



# Real-time crowd behavior recognition in surveillance videos based on deep learning methods

Fariba Rezaei<sup>1</sup> · Mehran Yazdi<sup>1</sup>

Received: 28 January 2021 / Accepted: 17 April 2021 / Published online: 3 May 2021  
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

## Abstract

Automatic video surveillance in public crowded places has been an active research area for security purposes. Traditional approaches try to solve the crowd behavior recognition task using a sequential two-stage pipeline as low-level feature extraction and classification. Lately, deep learning has shown promising results in comparison to traditional methods by extracting high-level representation and solving the problem in an end-to-end pipeline. In this paper, we investigate a deep architecture for crowd event recognition to detect seven behavior categories in PETS2009 event recognition dataset. More especially, we apply an integrated handcrafted and Conv-LSTM-AE method with optical flow images as input to extract a high-level representation of data and conduct classification. After achieving a latent representation of input optical flow image sequences in the bottleneck of autoencoder(AE), the architecture is split into two separate branches, one as AE decoder and the other as the classifier. The proposed architecture is jointly trained for representation and classification by defining two different losses. The experimental results in comparison to the state-of-the-art methods demonstrate that our algorithm can be promising for real-time event recognition and achieves a better performance in calculated metrics.

**Keywords** Crowd behavior recognition · Deep learning · PETS2009 dataset · CONV-LSTM-AE

## 1 Introduction

Currently, the study of crowd behavior in public places has been one of the interesting topics for crowd safety. As a huge number of CCTV cameras have been installed in crowded places for video surveillance, human-based detection of potentially dangerous situations may probably cause a high-risk errors. Therefore, research attention is towards automatic surveillance. Generally, crowd behavior analysis can be classified into three broad categories: crowd counting, anomaly detection, and behavior recognition. To perform crowd behavior analysis, video signals should be analyzed both spatially and temporally. Generally, crowd behavior recognition can be done through three main category approaches Holistic, Object-based, and hybrid methods [1]. Holistic methods [2] consider the scene as a single entity

instead of targeting an individual's behavior. So the main focus will be the extraction of low/medium level attributes such as optical flow fields or spatio-temporal patch-based features. Object-based methods [3] consider the crowd as an aggregation of several individuals and rely on detection and tracking of persons. The crowd behaviors are inferred by analyzing extracted trajectories. Due to challenges such as occlusion, for the large number of human beings in the scene, these methods will encounter some difficulties. Hybrid methods take advantages of both holistic and object-based methods. Most previous researches investigated extracting low-level features from the appearance or/and from the motion of the scene and subsequently applied various classifiers based on the task in hand and the extracted attributes. Recently the successful emergence of deep neural networks has persuaded researchers to employ various deep architectures to make progress in their researches. In this paper, we investigate the application of a deep learning technique to recognize different crowd behaviors happening in the scene captured by a video camera. We will evaluate our proposed architecture on PETS2009 crowd video dataset [4, 5]. PETS2009 S3- HL Event Recognition is one of the

✉ Mehran Yazdi  
yazdi@shirazu.ac.ir  
Fariba Rezaei  
rezaei.f.66@gmail.com

<sup>1</sup> School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran

well-known datasets in this domain which consists of seven crowd events defined as follows:

1. Walk: crowd moving at low speed (w.r.t a threshold).
2. Run: crowd moving at high speed (w.r.t a threshold).
3. Evacuation: rapidly dispersing crowd, moving in different directions. Attributes such as direction and crowd density would be helpful to recognize this event.
4. Split: localised movement of people within a crowd away from a given panic situation.
5. Merge: formation of a crowd of individuals, through convergence from multiple directions. Crowd density and the distance between the principal directions are significant features to discriminate this event
6. Dispersion: cohesive crowd splitting into two or more entities (multiple diverging flows).
7. Loitering: the static appearance of a crowd in the scenes with a little fluctuation.

The contribution of our work is as follows:

- I We propose an integrated handcrafted and deep architecture for the classification of seven crowd behaviors by extracting optical flow images and autoencoder features calculated on the sequence of optical flow images. More specially, we propose CONV-LSTM-AE architecture together with global average pooling (GAP) layer, where instead of raw frames the input is volume of OF images.
- II We also apply a new training strategy by integrating two different losses in our proposed one-input-two-output architecture. In other words, training is jointly done to achieve high-level representation and classification. The high-level feature representation is attained in AE bottleneck and classification is obtained in one output branch of the architecture.

Our proposed architecture takes advantages of three common blocks in deep learning, namely convolution, LSTM and autoencoder. Moreover, extracting motion-based handcrafted features to use as input for CONV-LSTM-AE help to emphasize on motion. single input multiple output architecture will also help saving time for feature extraction and classification. We evaluate our proposed method with respect to common metrics in classification such as confusion matrix, precision, recall, F1-score, accuracy, and Dice Score. Also, time complexity of our proposed approach is investigated for real-time computation. Experimental results shows comparable performance of our proposed algorithm in comparison to previous approaches.

The remainder of the paper is organized as follows: in Sect. 2, we present related works on crowd behavior analysis. Our proposed deep learning architecture is introduced

in Sect. 3. A detailed evaluation of our work is followed in Sect. 4. Finally, we conclude and make a suggestion for possible future works in Sect. 5.

## 2 Related work

When it comes to the crowd analysis as a subdomain problem of video analysis, both spatial and temporal information should be considered to boost performance. Hence, most researches in this area get involved with the extraction of visual features from the spatial domain [6–8] or consider short-term or long-term dynamic features through regular motion patterns [9–13]. There have been also some approaches that jointly consider spatio-temporal features. In [14], a dynamic texture model was used to represent holistic motion flow in the video crowd. In [15], low-level color, texture, and shape features were extracted followed by multi-layer perceptron classifiers per event. Benabbas et al. [16] used both direction and magnitude of optical flow vectors to model the global motion of the scene. They applied a region-based segmentation to group local blocks based on the similarity in motion direction and speed. They detected crowd events, such as walk, run, merge, split, dispersion, and evacuation by comparing these motion patterns against the learned models. Concatenation of three histograms extracted from local density, speed, and flow direction were employed in [9, 10] for crowd representation.

Briassouli et al. [17] applied the Fourier transform for modeling random crowd motion without optical flow estimation, parametric modeling, or extensive training. Real-time locally/globally detection of dynamic changes was achieved through statistical CUSUM method [18]. Mehran et al. [19] proposed social force model (SFM) for the detection of abnormal behaviors in crowds by considering interaction forces between pedestrians. Combination of social force graph and streak flow [20] attributes were proposed in [21] to capture the global spatiotemporal changes as well as the local motion of crowd video. Histograms of motion direction alongside an indication of motion speed [22] and histogram of oriented tracklets [23] was investigated for recognizing abnormal situations in crowded scenes. A combination of streakline based on fluid mechanics and a high-accurate variational optical flow model was proposed in [24] for crowd behavior identification.

The high-dimensional signal property, makes the low-dimensional embedding techniques promising for computational complexity purposes [11, 25]. There are linear and nonlinear methods for this purpose. Linear PCA and nonlinear ISOMAP, MDS, and AutoEncoder(AE) [26] to name a few. In [11], a combination of ISOMAP on texture features and SVM was applied for event recognition. A manifold learning algorithm with temporal constraints was proposed

in [25] to embed video signal to accurately reduce the data dimensions and preserve spatial–temporal content of the video. Video trajectory in the manifold space was also used for recognition.

In contrast to previous particle flow-based methods, [27] discussed group level representation of crowd. By clustering particle trajectories and group formation, they connect nodes in each group as a trajectory graph and at the high level, a bag of trajectory graph (BoTG) as a global feature of the scene clip was extracted. Three informative features are graph structure, group attribute and dynamic motion encode the graph. Delaunay triangulation was employed in [28] to approximate neighborhood interactions on an evolving crowd graph constructed from tracklets as nodes to extract various mid-level representations. Rao et al. [29] proposed a probabilistic detection framework of crowd events based on Riemannian manifolds on optical flow. A robust and effective spatio-temporal viscous fluid field was proposed in [30] to investigate appearance and interaction among pedestrians and thereby model crowd motion patterns. Stability analysis for dynamical systems was also proposed for identifying five crowd behaviors as bottlenecks, fountainheads, lanes, arches, and blocking in visual scenes, without the need for object detection, tracking, or training [31]. Linear approximation of the dynamical system provides behavior analysis through the Jacobian matrix. To consider long-term temporal sequences and compensate the imperfect effect of camera motion, an improved dense trajectory was employed in [32]. HOG and MBH features were extracted along the  $L$  length dense trajectory and one-vs-rest linear classifier was trained for modeling the events. Shuaibu [33] proposed a novel spatio-temporal dictionary learning-based sparse coding representation with  $k$ -means SVD for robust classification of crowd behaviors. In [34], the combination of optical flow and spatio-temporal methods was utilized for crowd analysis. Flow fields in spatio-temporal elements were considered as 2D distribution parameterizing by Mixture of Gaussian. To initialize the mixture model, they applied  $K$ -means, and to optimize the model, they applied EM. Then, a conditional random field was learned for classification.

To sum up, traditional crowd analysis, constitute up of two separate stages: low or mid-level feature engineering

$$A = f(I) \quad (1)$$

and model learning

$$y = g(A, \theta) \quad (2)$$

where  $I$  is the input visual data (either 2D or 3D),  $A$  is features extracted from  $I$ ,  $\theta$  is the model parameters,  $g$  is the chosen classifier and  $y$  is the predicted class label.

However, recently deep learning (DL) has become the turning point of researches in the machine learning

domain, and computer vision as well. In comparison to traditional ML, DL considers the whole process in one unique stage:

$$y = G(I, \theta) \quad (3)$$

where  $G$  acts as both hierarchical level feature extractor and classifier. Supervised deep learning algorithms, act as a function approximation through a given training set where backpropagation is used within the network to estimate the weights. After the successful emergence of deep learning for image analysis, investigations around its performance on video signals embarked. Both spatial and temporal information of video contains useful features for the task at hand. Several researches tried to take into account the temporal information in addition to spatial information [35–45]. Karpathy et al. [36] investigated operating CNN to both individual video frames and stack of frames and discussed the effect of temporal data at different fusion levels. Three-dimensional CNN was introduced in [37] for action recognition, where feature extraction is performed through 3D convolution to capture simultaneously both spatial and temporal information, which suffer from large number of parameters tuning in comparison to 2D CNN. Simonyan et al. [39] proposed two stream architecture which was discriminately trained on still frames and stack of optical flow frames, combining them at a higher level, resulted in competitive extracted features in comparison to the state-of-the-art hand-crafted features.

Meanwhile, [38] and [45] investigated the power of two streams for crowd anomaly detection. Wei et al. [38] employed fully convolutional neural networks (FCN), instead of CNN with fully connected layer in original framework [39]. FCNs were pre-trained on ImageNet to facilitate the weight updating procedure with fewer parameters. Then output feature maps from FCN were used to compute the abnormal coefficient for each frame. In [45], a two-stream residual network (TSRN) was proposed to aggregate appearance and motion features where motion stream was generated from three scene-independent motion maps: collectiveness, stability, and conflict. In [46], the combination of neural network and traditional statistical classification approaches were investigated and resulted in better accuracies. Wang, L et al. [12] combined the advantages of deep learning and traditional methods for video representation. In particular, they learned discriminative convolutional feature maps followed by trajectory constrained pooling to aggregate convolutional features into more effective descriptors. Combination of corner optical flow and CNN [13], Combination of tracklets and DBN [47], and combination of tracklets/trajectories with CNN and LSTM [48] were also investigated and have been used for crowd event recognition. Zhuang

et al. [49] proposed an end-to-end deep architecture, convolutional DLSTM (ConvDLSTM), for crowd analysis. ConvDLSTM is comprised of GoogleNet Inception V3 CNN and stacked differential long short-term memory (LSTM) with raw image sequence as the input. Application of unsupervised learning for crowds has been studied in [50, 51]. Erfani et al. [50], proposed a hybrid model of DBN for dimensionally reduction and one-class SVM for detection. Chong et al [51] trained a deep pipeline consisting of Convolution and Convolutional LSTM to extract both spatial and temporal information. At the middle of the pipeline, they achieved to a low-dimensional representation of input signals. They argued that ConvLSTM layer preserves the advantages of FC-LSTM, and is suitable for spatiotemporal data.

Recently, a kernel-based relevance analysis was proposed in [52] for social behavior recognition consisting of a feature ranking based on centered kernel alignment and a classification stage to perform the behavior prediction. Deng et al [53], considered video-based crowd behavior recognition as a multi-label classification task with imbalanced samples issue and tackled it by proposing a classifier based on associative subspace. Crowd psychology was investigated for predicting crowd behaviors in [54] and determining nine diverse crowd behaviors. The approach was a combination of two cognitive deep learning frameworks and a psychological fuzzy computational model.

A bidirectional recurrent prediction model with a semantic aware attention mechanism was proposed in [55] to explore the spatio-temporal features and semantic relations between attributes. The ConvLSTM was introduced for feature representation to capture the spatio-temporal structure of crowd videos and facilitate visual attention. Then, a bidirectional recurrent attention module was proposed for sequential attribute prediction by associating each subcategory attributes to corresponding semantic-related regions iteratively.

A comprehensive survey of convolution neural network based methods for crowd behaviour analysis was studied in [56] with various topics of optimization methods, architectures, temporal dimension considerations, etc.

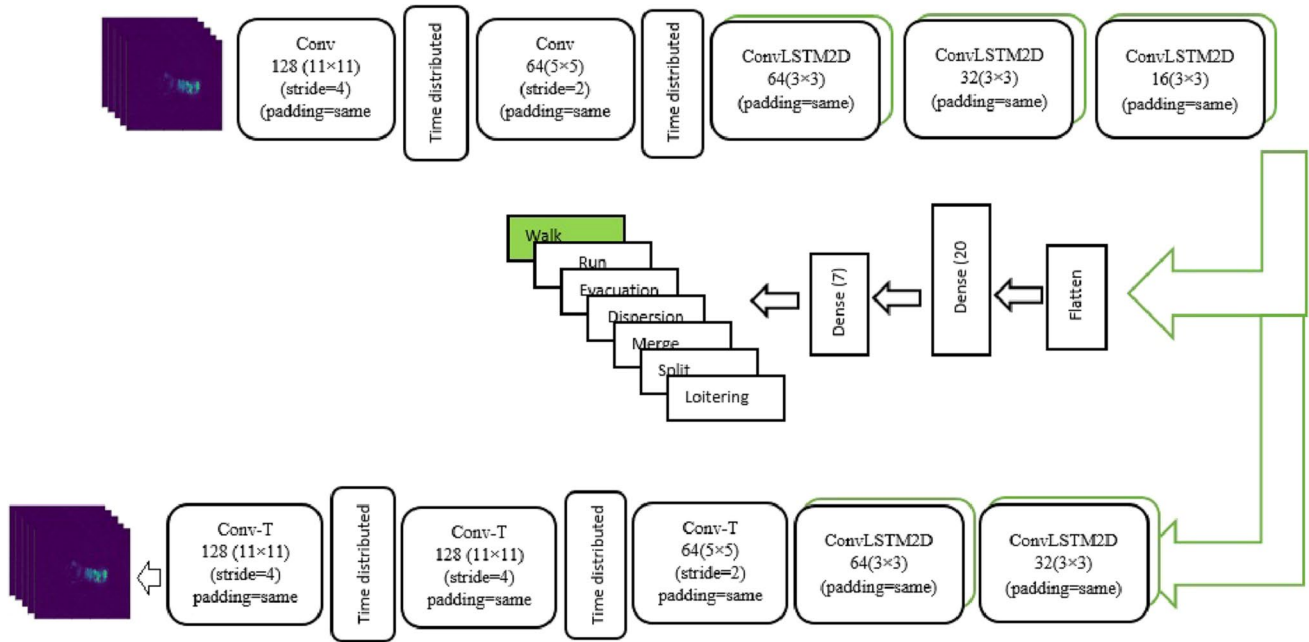
There have been significant researches about anomaly detection using deep learning and auto-encoders [57]. Since anomaly detection can be thought of as outlier detection, AE can be employed to find those signals whose reconstruction errors have a high difference w.r.t the input signal as an anomaly. Crowd event detection however is thought of as multi-class classification. To apply auto-encoders for multi-class cases, one can establish several AE architectures same as the count of the classes, and evaluate the reconstruction error per AE to decide on the type of events. But it may be time-consuming. In this paper, we investigate a deep architecture for this multiclass classification problem.

### 3 Proposed approach

An illustration of our proposed architecture for crowd behavior recognition is shown in Fig. 1. It has two main stages: Preprocessing and deep Conv-LSTM-AE architecture, which are discussed in the following.

*Preprocessing* In this paper, we evaluate our proposed architecture for crowd event recognition on PETS2009 dataset. At first, we resize the original frames to square smaller frames for reducing computation complexity and making them in appropriate size as the input for our deep architecture. Also, we convert the colored images to gray-scale since we believe that it is not a significant attribute in color for recognition. Besides, we normalize the frame intensity from [0–255] to [0–1]. Motion between neighbor frames is estimated through Farneback optical flow [58] method. We chose Farneback OF since it is more robust to noise than the other basic optical flow approaches. We consider dense optical flow estimation to calculate the motion for each pixel of the images. Then, we take the magnitude of OF images. As the optical flow computes only the motion between two consecutive frames, to take into account a bit longer period of motion, we consider concatenation of  $T$  OF magnitude images (which is calculated from  $T + 1$  consecutive original frames). Then, these short-term sequences of OF images are employed as the input to a deep neural network architecture at the subsequent stage.

*Conv-LSTM-AE architecture* Our deep learning structure takes advantage of three types of deep networks. First, as the state-of-the-art network design for vision tasks is CNN, we utilize convolutional layers to preserve spatial information. Due to the shared layers in CNN, very few parameters should be tuned in comparison to a fully connected network. Second, since we are dealing with video signals captured by high-resolution cameras, the actual data size is very large. Therefore, dimensional reduction techniques can help us to come up with this challenge. Auto-encoder (AE) will be used for compressing the input volumes to a much lower representation. Indeed, AE is generally used as an unsupervised approach, with the same inputs and outputs, and aims at extracting principal hidden information through minimizing reconstruction error between input and output. Pure AE is trained to learn an identity function through nonlinear transformations. The bottleneck layer in AE also gives a suitable feature representation. It has been largely used for anomaly detection as we mentioned in Sect. 2. One way to utilize it for classification purposes is to learn encoder and decoder weights of AE in end-to-end training, then throw away the decoder part and train it again in a supervised way by dense layer with softmax activation function or any



**Fig. 1** Proposed architecture for crowd event recognition. Input to the structure is a sequence of five optical flow images. After a series of layers and producing latent representation of the input, at the bottle-

neck, there will be two separate branches, the top for making classification and the bottom for input reconstruction. The network is trained jointly

other classic classifier such as SVM, decision trees, etc. We will use the latent representation with a dense layer for classification. Third, since we deal with time sequence signals, the RNN scheme could be beneficial. Here, we apply LSTM to cope with vanishing or exploding gradients problems. In particular, we utilize convolutional LSTM to take advantage of both convolutional and recurrent networks. In contrast to the original LSTM, which uses a fully connected layer, in ConvLSTM [59], convolution operation is applied which results in fewer parameters. The formulation of ConvLSTM can be summarized as follows:

$$\begin{aligned}
 f_t &= \sigma(W_f \times [h_{t-1}, x_t, C_{t-1}] + b_f) \\
 i_t &= \sigma(W_i \times [h_{t-1}, x_t, C_{t-1}] + b_i) \\
 \hat{C}_t &= \tanh(W_C \times [h_{t-1}, x_t] + b_C) \\
 C_t &= f_t \otimes C_{t-1} + i_t \otimes \hat{C}_t \\
 o_t &= \sigma(W_o \times [h_{t-1}, x_t, C_{t-1}] + b_o) \\
 h_t &= o_t \otimes \tanh(C_t)
 \end{aligned}
 \tag{4}$$

where,  $i_t, f_t, C_t, W(W_{f,i,C}), b_t, h_t, o_t$  are defined as input vector, forget gate, cell state, Trainable weight matrices bias, hidden state and output vector at time t, respectively.  $\sigma$  is a nonlinear activation function. Symbol  $\otimes$  denotes Hadamard product. Images are fed into the network and the weights for each connection replaced to convolution filters. The pseudo-code for our proposed approach is shown in Algorithm 1.

**Algorithm 1** Pseudo code for event recognition

- 1: **procedure** DETECT EVENT FOR FRAME  $i(state)$
- 2: split dataset to train and test set
- 3: **for** each Frame  $f_i$  **do**
- 4: calculate OF between  $f_i$  and  $f_{i-1}$  as  $OF_i$
- 5: calculate OF magnitude as  $OF_i^{mag}$
- 6: stack OF mag images  $OF_{i-T}^{mag} : OF_i^{mag}$  as  $OF_i^{clip}$
- 7: **for** training set as the number of epoches **do**
- 8: initialize network parameters
- 9: Input batches of  $OF_i^{clip}$  to ConvLSTM2D (Fig 1)
- 10: Train to Min MSE and categorical cross-entropy
- 11: Save the Model and Parameters
- 12: **for** test set **do**
- 13: Input  $OF_i^{clip}$  in Test Set
- 14: pass it through the network
- 15: compute Max Prob of 7 classes for  $OF_i^{clip}$  as predicted label for frame  $f_i$
- 16: **Return** Metrics on test data

In contrast to the CNN framework, where step by step we enlarge the number of filters and generate feature maps, to have a more powerful model in capturing more features, first we reduce the number of filters up to latent layer representation, and afterward like a mirror, we increase it again. Instead of throwing away the decoder part and training the network again for classification purposes, we split the architecture from the latent representation layer and append

a new branch for multiclass classification. Therefore, we will have a single-input-two-outputs architecture that could be jointly trained by hybrid supervised and unsupervised regimes. In other words, reconstruction of input is conducted unsupervised to get a high-level feature representation at the bottleneck of AE in one branch, and classification is done supervisedly through provided ground truth labels by dense layer with Softmax activation layer in the other branch. One of the outputs will be trained to reconstruct the input signal, the other will be trained for classification. We integrate two losses for training our two-branched network, i.e. mean square error and categorical cross-entropy. Since classification is more important to us, we consider a higher value for the weight of the classification loss.

$$\text{Loss} = w_1 \sum_i (X_i - \hat{X}_i)^2 - w_2 \sum_i (y_i \log C(X_i)), \quad (5)$$

where  $X_i, \hat{X}_i, y_i, C(X_i)$  show input clip, reconstructed input clip, output label and predicted class label for frame  $i$ , respectively.  $w_1$  and  $w_2$  are the weights we defined for two losses ( $w_2 > w_1$ ). Jointly training the network will result in less computational time. We target to detect seven crowd events in PETS2009. If our purpose was just to detect three events of walking, running, and loitering, since the most important feature is related to the speed, the corresponding features would be extracted just from OF between two adjacent frames. However, we more aim at the recognition of some events like dispersion, evacuation, splitting, and merging, so a longer time window should be considered instead of just two frames. Therefore, we input a sequence of  $T$  frames in our proposed architecture. To capture both spatial and temporal features, we choose ConvLSTM [60].

### 3.1 Real-time design

Here we analyze the processing time complexity of our proposed approach. Since our method is an integration of handcrafted optical flow computation and a deep architecture, the processing time of these two parts should be added. We applied Farneback OF method, however, there are also other real-time OF computation methods in the literature. The evaluation of processing time for deep architectures is considered in two different stages: training time, which includes forward and backward pass, and testing time, which only consists of forwarding pass through trained parameters.

Suppose  $T$  is frame rate of input images. The algorithm is said to be real-time if feature extraction time,  $T_{FE}$ , and forwarding pass through deep architecture,  $T_{rmFF}$  satisfied the following equation:

$$T_{rmFE} + T_{rmFF} < 1/T. \quad (6)$$

## 4 Experimental results

We chose PETS2009 dataset [4] for our experimental analysis. PETS dataset has been proposed for various crowd analysis tasks such as people counting, density estimation, event recognition, and so on in outdoor scenarios. Here, we assess our proposed approach on S3 HL sequences in PETS2009, which were provided for analyzing crowd event recognition. This dataset is comprised of four video sequences captured at the following time-stamps 14:16, 14:27, 14:31, and 14:33. Each sequence has been captured from four different views. Figure 2 shows a sample frame captured from four views. Some sequences are composed of 2 video clips, this is the case of 14:16, 14:27, and 14:33, which results totally in seven video sequences [10]. The durations of these seven videos are given in Table 1. The crowd events to be recognized are walking, running, formation (merging), splitting, evacuation, dispersion, and loitering. It should be mentioned that not all the previous methods tried to detect all these seven events. Some authors did not consider 'loitering' event (like [10, 14, 32]). As mentioned, 'loitering' corresponds to a scene where pedestrians are standing with a little fluctuation between them. We also considered this event to achieve more accuracy in the detection of different kinds of classes, and to make our labeled dataset a bit larger. Moreover, we

**Table 1** Frame numbers for each video clip

Sequence name	First frame	Last frame
14:16-a	0	107
14:16-b	108	222
14:27-a	0	184
14:27-b	185	333
14:31	0	130
14:33-a	0	310
14:33-b	311	377

**Fig. 2** Sample frame 26 from 4 different views





**Fig. 3** Sample frames for each event. Respectively from left to right: walk, run, split, merge, evacuation, dispersion, loitering

used  $k$ -fold cross-validation to partly overcome the small size of the dataset ( $k = 5$ ). Figure 3 depicts seven classes of crowd events in different frames. We manually annotated the frames based on the definition in Sect. 1 as brought in Table 2.

The original size of the video frames is  $768 \times 576$ . After resizing original frames to  $256 \times 256$ , grayscale transformation, and normalization, Farneback optical flow method was applied to each of  $28 = (7\text{video clips}) \times (4\text{views})$  sequences to densely estimate OF vector for each pixel. We took the magnitude of OF and created a sequence of OF images. After that, for each OF image at time  $t$ , we created a  $T$  length clip consisting of a sequence of  $t - T$  to  $t$  OF images. We considered  $T = 5$  as compared to [51] with  $T = 10$ . This smaller value for  $T$  is sufficient for recognizing events. Moreover, small  $T$  reduces the computational complexity of the algorithm. The label for the clip was defined according to the label for frame  $t$ . This process was done through all the sequences.

We considered 70/30 percent of data as training/testing set and 20% of training data for validation to tune network hyperparameters. Our deep learning architecture is the concatenation of layers as shown in Fig. 1.

In contrast to [49] whose input is a stack of raw frames, here we entered stack of OF frames to the network, since the events we are going to detect have high information on motion images. Besides, we added an extra layer in the bottleneck of the network proposed by [51], to get a smaller latent representation and reduce the number of parameters. Instead of flattening the bottleneck representation into the vector before dense layer, we apply global

average pooling to reduce the number of parameters. We considered the units in this dense layer as small as possible (20). Subsequently, a dense layer with seven units and a softmax activation function was used for classification. In Softmax function, the highest probability between the seven units outputs, shows the event label.

The numbers of Epochs and batch size are set to 50 and 16 respectively. Small size of the batch was chosen to avoid getting out of RAM. Early stopping and dropout(20%) was used as regularization. As the main purpose of our proposed approach is classification, we fixed  $w_2 = 1$  for categorical cross entropy loss, and changed  $w_1$  for mean square error (MSE) loss to see how it affected the performance (minimum overall loss) which resulted in chosen  $w_1 = 0.25$ . We conducted our experiments using Keras framework on Google Colab Tesla K80 GPU. The results of the proposed approach were evaluated through different metrics namely, precision, recall, F1-score, confusion matrix, and time complexity of the algorithm.

Confusion matrix is shown in Table 3. Confusion matrix is defined from TP, TN, FP, and FN numbers. Meanwhile, TP shows the number of correct predictions. From Table 3 we can see that all of the events can be predicted with true positive (TP) values more than 88%. The best previous results were reported by [29] in which TP is as low as 86%.

We show precision, recall and F1-score metrics at once in Table 4. Precision is defined as the number of true positives divided by the number of true positives plus the number of false positives. Recall is defined as the number of true positives divided by the number of true positives plus the number of false negatives. Precision  $PR_i$  and Recall  $RC_i$  for multiclass classification are calculated as follows:

**Table 2** Manual ground truth for video frames on PETS2009 event recognition dataset

Events	seq.14:16-a	seq.14:16-b	seq.14:27-a	seq.14:27-b	seq.14:31	seq.14:33-a	seq.14:33-b
Walking	[0–35]	[108–162]	–	–	[0–47]	[151–188], [289–310]	–
Running	[36–107]	[163–222]	–	–	–	–	–
Evacuation	–	–	–	–	–	–	[334–377]
Dispersion	–	–	[96–144]	[271–303]	–	–	–
Formation(Merge)	–	–	–	–	–	[0–150]	–
Splitting	–	–	–	–	[48–130]	–	–
Loitering	–	–	[0–95], [145–184]	[185–270], [304–333]	–	[189–288]	[311–333]

**Table 3** Confusion matrix For crowd event recognition using PETS2009 dataset

	Walk	Run	Evacuation	Dispersion	Merge	Split	Loitering
Walk	<b>96.08</b>	3.35	0	0	0	0	0.55
Run	4.45	<b>94.26</b>	0	0	0	0	1.27
Evacuation	0	2.00	<b>96</b>	0	2.00	0	0
Dispersion	0	0	0	<b>88.57</b>	0	0	11.43
Merge	0	0	0	0	<b>98.89</b>	0	1.10
Split	7.4	0	0	0	0	<b>92.59</b>	0
Loitering	0.2	0	0.4	.6	0	0	<b>98.80</b>

The bold values are the best performance achieved by the methods

**Table 4** Comparison of precision, recall and F1-score metrics for different approaches

Events	Measure	Holistic [14]	Random forest [16]	Motion pattern [16]	OF manifold [29]	Proposed approach
Walk	Precision	0.87	0.96	<b>0.97</b>	0.73	0.89
	Recall		<b>0.99</b>	0.96	0.92	0.96
	F1-score		<b>0.97</b>	0.96	0.81	0.93
Run	Precision	0.75	0.86	0.75	0.93	<b>0.94</b>
	Recall		0.68	0.81	0.84	<b>0.94</b>
	F1-score		0.75	0.77	0.89	<b>0.94</b>
Evacuation	Precision	0.94	0.83	0.69	0.85	<b>1.00</b>
	Recall		<b>1.0</b>	0.82	0.84	0.96
	F1-score		0.90	0.74	0.85	<b>0.98</b>
Dispersion	Precision	0.8	0.58	0.67	0.94	<b>0.99</b>
	Recall		0.48	0.45	<b>0.98</b>	0.89
	F1-score		0.52	0.53	<b>0.96</b>	0.94
Merge	Precision	0.68	0.65	0.59	0.92	<b>0.98</b>
	Recall		0.46	0.45	0.98	<b>0.99</b>
	F1-score		0.53	0.51	0.9	<b>0.99</b>
Split	Precision	0.74	0.73	0.47	0.93	<b>1.</b>
	Recall		0.92	0.47	0.95	0.93
	F1-score		0.81	0.47	0.94	<b>0.96</b>
Loitering	Precision	–	–	0.48	–	<b>0.87</b>
	Recall	–	–	0.85	–	<b>0.99</b>
	F1-score	–	–	0.61	–	<b>0.93</b>

Our proposed approach beats the state of the art methods

The bold values are the best performance achieved by the methods

$$PR_i = \frac{M_{ii}}{\sum_j M_{ji}} \quad (7)$$

$$RC_i = \frac{M_{ii}}{\sum_j M_{ij}}, \quad (8)$$

where  $PR_i$  and  $RC_i$  are precision and recall for class  $i$  respectively.  $M_{ii}$  are diagonal elements of Confusion matrix that show the number of true positive for class  $i$ .

F1-score is another measure of the test's accuracy. It considers both precision and recall, by computing harmonic mean between them as the following formula. Here we define this measure for each event  $i$  as  $F1_i$ .

$$F1_i = 2 \cdot \frac{PR_i \times RC_i}{PR_i + RC_i}. \quad (9)$$

From Table 4, we can see that for all events except walking, our proposed algorithm results in the best precision. Also, the best result for F1-score has been achieved by the proposed method for all events except walk and dispersion.

In another evaluation, we computed the average accuracy between all classes. Accuracy can be defined as the total number of correct predictions divided by the total number of samples in a test set. It is defined as follows for multiclass classification:

$$\text{Accuracy} = \frac{\sum_i M_{ii}}{\sum_{i,j} M_{ij}}, \quad (10)$$



We should note that accuracy is not a suitable metric when dealing with imbalanced data as we have encountered in PETS2009 crowd dataset. Better evaluation can be done through precision, recall, and F1-score. The proposed approach’s accuracy has been computed and compared with that of some previous methods in Table 5. Accuracy was compared with reported values in [14, 25, 27, 29, 32]. It can be interpreted from this Table that among all the previous methods, the approach based on dense trajectory [32] has the highest accuracy. The high amount of accuracy in this method is due to the application of trajectory attributes in the classification phase. All of these methods used low or mid-level features. Our deep architecture can learn high-level features through series of layers and has achieved a new state-of-the-art accuracy of 96.07%.

Besides, we calculated dice score,

$$Dice_k = \frac{2TP_k}{2TP_k + FP_k + FN_k}, \tag{11}$$

where  $k$  is indicator for events. The results shown in Table 6 which indicate high performance of our proposed method.

### 4.1 Real-time evaluation

Finally, to show the real-time aspect of our proposed approach, we evaluated the test time complexity and compared it with the previous researches which claims online recognition. Deep learning architecture, although taking much time for training due to large number of parameters’ tuning, at test time acts as real-time, since the test sample just needs to pass through some sequential layers whose weights have been set at the training stage. As mentioned in Sect. 3.1, whole test processing time for a test image is the integration of OF computation and its forward processing. Farneback OF takes 0.04 s for a pair of images to calculate (25 frames per second). Besides, our proposed algorithm can process a test OF image through DL architecture within 30 milliseconds per frame (33 frames per second). The time for calculation of optical flow images can be negligible through the application of real time optical flow methods. Therefore, overall processing a test frame happens at 14 fps rate, which can be considered as real-time since PETS2009 frame rate is 7 fps [4]. Figure 4 shows the time complexity of our algorithm. Benabbas et al. [16], reported a time complexity of four frames per second. They used two classifiers (random forest) to detect two categories of events(walk/run) and (split/merge/dispersion/evacuation/loitering). The

**Table 5** Comparison of accuracy metric for different approaches

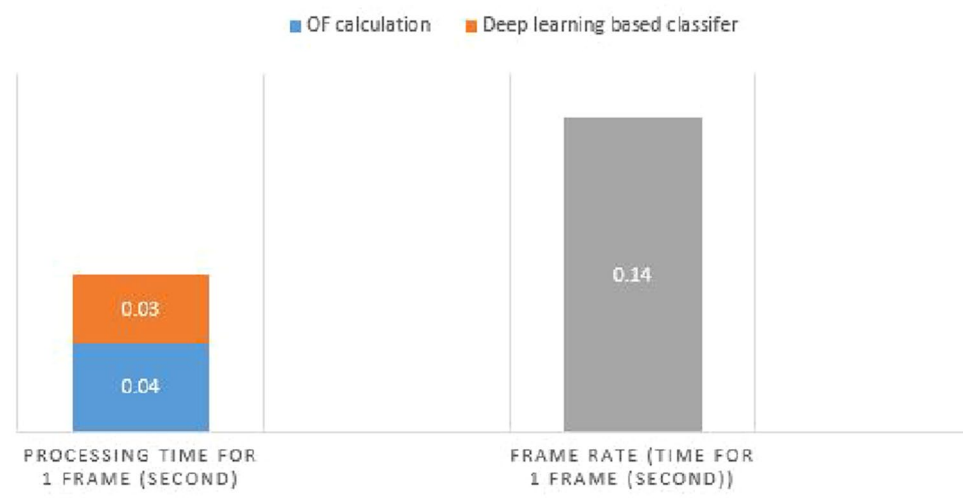
Method	Holistic [14]	Video manifold [25]	BoTG [27]	OF manifold [29]	Dense trajectory [32]	Proposed approach
Accuracy	81.0%	90.0%	90.0%	91.2%	93.0%	<b>96.07%</b>

The bold values are the best performance achieved by the methods

**Table 6** Comparison of dice metric for different events

Event	Walk	Run	Evacuation	Dispersion	Merge	Split	Loitering
Dice score	0.92	0.94	0.97	0.93	0.98	0.96	0.92 %

**Fig. 4** Time complexity evaluation for a test frame in second for PETS2009 dataset



application of two separate classifiers needs more time for learning, so it is time-consuming. They argued in their paper that holistic approach [14] is slower. Overall, our deep architectures do not need to do time-consuming background subtraction, object detection, and tracking and in this way, they are time efficient in test time.

## 5 Conclusion and future works

In this paper, we investigated the application of a deep learning architecture for crowd behavior analysis to recognize seven events as walk, run, evacuation, dispersion, merge, split, and loitering. The proposed Conv-LSTM-AE technique achieves higher performance than the previous hand-crafted-based methods both in accuracy, confusion matrix, precision, recall, and F1-measure. Experimental results on PETS2009 crowd event recognition approved the success of the proposed architecture.

Despite the rapidly growing success in the image domain, the scarcity of labeled video data has decreased the slope of the growth of Deep learning for visual sequences. New regularization methods can partly compensate for these deficiencies. The application of multitask learning [61], few-shot learning, and self-supervised learning for other large-scale datasets such as WWW crowd datasets [61], can be studied in future.

## References

- Li, T., Chang, H., Wang, M., Ni, B., Hong, R., Yan, S.: Crowded scene analysis: a survey. *IEEE Trans. Circ. Syst. Video Technol.* **25**(3), 367–386 (2014)
- Khan, M.T., Ali, A., Durrani, M.Y., Siddiqui, I.: Survey of holistic crowd analysis models. *J. Comput. Sci. Commun.* **1**(1), 1–9 (2015)
- Yuan, Y., Fang, J., Wang, Q.: Online anomaly detection in crowd scenes via structure analysis. *IEEE Trans. Cybern.* **45**(3), 548–561 (2014)
- Ferryman, J.: PETS 2009 benchmark data (2009). <http://www.cvg.rdg.ac.uk/PETS2009/a.html>
- Ferryman, J., Shahrokni, A.: PETS2009: dataset and challenge. In: 2009 Twelfth IEEE international workshop on performance evaluation of tracking and surveillance, pp. 1–6. IEEE (2009)
- Shi, J.: Good features to track. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600. IEEE (1994)
- Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **110**(3), 346–359 (2008)
- Tomasi, C., Kanade, T.: Detection and tracking of point features. Technical report CMU-CS-91-132, CMU Google Scholar (1991)
- Fradi, H., Dugelay, J.L.: Spatial and temporal variations of feature tracks for crowd behavior analysis. *J. Multimodal User Interfaces* **10**(4), 307–317 (2016)
- Fradi, H., Dugelay, J.L.: Sparse feature tracking for crowd change detection and event recognition. In: 22nd International Conference on Pattern Recognition, pp. 4116–4121. IEEE (2014)
- Rao, A.S., Gubbi, J., Palaniswami, M.: An improved approach to crowd event detection by reducing data dimensions. In: Advances in Signal Processing and Intelligent Recognition Systems, pp. 85–96. Springer, Cham (2016)
- 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, pp. 4305–4314 (2015)
- Zhang, W., Hou, Y., Wang, S.: Event recognition of crowd video using corner optical flow and convolutional neural network. In: 8th International Conference on Digital Image Processing (ICDIP 2016), vol. 10033, p. 100335K. International Society for Optics and Photonics (August)
- Chan, A.B., Morrow, M., Vasconcelos, N.: Analysis of crowded scenes using holistic properties. In: Performance Evaluation of Tracking and Surveillance workshop at CVPR, pp. 101–108 (2009)
- Cermeno, E., Mallor, S., Sigüenza, J.A.: Learning crowd behavior for event recognition. In: IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS), pp. 1–5. IEEE (2013)
- Benabbas, Y., Ihaddadene, N., Djeraba, C.: Motion pattern extraction and event detection for automatic visual surveillance. *EURASIP J. Image Video Process.* **2011**(1), 163682 (2011)
- Briassouli, A., Kompatsiaris, I.: Spatiotemporally localized new event detection in crowds. In: IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 928–933. IEEE (2011)
- Grigg, O.A., Farewell, V.T., Spiegelhalter, D.J.: Use of risk-adjusted CUSUM and RSPRTcharts for monitoring in medical contexts. *Stat. Methods Med. Res.* **12**(2), 147–170 (2003)
- Mehran, R., Oyama, A., Shah, M.: Abnormal crowd behavior detection using social force model. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 935–942. IEEE (2009)
- Mehran, R., Moore, B.E., Shah, M.: A streakline representation of flow in crowded scenes. In: European Conference on Computer Vision, pp. 439–452. Springer, Berlin (2010)
- Huang, S., Huang, D., Khuhro, M.A.: Crowd motion analysis based on social force graph with streak flow attribute. *J. Electr. Comput. Eng.* **2015**, 52 (2015)
- Dee, H.M., Caplier, A.: Crowd behaviour analysis using histograms of motion direction. In: IEEE International Conference on Image Processing, pp. 1545–1548. IEEE (2010)
- Mousavi, H., Mohammadi, S., Perina, A., Chellali, R., Murino, V.: Analyzing tracklets for the detection of abnormal crowd behavior. In: IEEE Winter Conference on Applications of Computer Vision, pp. 148–155. IEEE (2015)
- Wang, X., He, Z., Sun, R., You, L., Hu, J., Zhang, J.: A crowd behavior identification method combining the streakline with the high-accurate variational optical flow model. *IEEE Access* **7**, 114572–114581 (2019)
- Thida, M., Eng, H.L., Monekosso, D.N., Remagnino, P.: Learning video manifolds for content analysis of crowded scenes. *IPSN Trans. Comput. Vis. Appl.* **4**, 71–77 (2012)
- Ghods, A.: Dimensionality reduction a short tutorial, vol 37, p 38. Department of Statistics and Actuarial Science, Univ. of Waterloo, Ontario (2006)
- Zhang, Y., Huang, Q., Qin, L., Zhao, S., Yao, H., Xu, P.: Representing dense crowd patterns using bag of trajectory graphs. *Signal Image Video Process* **8**(1), 173–181 (2014)
- Fradi, H., Luvison, B., Pham, Q.C.: Crowd behavior analysis using local mid-level visual descriptors. *IEEE Trans. Circ. Syst. Video Technol.* **27**(3), 589–602 (2016)

29. Rao, A.S., Gubbi, J., Marusic, S., Palaniswami, M.: Crowd event detection on optical flow manifolds. *IEEE Trans. Cybern.* **46**(7), 1524–1537 (2015)
30. Su, H., Yang, H., Zheng, S., Fan, Y., Wei, S.: The large-scale crowd behavior perception based on spatio-temporal viscous fluid field. *IEEE Trans. Inf. Forensics Secur.* **8**(10), 1575–1589 (2013)
31. Solmaz, B., Moore, B.E., Shah, M.: Identifying behaviors in crowd scenes using stability analysis for dynamical systems. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(10), 2064–2070 (2012)
32. Khokher, M. R., Bouzerdoum, A., Phung, S.L.: Crowd behavior recognition using dense trajectories. In: *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–7. IEEE (2014)
33. Shuaibu, A.N., Faye, I., Ali, Y.S., Kamel, N., Saad, M.N., Malik, A.S.: Sparse representation for crowd attributes recognition. *IEEE Access* **5**, 10422–10433 (2017)
34. Pathan, S.S., Al-Hamadi, A., Michaelis, B.: Crowd behavior detection by statistical modeling of motion patterns. In: *International Conference of Soft Computing and Pattern Recognition*, pp. 81–86. IEEE (2010)
35. Hu, X., Hu, S., Huang, Y., Zhang, H., Wu, H.: Video anomaly detection using deep incremental slow feature analysis network. *IET Comput. Vis.* **10**(4), 258–267 (2016)
36. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1725–1732 (2014)
37. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 221–231 (2012)
38. Wei, H., Xiao, Y., Li, R., Liu, X.: Crowd abnormal detection using two-stream Fully Convolutional Neural Networks. In: *10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pp. 332–336. IEEE (2018)
39. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: *Advances in neural information processing systems*, pp. 568–576 (2014)
40. Xu, Y., Lu, L., Xu, Z., He, J., Wang, J., Huang, J., Lu, J.: Towards intelligent crowd behavior understanding through the STFD descriptor exploration. *Sens. Imaging* **19**(1), 17 (2018)
41. Fang, Z., Fei, F., Fang, Y., Lee, C., Xiong, N., Shu, L., Chen, S.: Abnormal event detection in crowded scenes based on deep learning. *Multimed. Tools Appl.* **75**(22), 14617–14639 (2016)
42. Khan, G., Farooq, M.A., Hussain, J., Tariq, Z., Khan, M.U.G.: Categorization of crowd varieties using deep concurrent convolution neural network. In: *2nd International Conference on Advancements in Computational Sciences (ICACS)*, pp. 1–6. IEEE (2019)
43. Burney, A., Syed, T.Q.: Crowd video classification using convolutional neural networks. In: *International Conference on Frontiers of Information Technology (FIT)*, pp. 247–251. IEEE (2016)
44. Borja-Borja, L.F., Saval-Calvo, M., Azorin-Lopez, J.: A short review of deep learning methods for understanding group and crowd activities. In: *International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE (2018)
45. Li, P., Jiang, X., Sun, T., Xu, K.: Crowded scene understanding algorithm based on two-stream residual network. In: *10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1–6. IEEE (2017)
46. Roli, F., Giacinto, G., Vernazza, G.: Comparison and combination of statistical and neural network algorithms for remote-sensing image classification. In: *Neurocomputation in remote sensing data analysis*, pp. 117–124. Springer, Berlin (1997)
47. Wang, C., Zhao, X., Shou, Z., Zhou, Y., Liu, Y.: A discriminative tracklets representation for crowd analysis. In: *IEEE International Conference on Image Processing (ICIP)*, pp. 1805–1809. IEEE (2015)
48. Li, Y.: A deep spatiotemporal perspective for understanding crowd behavior. *IEEE Trans. Multimed.* **20**(12), 3289–3297 (2018)
49. Zhuang, N., Ye, J., Hua, K.A.: Convolutional DLSTM for crowd scene understanding. In: *IEEE International Symposium on Multimedia (ISM)*, pp. 61–68. IEEE (2017)
50. Erfani, S.M., Rajasegarar, S., Karunasekera, S., Leckie, C.: High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recogn.* **58**, 121–134 (2016)
51. Chong, Y.S., Tay, Y.H.: Abnormal event detection in videos using spatiotemporal autoencoder. In: *International Symposium on Neural Networks*, pp. 189–196. Springer, Cham (2017)
52. Fernández-Ramírez, J., Álvarez-Meza, A., Pereira, E.M., Orozco-Gutiérrez, A., Castellanos-Dominguez, G.: Video-based social behavior recognition based on kernel relevance analysis. *Vis. Comput.* **36**(8), 1535–1547 (2020)
53. Deng, C., Kang, X., Zhu, Z., Wu, S.: Behavior recognition based on category subspace in crowded videos. *IEEE Access* **8**, 222599–222610 (2020)
54. Varghese, E., Thampi, S.M., Berretti, S.: A psychologically inspired fuzzy cognitive deep learning framework to predict crowd behavior. In: *IEEE Transactions on Affective Computing* (2020)
55. Li, Q., Zhao, X., He, R., Huang, K.: Recurrent prediction with spatio-temporal attention for crowd attribute recognition. *IEEE Trans. Circ. Syst. Video Technol.* **30**(7), 2167–2177 (2019)
56. Tripathi, G., Singh, K., Vishwakarma, D.K.: Convolutional neural networks for crowd behaviour analysis: a survey. *Vis. Comput.* **35**(5), 753–776 (2019)
57. Kiran, B.R., Thomas, D.M., Parakkal, R.: An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. *J. Imaging* **4**(2), 36 (2018)
58. Pérez, J.S., Meinhardt-Llopis, E., Facciolo, G.: TV-L1 optical flow estimation. *Image Process. On Line* **2013**, 137–150 (2013)
59. Xingjian, S.H.L., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional LSTM network: a machine learning approach for precipitation nowcasting. In: *Advances in neural information processing systems*, pp. 802–810 (2015)
60. Yang, M., Rajasegarar, S., Erfani, S. M., Leckie, C.: Deep learning and one-class SVM based anomalous crowd detection. In: *International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE (2019)
61. 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*, pp. 4657–4666 (2015)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Fariba Rezaei** is PhD student in the Faculty of Electrical and computer Engineering, Shiraz University, Shiraz, Iran. She received Master's degree in Electrical Engineering from Shahid Beheshti University in 2012. Her major research interests are in the field of video image processing.

**Mehran Yazdi** is professor in the Faculty of Electrical and computer Engineering, Shiraz University, Shiraz, Iran. He received Master and PhD from Laval University, Québec, Canada in Electrical Engineering in 1997 and 2005 respectively. His major research interests are medical image and video processing.