# Convolutional DLSTM for Crowd Scene Understanding

Naifan Zhuang
Department of Computer Science
University of Central Florida
Orlando, USA
Email: zhuangnaifan@knights.ucf.edu

Jun Ye
Microsoft
Redmond, USA
Email: jye@cs.ucf.edu

Kien A. Hua
Department of Computer Science
University of Central Florida
Orlando, USA
Email: kienhua@cs.ucf.edu

***Abstract*—With the growth of crowd phenomena in the real world, crowd scene understanding is becoming an important task in anomaly detection and public security. Visual ambiguities and occlusions, high density, low mobility and scene semantics, however, make this problem a great challenge. In this paper, we propose an end-to-end deep architecture, Convolutional DLSTM (ConvDLSTM), for crowd scene understanding. ConvDLSTM consists of GoogleNet Inception V3 convolutional neural networks (CNN) and stacked differential long short-term memory (DLSTM) networks. Different from traditional non-end-to-end solutions which separate the steps of feature extraction and parameter learning, ConvDLSTM utilizes a unified deep model to optimize the parameters of CNN and RNN hand in hand. It thus has the potential of generating a more harmonious model. The proposed architecture takes sequential raw image data as input, and does not rely on tracklet or trajectory detection. It thus has clear advantages over the traditional flow-based and trajectory-based methods, especially in challenging crowd scenarios of high density and low mobility. Taking advantage of the semantic representation of CNN and the memory states of LSTM, ConvDLSTM can effectively analyze both the crowd scene and motion information. Existing LSTM-based crowd scene solutions explore deep temporal information and are claimed to be "deep in time". ConvDLSTM, however, models the spatial and temporal information in a unified architecture and achieves "deep in space and time". Extensive performance studies on the Violent-Flows and CUHK Crowd datasets show that the proposed technique significantly outperforms state-of-the-art methods.**

## I. Introduction

With the increase of world population and various human activities, crowd phenomena are growing more rapidly than ever before. To ensure public security and safety, understanding crowd scenes, especially abnormal crowd behaviors and emotions, is becoming increasingly urgent and important [1], [2]. Although human observers are able to monitor behavior patterns and detect unusual crowd activities in the surveillance area, the wide use of video surveillance in the last decade has led to huge amounts of video data which are beyond the capability of human observers [3]. In addition, psychophysical research suggests that humans' ability to monitor simultaneous signals deteriorates after long-term monitoring because extremely crowded scenes exhibit excessive numbers of individuals and their activities [1]. These issues make humans poor-performing observers of crowd interactions and anomaly events.

In the last decade, researchers from the computer vision community have shown much interest in developing automated crowd scene understanding systems. Video analysis for uncrowded scenes usually involves object detection, object tracking and behavior recognition. Such solutions, however, are not suitable for crowded scenes; and special considerations must be taken into account. As a crucial basis, appropriate feature representation for crowded scenes is necessary. In terms of representation level, previous crowd features can be divided into the following three categories [1]: flow-based features, local spatio-temporal features and trajectory/tracklet features.

Flow-based features are extracted densely on the pixel level and are suitable for highly dense crowded scenes when tracking each person in the videos is impracticable. Several flow-based features have been presented in recent years [4], [5], [6], [7]. Their methods achieved success in addressing dense and complex crowd flows by avoiding tracking at the macroscopic level. However, flow-based features ignore the scene information and tend to fail in crowd videos with less mobility.

Local spatio-temporal features exploit the dense local motion features created by the subjects and model their spatio-temporal relationships to represent the underlying intrinsic structure formed in the video. Some related works use histogram functions [8], [9], and spatio-temporal gradients [10], [11]. Local spatio-temporal features, however, analyze local features of crowd dynamics and are sub-optimal for complex crowd behaviors with long-range dependency.

Most recent methods for crowd scene understanding mostly analyze crowd activities based on motion features extracted from trajectories/tracklets of objects [12], [13], [14], [15], [16], [17], [18]. The trajectory/tracklet feature contains more semantic information, but the accuracy of trajectories/tracklets dictates the performance of crowd scene analysis. In extremely crowded areas, tracking algorithms could fail and generate inaccurate trajectories.

The disadvantages of the above existing crowd representation methods motivate us to explore a new representation for crowded scenes, which is simple yet can still maintain the raw information of the source video as much as possible. We are inspired by the success of convolutional neural networks [19]

978-1-5386-2937-6/17 $31.00 © 2017 IEEE
DOI 10.1109/ISM.2017.19

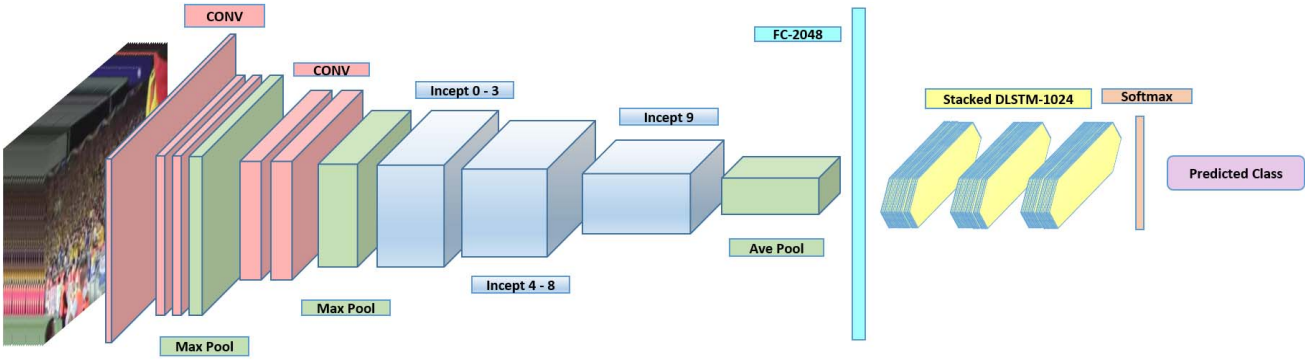


Fig. 1. An illustration of Convolutional DLSTM. It takes crowd image sequences as input and directly outputs predicted crowd class. The model consists of convolutional (red), pooling (green), mixed inception blocks (blue), fully-connected (cyan), DLSTM (yellow) and softmax (brown) layers. **Best viewed in color**.

to explore the use of raw input data for crowd scenes.

Human crowds exhibit complex temporal dynamics and psychological characteristics [20]. To model such complex dynamics, Long Short-Term Memory (LSTM) [21] was proposed to learn the dynamical evolution of a long sequence. LSTM possesses the potential to model various sequential data, where the current hidden state has to be considered in the context of the past hidden states. This property makes LSTM an ideal choice to learn the complex dynamics of human activities [22]. Alexandre *et al.* [23] viewed the human trajectory prediction in crowded spaces as a sequence generation task and used an LSTM model to learn general human movement and predict their future trajectories. Su *et al.* [13] explored Coherent LSTM to model the nonlinear characteristics and spatio-temporal motion patterns in crowd behaviors. Using two stacked LSTMs, their model is better at learning deep temporal information and they claimed their model to be "deep in time". In this paper, we aim to construct a unified deep model exploring spatial and temporal information concurrently for crowd scene understanding and investigate the possibility of achieving "deep in space and time".

To address the challenge of video data processing, we have introduced the dynamic temporal quantization [24] and differential long-short term memory [25] in our early work to achieve a fixed-length representation of video data with varied length. However, these two methods are not designed for crowd scene analysis. In addition, neither of them is an end-to-end solution and does not fully explore the capability of spatial and temporal representation in deep neural networks. In this paper, we propose a novel end-to-end convolutional DLSTM (ConvDLSTM) network for crowd scene understanding. The architecture connects GoogleNet Inception V3 [26] convolutional neural networks and stacked Differential Long Short-Term Memory [22]. The deep neural network directly takes the sequential raw image data as input, and outputs the predicted crowd scene label. It differs from the three existing categories of feature representations by directly using the raw image sequences. The proposed technique has the following advantages over existing methods.

Firstly, when dealing with highly dense crowd scenes, trajectory/tracklet methods tend to perform poorly. ConvDLSTM has no such problem because it does not rely on trajectory detection. Secondly, flow-based and trajectory-based methods assume crowd mobility when extracting the flow and trajectory representations. The convolutional neural network layers in the proposed ConvDLSTM can model the scene semantics and perform reasonable analysis and do not require motion information from the crowd. Thirdly, different from existing LSTM-based crowd scene solutions which are "deep in time", ConvDLSTM models the spatial and temporal information in a unified architecture and achieves "deep in space and time".

We extensively evaluate the performance of the proposed deep architecture on two public crowd understanding datasets, Violent-Flows [27] and CUHK Crowd [12]. Experimental results show that the proposed technique significantly outperforms the conventional flow-based and trajectory/tracklet-based methods by a great margin. We also show that our Convolutional DLSTM model can outperform the LSTM-based methods by achieving deep in both space and time. Our main contributions are summarized as follows:

1) We propose an end-to-end deep architecture ConvDLSTM. It consists of convolutional neural networks and stacked DLSTMs and achieves "deep in space and time";
2) ConvDLSTM does not require any handcrafted feature representation such as flow or trajectory, and can handle crowd scenes of high density and low mobility;
3) Extensive experimental studies demonstrate the superior performance of ConvDLSTM over state-of-the-art methods.

We structure the remainder of this paper as follows. In Section II, we talk about related work. The proposed ConvDLSTM model is presented in Section III. Performance studies on two public crowd datasets, Violent-Flows and CUHK Crowd, are given in Section IV. Finally, we conclude the paper in Section V.

## II. Related Work

Due to the high-density and low-resolution characteristics of crowd scene videos, feature representation for crowd analysis is a crucial basis for this problem. Previous crowd analysis methods can be divided into three categories according to the feature representation level: flow-based features, local spatio-temporal features and trajectory/tracklet features.

Crowd scenes often present highly sense scenario, which makes it very difficult to track each individual in the videos. Flow-based crowd analysis techniques extract densely features on the pixel level, avoiding the issue of tracking each object. For flow-based features, the rationale is as follows: as specific actions of individuals may be relatively random, but the overall dynamics of the crowd can still be convincing. Several works [28], [29] use optical flow to compute pixel-wise instantaneous motion between consecutive frames and apply to crowd motion detection. Based on the Lagrangian framework of fluid dynamics [30], partical flow was introduced to handle abnormal crowd behavior detection [5], [31]. To acquire accurate representation of the crowd motion flow, Mehran *et.al.* [6] introduced streakline to compute the motion field for crowd scene analysis. Flow-based methods achieved success in addressing dense and complex crowd flows by avoiding tracking at the macroscopic level. However, flow-based features ignore the scene information and tend to fail in crowd videos with less mobility.

Some extremely crowded scenes, though similar in density, are less structure due to the high variability of individual movements. In this case, the motion within each local are may be non-uniform and flow-based features, such as optical flow, would not provide enough information. One solution is to exploit the dense local motion patterns created by the subjects and model their spatio-temporal relationships. The related methods use spatio-temporal gradients [10], [11], and histogram functions [8], [9]. Kratz *et.al.* [10], [11] use the distribution of spatio-temporal gradients as the base representation to detect unusual activities. Motion histograms can be considered as one kind of motion information defined on local regions. Jodoin *et.al.* [8] proposed a feature called orientation distribution function, which has advantage in computation for the upcoming motion pattern learning. Cong *et.al.* [9] proposed a novel feature descriptor called multi-scale histogram of optical flow. It preserves both motion information and spatial contextual information, and performs well on abnormal event detection. The downside of local spatio-temporal features is since these methods analyze local features of crowd dynamics and are sub-optimal for complex crowd behaviors with long-range dependency.

Compared with the above two types of feature representations, trajectories/tracklet is more semantic and seems to be more popular in recent computer vision research community. Since the density of the crowd increases and the scene cluttter becomes severe, traditional object detection and tracking can hardly be performed accurately. A new motion feature called tracklet has been proposed. As a fragment of a trajectory obtained by the tracker within a short period, tracklets terminate when ambiguities occur. Tracklets thus have been used to complete trajectories for tracking. Several tracklet based approaches [15], [16], [17] were proposed to learn semantic regions and clustering trajectories. Recent methods for crowd scene understanding mostly analyze crowd activities based on motion features extracted from trajectories/tracklets of objects [12], [27], [13], [14], [18]. Marsden et al. [3] studied scene-level holistic features using tracklets to solve real-time crowd behavior anomaly detection problem. Su et al. [13] used tracklet-based features and explored Coherent LSTM to model the nonlinear characteristics and spatio-temporal motion patterns in crowd behaviors. These two methods hold state-of-the-art performances for the Violent-Flows and CUHK Crowd datasets, respectively. The trajectory/tracklet feature contains more semantic information, but the accuracy of trajectories/tracklets dictates the performance of crowd scene analysis. In extremely crowded areas, tracking algorithms could fail and generate inaccurate trajectories.

## III. Convolutional DLSTM

In this section, we elaborate upon our proposed Convolutional DLSTM (ConvDLSTM) framework for crowd scene analysis. ConvDLSTM connects GoogleNet Inception V3 [26] and stacked DLSTMs [21] into an end-to-end model. Fig. 1 shows the diagram of the proposed ConvDLSTM architecture. As shown in Fig. 1, ConvDLSTM takes the sequential raw RGB image data as input and outputs the predicted crowd scene class. The model is composed of first few convolutional and max-pooling layers, ten mixed Inception blocks with the last block Mixed 9 containing two identical inception blocks, one average-pooling layer, dropout and fully-connected layer, three stacked LSTMs and lastly, a softmax layer. In ConvDLSTM, each frame of a crowd video is first processed by GoogleNet Inception V3 convolutional neural network (CNN) to generate frame-level representations, which are then allowed to flow between time-steps using stacked DLSTMs. By doing so, ConvDLSTM possesses the potential of analyzing the spatial and temporal information in a unified model and achieves "deep in space and time".

### A. GoogleNet Inception V3 Convolutional Neural Networks

In the proposed ConvDLSTM model, we adopt GoogleNet Inception V3 as the prototype of the convolutional neural network part. The Inception micro-architecture was first introduced by Szegedy et al. [32]. The goal of the inception module is to act as a multi-level feature extractor by computing $1 \times 1$, $3 \times 3$, and $5 \times 5$ convolutions within the same module of the network. The output of these filters are then stacked along the channel dimension before being fed into the next layer in the network. The Inception V3 architecture generally includes the following disciplines. It reduces the number of convolutions to maximum $3 \times 3$ blocks. In addition it increases the general depth of the networks. Lastly, Inception V3 uses the width increase technique at each layer to improve feature combination. The Inception V3 architecture further

boosts ImageNet classification accuracy. In the mean time, it keeps the number of parameters for the whole architecture smaller, enabling the high efficiency for training and testing the network.

GoogleNet Inception V3 originally consists of first few regular convolutional and max-pooling layers, ten mixed Inception blocks and one average-pooling, dropout and fully-connected layer. For our proposed ConvDLSTM model, we remove the top fully-connected layer FC-1000 since it corresponds the 1000 ImageNet class label probabilities, which do not specifically correlate with our crowd scene analysis tasks. In Fig. 1, due to page limit, we use one blue block to demonstrate the same inception blocks. For more details for the Inception V3 architecture, please refer to [26].

For notational simplicity, we refer to the modified GoogleNet Inception V3 network as $\mathbf{x} = V(m)$, which takes the raw RGB data of an image $m$ as input and produces a 2048-dimension representation. Denote $\{m_1, m_2, ..., m_T\}$ as a crowd image sequence of length $T$, where $m_t$ indicates the image frame at time $t$. The sequential image data for a crowd video are passed through the Inception V3 network frame by frame to produce $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T\}$, which serve as the input sequence to stacked DLSTMs. At time $t$, modified Inception V3 takes the input image $m_t$ and computes $\mathbf{x}_t$ via:

$$\mathbf{x}_t = V(m_t). \tag{1}$$

### B. Differential Long Short-Term Memory

Due to exponential decay, traditional RNNs are limited in learning long-term sequences. Hochreiter *et al.* [21] designed Long Short-Term Memory (LSTM) to exploit long-range dependency. According to a recent study, the Derivative of States (DoS) in differential long short-term memory (DLSTM) [8] can explicitly model spatio-temporal structure and better learn salient patterns within. Replacing internal state with the DoS in the gate units in LSTM, the DLSTM has the following updated equations:

(i) Input gate $\mathbf{i}_t$ regulates how much input information would enter the memory cell to affect its internal state $\mathbf{s}_t$ at time-step $t$. The activation of the input gate has the following recurrent form:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{id}\frac{d\mathbf{s}_{t-1}}{dt} + \mathbf{W}_{ih}\mathbf{h}_{t-1} + \mathbf{W}_{ix}\mathbf{x}_t + \mathbf{b}_i), \tag{2}$$

where $\sigma(\cdot)$ stands for a sigmoid activation function in the range [0, 1].

(ii) Forget gate $\mathbf{f}_t$ modulates the contribution of the previous state $\mathbf{s}_{t-1}$ to the current state. It is defined by the following equation as:

$$\mathbf{f}_t = \sigma(\mathbf{W}_{fd}\frac{d\mathbf{s}_{t-1}}{dt} + \mathbf{W}_{fh}\mathbf{h}_{t-1} + \mathbf{W}_{fx}\mathbf{x}_t + \mathbf{b}_f). \tag{3}$$

The internal state $\mathbf{s}_t$ of each memory cell can be updated using the input and forget gate units, which is shown in the equation below:

$$\mathbf{s}_t = \mathbf{f}_t \odot \mathbf{s}_{t-1} + \mathbf{i}_t \odot \widetilde{\mathbf{s}}_t, \tag{4}$$

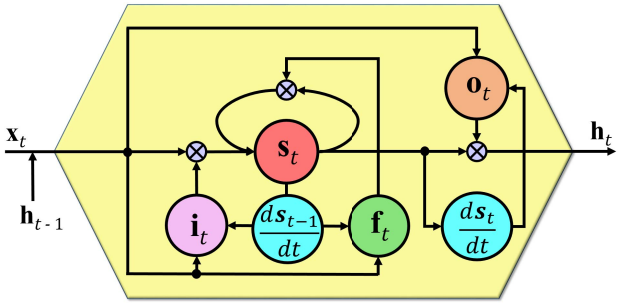


Fig. 2. An illustration of LSTM architecture at time $t$.

where $\odot$ stands for element-wise product. Pre-state $\widetilde{\mathbf{s}}_t$ is defined as:

$$\widetilde{\mathbf{s}}_t = tanh(\mathbf{W}_{sh}\mathbf{h}_{t-1} + \mathbf{W}_{sx}\mathbf{x}_t + \mathbf{b}_s).$$

(iii) Output gate $\mathbf{o}_t$ gates the information output from a memory cell and it affects the future states of DLSTM cells. The output gate can be expressed as:

$$\mathbf{o}_t = \sigma(\mathbf{W}_{od}\frac{d\mathbf{s}_t}{dt} + \mathbf{W}_{oh}\mathbf{h}_{t-1} + \mathbf{W}_{ox}\mathbf{x}_t + \mathbf{b}_o). \tag{5}$$

Then the hidden state of a memory cell is output as:

$$\mathbf{h}_t = \mathbf{o}_t \odot tanh(\mathbf{W}_{hs}\mathbf{s}_t + \mathbf{b}_h). \tag{6}$$

Fig. 2 gives an illustration of the architecture of DLSTM at time-step $t$.

### C. Stacked DLSTMs

As shown in Fig. 1, to investigate the temporal information in deep architectures, we adopt stacked DLSTMs up to three layers for ConvDLSTM. The advantage of stacked DLSTMs over single layer of DLSTM is as follows: for single layer of DLSTM, the input of DLSTM is the output of CNN, which contains only the spatial information; for deeper layers of DLSTMs in stack DLSTM setup, they take the output of previous DLSTMs as input sequences. Such inputs as each time-step contain both spatial and temporal information. In other words, the hidden states from previous layer of LSTM serve as the input sequence to the next layer, higher LSTM layers can capture abstract concepts in the sequences, which helps the whole system to better interpret the complex scene semantics and crowd dynamics.

For the last layer of DLSTMs, we consider mean-pooling, for the hidden states to generate a video-level representation. The last time-step hidden state was frequently used to denote a sequence level representation, *e.g.* [22]. As LSTMs process the input frames sequentially, hidden state information learned from previous frames decays gradually over a very long sequence. For crowd scene videos with large number of frames, it tends to be sub-optimal to only use the last time-step hidden state to learn the parameters of the neural networks. Max pooling is better at selecting salient signals and often generates a more discriminative representation. The problem of max pooling is it tends to be affected by motion noise. Mean

pooling averages all time-step hidden states, and statistically summarizes the information collected from all previous frames and thus has a more stable representation of the sequences.

Using mean pooling method, we acquire $\mathbf{h}_\tau$ from last layer of stacked DLSTMs as a video-level representation of a crowd video. $\mathbf{h}_\tau$ denotes a 1-of-$k$ encoding of the confidence scores on $k$ classes of crowd scenes. The confidence scores are then transformed to a vector of probabilities $\mathbf{p}$ by the softmax function:

$$p_c = \frac{\exp h_{T,c}}{\sum_{m=1}^{k} \exp(h_{T,m})}, \tag{7}$$

where each entry $p_c$ is the probability of input crowd video belonging to class $c \in \{1, 2, ..., k\}$.

### D. ConvDLSTM at time-step t and Learning Strategy

Given a crowd image sequence $\{m_1, m_2, ..., m_T\}$, Convolutional DLSTM proceeds as shown in Algorithm 1 at time step $t$. After $T$ time steps, $\mathbf{h}_\tau$ for the last layer of stacked DLSTMs is computed with mean pooling method. Given the video-level class $c$ of this crowd scene, compute crowd scene label probability $p_c$ by applying the softmax function Eq.(7). ConvDLSTM can then be trained by minimizing the loss function below, *i.e.*

$$\ell(\mathbf{p}, c) = -\log p_c.$$

The loss function can be minimized by Back Propagation Through Time (BPTT) [33], which unfolds an LSTM model over several time steps and then runs the back propagation algorithm to train the model. To prevent back-propagated errors from decaying or exploding exponentially, we use truncated BPTT according to Hochreiter *et al.* [21] to learn the model parameters.

**Algorithm 1** Convolutional DLSTM at time step $t$

1: Given image frame $m_t$, compute $\mathbf{x}_t$ via Eq.(1)
2: Compute input gate activation $\mathbf{i}_t$ and forget gate activation $\mathbf{f}_t$ by Eq. (2) and Eq. (3)
3: Update state $\mathbf{s}_t$ with $\mathbf{i}_t$ and $\mathbf{f}_t$ by Eq. (4)
4: Compute output gate $\mathbf{o}_t$ by Eq. (5)
5: Output $\mathbf{h}_t$ gated by $\mathbf{o}_t$ from memory cell by Eq. (6)
6: If there exists a deeper layer of LSTM, set $\mathbf{x}_t = \mathbf{h}_t$ for the following stacked LSTM and repeat steps 2 - 6

### E. Rationale behind Deep in Space and Time

ConvDLSTM optimizes the information flow of the spatial and temporal crowd scene dynamics in a unified model thus achieves "deep in space and time". The CNN layers in ConvDLSTM can model the scene information of crowd videos. LSTM can take advantage of the semantic CNN representation and analyze the long-term temporal dependency of crowd dynamics. Different from traditional non-end-to-end solutions which separate the steps of feature extraction and parameter learning, ConvDLSTM utilizes a unified deep model to optimize the parameters of CNN and LSTM hand in hand. It thus has the potential of generating a more harmonious model.

| | Class Name |
|---|---|
| 1 | Highly mixed pedestrian walking |
| 2 | Crowd walking following a mainstream and well organized |
| 3 | Crowd walking following a mainstream but pooly organized |
| 4 | Crowd merge |
| 5 | Crowd split |
| 6 | Crowd crossing in opposite directions |
| 7 | Intervened escalator traffic |
| 8 | Smooth escalator traffic |

TABLE I
LIST OF CROWD VIDEO CLASSES FOR THE CUHK CROWD DATASET.

The advantage of stacked LSTMs over single-layer of LSTM is also intuitive. For stacked LSTMs, first layer of stacked LSTMs takes the output of CNN. Such input is the same as single-layer LSTM and contains only the spatial information. Deeper layers of stacked LSTMs, however, take the output of previous LSTM as input sequences. Different from single-layer of LSTM, such inputs at each time step contain both spatial and temporal information thus are more comprehensive for understanding complex spatio-temporal structures. To conclude, ConvDLSTM is better at learning crowd scenes than non-end-to-end structures. In addition, stacked LSTMs possess superiority over single layer of LSTM. We will demonstrate these claims by experimental results.

## IV. EXPERIMENTS

In this section, we extensively evaluate the performances of the proposed method for crowd scene understanding on two public video datasets: Violent-Flows and CUHK Crowd. The Violent-Flows dataset [27] is a real-world video footage of crowd violence, along with standard benchmark protocols designed for violent/non-violent classification. The Violent-Flows dataset includes 246 real-world videos downloaded from YouTube. The shortest clip duration is 1.04 seconds, the longest clip is 6.52 seconds, and the average length of the videos is 3.6 seconds, with shortest/longest 1.04/6.52 seconds. The Violent-Flow dataset is designed for a five-fold cross-validation. Specifically, the video set is split into five sets: half the videos in each set portray are violent crowd behavior and half non-violent behavior. Five tests are performed: in each test, four of sets are used for training and the fifth set is used for testing. The CUHK Crowd dataset [12] consists of 474 crowd videos from over 200 crowded scenes, which were collected from many different environments, *e.g.* streets, shopping malls, airports, and parks. The videos are manually annotated into 8 different classes and the human trajectories are provided by the dataset. Details of 8 different classes of the CUHK Crowd dataset is shown in Table I.

### A. Experiment Setting and Training Strategy

To achieve better performance, the ConvDLSTM architecture is initialized with pre-trained parameters. We initialize GoogleNet Inception V3 in ConvDLSTM with parameters [26] trained on ImageNet in the ILSVRC-2014 competition. To initialize the parameters of stacked DLSTMs, we first freeze the Inception V3 weights in ConvDLSTM and then train the

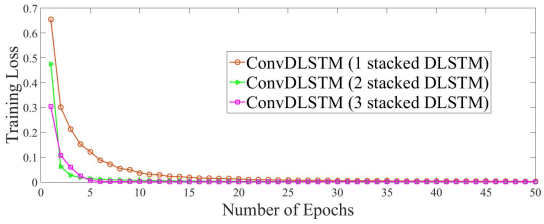


Fig. 3. Training loss vs. epochs on the Violent-Flows dataset.

model on unaugmented crowd videos. The number of memory cells for each DLSTM layer is determined as 1024 determined by cross-validation. RMSprop adaptive learning optimizer [1] with initial learning rate $1e-3$ is employed for pre-training the stacked DLSTMs. Original dataset is used for this step.

Since the ConvDLSTM architecture is large and complex, it suffers from high chance of overfitting. We thus perform data augmentation for both two datasets to increase the diversity of the training sequences. Random rotation, shear, zoom, horizontal flip, width, height and channel shifts are conducted on training instances. Given a crowd video sequence, the same augmentation is applied to all frames. The datasets Violent-Flows and CUHK Crowd are enlarged to 15 and 20 times of original sizes, respectively. All videos are resized to $299 \times 299$ pixels.

To train ConvDLSTM, we first initialize the weighting parameters as mentioned above and then unfreeze the weights in Mixed Inception Block 9, fully-connected layer and stacked DLSTM layers. The network is trained on augmented crowd image sequences using stochastic gradient descent method with mini-batch and a learning rate $1e-4$. Fig. 3 shows an example of the training loss through the iteration epochs on the Violent-Flows dataset. The implementation is done with Keras [34].

### *B. Experiments on the Violent-Flows Dataset*

To evaluate our method on the Violent-Flows dataset, we follow the standard 5-fold cross-validation protocol in [27] and report the results in terms of mean accuracy in Table II. It can be seen that our method achieves over 93% accuracy, outperforming state-of-the-art by more than 8%. Note that since Violent-Flows was released in 2012, several methods have been proposed to address the crowd understanding problem on this dataset, but only 4% performance improvement has been made over these years. Such an increase in performance achieved from our model demonstrates the effectiveness of ConvDLSTM.

We also compare in Table II the performances of ConvLSTM and ConvDLSTM. In ConvLSTM, we use traditional LSTMs instead of DLSTMs. Since the Derivative of States in DLSTM can explicitly model the information gain between successive frames, it is reasonable that ConvDLSTM outperforms ConvLSTM.

[1] http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

| Methods | Accuracy (%) |
|---|---|
| Violent Flows [27] | 81.30 |
| Common Measure [35] | 81.50 |
| Hist of Tracklet [14] | 82.30 |
| Substantial Derivative [36] | 85.43 |
| Holistic Features [3] | 85.53 |
| ConvLSTM | 91.34 |
| ConvDLSTM | **93.59** |

TABLE II
PERFORMANCE COMPARISON WITH EXISTING METHODS ON THE VIOLENT-FLOWS DATASET. FOR CONVLSTM AND CONVDLSTM, WE USE GOOGLENET INCEPTION V3 AS THE COVOLUTIONAL PART, THREE STACKED LSTM/DLSTMS, AND MEAN-POOLING FOR THE VIDEO-LEVEL REPRESENTATION.

Table III summarizes the performances of variants of ConvDLSTM architectures on the Violent-Flows dataset. To validate the advantage of the end-to-end ConvDLSTM model, we compare its performance with the solution using pre-trained CNN features as the input to DLSTMs. Results show that ConvDLSTM achieves higher performance than the non-end-to-end deep structures containing only DLSTMs. End-to-end solution employs a unified scheme to optimize the parameters of CNN and RNN hand in hand, thus generates a more harmonious model.

Table III also shows that larger number of stacked DLSTM achieves better performance. This can be explain as follows. For single-layer DLSTM, the input is the output of corresponding convolutional neural networks. Such input in certain time-step contains only spatial information. For deeper layers of stacked DLSTM, the input is the output of previous DLSTM, which in each time-step contains both spatial and temporal representations. Such high-level comprehensive representation contributes to stacked DLSTMs' better ability to understand complex crowd behaviors. We notice that the margin between two and three stacked DLSTMs is very small. The reason is with the increase of DLSTM layers, ConvDLSTM has higher chance of overfitting. The choice of two or three stacked DLSTMs depends on the trade-off of the selection of higher accuracy or faster detection.

We consider another variant in ConvDLSTM architecture that uses VGG-16 [37] instead of Inception V3 as the convolutional part, shown in Fig 4. We can see that ConvDLSTM with Inception V3 architecture constantly achieve higher performance than its variant using VGG architecture. Since the Inception blocks have lower number of parameters, training and testing for ConvDLSTM using Inception V3 are also faster.

### *C. Experiments on the CUHK Crowd Dataset*

We also evaluate our model on the CUHK Crowd dataset and report the results in Table IV. ConvDLSTM outperforms Collective Transition [12] and Coherent LSTM [13] for 10% and 4%, respectively. These two methods take human trajectories as input, which were provided by the dataset after manual correction. Considering that these two methods use "ground truth" human trajectories as input, ConvDLSTM's

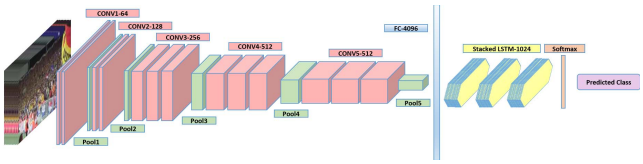


Fig. 4. An illustration of a variant of Convolutional DLSTM using VGG-16 as the convolutional part. The model consists of convolutional (red), pooling (green), fully-connected (blue), DLSTM (yellow) and softmax (brown) layers. **Best viewed in color**.

| | Methods | IV3 | VGG |
|---|---|---|---|
| **Non-End-to-End** | CNN feature + 1 stacked DLSTM | 86.34 | 84.75 |
| **Non-End-to-End** | CNN feature + 2 stacked DLSTM | 86.98 | 84.15 |
| **Non-End-to-End** | CNN feature + 3 stacked DLSTM | 87.03 | 85.18 |
| **End-to-End** | ConvDLSTM (1 stacked DLSTM) | 91.85 | 89.86 |
| **End-to-End** | ConvDLSTM (2 stacked DLSTM) | 93.50 | 91.09 |
| **End-to-End** | ConvDLSTM (3 stacked DLSTM) | 93.59 | 91.13 |

TABLE III
PERFORMANCE COMPARISON OF VARIANTS OF CONVDLSTM ARCHITECTURES ON THE VIOLENT-FLOWS DATASET. IV3 INDICATES INCEPTION V3.

performance is impressive as it does not require trajectory clues. Comparing the performances of Coherent LSTM and ConvDLSTM, we can see that ConvDLSTM has better ability in understanding crowd dynamics by achieving "deep in space and time".

Table IV also shows that ConvLSTM achieves slightly lower performance than ConvDLSTM. The conventional LSTMs do not consider the impact of spatio-temporal dynamics corresponding to the give salient motion patterns, when they gate the information that ought to be memorized through time. The weakness is addressed by the Derivative of States in DLSTMs.

Table V summarizes the performance of variants of ConvDLSTM architectures on the CUHK Crowd dataset. Firstly, similar to the Violent-Flows dataset, ConvDLSTM outperforms the non-end-to-end solution. Non-end-to-end solutions separates the steps feature extraction and parameter learning. ConvDLSTM unifies these two processes and thus generates better performances. Secondly, deeper stacked LSTMs acquire higher classification accuracy. Deeper layers of stacked DLSTM take the output of previous DLSTM layer, which contains spatio-temporal information in each step. Such information is more comprehensive in understanding complex crowd dynamics. Thirdly, ConvDLSTM using VGG-16 achieves lower performance than the one using GoogleNet Incpetion V3. VGG-16 architecture. The above results are consistent with the results on the Violent-Flows dataset.

| Methods | Accuracy (%) |
|---|---|
| Collective Transition [12] | 70.00 |
| Un-coherent LSTM [13] | 73.82 |
| Coherent LSTM [13] | 76.50 |
| ConvLSTM | 78.65 |
| ConvDLSTM | **80.33** |

TABLE IV
PERFORMANCE COMPARISON WITH EXISTING METHODS ON THE CUHK CROWD DATASET. FOR CONVLSTM AND CONVDLSTM, WE USE GOOGLENET INCEPTION V3 AS THE COVOLUTIONAL PART, THREE STACKED LSTM/DLSTMS, AND MEAN-POOLING FOR THE VIDEO-LEVEL REPRESENTATION.

| | Methods | IV3 | VGG |
|---|---|---|---|
| **Non-End-to-End** | CNN feature + 1 stacked DLSTM | 73.35 | 71.06 |
| **Non-End-to-End** | CNN feature + 2 stacked DLSTM | 73.68 | 72.94 |
| **Non-End-to-End** | CNN feature + 3 stacked DLSTM | 75.83 | 73.43 |
| **End-to-End** | ConvDLSTM (1 stacked DLSTM) | 78.32 | 76.15 |
| **End-to-End** | ConvDLSTM (2 stacked DLSTM) | 79.68 | 77.48 |
| **End-to-End** | ConvDLSTM (3 stacked DLSTM) | 80.33 | 78.03 |

TABLE V
PERFORMANCE COMPARISON OF VARIANTS OF CONVDLSTM ARCHITECTURES ON THE CUHK CROWD DATASET. IV3 INDICATES INCEPTION V3.

| Methods | Violent-Flows | CUHK Crowd |
|---|---|---|
| Holistic Features [3] | 85.53 | 70.75 |
| Coherent LSTM [13] | 84.23 | 76.65 |
| Our method | 93.59 | 80.33 |

TABLE VI
EVALUATION OF MODEL GENERALIZATION CAPABILITY.

### *D. Evaluation of Model Generalization Capability*

To evaluate ConvDLSTM's capability of understanding various crowd scene scenarios, we carefully implement the Holistic Features [3] and Coherent LSTM [13] methods and compare them with ConvDLSTM on both the above two datasets. For Coherent LSTM, we use the same gKLT tracker [38] as the CUHK Crowd dataset to generate the human trajectories for the Violent-Flows dataset.

As shown in Table VI, our ConvDLSTM outperforms the two methods on both two datasets. Holistic Features [3] performs uncomparably on the CUHK Crowd dataset because it uses simple handcrafted features of only four dimensions, which are not optimized for different crowd scenarios. Although Coherent LSTM [13] works decently on CUHK dataset, it performs unsatisfactorily on the Violent-Flows dataset. In the Violent-Flows dataset, crowd density is much higher. This could result in generating inaccurate human trajectories, which then serve as inputs for Coherent LSTM. In addition, there exists less crowd motion flow in Violent-Flows. As a trajectory-based method, Coherent LSTM makes no use of scene information and tends to be less effective. ConvDLSTM, however, requires no trajectory detection and takes into consideration of scene semantics, thus possesses better generalization capability for crowd scene understanding.

## V. CONCLUSION

In this paper, we propose an end-to-end deep architecture Convolutional LSTM (ConvDLSTM) for crowd scene understanding. Our model consists of convolutional neural networks and stacked long short-term memory recurrent neural networks. ConvDLSTM directly takes the raw image sequences as the input and does not require additional handcrafted flow-based or trajectory-based feature representation and works

with crowds of high density and low mobility. Performance studies on two public crowd datasets have shown that the proposed technique significantly outperforms state-of-the-art methods.

## REFERENCES


[1] Teng Li, Huan Chang, Meng Wang, Bingbing Ni, Richang Hong, and Shuicheng Yan, "Crowded scene analysis: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 367–386, 2015.

[2] Tuoerhongjiang Yusufu, Naifan Zhuang, Kai Li, and Kien A Hua, "Relational learning based happiness intensity analysis in a group," in *IEEE International Symposium on Multimedia (ISM)*. IEEE, 2016, pp. 353–358.

[3] Mark Marsden, Kevin McGuinness, Suzanne Little, and Noel E O'Connor, "Holistic features for real-time crowd behaviour anomaly detection," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 918–922.

[4] Ramin Mehran, Alexis Oyama, and Mubarak Shah, "Abnormal crowd behavior detection using social force model," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 935–942.

[5] Shandong Wu, Brian E Moore, and Mubarak Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2054–2060.

[6] Ramin Mehran, Brian E Moore, and Mubarak Shah, "A streakline representation of flow in crowded scenes," in *European Conference on Computer Vision*. Springer, 2010, pp. 439–452.

[7] Xiaofei Wang, Xiaomin Yang, Xiaohai He, Qizhi Teng, and Mingliang Gao, "A high accuracy flow segmentation method in crowded scenes based on streakline," *Optik-International Journal for Light and Electron Optics*, vol. 125, no. 3, pp. 924–929, 2014.

[8] Pierre-Marc Jodoin, Yannick Benezeth, and Yi Wang, "Meta-tracking for video scene understanding," in *Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on*. IEEE, 2013, pp. 1–6.

[9] Yang Cong, Junsong Yuan, and Ji Liu, "Abnormal event detection in crowded scenes using sparse representation," *Pattern Recognition*, vol. 46, no. 7, pp. 1851–1864, 2013.

[10] Louis Kratz and Ko Nishino, "Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 5, pp. 987–1002, 2012.

[11] Louis Kratz and Ko Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1446–1453.

[12] Jing Shao, Chen Change Loy, and Xiaogang Wang, "Scene-independent group profiling in crowd," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2219–2226.

[13] Hang Su, Yinpeng Dong, Jun Zhu, Haibin Ling, and Bo Zhang, "Crowd scene understanding with coherent recurrent neural networks.," in *IJCAI*, 2016, pp. 3469–3476.

[14] Hossein Mousavi, Sadegh Mohammadi, Alessandro Perina, Ryad Chellali, and Vittorio Murino, "Analyzing tracklets for the detection of abnormal crowd behavior," in *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2015, pp. 148–155.

[15] Bolei Zhou, Xiaogang Wang, and Xiaoou Tang, "Random field topic model for semantic region analysis in crowded scenes from tracklets," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3441–3448.

[16] Bolei Zhou, Xiaogang Wang, and Xiaoou Tang, "Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2871–2878.

[17] Wang Chongjing, Zhao Xu, Zou Yi, and Liu Yuncai, "Analyzing motion patterns in crowded scenes via automatic tracklets clustering," *china communications*, vol. 10, no. 4, pp. 144–154, 2013.

[18] Chongjing Wang, Xu Zhao, Zhe Wu, and Yuncai Liu, "Motion pattern analysis in crowded scenes based on hybrid generative-discriminative feature maps," in *Image Processing (ICIP), 2013 20th IEEE International Conference on*. IEEE, 2013, pp. 2837–2841.

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[20] Saad Ali, *Taming crowded visual scenes*, Ph.D. thesis, Citeseer, 2008.

[21] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] Vivek Veeriah, Naifan Zhuang, and Guo-Jun Qi, "Differential recurrent neural networks for action recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4041–4049.

[23] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese, "Social lstm: Human trajectory prediction in crowded spaces," 2016.

[24] Jun Ye, Kai Li, Guo-Jun Qi, and Kien A Hua, "Temporal order-preserving dynamic quantization for human action recognition from multimodal sensor streams," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. ACM, 2015, pp. 99–106.

[25] Naifan Zhuang, Jun Ye, and Kien A Hua, "Dlstm approach to video modeling with hashing for large-scale video retrieval," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE, 2016, pp. 3222–3227.

[26] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[27] Tal Hassner, Yossi Itcher, and Orit Kliper-Gross, "Violent flows: Real-time detection of violent crowd behavior," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2012, pp. 1–6.

[28] Min Hu, Saad Ali, and Mubarak Shah, "Detecting global motion patterns in complex videos," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–5.

[29] HU Min, S ALI, and M SHAH, "Learning motion patterns in crowded scenes using motion flow field," in *International Conference on Pattern Recognition (ICPR)*, 2008.

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

[31] Saad Ali and Mubarak Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–6.

[32] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[33] Manuel P Cuéllar, Miguel Delgado, and MC Pegalajar, "An application of non-linear programming to train recurrent neural networks in time series prediction problems," in *Enterprise Information Systems VII*, pp. 95–102. Springer, 2007.

[34] François Chollet et al., "Keras," https://github.com/fchollet/keras, 2015.

[35] Hossein Mousavi, Moin Nabi, Hamed Kiani, Alessandro Perina, and Vittorio Murino, "Crowd motion monitoring using tracklet-based commotion measure," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2354–2358.

[36] Sadegh Mohammadi, Hamed Kiani, Alessandro Perina, and Vittorio Murino, "Violence detection in crowded scenes using substantial derivative," in *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*. IEEE, 2015, pp. 1–6.

[37] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[38] Bolei Zhou, Xiaoou Tang, and Xiaogang Wang, "Measuring crowd collectiveness," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3049–3056.