Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 169 – 185
© TÜBİTAK
doi:10.3906/elk-2004-14

Characterization of different crowd behaviors using novel deep learning framework

Abdullah J. ALZAHRANI[1], **Sultan Daud KHAN**[2,*]
[1]College of Computer Science and Software Engineering, University of Ha'il, Ha'il, Saudi Arabia
[2]Department of Computer Science, National University of Technology, Islamabad, Pakistan

Received: 03.04.2020 • **Accepted/Published Online:** 01.07.2020 • **Final Version:** 27.01.2021

Abstract: Crowd behavior understanding is recognized as a complex problem due to unpredictable behavior of humans and complex interactions of individuals in groups. For crowd managers, it is crucial to understand the crowd dynamics to manage the crowd efficiently and effectively. Current practice of crowd management is based on manual analysis of the scene. Such manual analysis of the scene is a tedious job and usually prone to errors due to limited human capabilities. Therefore, the task of automatizing crowd analysis has received tremendous attention from the research community during the recent years. In this paper, we propose a deep model framework that automatically characterizes different crowd behaviors based on motion and appearance. We first extract dense trajectories from the input video segment and then generate trajectory image by projecting trajectories on to image plane. Trajectory image effectively captures relative motion in the scene. We use stack of trajectory images to train deep convolutional network that learns compact and powerful representation of motion in the scene. We evaluate our approach on UCF, CUHK, and Crowd-11 benchmark datasets. From the experiment results, we demonstrate, both in quantitative and qualitative ways, that the proposed framework outperforms other existing methods by a great margin.

Key words: Crowd analysis, crowd segmentation, crowd behavior

1. Introduction

With growing urbanization, public safety in crowded events is the prime concern of crowd managers and dwellers. Typically in religious and political gatherings, a large number of people gather in a restricted environment. Although such gatherings serve peaceful purposes, crowd disasters still occur. To control and avoid crowd disasters, usually, surveillance cameras are mounted in different areas of the crowded scene. These surveillance cameras provide a significant amount of information that can be utilized for crowd analysis. Usually, security personnel manually analyze live streams to detect suspicious activities and behaviors. Such manual analysis is a cumbersome job and usually prone to errors due to limited human capabilities. Therefore, automatic analysis is required to efficiently characterize different crowd behaviors.

Automated crowd behavior understanding has a wide range of applications, such as congestion detection and anomaly detection. The goal of this work is to develop a framework that automatically identifies specific crowd behaviors. Due to the complexity of the problem, a few strides have been made to address the problem in recent years. Most of the existing methods focus on detecting abnormal crowd behaviors [1–6], counting people in crowds [7–13], characterizing different motion flows, and segmentation [14–16].

[*]Correspondence: sultandaud@nutech.edu.pk

For automated crowd analysis, motion representation plays a vital role in action recognition systems. Traditionally, motion information (trajectories) is extracted from the videos via detection and tracking. This conventional method works well in low-dense situations; however, it suffers a significant setback in crowded scenes. This reduces performance attributes to partial/full occlusion of human body that limits the performance of pedestrian detectors. Several methods are reported in the literature to capture motion information from the video. For example, [18–59] achieved high performance by learning low-level visual features using optical flow fields. Histogram of oriented gradients [21] and histogram of optical flow were extracted in [22] and features were then encoded with bag of features. To accurately capture motion information, Wang et al. [59] extracted dense trajectories by tracking dense points at multiple scales. The method was further improved in [18] by incorporating camera motion estimation. In [23], spatial and temporal extents were learned by employing dense trajectories. The abovementioned motion extraction methods achieve noticeable success in recognition of short duration actions. However, these methods cannot model long-term behaviors. Therefore, as a solution, we extract motion information by adopting a holistic approach and employ optical flow computation method to obtain global motion information from input videos.

Convolutional neural network (CNN) achieves tremendous success in object detection, recognition, and segmentation tasks. The features learned by CNN are robust and generic compared to hand-craft features. For this reason, researchers have employed various CNN architectures to learn motion representation from videos. In order to learn motion from videos, several methods [24–27] have been proposed. To model motion representation from videos, Ji et al. [28] used 3D-CNN that performs 3D convolutions on multiple channels of the input. In the same way, Simonyan et al. [19] proposed a two-stream network that extracts spatial and motion features. Unlike these models, Wang et al. [27] proposed a deep trajectory model that exploits deep and hand-craft features. Hasan et al. [29] proposed a dynamic learning strategy for streaming videos. However, these models cannot capture long-term motion information from video sequences.

In this paper, we propose an approach of identifying multiple crowd behaviors. We first extract spatial and temporal motion information based on low-level motion features, i.e. optical flow. After extracting motion information, we generate trajectory images (TIs) by projecting trajectories on a 2D plane. We then use multiple TIs to train a CNN that learns long-term representations of motion. The overview of our framework is depicted in Figure 1. Our contributions can be summarized as follows:

- For motion extraction, our method eliminates the prevalent paradigm of detection and tracking and uses low-level local features combined with high global-level motion information.
- Our model learns compact representations of motion by learning from TIs and avoids the need of learning directly from optical flow.
- Our approach, in contrast to Solmaz et al.'s [30], identifies multiple crowd behaviors.
- Our approach identifies multiple crowd behaviors in a single scene and is not restricted to isolated activities as in [31].
- We evaluate our method on Crowd-11 [32], UCF [30], and CUHK [33] datasets. The experiment results show that our proposed method outperforms the state-of-the-art methods.

The rest of the paper is organized as follows: In Section 3, we discuss the proposed methodology of extracting motion information from videos. Section 4 discusses TI generation. Section 5 discusses experiment and comparative results. Finally, Section 6 concludes the paper.

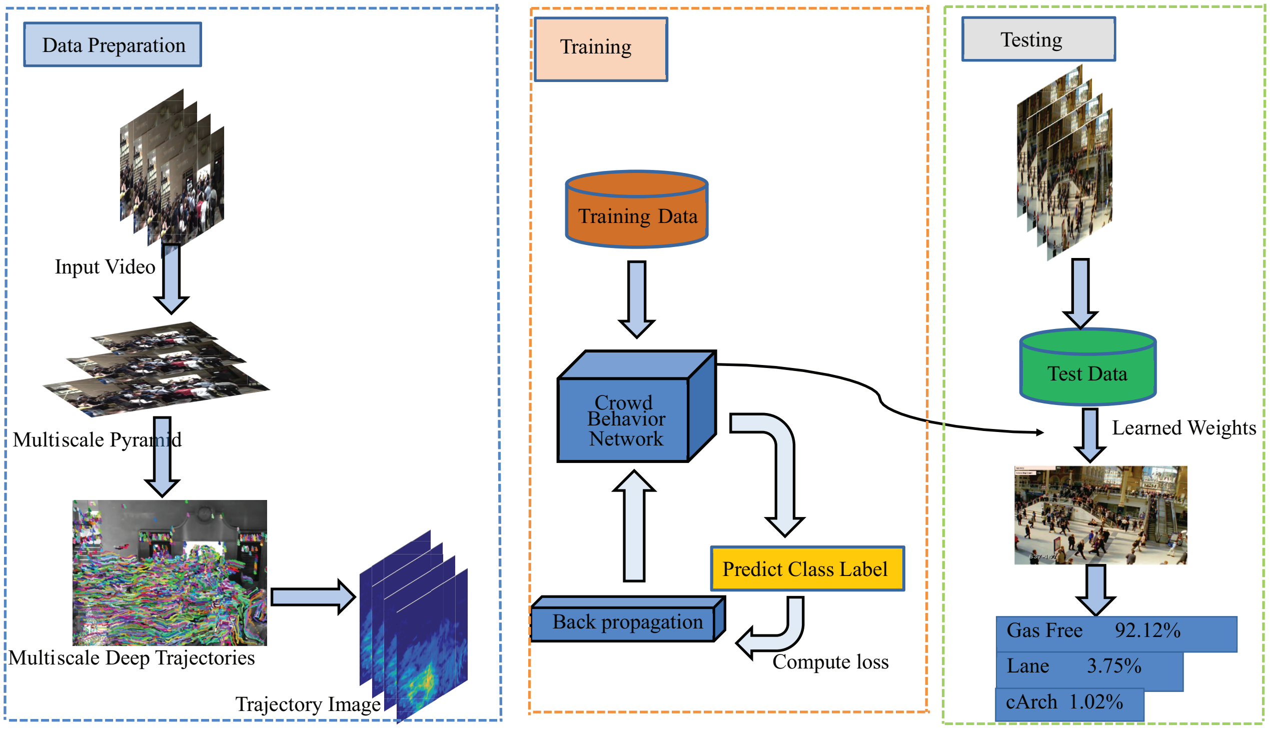


Figure 1. Overview of the proposed framework. Trajectories are extracted from input video using multiplescales. TIs are then generated using extracted trajectories. TIs are stacked together and applied as input to the network. The learned model is then used to test the new video and to assign behavior class probabilities.

2. Related work

A considerable amount of work is reported in the literature regarding vision-based crowd analysis. Most of the existing methods mainly focus on finding the correlation among the individuals and employ different techniques to estimate coherent motion patterns [34, 35]. Furthermore, these methods exploit low-level motion information, i.e. optical flow, to find independent and dominant motion patterns [36, 37]. Li et al. [38] reported a comprehensive survey about crowd flow segmentation methods.

As mentioned above, most of the existing methods focus on crowd analysis as a whole and do not particularly focus on characterizing crowd behaviors in the scene. Therefore, we find limited work on detecting and identifying crowd behaviors. Andrade et al. [39] and Hu et al. [40] adopt a holistic approach to extract motion information from the scene. Later on, the extracted motion information is utilized to understand the crowd scene. Other methods, such as Sultani et al.'s [41], focus on detecting abnormal behaviors in the scene. In contrast to these methods, our method is mainly concerned with identifying different crowd behaviors. Widhalm et al. [42] and Li et al. [43] proposed models to learn motion patterns from the crowded scenes. Rao et al. [44] proposed a method that learns the orientation field from the optical flow to detect critical locations [45]. Helbing et al. [46] proposed a model based on fluid dynamics for simulating the crowd movement and demonstrate how the phenomenon of lane formation occurs in high-density crowds.

For understanding crowds, various methods exploit low-level features to describe crowd motion. Mehran et al. [47] incorporated theoretical social force model for understanding dynamic interactions among the individuals in the scene. Wu et al. [48] measured motion characteristics by Lyapunov exponent that measures the correlation among the individuals. For identifying different crowd behaviors, Solmaz et al. [30] computed jacobian matrix, where the eigenvalues of jacobian are used to identify different crowd behaviors.

To intuitively describe the crowd characteristics, several methods have been reported to measure intrinsic and extrinsic characteristics of the crowd, such as crowd complexity, crowd collectives, and stability analysis. Ali et al. [36] proposed a particle-based dynamic system to compute the complexity and stability of crowded scenes. Zhou et al. [33] measured the degree of collectiveness of the crowd with a crowd descriptor by exploiting correlation among different motion paths. Shao et al. [49] proposed a model that measures collectiveness, conflict, and stability using a single-stage model. Yi et al. [50] detected stationary groups of pedestrians by using 3D stationary maps. Also, some methods exploited contextual information [51, 52] to classify different human actions in videos.

Generally, most of the existing methods propose various solutions for crowd counting, crowd density estimation, crowd tracking, and anomaly detection. However, a limited amount of work has been reported in the literature on characterizing crowd behaviors. One of the reasons is the lack of proper datasets, which are challenging to acquire. Most datasets aimed to study a specific behavior, such as panic detection [53], opposite flow [54], and violence flow [55]. These datasets do not cover natural crowd behaviors. Furthermore, the size of these datasets is small, and for a covolutional neural network, it is impossible to learn the representations of different behaviors. Most related studies, [56] and [49], propose models that can identify limited crowd behaviors, i.e. merging, divergence, bottleneck, and crossing.

3. Extracting motion information

Trajectories capture local spatial and temporal information from videos. To provide good coverage of foreground moving objects, long and dense trajectories are highly desirable. There are two ways to capture motion information: (1) sparse features (like corner points and SIFT features), (2) dense optical flow. In the first type, interest points are extracted from the initial frame of the video segment and then tracked through multiple frames. The most commonly used method for obtaining sparse trajectories are KLT [57] and SIFT flow [58]. The trajectories obtained through these methods are sparse and cannot provide detailed information for crowd behavior understanding. In the dense optical flow method, optical flow is computed for every pixel and dynamical system is initialized with the grid of particles overlaid on the initial frame of the video. Trajectories are then extracted through time integration of the dynamical system. The trajectories obtained through this method are long and dense enough to capture long-term motion information. However, these trajectories are not reliable as the optical flow is sensitive to the illumination changes. Nonetheless, trajectories achieved through this method are reliable when extracted from structured crowd scenes, where the pedestrians perform the same behavior like a group of people moving in the same direction. However, in the case of unstructured crowds, the trajectories are unreliable, since people display different behaviors like people moving in arbitrary directions. In these cases, particles cease to follow a stable path and mix with different motion patterns. Therefore, as a solution, we proposed a new approach that can extract dense, long, and reliable trajectories for extracting motion information from the scene.

3.1. Reliable descriptive motion information

The framework takes a sequence of video frames as input. The input video sequence is divided into N number of temporal segments where the size of each segment is k. For each segment $\mathbb{S}$, we first compute the dense optical flow between two consecutive frames of segment $\mathbb{S}$. Consider a particle i at time t of segment $\mathbb{S}$. Let $\mathbb{X}_{i,t}$ be the spatial location $X_{i,t} = (x_{i,t}, y_{i,t})$ of particle i, and $\mathbb{V}_{i,t}$ be the flow vector that encodes the change in the horizontal and vertical position, which is given by Equation 1.

$$\mathbb{V}_{i,t} = \mathbb{O}(\mathbb{X}_{i,t}) \tag{1}$$

We launch a grid $\mathbb{G}$ of particles over the first optical flow field at initial time t_1 of the segment $\mathbb{S}$. We keep the resolution of the grid the same as the resolution of the image in order to capture dense motion information. Let the initial location of particle i at time t be $\mathbb{X}_{i,t} = (x_{i,t}, y_{i,t})$, its next position at time $t+1$ is computed by using Equation 2.

$$\mathbb{X}_{(i,t+1)} = \mathbb{O}(\mathbb{X}_{(i,t)}) + \mathbb{X}_{(i,t)} \tag{2}$$

During the particle advection process, we generate and maintain a pair of motion maps, ψ_x and ψ_y. These motion maps contain the initial and all subsequent locations of all particles of the grid. However, trajectories extracted through Equation 2 cannot provide reliable motion information particularly in unstructured crowd scenes. This is because in unstructured crowd scenes, motion particles drift from the original flow and become part of another flow which may have a different direction. In order to avoid this defect, we introduce a binary indicator that will allow the particle to stop the advection process to avoid drifting. For this purpose, we modify Equation 2 as Equation 3.

$$\mathbb{X}_{(i,t+1)} = \mathbb{X}_{(i,t)} + \mathbb{O}(\mathbb{X}_{(i,t)}) * \mathbb{B}_i \tag{3}$$

$$\mathbb{B}_i = \begin{cases} 1, & \text{if } \| \theta_{i,1} \text{ - } \theta_{i,t} \|_2 < \xi \\ 0 & \text{otherwise} \end{cases}$$

The particle, as mentioned above, will stop the advection process if the circular distance between its position at t and its next position at $t+1$ is greater than a threshold ξ. After particle advection, some of the trajectories belong to the background and are generated due to noise. In order to suppress these trajectories, we simply compute trajectory length, which is the euclidean distance between the start and end points of the trajectory. We then suppress those trajectories for which $\| \ (x_i^1, y_i^1) \text{ - } (x_i^T, y_i^T) \ \|_2 < \delta$. The trajectories obtained through this method are long and dense and provide reliable information for further analysis of crowd behavior.

3.2. Motion and structural descriptors

After obtaining dense, long, and reliable trajectories, the next step is to compute descriptors that encode the motion and structural information of the trajectories. However, before computing descriptors, we encode local motion patterns by computing the shape of trajectory. Consider a trajectory of length L, we describe its shape by a set of displacement vectors as $(\Delta P_t \ldots \Delta P_{t+L-1})$, where ΔP_t is a displacement vector and computed as $\Delta P_t = (\ P_{t+1} - P_t \ )$.

Let $\Omega(V) = \{\omega_1, \omega_2, \ldots \omega_k\}$ represent the set of trajectories extracted from the input video V. Each trajectory ω_i is represented by (i, j, u, v), where i and j represent spatial coordinates of the video frame and

u, v represent the displacement vector along the x-axis and y-axis, respectively. These trajectories will be used in constructing TIs.

4. Generating trajectory images

For action recognition tasks, most of the existing work relies on histogram of optical flow, histogram of gradient, and motion boundary histogram [59]. These descriptors utilize trajectory-level information and extract different features which are then encoded using bag of words or fisher vector. However, these descriptors cause high computation costs for long-duration actions. In order to overcome this problem, we propose a novel way of trajectory motion representation. We convert the 3D long-term motion information to two-dimensional space that can be effectively and efficiently processed through CNNs.

Given a video sequence S of an action, we extract trajectories Ω from video sequence S. We then convert $\Omega(S)$ to image plane $I(S)$ using Equation 4

$$I_{(i,j)} = \begin{cases} \sqrt{u_t^2 + v_t^2}, & \text{if } i_t = \text{i and } j_t = j \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The above equation converts 3D trajectories to euclidean space. However, we observed that trajectories overlap in such dense representations. These overlapping trajectories cause serious problems in the action recognition process. In order to reduce the effect of overlapping trajectories, we extend the equation above by including an overlapping constraint. In order to incorporate the overlapping constraint, we define two terms, Ψ in Equation 5 and Υ in Equation 6 for trajectory image I.

$$\Psi_{(i,j)} = \quad 1, \qquad \text{if } I_{(i,j)} \neq 0 \tag{5}$$

$$\Upsilon^t_{(i,j)} = \Big\{ \ \Upsilon^{t-1}_{(i,j)} + \textstyle\sum_{n=1}^{N} \sum_{t=1}^{T} \Psi_{(i,j)}, \tag{6}$$

where Ψ checks the nonzero value in TI at location (i,j) after trajectory conversion and Υ represents the number of overlapped trajectories. We set a threshold value of 0.6 and if the value of Υ is greater than 0.6, we reconstruct TI.

5. Experiment results

In this section, we present quantitative and qualitative evaluations of different state-of-the-art methods. For evaluation purpose, we use three publicly available benchmark datasets, namely, UCF [30], CUHK, and Crowd-11. We discuss the details of each dataset below.

UCF dataset was initially proposed by Solmaz et al. [30]. This dataset covers five different crowd behaviors, namely, Lane, cArch, ccArch, fountainhead, blocking, and bottleneck. The dataset was collected from different sources, for example, Youtube, Thought Equity, and BBC Motion gallery. The dataset covers different scenarios with different view points, resolutions, frame rates, and duration. The dataset contains 66 videos of Lane, 20 of ccArch, 8 of cArch, 29 of fountainhead, and 20 of bottleneck. We do not consider the blocking behavior during performance evaluation due to limited number of samples.

CUHK crowd dataset was proposed by Zhou et al. [33]. It contains 95 videos that were collected from different indoor and outdoor scenes. This dataset also covers five behaviors, i.e. Lane, cArch, ccArch, fountainhead, and bottleneck. The dataset contains 91 videos of Lane, 20 of cArch, 18 of ccArch, 9 of fountainhead, and 20 of bottleneck behaviors.

UCF and CUHK datasets are predominately used for evaluating crowd behavior models. However, these datasets suffer from the following limitations. (1) They have a limited number of samples per class, which leads to inefficient modeling of crowd behaviors. (2) Definitions of classes are ambiguous. Definition of different crowd behaviors contain many discrepancies that lead to poor learning of the models. Deep learning requires a large amount of data to learn the best representations of different classes. However, we observed that the UCF and CUHK are small enough to train a deep neural network.

To train an efficient crowd behavior model, a substantial amount of data is needed. Moreover, the dataset should incorporate more crowd behaviors.

To address the above problems, Dupont et al. [32] proposed Crowd-11 dataset that covers crowd behaviors that have representation in daily life. This dataset contains 11 classes of crowd behaviors and provides sufficient amount of data to train a deep neural network. The dataset was collected from other datasets such as Violent-Flow [55], WorldExpo'10 [60], Agoraset [61], PETS [62], and Hockey Fight [63].The comparison of datasets is presented in Table 1. The summary of Crowd-11 dataset is presented in Table 2.

Crowd-11 dataset has the following class labels: gas free, gas jammed, laminar flow, turbulent flow, crossing flow, merging flow, diverging flow, static calm, static agitated, interacting crowd, and no crowd. Laminar flow represents the smooth motion of the crowd. Turbulent flow occurs in unstructured crowds, where pedestrians move in different directions, obstructing each other's flow. Crossing flow occurs when pedestrians move in opposite directions but do not obstruct the motion of each other. Merging flow occurs when groups of pedestrians from different locations of the scene merges at one location, for example, a train station. Diverging flow occurs when pedestrians move in different directions. Gas free flow represents situations where pedestrians can freely move around the environment without any obstruction. Gas jammed flow represents situations where pedestrians gather in a constrained environment and increase crowd density to a critical level. Static calm occurs when pedestrians are static and do not move. Interactive crowd refers to the behavior when pedestrians move opposite to each other in a violent manner. No crowd contains the video sequence with no crowd and contains vehicles and background.

Table 1. Summary of crowd behavior datasets.

Dataset	Total number of videos	Total number of frames	Number of behaviors	Resolution
UCF [30]	126	-	5	Various
CUHK [33]	474	60,384	5	Various
Crowd-11 [32]	6272	621,196	11	Various

For the implementation of the proposed framework, we use caffe library. We train and test our model on Nvidia Titan Xp GPU. After generating TIs, we use a pretrained model of GoolgeNet to learn compact representation of crowd motion. We use stacked TIs and provide input to the network. Our network is composed of five convolution layers, three normalization layers, two pooling layers, and one fully connected layer. Early convolution layers learn the features from the local neighborhood while last convolution layers learn the context associated with the action/behavior.

We use different state-of-the-art methods for comparison. Since the source code of most of the state-of-the-art methods are not publicly available, we followed our own implementation of these models. The existing

Table 2. Number of videos per each crowd behavior in Crowd-11 dataset.

Behavior class	Interacting crowd	Static agitated	Gas free	Gas jammed	Divergence flow	Crossing Flow	Laminar flow	Turbulent flow	Static calm	Not crowd	Merging flow
Number of videos	248	410	529	520	184	763	1304	892	737	390	295

related baseline methods are briefly discussed as follows:

1. ER is the eigenvalue ratio [30] method proposed by Solmaz et al. The method starts by first extracting motion information using optical flow and then particle advection is employed to extract long and dense trajectories. Jacobian matrix is employed to find the candidate regions, where the eigenvalues of jacobian matrix are used to classify different behaviors of the crowd.

2. ER-G is the variant of ER that follows the same pipeline of extracting motion information; however, this method uses ground truth and predicted points for computation of eigenvalues of the jacobian matrix.

3. Two-SCNN is the two-stream architecture (two-streamCNN) [19] which computes optical flow from a temporal segment of video. The stack of consecutive optical flow frames is combined with the corresponding stack of RGB images fed to convolutional neural network. The authors use the architecture of AlexNet. At the end of two stream network, the features from both branches of the network are combined by averaging corresponding scores. For the comparison, we use the publicly available code of this method.

4. C3D is the 3D convolution (C3D), where the first and second dimension correspond to horizontal and vertical axis of the image and the third dimension corresponds to stack of optical flows for each temporal segment. Here temporal segment represents a group of 16 consecutive color images. The basic architecture of the network consists of five 3D convolution + pooling layers (3D), and fully connected layers are added to end of the network for classification.

5. V3G is a blend of C3D [64] and VGG network along with batch normalization [65]. The network follows the baseline architecture of VGG with a modification of using 3D convolutions and pooling layers instead of 2D. Batch normalization layer is added after 3D pooling layer. Moreover, the network is converted into a full convolutional network by converting fully connected layers to 1 x 1 convolutional layers. Dropout with the ratio of 0.7 is added to enhance the generalization capability of the network and avoid overfitting.

We evaluate and compare the performance of different methods on all datatsets in Tables 3–5. We use true-positive and false-positive values (based on fixed threshold) as the evaluation metric. We used two variants, i.e. appearance network and motion network of two-SCNN [19]. These networks are learned independently. We observed from experiments that appearance (RGB)-based network performs better than motion-based network. Motion-based network suffers from the overfitting problem due to utilization of large amount of complex motion information, which ultimately increases the complexity of the network due to increased number of parameters. However, we observed that fusion model (concatenation of feature maps of both appearance and motion network) yields better results. We also observed that this configuration yields far better results than simply averaging the score of both network as proposed in the original work [19]. This is because concatenating feature maps from both networks capture correlated information that cannot be captured by simply averaging the scores of both networks. We report the results of different configuration on all datasets in Figure 2.

Table 3. Summary of classification performances of different methods using Crowd-11 dataset. TP is the true-positive rate and FP is the false-positive rate. Our proposed method achieves higher a TP rate and a lower FP rate.

Behavior class	ER		ER-G		Two-SCNN		C3D		V3G		Proposed	
	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP
Static agitated	45.21%	29.74%	36.72%	41.09%	65.87%	35.19%	72.42%	29.34%	68.72%	45.19%	73.26%	27.34%
Gas free	37.42%	64.79%	42.62%	32.48%	60.27%	42.94%	68.84%	40.69%	60.66%	50.36%	72.92%	34.37%
Gas jammed	46.22%	43.10%	46.37%	42.62%	67.49%	33.27%	62.76%	35.19%	63.74%	34.13%	68.25%	23.71%
Divergence flow	62.34%	32.43%	55.39%	29.30%	72.35%	33.67%	75.12%	34.22%	67.96%	36.62%	75.27%	29.33%
Crossing flow	55.79%	34.10%	52.71%	33.95%	65.21%	38.64%	69.63%	24.16%	65.15%	29.63%	71.25%	28.62%
Laminar flow	45.72%	39.62%	52.37%	40.74%	69.49%	29.38%	69.32%	25.79%	68.22%	33.68%	70.10%	22.43%
Turbulent flow	37.16%	45.32%	39.20%	34.02%	55.23%	37.00%	59.64%	30.97%	57.65%	28.77%	65.73%	27.88%
Static calm	55.24%	33.16%	52.39%	29.64%	62.12%	29.64%	65.00%	25.13%	64.36%	27.34%	69.65%	23.60%
No crowd	65.30%	24.10%	63.03%	27.39%	71.09%	28.62%	67.97%	30.11%	65.43%	32.46%	72.94%	26.30%
Merging flow	67.63%	33.16%	65.32%	30.76%	73.19%	28.62%	75.17%	30.13%	63.29%	38.90%	74.09%	29.62%
Interacting crowd	54.23%	37.13%	55.10%	35.12%	67.29%	36.73%	71.08%	26.64%	67.33%	34.02%	72.04%	32.01%

Table 4. Summary of classification performances of different methods using UCF dataset. TP is true-positive rate and FP is the false-positive rate. Our proposed method achieves a higher TP rate and a lower FP rate.

Behavior class	ER		ER-G		Two-SCNN		C3D		V3G		Proposed	
	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP
Lane	84.85%	24.44%	75.38%	32.10%	85.26%	32.14%	87.49%	32.16%	85.67%	27.43%	88.10%	27.34%
cArch	82.14%	13.33%	50.00%	31.36%	82.16%	21.06%	85.67%	25.16%	84.32%	24.16%	87.22%	20.13%
Fountainhead	79.31%	11.11%	55.17%	17.09%	82.74%	32.76%	84.04%	20.12%	80.73%	34.97%	86.70%	25.13%
Bottleneck	80.00%	6.67%	52.38%	16.00%	81.46%	20.78%	82.16%	26.04%	78.34%	30.76%	82.79%	16.37%

Table 5. Summary of classification performances of different methods using CUHK dataset. Our proposed method achieves a higher TP rate and a lower FP rate.

Behavior class	ER		ER-G		Two-SCNN		C3D		V3G		Proposed	
	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP
Lane	65.12%	23.10%	63.78%	26.34%	71.02%	25.75%	73.19%	19.64%	72.32%	25.78%	74.67%	15.10%
cArch	55.67%	23.49%	45.17%	45.92%	60.79%	26.37%	67.19%	23.74%	65.44%	27.10%	69.17%	20.99%
Fountainhead	62.37%	24.30%	57.62%	29.61%	65.42%	20.32%	65.97%	15.16%	66.42%	35.73%	67.10%	20.93%
Bottleneck	65.45%	35.60%	62.00%	30.44%	65.27%	15.32%	66.43%	14.13%	64.83%	27.94%	68.21%	20.39%

For the C3D method, we use two variants of the model. The first variant of the model is trained directly using the Crowd-11 data set. In the second variant, we first train the model on UCF dataset and then fine-tuned the model on Crowd-11 dataset. From the experiment results, we observed that the first variant achieved 49.71% accuracy while the second variant achieved 52.31% accuracy. This is because the second variant already

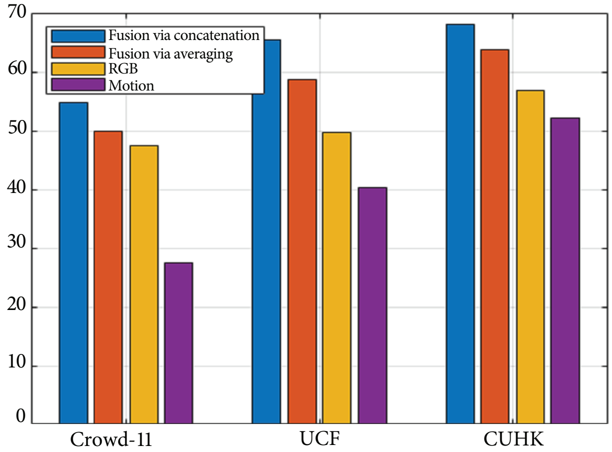


Figure 2. Performanced of different configurations of Two-SCNN [19]. Four variants are used to perform the comparison. First variant (fusion via concatenation) concatenates the features from both appearance and motion network. The second variant (fusion via averaging) fuses features from appearance and motion model. The third variant (RGB) is trained to learn only appearance features from video clips, while the fourth variant (motion) learns motion features from video clips.

learned features from UCF dataset and was able to precisely classify both dynamic and static behavior of the crowd in Crowd-11 dataset.

We also generate two versions of V3G in the same way. In this case, we also observed that model trained on Crowd-11 dataset achieves lower accuracy than training the model on UCF dataset first and then fine tunned on Crowd-11 dataset.

We select the best-performing variant of reference methods and report the comparison results in Tables 3–5 for each dataset.

From Table 3, it is obvious that the proposed method beats reference methods by producing higher TP and lower FN values. The reference method C3D produces comparable results and achieves better performance compared to V3G, two-CNN, and other reference methods. The superior performance of C3D can be attributed to adoption of 3D convolutional and pooling layers. Furthermore, the C3D model fuses appearance and motion features in the early stage, which further increases the performance of the model. However, we noticed that C3D network got confused in classifying merging flows with those of dense crowds, since the movement is almost zero. Furthermore, the C3D method incurs high computation costs due adoption of 3D convolutional and pooling layers. We also observed that the ER and ER-G methods produce relatively low TP values and high FP values compared to other reference methods. These methods are based on weak motion feature, i.e. optical flow that is susceptible to illumination changes and noise. This is why these methods could not precisely detect crowd behaviors that have real representation in daily life. Compared to other behavior classes, we observed that all methods recognize ”no crowd” class by producing higher TP and lower FP values. However, some of the reference methods still get confused by vehicle motions, which leads to increased FP values.

Table 4 shows the performances of all methods on UCF dataset. From the table, it is obvious that the proposed method achieves better performance compared to the reference methods by producing higher TP and lower FP values. For Lane behavior, the proposed method performs better than the other related methods. Furthermore, the proposed method correctly distinguishes clockwise and anticlockwise motion patterns, which allows the proposed method to correctly classify cArch behavior. The performances of ER and ER-G are worse than the other methods. This can be attributed to the fact that these methods are based on optical flow feature that is not properly fine-tuned for each video sequence. The reference method C3D produces comparable results for all four behaviors.

Table 5 shows the performance of methods on CUHK dataset. These results indicate that the proposed method outperforms the reference methods by detecting four crowd behaviors.

The results reported in Tables 3–5 are generated using a fixed threshold value (0.5) for deciding whether a behavior class is detected or not. However, we observed that using a single threshold value cannot provide conclusive information about the performance of all methods. Therefore, for a comprehensive evaluation, we use the receiver operating characteristic (ROC) curve as shown in Figure 3.

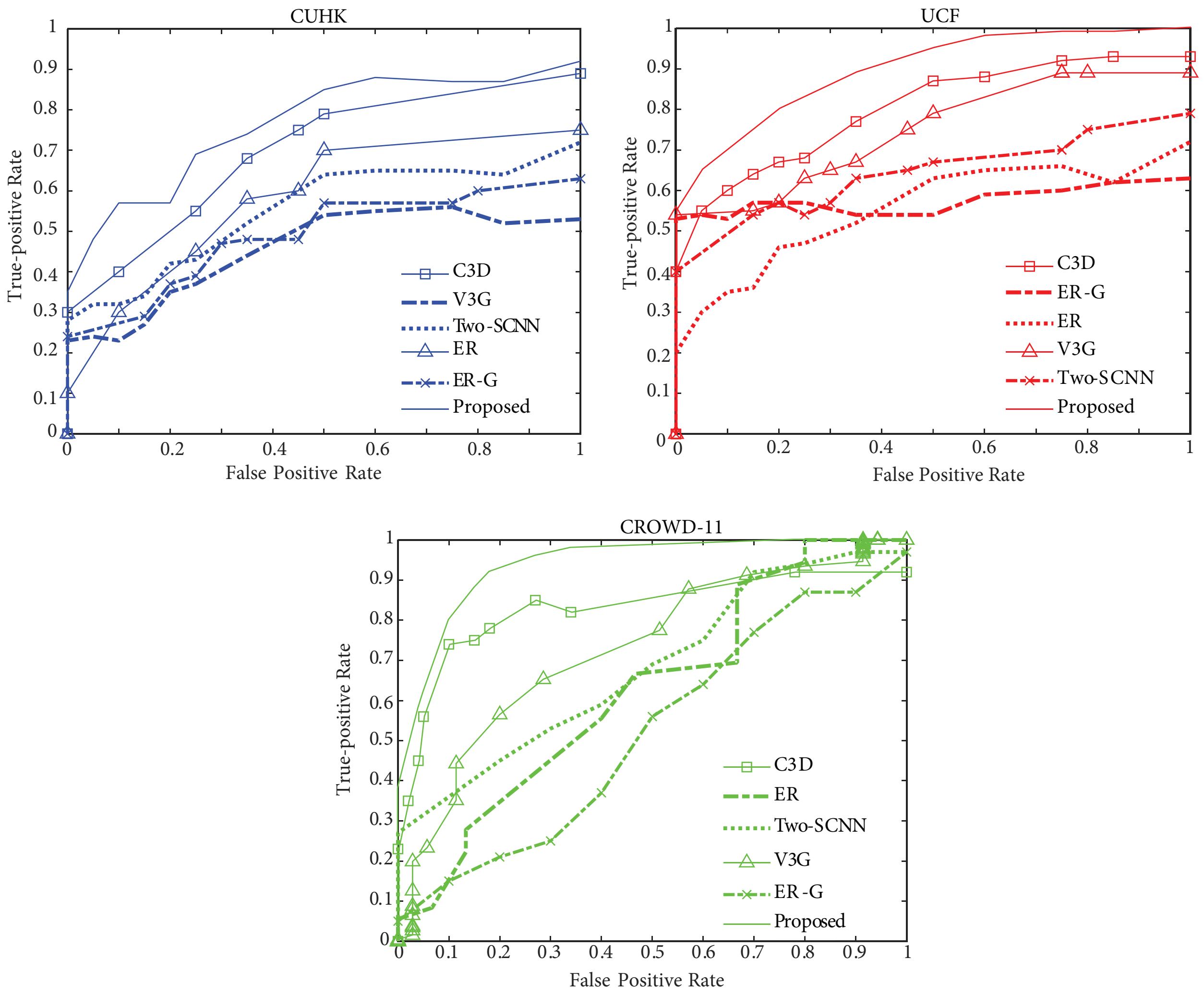


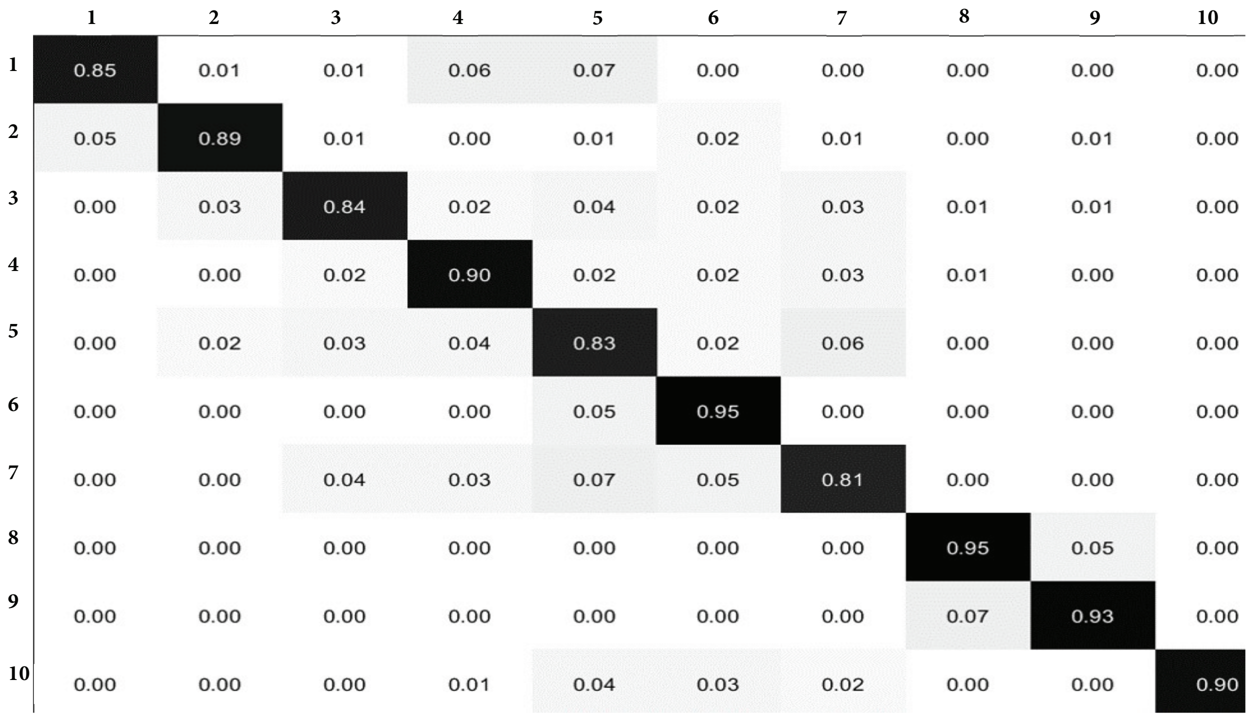
Figure 3. ROC curves of different methods. Our proposed method outperforms other state-of-the-art methods by a significant margin.

ROC provides a more detailed evaluation than fixed threshold evaluation metrics. One of the limitations of fixed threshold metrics is that they do not provide an overview of the range of performance with varying the thresholds. Although using a fixed threshold divides the given dataset into positive and negative classes and it may be reasonable for some particular applications, it is difficult to find the correct value of the threshold. The alternative powerful solution is to use threshold-free measures, i.e. ROC.

ROC curve is graphical plot between true-positive rate (TPR) and falsepositive rate (FPR). The graph is plotted between TPR and FPR at different thresholds. (TPR) is computed as $\frac{TP}{TP+FN}$ and (FPR) is measured as $\frac{FP}{FP+TN}$. True-positive (TP) represents the number of correctly identified behaviors. False-negative represents the count of incorrectly identified behaviors. We compute the ROC for all the methods and for the all datasets and the results are shown in Figure 3. From Figure 3, it is obvious that our proposed framework outperforms other reference methods by a significant margin.

We observed from the experiments that the proposed model is robust in Figure 4. In Figure 4, we evaluate the discriminating power of the proposed method.

For this purpose, we used Crowd-11 dataset, since it contains more realistic behaviors compared to the other datasets. We trained the proposed method independently for each behavior and tested the model on other behaviors and report results in the form of confusion matrix in Figure 4.



	1	2	3	4	5	6	7	8	9	10
1	0.85	0.01	0.01	0.06	0.07	0.00	0.00	0.00	0.00	0.00
2	0.05	0.89	0.01	0.00	0.01	0.02	0.01	0.00	0.01	0.00
3	0.00	0.03	0.84	0.02	0.04	0.02	0.03	0.01	0.01	0.00
4	0.00	0.00	0.02	0.90	0.02	0.02	0.03	0.01	0.00	0.00
5	0.00	0.02	0.03	0.04	0.83	0.02	0.06	0.00	0.00	0.00
6	0.00	0.00	0.00	0.00	0.05	0.95	0.00	0.00	0.00	0.00
7	0.00	0.00	0.04	0.03	0.07	0.05	0.81	0.00	0.00	0.00
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.95	0.05	0.00
9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.93	0.00
10	0.00	0.00	0.00	0.01	0.04	0.03	0.02	0.00	0.00	0.90

Figure 4. Confusion matrix of our proposed method on Crowd-11 dataset. The classes are labeled as 1: Gas free, 2: Gas jammed, 3: laminar flow, 4: turbulent flow, 5: crossing flow, 6: merging flow, 7: diverging flow, 8: static calm, 9: static agitated, 10: interacting crowd.

From Figure 4, it is clear that the proposed method can effectively discriminate different behaviors. For example, the first row of Figure 4 shows the performance of the proposed method trained on samples of 1:Gas free behavior. As obvious from the first row of Figure 4, the model discriminates gas free behavior from other behaviors. The model (trained on gas free samples) yields a high score when tested on sample from the same class and produces lower values when tested on other behaviors. Furthermore, we also observed that proposed model can discriminate the behaviors that involve motion from the behaviors that involve no motion, for example static calm and static agitated.

We also report qualitative results of the proposed method in Figure 5. From the figure, it is obvious that the proposed framework precisely identifies the crowd behaviors in given complex crowd scenes. The superior performance of our proposed model can be attributed to the better representation of spatio-temporal features embedded in trajectory images.

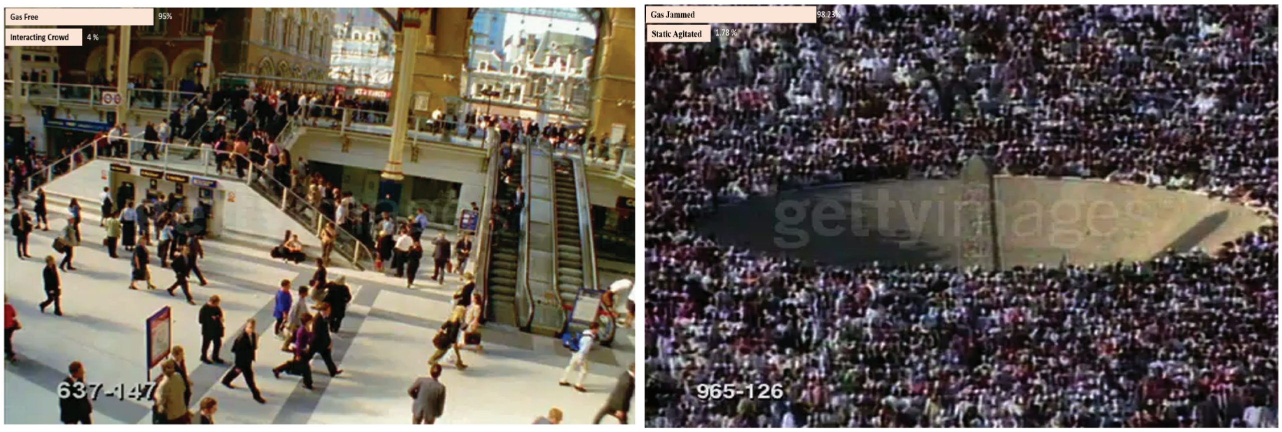


Figure 5. Qualitative results of our proposed approach. Our network classifies and assigns top two class probabilities to each input video. (best view in zoom)

6. Conclusion

In this paper, we proposed an effective framework for characterizing motion behaviors in complex scenes. We extract motion information from videos using point trajectories. These trajectories are then projected on a 2D plane to generate TIs. The TIs are then used to train the CNN model. Our approach achieves a state-of-the-art performance on all benchmark datasets, and beats the existing methods by a considerable margin.

In this paper, we proved the significance of modeling long-term motion using TIs. The current method can be improved further by incorporating an attention module that will process important frames instead of processing all the frames of a video. This strategy will boost the speed and will enable the current framework to be applied in real-time surveillance setup. Furthermore, we will train the framework on a large dataset.

7. Acknowledgment

This research was supported by the Deanship of Scientific Research at the University of Hail - Kingdom of Saudi Arabia with the number “BA-1919”.

References

[1] Ullah H, Altamimi AB, Uzair M, Ullah M. Anomalous entities detection and localization in pedestrian flows. Neurocomputing 2018; 290: 74-86.

[2] Kratz L, Nishino K. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition; Miami, USA; 2009. pp. 1446-1453.

[3] Mahadevan V, Li W, Bhalodia V, Vasconcelos N. Anomaly detection in crowded scenes. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; New York, USA; 2010. pp. 1975-1981.

[4] Ullah H, Ullah M, Conci N. Real-time anomaly detection in dense crowded scenes. In: Video Surveillance and Transportation Imaging Applications 2014. vol. 9026. International Society for Optics and Photonics; 2014. pp.35 902608.

[5] Ullah H, Tenuti L, Conci N. Gaussian mixtures for anomaly detection in crowded scenes. In: Video Surveillance and Transportation Imaging Applications. vol. 8663. International Society for Optics and Photonics; 2013.

[6] Ullah H, Ullah M, Afridi H, Conci N, De Natale FG. Traffic accident detection through a hydrodynamic lens. In: Image Processing (ICIP), 2015 IEEE International Conference; Quebec, Canada; 2015. pp. 2470-2474.

[7] Rabaud V, Belongie S. Counting crowded moving objects. In: Computer Vision and Pattern Recognition; New York, USA; 2016. pp. 705-711.

[8] Idrees H, Saleemi I, Seibert C, Shah M. Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Portland, Oregon, USA; 2013. pp. 2547-2554.

[9] Arif M, Daud S, Basalamah S. Counting of people in the extremely dense crowd using genetic algorithm and blobs counting. IAES International Journal of Artificial Intelligence 2013; 2(2): 51.

[10] Arif M, Daud S, Basalamah S. People counting in extremely dense crowd using blob size optimization. Life Science Journal 2012;9(3): pp. 1663-1673.

[11] Saqib M, Khan SD, Blumenstein M. Texture-based feature mining for crowd density estimation: A study. In: International Conference on Image and Vision Computing; Auckland, New Zealand; 2016. pp. 1-6.

[12] Khan S, Vizzari G, Bandini S, Basalamah S. Detecting dominant motion flows and people counting in high density crowds. Journal of WSCG 2014; 22(1): 21-30.

[13] Marsden M, McGuinness K, Little S, O'Connor NE. ResnetCrowd: A residual deep learning architecture for crowd counting, violent behaviour detection and crowd density level classification. In: 14th IEEE International Conference Advanced Video and Signal Based Surveillance; Lecce, Italy; 2017. pp. 1-7.

[14] Khan SD, Vizzari G, Bandini S. Identifying sources and sinks and detecting dominant motion patterns in crowds. Transportation Research Procedia 2014; 1: 195-200.

[15] Saqib M, Khan SD, Sharma N, Blumenstein M. Extracting descriptive motion information from crowd scenes. In: International Conference on Image and Vision Computing New Zealand; Auckland, New Zealand; 2017. pp. 1-6.

[16] Saqib M, Khan SD, Blumenstein M. Detecting dominant motion patterns in crowds of pedestrians. In: Eighth International Conference on Graphic and Image Processing; Qingdao, China; 2017. pp. 1-6.

[17] Wang H, Klaser A, Schmid C, Liu CL. Action recognition by dense trajectories. In: International Conference on Computer Vision and Pattern Recognition; Colorado, USA; 2011. pp. 3169-3176.

[18] Wang H, Schmid C. Action recognition with improved trajectories. In: Proceedings of the IEEE International Conference on Computer Vision; Portland, Oregon, USA; 2013. pp. 3551-3558.

[19] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems; Montreal, Canada; 2014. pp. 568-576.

[20] Donahue J, Anne Hendricks L, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K et al. Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Boston, USA; 2015. pp. 2625-2634.

[21] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: 2005 IEEE Conference on Computer Vision and Pattern Recognition; San Diego, USA; 2005. pp. 886-893.

[22] Chaudhry R, Ravichandran A, Hager G, Vidal R. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition; Florida, USA; 2009. pp. 1932-1939.

[23] Zhou Z, Shi F, Wu W. Learning spatial and temporal extents of human actions for action detection. IEEE Transactions on Multimedia 2015;17(4): 512-525.

[24] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Las Vegas, USA; 2016. pp. 770-778.

[25] Wu Z, Wang X, Jiang YG, Ye H, Xue X. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In: Proceedings of the 23rd ACM International Conference on Multimedia; Brisbane Australia; 2015. pp. 461-470.

[26] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems; Montreal, Canada; 2014. pp. 3104-3112.

[27] Wang L, Qiao Y, Tang X. Action recognition with trajectory-pooled deep-convolutional descriptors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Boston, USA; 2015. pp. 4305-4314.

[28] Ji S, Xu W, Yang M, Yu K. 3D convolutional neural networks for human action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 2012; 35(1): 221-231.

[29] Hasan M, Roy-Chowdhury AK. A continuous learning framework for activity recognition using deep hybrid feature models. IEEE Transactions on Multimedia 2015; 17(11): 1909-1922.

[30] Solmaz B, Moore BE, Shah M. Identifying behaviors in crowd scenes using stability analysis for dynamical systems. IEEE Transactions on Pattern Analysis and Machine Intelligence 2012; 34(10): 2064-2070.

[31] Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R et al. Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition; Ohio, USA; 2014. pp. 1725-1732.

[32] Dupont C, Tobias L, Luvison B. Crowd-11: A dataset for fine grained crowd behaviour analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops; Hawaii, USA; 2017. pp. 9-16.

[33] Zhou B, Tang X, Wang X. Measuring crowd collectiveness. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Portland, Oregon; 2013. pp. 3049-3056.

[34] Wang W, Lin W, Chen Y, Wu J, Wang J, Sheng B. Finding coherent motions and semantic regions in crowd scenes: A diffusion and clustering approach. In: European Conference on Computer Vision; Zurich, Switzerland; 2014. pp. 756-771.

[35] Wu S, San Wong H. Crowd motion partitioning in a scattered motion field. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 2012; 42(5):1443-1454.

[36] Ali S, Shah M. A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition; Minneapolis, USA; 2007. pp. 1-6.

[37] Saleemi I, Hartung L, Shah M. Scene understanding by statistical modeling of motion patterns. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; San Francesco, USA; 2010. pp. 2069-2076.

[38] Li T, Chang H, Wang M, Ni B, Hong R et al. Crowded scene analysis: A survey. IEEE Transactions on Circuits and Systems for Video Technology 2014; 25(3): 367-386.

[39] Andrade EL, Blunsden S, Fisher RB. Modelling crowd scenes for event detection. In: 18th International Conference on Pattern Recognition; Hong Kong, China; 2006. pp. 175-178.

[40] Hu M, Ali S, Shah M. Learning motion patterns in crowded scenes using motion flow field. In: 2008 19th International Conference on Pattern Recognition; Tampa, Florida; 2008. pp. 1-5.

[41] Sultani W, Chen C, Shah M. Real-world anomaly detection in surveillance videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Utah, USA; 2018. pp. 6479-6488.

[42] Widhalm P, Brandle N. Learning major pedestrian flows in crowded scenes. In: 2010 20th International Conference on Pattern Recognition; Istanbul, Turkey; 2010. pp. 4064-4067.

[43] Li R, Chellappa R. Group motion segmentation using a spatio-temporal driving force model. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; San Francisco, USA; 2010. pp. 2038-2045.

[44] Rao AR, Jain RC. Computerized flow field analysis: Oriented texture fields. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1992;(7): 693-709.

[45] Ford RM. Critical point detection in fluid flow images using dynamical system properties. Pattern Recognition 1997; 30(12): 1991-2000.

[46] Helbing D. A fluid dynamic model for the movement of pedestrians. arXiv preprint cond-mat/9805213. 1998.

[47] Mehran R, Oyama A, Shah M. Abnormal crowd behavior detection using social force model. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition; Florida, USA; 2009. pp. 935-942

[48] Wu S, Moore BE, Shah M. Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; San Francisco, USA; 2010. pp. 2054-2060.

[49] Shao J, Change Loy C, Wang X. Scene-independent group profiling in crowd. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Ohio, USA; 2014. pp. 2219-2226.

[50] Yi S, Li H, Wang X. Understanding pedestrian behaviors from stationary crowd groups. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Boston, USA; 2015. pp. 3488-3496.

[51] Shao J, Kang K, Change Loy C, Wang X. Deeply learned attributes for crowded scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Boston, USA; 2015. pp. 4657-4666.

[52] Shao J, Loy CC, Kang K, Wang X. Slicing convolutional neural network for crowd video understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Las Vegas, USA; 2016. pp. 5620-5628.

[53] Zhang X, Shu X, He Z. Crowd panic state detection using entropy of the distribution of enthalpy. Physica A: Statistical Mechanics and its Applications 2019; 525: 935-945.

[54] Bera A, Kim S, Manocha D. Realtime anomaly detection using trajectory-level crowd behavior learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops; Las Vegas, USA; 2016. pp. 50-57.

[55] Hassner T, Itcher Y, Kliper-Gross O. Violent flows: Real-time detection of violent crowd behavior. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops; Rhode Island, USA; 2012. pp. 1-6.

[56] Fradi H, Luvison B, Pham QC. Crowd behavior analysis using local mid-level visual descriptors. IEEE Transactions on Circuits and Systems for Video Technology 2016; 27(3): 589-602.

[57] Kim JS, Hwangbo M, Kanade T. Realtime affine-photometric KLT feature tracker on GPU in CUDA framework. In: 2009 IEEE 12th International Conference on Computer Vision Workshops; Kyoto, Japan; 2009. pp. 886-893.

[58] Liu C, Yuen J, Torralba A, Sivic J, Freeman WT. Sift flow: Dense correspondence across different scenes. In: European Conference on Computer Vision; Marseille, France; 2008. pp. 28-42.

[59] Wang H, Kl¨aser A, Schmid C, Liu CL. Dense trajectories and motion boundary descriptors for action recognition. International Journal of Computer Vision. 2013; 103(1): 60-79.

[60] Zhang C, Li H, Wang X, Yang X. Cross-scene crowd counting via deep convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Boston, USA; 2015. pp. 833-841.

[61] Allain P, Courty N, Corpetti T. AGORASET: a dataset for crowd video analysis. In: ICPR International Workshop on Pattern Recognition and Crowd Analysis; Tsukuba, Japan; 2012. pp. 1-6.

[62] Patino L, Cane T, Vallee A, Ferryman J. Pets 2016: Dataset and challenge. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops; Las Vegas, USA; 2000. pp. 1-8.

[63] Nievas EB, Suarez OD, Garcia GB, Sukthankar R. Violence detection in video using computer vision techniques. In: International Conference on Computer Analysis of Images and Patterns; Seville, Spain; 2011. pp. 332-339.

[64] Tran D, Bourdev L, Fergus R, Torresani L, Paluri M. Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision; Santiago, Chile; 2015. pp. 4489-4497.

[65] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:150203167. 2015.