial element that we will consider for increasing robustness is camera motion. To further improve the performance of the segmentation results, new features will be researched by utilizing other feature descriptors. Moreover, future research can be implemented by investigating the performance of the Kalman filter and YOLO (You Only Look Once) for VMS.

Acknowledgement: The authors thank the Deputyship of Research & Innovation, Ministry of Education in Saudi Arabia, for funding this research work through Project Number 758. The authors also would like to thank the Research Management Center of Universiti Teknologi Malaysia for managing this fund under Vot. No. 4C396.

Funding Statement: This research work is supported by the Deputyship of Research & Innovation, Ministry of Education in Saudi Arabia (Grant Number 758).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Anthwal and D. Ganotra, "An overview of optical flow-based approaches for motion segmentation," *The Imaging Science Journal*, vol. 67, no. 5, pp. 284–294, 2019.

[2] M. Munsif, H. Afridi, M. Ullah, S. D. Khan, F. A. Cheikh *et al.*, "A lightweight convolution neural network for automatic disasters recognition," in *The 10th European Workshop on Visual Information Processing (EUVIP)*, Lisbon, Portugal, pp. 1–6, 2022.

[3] A. Al-Dhamari, R. Sudirman and N. H. Mahmood, "Abnormal behavior detection using sparse representations through sequential generalization of K-means," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 29, no. 1, pp. 152–168, 2021.

[4] A. Raza, M. Rafiq, J. Awrejcewicz, N. Ahmed and M. Mohsin, "Dynamical analysis of coronavirus disease with crowding effect, and vaccination: A study of third strain," *Nonlinear Dynamics*, vol. 107, no. 4, pp. 3963–3982, 2022.

[5] S. Wu and H. San Wong, "Crowd motion partitioning in a scattered motion field," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 1, pp. 1443–1454, 2012.

[6] J. Shao, C. C. Loy and X. Wang, "Learning scene-independent group descriptors for crowd understanding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 6, pp. 1290–1303, 2017.

[7] A. Hafeezallah, A. Al-Dhamari and S. A. R. Abu-Bakar, "Multi-scale network with integrated attention unit for crowd counting," *Computers, Materials & Continua*, vol. 73, pp. 3879–3903, 2022.

[8] A. Hafeezallah, A. Al-Dhamari and S. A. R. Abu-Bakar, "U-ASD net: Supervised crowd counting based on semantic segmentation and adaptive scenario discovery," *IEEE Access*, vol. 9, pp. 127444–127459, 2021.

[9] S. Ali and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, pp. 1–6, 2007.

[10] J. Shao, C. Change Loy and X. Wang, "Scene-independent group profiling in crowd," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 2219–2226, 2014.

[11] J. Bian, D. Tian, Y. Tang and D. Tao, "A survey on trajectory clustering analysis," *arXiv Prepr. arXiv1802.06971*, 2018.

[12] H. Li, J. Liu, R. W. Liu, N. Xiong, K. Wu *et al.*, "A dimensionality reduction-based multi-step clustering method for robust vessel trajectory analysis," *Sensors*, vol. 17, no. 8, pp. 1792, 2017.

[13] M. Irfan and M. Munsif, "Deepdive: A learning-based approach for virtual camera in immersive contents," *Virtual Reality & Intelligent Hardware*, vol. 4, no. 3, pp. 247–262, 2022.

[14] H. Ullah, M. Ullah and M. Uzair, "A hybrid social influence model for pedestrian motion segmentation," *Neural Computing and Applications*, vol. 31, no. 11, pp. 7317–7333, 2019.

[15] T. M. Nguyen and Q. J. Wu, "A consensus model for motion segmentation in dynamic scenes," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 12, pp. 2240–2249, 2015.

[16] M. Seyedhosseini and T. Tasdizen, "Semantic image segmentation with contextual hierarchical models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 5, pp. 951–964, 2015.

[17] M. Narayana, A. Hanson and E. Learned-Miller, "Coherent motion segmentation in moving camera videos using optical flow orientations," in *Proc. of the IEEE Int. Conf. on Computer Vision*, Sydney, NSW, Australia, pp. 1577–1584, 2013.

[18] E. Meunier, A. Badoual and P. Bouthemy, "EM-driven unsupervised learning for efficient motion segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4462–4473, 2022.

[19] S. Choudhury, L. Karazija, I. Laina, A. Vedaldi and C. Rupprecht, "Guess what moves: Unsupervised video and image segmentation by anticipating motion," *arXiv Prepr. arXiv2205.07844*, 2022.

[20] D. Mukherjee and Q. M. J. Wu, "Streaming spatio-temporal video segmentation using Gaussian mixture model," in *IEEE Int. Conf. on Image Processing (ICIP)*, Paris, France, pp. 4388–4392, 2014.

[21] A. Milan, L. Leal-Taixé, K. Schindler and I. Reid, "Joint tracking and segmentation of multiple targets," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Boston, MA, USA, pp. 5397–5406, 2015.

[22] S. C. Shadden, F. Lekien and J. E. Marsden, "Definition and properties of lagrangian coherent structures from finite-time lyapunov exponents in two-dimensional aperiodic flows," *Physica D: Nonlinear Phenomena*, vol. 212, no. 3–4, pp. 271–304, 2005.

[23] Y. Tang, Z. Pan, W. Pedrycz, F. Ren and X. Song, "Viewpoint-based kernel fuzzy clustering with weight information granules," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 7, no. 2, pp. 342–356, 2022.

[24] B. Zhou, X. Tang, H. Zhang and X. Wang, "Measuring crowd collectiveness," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1586–1599, 2014.

[25] C. Tomasi and T. Kanade, "Detection and tracking of point," *Int. J. Comput. Vis.*, vol. 9, pp. 137–154, 1991.

[26] Y. Wu, Y. Ye and C. Zhao, "Coherent motion detection with collective density clustering," in *Proc. of the 23rd ACM Int. Conf. on Multimedia*, Canada, pp. 361–370, 2015.

[27] W. Lin, Y. Mi, W. Wang, J. Wu, J. Wang *et al.*, "A diffusion and clustering-based approach for finding coherent motions and understanding crowd scenes," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1674–1687, 2016.

[28] A. K. Pai, A. K. Karunakar and U. Raghavendra, "Scene-independent motion pattern segmentation in crowded video scenes using spatio-angular density-based clustering," *IEEE Access*, vol. 8, pp. 145984–145994, 2020.

[29] S. R. Saufi, Z. A. Bin Ahmad, M. S. Leong and M. H. Lim, "Low-speed bearing fault diagnosis based on ArSSAE model using acoustic emission and vibration signals," *IEEE Access*, vol. 7, pp. 46885–46897, 2019.

[30] M. Rodriguez, J. Sivic, I. Laptev and J. -Y. Audibert, "Data-driven crowd analysis in videos," in *2011 Int. Conf. on Computer Vision*, Barcelona, Spain, pp. 1235–1242, 2011.

[31] B. Zhou, X. Wang and X. Tang, "Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Providence, RI, USA, pp. 2871–2878, 2012.

[32] M. J. Zaki and W. Meira Jr, *Data Mining and Machine Learning: Fundamental Concepts and Algorithms*. Cambridge, UK: Cambridge University Press, 2020.

[33] Q. Wang, M. Chen, F. Nie and X. Li, "Detecting coherent groups in crowd scenes by multiview clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 1, pp. 46–58, 2018.

[34] W. Ge, R. T. Collins and R. B. Ruback, "Vision-based analysis of small groups in pedestrian crowds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 1003–1016, 2012.

[35] B. Zhou, X. Tang and X. Wang, "Coherent filtering: Detecting coherent motions from crowd clutters," in *European Conf. on Computer Vision*, Florence, Italy, pp. 857–871, 2012.

[36] M. Chen, Q. Wang and X. Li, "Anchor-based group detection in crowd scenes," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, USA, pp. 1378–1382, 2017.