MSCDP: Multi-Step Crowd Density Predictor in Indoor Environment

Shuyu Wang
Southeast University
Yan Lyu (lvyanly@seu.edu.cn)
Southeast University
Yuhang Xu
Southeast University
Weiwei Wu
Southeast University

Research Article

Keywords: indoor environment, crowd density distribution, multi-step prediction, video frames, density heatmaps, optical flow

Posted Date: October 6th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2119562/v1

MSCDP: Multi-Step Crowd Density Predictor in Indoor Environment

Shuyu Wang[1,3*], Yan Lyu[2*], Yuhang Xu[2†] and Weiwei Wu[2†]

[1*]School of Cyber Science and Engineering, Southeast University, Nanjing, 211189, Jiangsu, China.
[2]School of Computer Science and Engineering, Southeast University, Nanjing, 211189, Jiangsu, China.
[3]School of Information Engineering, Xizang Minzu University, Xian'yang, 712082, Shaanxi , China.

*Corresponding author(s). E-mail(s): shywang@seu.edu.cn; lvyanly@seu.edu.cn;
Contributing authors: yuhang_xu@seu.edu.cn; weiweiwu@seu.edu.cn;
†These authors contributed equally to this work.

Abstract

Monitoring and predicting crowd movements in indoor environments are of great importance in crowd management to prevent crushing and trampling. Existing works mostly focused on individual trajectory forecasting in a less crowded scene, or crowd counting and density estimation. Only a very few works predict the crowd density distribution. However, this study is failing to realize multi-step prediction or exploiting only density heatmaps modality ignores the information complementation with corresponding video frames. Therefore, we are motivated to predict crowd density distribution in multiple time steps to facilitate long-term prediction. In this paper, a Multi-Step Crowd Density Predictor (MSCDP) to fuse video frame sequences and corresponding density heatmaps, is proposed to accurately forecast the future crowd density heatmaps. To capture long-term periodic movement features, the long-term optical flow context memory (LOFCM) module is designed to store learnable patterns. We conducted extensive experiments on two real-world datasets. Evaluation results show that our MSCDP outperforms the state-of-the-art baseline techniques and MSCDP variants in terms

of various prediction errors, demonstrating the effectiveness of MSCDP and each of its key components in multi-step crowd density prediction.

Keywords: indoor environment, crowd density distribution, multi-step prediction, video frames, density heatmaps, optical flow

1 Introduction

Monitoring and predicting crowds movements in indoor environments such as subway or railway stations [1–3], airport terminals [4] and shopping malls [5] is of great importance in crowd management to prevent crushing and trampling [6], and also help with interior design [7, 8]. With the help of surveillance cameras, how the crowd distributed in an indoor environment can be easily estimated in real-time with techniques of crowd counting [9–15] or crowd density estimation [14]. However, this only helps with monitoring the current situation, accurately predicting crowd density in the future will be more useful to take early action to avoid overcrowding [16–21].

Although there have been works on predicting individual trajectory in a less crowd scene [22–26], these methods can hardly be applied to predict the dense crowd due to the difficulty in tracking individuals in a dense crowd. Only a few works predict the crowd density distribution. Niu et. al [18] proposed a crowd density prediction framework. To have a long-term prediction, they sampled frames with equal long-term time intervals, failing to generalize to a longer time interval (e.g., 5 minutes) prediction. The other work PDFN-ST [19] forecasts future crowdedness using 3D convolutions to learn local crowd density dynamics only with historical density heatmaps, failing to include the detailed pedestrian motions from video frames that may also help predict crowd movements. Therefore, we are motivated to predict crowd density distribution in multiple time steps to facilitate long-term prediction. Specifically, we focus on an indoor environment with dense crowds in which individuals are hard to track. This is challenging three-fold: 1) Surveillance crowd videos usually have crowded pedestrian objects that are small, without clear appearance and posture, and occluded with each other. The crowd is also in dynamic and arbitrary shape without a clear boundary. It is difficult to mathematically model crowd motions like individual motion [27, 28]. 2) Crowd movements are usually highly dynamic with randomness in both spatial and temporal dimensions. Pedestrians in an indoor environment don't have lanes to walk along, the trajectories can be arbitrary and goals are quite diverse compared to those walking on the streets. 3) The multiple-step prediction problem itself is challenging, as the prediction error could accumulate when predicting step by step. Given the high dynamic and arbitrary crowd movement, it is even more difficult to have an accurate prediction in the multiple future time step.

In this paper, we propose a Multi-Step Crowd Density Predictor (MSCDP) to predict crowd density distribution (i.e., density heatmap) in multiple time

steps to facilitate long-term prediction. Instead of directly modeling crowd movements themselves, we leverage the density estimation techniques [14, 29–32] to capture historical crowd spatial and temporal distribution dynamics. We also utilize *optical flow* between subsequent video frames as well as that between density maps to capture local temporal dynamics. Although crowd movement could be dynamic with some randomness, we still observed some movement patterns appears periodically, i.e., a similar bundle of trajectories start in close locations to similar directions (see Fig. 1). These patterns can be quite common in railway stations where passengers periodically get off the train and walk to exits, and shopping malls where customers periodically get off the elevator and walk directly to popular shops. This motivates us to utilize such long-term patterns to help with density prediction. Particularly, we leverage a long-term motion context memory technique [33] to capture the long-term crowd movement patterns and adapt the memory alignment learning network [33] to improve prediction accuracy for detailed density heatmap. Different from the original network input, we apply the network to the input of original video frames and their corresponding density heatmaps, and propose attention-based feature fusion. We also propose to use optical flow, instead of frame difference, to represent motion dynamics in our MSCDP network. In summary, our contributions are

- Proposed a Multi-Step Crowd Density Predictor (MSCDP) that fuses motion features from both historical video frames and corresponding density heatmaps with gating and spatial attention techniques to predict crowd density heatmaps in future multiple time steps.
- Adapted the optical flow based long-term context memory learning to capture both long-term and short-term optical flow dynamics for both video frames and crowd density heatmaps.
- Evaluation on two real-world datasets shows that our MSCDP outperforms the state-of-the-art techniques and MSCDP variants in terms of prediction errors, demonstrating the effectiveness of MSCDP and each of its key components in predicting future crowd density heatmaps.

The rest of this paper is organized as follows: Section 2 reviews the related work. Section 3 formally defines multi-step prediction problems and proposes our MSCDP model and its critical components. We present evaluation in Section 4 and conclude this paper in Section 5.

2 Related Work

We discuss related works including video frame prediction, crowd counting and density estimation, and trajectory prediction in this section.

2.1 Video frame prediction

The task of video frame prediction is to use historical video sequences to predict upcoming video frames at the pixel level. Deep learning-based video

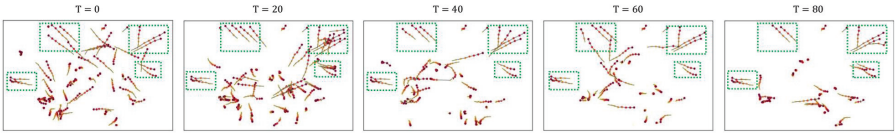

Fig. 1 Pedestrian trajectories from time T to $T+2.4s$, where T is the $0, 20, \cdots, 80$ seconds of the video in GC dataset [34]. Each trajectory is plotted by dots from small light orange to big dark red, indicating its movement direction. We can observe similar movement patterns appear periodically, i.e., similar bundles of trajectories (in green boxes) of a different group of pedestrians appear every 20 seconds in the video.

prediction methods have achieved some satisfactory results. It is first studied to predict future frames [35] based on RNNs. This kind of model [36] merely focuses on exploring the temporal dependency but ignores the spatial features of videos, and also the temporal receptive field of the predictive model is narrow. To solve the problem of ignoring the spatial features, some works [37–39] add spatial information processing modules. To solve the problem of the temporal receptive field, some works [40], [33] capture reliable inter-frame motion information by broadening the temporal receptive field. These kinds of video prediction tasks are usually a scene with a single target and simple background. Our approach is similar but different from such technologies. Since our task is to predict crowd density distribution in the future, we consider inputting frame sequences and corresponding crowd density maps to realize multi-step crowd density prediction. There are two kinds of data modalities to feed into the net in our task, so we also have to think about data fusion.

2.2 Trajectory prediction

Trajectory prediction is to predict a pedestrian's next location from a sequence of historical locations detected from videos. Social-LSTM [22] and social-GAN [24] proposed by Feifei Li's team are typical representatives of solving trajectory prediction tasks, which are different from Social Force Model [27] with artificially designed social interaction features. Unlike these studies, [22, 24], future trajectories are influenced not only by other pedestrians in the environment but also by the physical environment or their goals. Amir [41] proposed a trajectory prediction network named Sophie that considers the interaction of various agents and constraints of physical scenes at the same time. By learning BicycleGAN [42], Kosaraju [43] proposes a potential spatial encoder that can be used to explain the generation of multimodal agent trajectories. Y-net [44] models the epistemic un-certainty through multimodality in long-term goals and the aleatoric un-certainty through multimodality in waypoint & paths to predict trajectories with prediction horizons up to a minute. However, all these works assume that accurate pedestrian trajectories are given. This is not always a realistic assumption when one wants to forecast the behaviors of a dense crowd or tracking every pedestrian is computationally expensive and redundant for some particular applications. For example, some applications only need to know where and when crowd congestion occurs,

so that unsafe events caused by high crowd density can be avoided. Without any exact position or identity information of the individuals, resolving crowd density prediction is of great significance.

2.3 Crowd counting and density estimation

Crowd counting and density estimation have been studied to help with safety monitoring and crowd management. Detection-based counting methods deploy a detector to traverse the image, which localizes and counts the targets along the way [45, 46]. These methods are surpassed by the regression-based alternatives, as the detection performance is affected in the presence of overcrowded scenes. Nevertheless, regression-based methods [47, 48] forfeit localization capability such that they cannot perceive crowd distributions. Crowd counting methods based on density estimation are therefore developed by conducting pixel-wise regressions [14, 29–32]. The characteristic of this research is that the input is the current video frame, and its output is the crowd counting and density map estimation at the current moment. Our work is to predict crowd density distribution in future time steps. We need density maps of the corresponding video frames as our input sequences by these methods. Crowd density prediction can be viewed as a downstream task in this study area.

2.4 Crowd density prediction

There are only a few works on predicting crowd density in the future. Niu et al. [18] formulated a novel yet challenging crowd distribution prediction problem. To benefit long-term prediction, the provided frames of the crowd video are sampled over an equal interval. To utilize frame and density map sequences, they proposed a global-residual two-stream network for predicting only one-step crowd density. However, to achieve long-term prediction, by sampling video frames and corresponding density maps, critical information in the temporal dimension is lost. PDFN-ST [19] is a network to forecast future crowdedness by utilizing historical density maps without using the corresponding video frames, thereby losing detailed pedestrian motion information that also helps to improve prediction accuracy. In contrast, our MSCDP utilizes and fuses both video frames and estimated density heatmaps to predict crowd density in future multiple-time steps. It adapts the long-term context memory learning [33] to capture the long-term movement patterns of crowds to improve multi-step prediction accuracy.

3 Methodology

3.1 Problem Definition

Given a sequence of crowd video frames from a surveillance camera, the goal is to forecast crowd density in the next few time steps. To help with accurate prediction, we consider both original video frames as well as their corresponding *density heatmap* as input to facilitate output *density heatmap* in the future.

Density Heatmap. We estimate the crowd density of a video frame using the state-of-the-art crowd counting technique C^3F [32], then render the estimated densities into RGB maps as the input of density heatmaps.

Problem Definition. Let d_t be a crowd density heatmap extracted from the t-th camera video frame i_t of size (H, W), $i_t \in \mathbb{R}^{3\times H\times W}$ and $d_t \in \mathbb{R}^{3\times H\times W}$ are both saved in RGB format. We denote inputs sequences of length T_{in} as frame sequences $I_{in} = [i_{t-T_{in}+1}, ..., i_t] \in \mathbb{R}^{T_{in}\times 3\times H\times W}$ and density heatmap sequences $D_{in} = [d_{t-T_{in}+1}, ..., d_t] \in \mathbb{R}^{T_{in}\times 3\times H\times W}$, and denote output density heatmap sequences of length T_{out} as $\hat{D}_{out} = [\hat{d}_{t+1}, ..., \hat{d}_{t+T_{out}}] \in \mathbb{R}^{T_{out}\times 3\times H\times W}$, we can formulate the problem as:

$$\hat{D}_{out} = \mathbb{F}(I_{in}, D_{in}) \tag{1}$$

Table 1 Variables and explanations

Variables	Explanations
MSCDP	Multi-Step Crowd Density Predictor
I_{in}	Short-term video frame sequences, the length is T_{in}
D_{in}	Short-term Density heatmap sequences, the length is T_{in}
$\hat{D}_{out}$	Density heatmaps of MSCDP prediction, the length is T_{out}
D_{out}	Ground truth of density heatmaps
T_{in}	Input step size of raw frames in test phase
T_{out}	Output step size of density heatmaps
H_F	Spatiotemporal features extracted from the frame sequences
H_D	Spatiotemporal features extracted from the density heatmap sequences
F^{F_m}	motion feature from frame branch memory units
F^{D_m}	motion feature from density heatmap branch memory units
$\tilde{F}^{F\_m}$	Refined motion feature from frame branch memory units
$\tilde{F}^{D\_m}$	Refined motion feature from density heatmap branch memory units
F^{DF}	Gating feature
$\tilde{F}_{fusion}$	Fusion feature
LOFCM	Long-term optical flow context memory
I_N^{long}	Long-term frame sequences, the length is N
D_N^{long}	Long-term density heatmap sequences, the length is N
V^{F_LOFC}	Optical flow feature from long-term frame optical flow sequences
V^{D_LOFC}	Optical flow feature from long-term density heatmap optical flow sequences
M_F	The optical flow memory unit matrix for the frame branch
M_D	The optical flow memory unit matrix for the density heatmap branch
V^{F_SOFC}	Optical flow feature from short-term frame optical flow sequences
V^{D_SOFC}	Optical flow feature from short-term density heatmap optical flow sequences

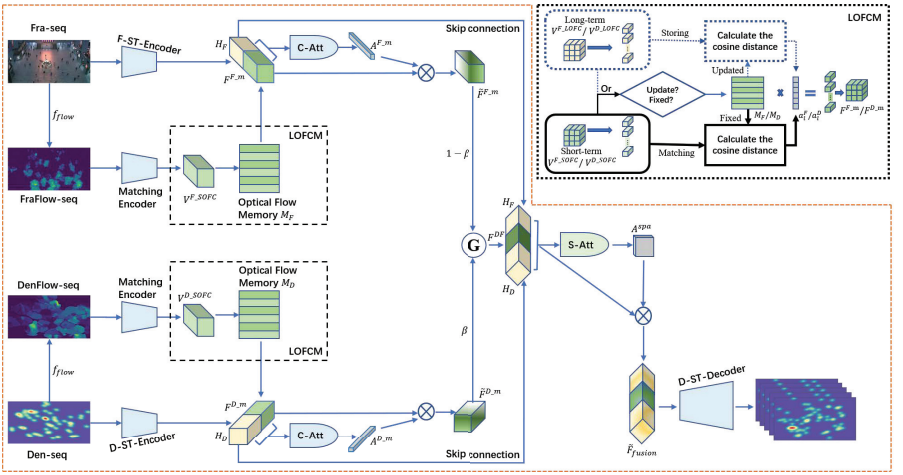

Fig. 2 Overall framework of MSCDP network for crowd density prediction at inference phase. The frame sequence and density heatmap sequence both have long-term optical flow context memory respectively. F-ST-Encoder and D-ST-Encoder have the same structure to extract spatial-temporal features. Optical flow matching encoder and optical flow encoder (training phase) have the same structure for matching and storing learnable dynamic optical flow motion patterns respectively. C-Att and S-Att are adopted to refine features. Gating is utilized to merge features. The dotted line shows the process that is only available during training in LOFCM.

3.2 Overview of Network Architecture

Fig. 2 shows the overall framework of the proposed MSCDP architecture at the inference phase. The frame and density heatmap sequence respectively is represented as Fra-seq and Den-seq. By respectively utilizing optical flow methods [49] to obtain optical flow sequence denoted as FraFlow-seq and DenFlow-seq. Fra-seq(FraFlow-seq) and Den-seq(DenFlow-seq) respectively go through two paths. One is for capturing spatial-temporal features H_F or H_D, and the other is for matching the LOPCM module to obtain optical flow context motion feature F^{F_m} or F^{D_m}. Then optical flow context motion feature F^{F_m} or F^{D_m} and spatial-temporal features H_F or H_D are concatenated, and pass through C-Att to make channel-wise attention respectively to obtain refined optical flow motion feature $\tilde{F}^{F\_m}$ and $\tilde{F}^{D\_m}$. And then Gating unit is applied to merge feature vector $\tilde{F}^{F\_m}$ and $\tilde{F}^{D\_m}$ to obtain gating feature F^{DF}. Subsequently, spatial-temporal features H_F, H_D and gating feature F^{DF} are concatenated and fed to a spatial attention module S-Att to obtain refined fusion feature $\tilde{F}_{fusion}$. At last the refined feature $\tilde{F}_{fusion}$ is fed to a density spatial-temporal decoder to generate future time steps density heatmaps. The variables and representations in the paper and their explanation are shown in Table 1.

3.3 Spatial-temporal Encoder

Our framework MSCDP first encodes spatial-temporal features from both video frames and the corresponding density heatmaps with two 3D convolution networks [50] as encoders, named F-ST-Encoder and D-ST-Encoder, respectively. Specifically, we adapt the well-known spatial-temporal 3D convolution network C3D [50]. C3D is well-suited for spatial-temporal feature learning, it convolves and pools in both the 2D spatial dimension and the temporal dimension at the same time to capture spatial-temporal features of crowd movements.

We adapt C3D into 6 convolution layers and 5 pooling layers for both F-ST-Encoder and D-ST-Encoder. The number of filters for 6 convolution layers from 1 to 6 are 64, 128, 256, 256, 512, 512,respectively. We also refer to 3D convolution and pooling kernel size $d \times k \times k$, where d is kernel temporal depth and k is kernel spatial size. All of these convolution layers are applied with appropriate padding (both spatial and temporal) and $stride = 1$. The first 4 pooling layers are max pooling with kernel size $2 \times 2 \times 2$ except for the first layer with kernel size $1 \times 2 \times 2$, the first pooling layer has kernel size $1 \times 2 \times 2$ with the intention of not to merge the temporal signal too early. The last pooling layer is adaptive average pooling to obtain spatial-temporal features. We denote F-ST-Encoder and D-ST-Encoder as E_{F_ST} and E_{D_ST}. The following Table 2 describes specific network settings. The input frame sequences and density heatmap sequences go through the F-ST-Encoder and D-ST-Encoder to obtain the spatial-temporal features, which can be formulated as

$$H_F = E_{F\_ST}(I_{in}) \tag{2}$$

$$H_D = E_{D\_ST}(D_{in}) \tag{3}$$

3.4 Long-term Optical Flow Context Memory

As crowd movement in an indoor environment has shown long-term and periodical patterns (Fig. 1), we adopt long-term context memory alignment leaning [33] to capture and utilize such periodical patterns to help with density prediction. The long-term context memory [33] was proposed to predict video frames by storing long-term motion patterns, such as walking and running, in a video into memory, and matching short-term motions, such as the movement of a pedestrian's leg to its long-term memory to identify its motion context. Both long-term and short-term motions, however, were encoded from frame difference, which may be unable to capture movement directions of small individual objects in crowd videos. Therefore we propose to use optical flow to represent motion dynamics for both long-term context memory and short-term memory alignments. Specifically, we propose Long-term Optical Flow Context Memory (LOFCM) module with the input of optical flow extracted from both video frames and their corresponding density heatmaps. We also train LOFCM with two phases: 1) *Storing Phase* to store long-term optical

Table 2 F-ST-Encoder/D-ST-Encoder module settings

Layer No.	Styles	Filters	Kernel size	Stride
1	3D-Convs	64	$3 \times 3 \times 3$	$1 \times 1 \times 1$
2	Relu		/	
3	Pooling	/	$1 \times 2 \times 2$	$1 \times 2 \times 2$
4	3D-Convs	128	$3 \times 3 \times 3$	$1 \times 1 \times 1$
5	Relu		/	
6	Pooling	/	$2 \times 2 \times 2$	$2 \times 2 \times 2$
7	3D-Convs	256	$3 \times 3 \times 3$	$1 \times 1 \times 1$
8	Relu		/	
9	3D-Convs	256	$3 \times 3 \times 3$	$1 \times 1 \times 1$
10	Relu		/	
11	Pooling	/	$2 \times 2 \times 2$	$2 \times 2 \times 2$
12	3D-Convs	512	$3 \times 3 \times 3$	$1 \times 1 \times 1$
13	Relu		/	
14	3D-Convs	512	$3 \times 3 \times 3$	$1 \times 1 \times 1$
15	Relu		/	
16	Pooling	/	$2 \times 2 \times 2$	$2 \times 2 \times 2$
17	AdaptiveAvgPool3d			

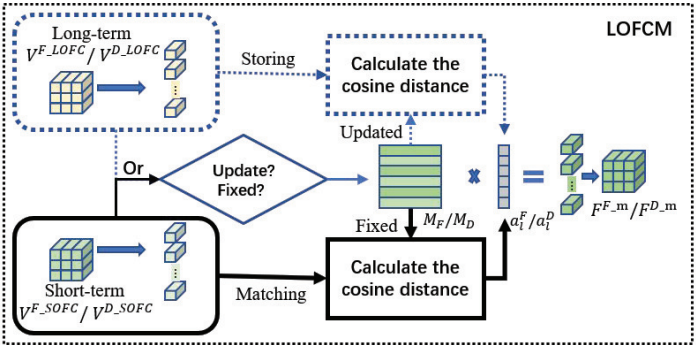

Fig. 3 The calculation process of Long-term Optical Flow Context Memory (LOFCM). F^{F_m} and F^{D_m} are optical flows from video frames and density heatmaps, respectively. In storing phase, the module takes a long sequence of optical flows from video frames or density heatmaps, and memorizes the patterns into matrix M_F or M_D; In the matching phase, a corresponding short-term optical flow sequence is used to match the memory matrix. The training process of the network consists of both storing and matching phases, but the matching phase is only used for outputting prediction results in testing.

flow context patterns into a memory matrix and 2) *Matching Phase* to match short-term optical flow to the long-term memory. Fig. 3 demonstrates a more detailed computation flow.

3.4.1 Storing Phase

At the storing phase, we offer two long-term sequences I_N^{long} and D_N^{long} with length N from the training data. After obtaining the optical flow sequences

from consecutive frames or density heatmaps, the optical flow context feature of the long-term sequence is captured by an optical flow context encoder. We utilize a typical feature extractor, C3D[50] with 3D-Convs for E_{F_LOFC} and E_{D_LOFC}. The long-term optical flow context feature $V^{F\_LOFC} = \{v_l^{F\_LOFC}\}_{l=1}^{w\times h} \in \mathbb{R}^{w\times h\times c}$ and $V^{D\_LOFC} = \{v_l^{D\_LOFC}\}_{l=1}^{w\times h} \in \mathbb{R}^{w\times h\times c}$ are divided into local vectors to exploit decomposed dynamics. The local vector $v_l^{F\_LOFC} \in \mathbb{R}^c$ and $v_l^{D\_LOFC} \in \mathbb{R}^c$ are used as a memory query individually. Calculate optical flow sequences from frames I_N^{long} and density heatmaps D_N^{long} denote as $f_{flow}(\cdot)$. The optical flow context features from two long-term sequences can be denoted as

$$V^{F\_LOFC} = E_{F\_LOFC}(f_{flow}(I_N^{long})) \tag{4}$$

$$V^{D\_LOFC} = E_{D\_LOFC}(f_{flow}(D_N^{long})) \tag{5}$$

Frame and density heatmap sequences both have LOPCM. The parameters of frame branch memory, $M_F = \{m_i^{fra}\}_{i=1}^{s} \in \mathbb{R}^{s\times c}$ with s slot size and c channels. The memory vector $m_i^{fra} \in \mathbb{R}^c$ denotes a item of M_F. The parameters of density heatmap branch memory, $M_D = \{m_i^{den}\}_{i=1}^{s} \in \mathbb{R}^{s\times c}$ with s slot size and c channels. The memory vector $m_i^{den} \in \mathbb{R}^c$ denotes a item of M_D.

The optical flow sequences are fed into optical flow context encoder E_{F_LOFC} and E_{D_LOFC}, to obtain query $v_l^{F\_LOFC}$ and query $v_l^{D\_LOFC}$, the weights of the frame branch memory slots denoted by $a_l^F = \{a_{l\_i}^{fra}\}_{i=1}^{s} \in \mathbb{R}^s$, the weights of the density heatmap branch memory slots denoted by $a_l^D = \{a_{l\_i}^{den}\}_{i=1}^{s} \in \mathbb{R}^s$. Memory mechanism can be formulated as

$$a_{l\_i}^{fra} = \frac{exp(d(v_l^{F\_LOFC}, m_i^{fra}))}{\sum_{j=1}^{s} exp(d(v_l^{F\_LOFC}, m_j^{fra}))} \tag{6}$$

$$a_{l\_i}^{den} = \frac{exp(d(v_l^{D\_LOFC}, m_i^{den}))}{\sum_{j=1}^{s} exp(d(v_l^{D\_LOFC}, m_j^{den}))} \tag{7}$$

where $d(\cdot,\cdot)$ indicates cosine similarity function and $exp(\cdot)/\sum exp(\cdot)$ denotes softmax function. The memory cell outputs optical flow context motion feature $f_l^{fra} \in \mathbb{R}^c (l = 1, 2, ..., w \times h)$ and $f_l^{den} \in \mathbb{R}^c (l = 1, 2, ..., w \times h)$for each location l as follows

$$f_l^{fra} = \sum_{i=1}^{s} a_{l\_i}^{fra} m_i^{fra} \tag{8}$$

$$f_l^{den} = \sum_{i=1}^{s} a_{l\_i}^{den} m_i^{den} \tag{9}$$

Finally, optical flow motion features can be denoted as

$$F^{F\_m} = \{f_l^{fra}\}_{l=1}^{w\times h} \in \mathbb{R}^{w\times h\times c} \tag{10}$$

$$F^{D\_m} = \{f_l^{den}\}_{l=1}^{w\times h} \in \mathbb{R}^{w\times h\times c} \tag{11}$$

where F^{F_m} and F^{D_m} are concatenated with spatial-temporal features H_F and H_D from frame sequences encoder and density heatmap sequences encoder individually to proceed next processing.

During training of storing phase, the parameters of optical flow memory units are updated through backpropagation. Long-term optical flow context motion patterns can be stored in the memory by training the networks from long-term input sequences (both frame sequences and density map sequences).

3.4.2 Matching Phase

At the matching phase, the network receives two short-term sequence I_{in} and D_{in} with length T_{in} (long-term length N, short-term length T_{in}, prediction length T_{out}, The relationship between them is $N \geq T_{in} + T_{out}$). The information in the optical flow memory is to be matched by two short-term optical context features. The short-term optical flow sequences are fed into optical flow matching encoder $E_{M\_F\_SOFC}$ and $E_{M\_D\_SOFC}$ to obtain optical flow motion features. $E_{M\_F\_SOFC}$, $E_{M\_D\_SOFC}$, E_{F_LOFC} and E_{D_LOFC} have the same C3D[50] structure, but do not share parameters with each other. The memory mechanism procedures are the same as the storing phase (Eq.4, Eq.5, Eq.6 and Eq.7). Calculate optical flow sequences from frames I_{in} and density heatmaps D_{in} denote as $f_{flow}(\cdot)$. The optical flow feature from two short sequences can be denoted as

$$V^{F\_SOFC} = E_{M\_F\_SOFC}(f_{flow}(I_{in})) \tag{12}$$

$$V^{D\_SOFC} = E_{M\_D\_SOFC}(f_{flow}(D_{in})) \tag{13}$$

Unlike the storing phase, the parameters of optical flow memory are not updated. Except for the optical flow memory, other parameters of the whole network are updated. During the testing, only short-term sequences (both frames and density heatmaps) I_{in} and D_{in} are fed into network MSCDP.

3.5 Adaptive Feature Fusion

3.5.1 Channel-wise Attention

Channel-wise Attention named C-Att, is designed to refine optical flow motion features F^{F_m} and F^{D_m}. Since spatial-temporal features H_F and H_D contain the dynamic information from the past to the present of the input sequences, we adopt H_F and H_D from the frame and density heatmap sequences respectively, to refine optical flow motion features for embedding the required dynamic information.

The left of Fig. 4 demonstrates a more detailed computation flow. The yellow block represents the spatial-temporal feature from the frame or density heatmap sequences, the green block represents the optical flow motion feature from the frame or density heatmap sequences. $\tilde{F}^{F\_m}$ and $\tilde{F}^{D\_m}$ are denoted refined optical flow motion feature. Spatial-temporal features H_F and H_D are concatenated with optical flow motion features F^{F_m} and F^{D_m} respectively. The concatenated features are first passed through an adaptive average pooling layer to decompose channel-wise feature vectors, and then fed into fully

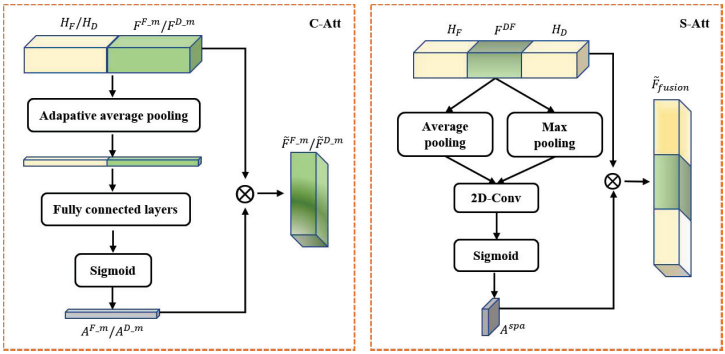

Fig. 4 The calculation process of channel-wise attention (left) and spatial attention (right).

connected layers to output channel-wise attention weights A^{F_m} and A^{D_m}. The refined optical flow motion features from frames and density heatmaps respectively can be formulated as

$$\tilde{F}^{F\_m} = A^{F\_m} \times F^{F\_m} \tag{14}$$

$$\tilde{F}^{D\_m} = A^{D\_m} \times F^{D\_m} \tag{15}$$

3.5.2 Gating

Gating is applied to merge feature vectors. First, we apply a gating unit to automatically tune how much the prediction relies on frame branch optical flow motion feature $\tilde{F}^{F\_m}$ or density heatmap branch optical flow memory feature $\tilde{F}^{D\_m}$. The gating feature takes a trade-off between two kinds of optical flow motion features with a trainable gate weight parameter β, i.e.,

$$F^{DF} = \beta \tilde{F}^{D\_m} + (1 - \beta)\tilde{F}^{F\_m} \tag{16}$$

3.5.3 Spatial Attention

Spatial Attention named S-Att is designed to obtain refined concatenated feature vectors. The right of Fig. 4 demonstrates a more detailed computation flow. The yellow block represents the spatial-temporal feature from the frame or density heatmap sequences, and the green block represents the gating feature F^{DF}. A^{spa} is denoted spatial attention weights. $\tilde{F}_{fusion}$ is denoted refined fusion feature. We apply average-pooling and max-pooling operations along the channel axis and concatenate them to generate an efficient feature vector. Applying pooling operations along the channel axis is shown to be effective in highlighting informative regions [51]. On the concatenated feature vector, we apply a convolution layer to generate a spatial attention map that encodes where to emphasize or suppress. Spatial-temporal features H_F and H_D are concatenated with gating features F^{DF}. Since spatial-temporal features H_F and H_D contain the dynamic information from the past to the present of the input sequences, we adopt H_F and H_D from the frame and density heatmap

sequences respectively by skip connection to remain dynamic information. We concatenate the features from the frame and density heatmap streams, as shown in Fig 2. The fusion method of direct splicing often introduces redundant information. The features pass through the S-Att module to obtain refined features. This process can be formulated as

$$A^{spa} = S\text{-}Att[H_F;\ F^{DF};\ H_D] \tag{17}$$

$$\tilde{F}_{fusion} = A^{spa} \times [H_F;\ F^{DF};\ H_D] \tag{18}$$

3.6 Density Prediction Decoder

The fusion feature $\tilde{F}_{fusion}$ is fed into a decoder named D-ST-Decoder denoted as D_{D_ST}. D-ST-Decoder with stacked ConvTranspose-2D layers is to generate corresponding future density heatmaps $\hat{d}_{t+1}$

$$\hat{d}_{t+1} = D_{D\_ST}(\tilde{F}_{fusion}) \tag{19}$$

The module has 6 ConvTranspose-2D layers. The number of filters for 6 ConvTranspose-2D layers from 1 to 6 are 512, 256, 128, 64, 32, 3, respectively. All of these ConvTranspose-2D layers are applied with appropriate padding and stride. Since the optical flow motion features can provide the prior of long-term optical flow context, and spatial-temporal features can provide the historical dynamic information for the current input, the generated density heatmap $\hat{d}_{t+1}$ as a new input to create the further future density heatmap sequences $\hat{D}_{out} = [\hat{d}_{t+1}, ..., \hat{d}_{t+T_{out}}]$.

3.7 Loss function

The loss function is referred to[52] named Huber loss, which is a loss function used in robust regression, which is less sensitive to outliers in data than the squared error loss. The predicted and ground truth density heatmaps are denoted as $\hat{D}_{out} = [\hat{d}_{t+1}, ..., \hat{d}_{t+T_{out}}]$ and $D_{out} = [d_{t+1}, ..., d_{t+T_{out}}]$. We define the loss function for the MSCDP network with the help of two sub-losses, referred to as 1) Consistency loss $\mathcal{L}_c$: to rebuff any inconsistency between $\hat{D}_{out}$ and D_{out}. 2) Forward-difference flow loss $\mathcal{L}_f$: to maintain the trend of forwarding density heatmaps flow.

$$\mathcal{L}_c = \left\| \hat{D}_{out} - D_{out} \right\|_{huber} \tag{20}$$

$$\mathcal{L}_f = \left\| \Delta\hat{D}_{out} - \Delta D_{out} \right\|_{huber} \tag{21}$$

$$\arg\min_{\theta} \mathcal{L} = \mathcal{L}_c + \mathcal{L}_f \tag{22}$$

In the above equations, Δ denotes forward-difference between multiple time-step. We compute the final loss $\mathcal{L}$ as the sum value of the two sub-losses.

If one step prediction network is training, $\mathcal{L}_f$ is removed. where θ is a set of all trainable parameters in the network. Note our method only takes short-term frame sequences and short-term density map sequences as inputs at test processing as shown in Fig. 2. The whole training procedure is described in Algorithm 1.

The training procedure of MSCDP: Given short-term and long-term density heatmap sequences $D_{in} = D_{t-T_{in}+1:t}$, $D_N^{long} = D_{t-N+1:t}$, and short-term and long-term frame sequences $I_{in} = I_{t-T_{in}+1:t}$, $I_N^{long} = I_{t-N+1:t}$. Through iterative training, the parameters of the MSCFP network are optimized to reach convergence to obtain optimum MSCDP(θ). Firstly initialize parameters of MSCDP(θ), frame branch memory size $M1$, and density heatmap branch memory size $M2$, then we train iteratively, each iteration consisting of two-phase: storing and matching.

In **storing phase**, we use long term frame and density heatmap sequences to extract spatial-temporal features H_F and H_D (line 4, Eq. 2 and 3). Then obtain long-term optical flow features V^{F_LOFC} and V^{D_LOFC} by optical flow encoders (line 5, Eq. 4 and 5). And next obtain optical flow motion features F^{F_m} and F^{D_m} (line 6, Eq. 10 and 11). Refine optical flow motion features to obtain $\tilde{F}^{F\_m}$ and $\tilde{F}^{D\_m}$ by channel-wise attention (line 7, Eq. 14 and 15). Then merge two branches' refined features by gating to obtain F^{DF} (line 8, Eq. 16). And then obtain refined fusion feature $\tilde{F}_{fusion}$ by skipping connection and spatial attention (line 9, Eq.17 and 18). Then the refined fusion feature feed into the decoder to obtain prediction density maps $\tilde{D}_{T_{out}}$ (line 10-12). Then adjust network's weights and biases by computing loss (line 13, Eq. 20, 21, 22). Last update parameters of MSCDP(θ, $M1$ and $M2$) (line 14)

In **matching phase**, we use short-term frame and density heatmap sequences to extract spatial-temporal features H_F and H_D (line 16, Eq.2 and 3). Then obtain short-term optical flow features V^{F_SOFC} and V^{D_SOFC} by optical flow matching encoders (line 17, Eq. 12 and 13). And next obtain optical flow motion features F^{F_m} and F^{D_m} (line 18, Eq. 10 and 11). Refine optical flow memory features to obtain $\tilde{F}^{F\_m}$ and $\tilde{F}^{D\_m}$ by channel-wise attention (line 19, Eq. 14 and 15). Then merge two branches refined features by gating to obtain F^{DF} (line 20, Eq. 16). And then obtain refined fusion feature $\tilde{F}_{fusion}$ by skipping connection and spatial attention (line 21, Eq. 17 and 18). Then the refined fusion feature feed into the decoder to obtain prediction density heatmaps $\tilde{D}_{T_{out}}$ (line 22-24). Then adjust network's weights and biases by computing loss (line 25, Eq. 20, 21, 22). Last update parameters of MSCDP(θ, except $M1$ and $M2$) (line 26).

4 Experiment

We conducted experiments on two real-world datasets taken by a fixed surveillance camera in an indoor environment and then compared our proposed networks with a set of baseline models as well as variants with different components. In the following, we first introduce the two datasets, pre-processing

Algorithm 1 Training Procedure of MSCDP

Require: Short-term and long-term density heatmap sequence $D_{in} = D_{t-T_{in}+1:t}$, $D_N^{long} = D_{t-N+1:t}$, short-term and long-term frame sequence $I_{in} = I_{t-T_{in}+1:t}$ and $I_N^{long} = I_{t-N+1:t}$, and learning rate α, where $N \geq T_{in} + T_{out}$

Ensure: Optimum MSCDP(θ)

Initialize parameters of MSCDP(θ), $M1$ and $M2$.
while each iteration **do**
  < **Phase 1: Storing** >
  Get $H_F = E_{F\_ST}(I_N^{long})$ and $H_D = E_{D\_ST}(D_N^{long})$
  Get storing features $V^{F\_LOFC} = E_{F\_LOFC}(f_{flow}(I_N^{long}))$ and $V^{D\_LOFC} = E_{D\_LOFC}(f_{flow}(D_N^{long}))$
  Get optical flow context memory features F^{F_m} and F^{D_m} by LOPCM
  Get refined optical flow context memory features $\tilde{F}^{F\_m}$ and $\tilde{F}^{D\_m}$ by C-Att unit
  Get gating feature $F^{DF} = \beta\tilde{F}^{D\_m} + (1-\beta)\tilde{F}^{F\_m}$
  Get $\tilde{F}_{fusion}$ =S-Att[H_F; F^{DF}; H_D]
  while $i = 0, 1, ..., T_{out}$ **do**
    Get $\tilde{D}_{T_{out}} = D_{D\_ST}(\tilde{F}_{fusion})$
  end while
  $\mathcal{L} \leftarrow \mathcal{L}^{pred}(\tilde{D}_{T_{out}}, D_{T_{out}})$
  Update $\theta(include\ M1, M2) \leftarrow \theta - \alpha\nabla_\theta\mathcal{L}$
  < **Phase 2: Matching** >
  Get $H_F = E_{F\_ST}(I_{in})$ and $H_D = E_{D\_ST}(D_{in})$
  Get matching features $V^{F\_SOFC} = E_{M\_F\_SOFC}(f_{flow}(I_{in}))$ and $V^{D\_SOFC} = E_{M\_D\_SOFC}(f_{flow}(D_{in}))$
  Get optical flow context memory features F^{F_m} and F^{D_m} by LOPCM
  Get refined optical flow context memory features $\tilde{F}^{F\_m}$ and $\tilde{F}^{D\_m}$ from short sequence by C-Att unit
  Get gating feature $F^{DF} = \beta\tilde{F}^{D\_m} + (1-\beta)\tilde{F}^{F\_m}$
  Get $\tilde{F}_{fusion}$ =S-Att[H_F; F^{DF}; H_D]
  while $i = 0, 1, ..., T_{out}$ **do**
    Get $\tilde{D}_{T_{out}} = D_{D\_ST}(\tilde{F}_{fusion})$
  end while
  $\mathcal{L} \leftarrow \mathcal{L}^{pred}(\tilde{D}_{T_{out}}, D_{T_{out}})$
  Update $\theta(except\ M1, M2) \leftarrow \theta - \alpha\nabla_\theta\mathcal{L}$
end while

data, baseline models, evaluation metrics, and implementation settings and discuss experimental results.

4.1 Datasets

1) **GC** [34] is a dataset collected from Grand Central Station in New York. The videos were originally collected by [53], and later the trajectories dataset

was created by [34]. The dataset is much larger to meet the requirements of deep learning, it contains both video frames and density heatmaps that can be converted from trajectory data. It is a crowd surveillance dataset that is difficult and challenging for vision tasks.

2) **Mall** [54] dataset was captured using a publicly accessible surveillance camera in a shopping mall. It also covers more diverse crowd densities from sparse to crowded, as well as different activity patterns (static and moving crowds) under a larger range of illumination conditions at different times of the day. The details of the two datasets are summarized in Table 3.

Table 3 Statistics of crowd datasets

Dataset	Mall	GC
Number of annotated frames	2000	5000
Resolution	640 × 480	1920 × 1080
Annotated frame rate [Hz]	<2	1.25
Total number of annotated persons	62,325	12,684

4.2 Data Preprocessing

Crowd Density Estimation. We adopt a pretrained network C^3 [32] with ShanghaiTech Part_B dataset[55] to estimate the degree of crowdedness per pixel directly from input video frames. We then render the estimated densities into RGB heatmaps as one of the inputs of our network.

Optical Flow Extraction. We adopt Lucas–Kanade [56] method to calculate optical flows from video frames and density heatmaps, respectively.

4.3 Baseline Methods

- **ConvLSTM** (2015) [57] is a recurrent network that also captures spatial dependencies within single recurrent unit. It has been widely used to predict video frames [58] as well as city-wide traffic flows [59].
- **SAM** (2020) [60] improves ConvLSTM with a self-attention memory and has achieved high prediction accuracy in predicting videos of human actions and traffic flows.
- **Alert** (2020) [18] predicts crowd density from both crowd video frames and the estimated density heatmaps. It achieves long-term prediction by having sparsely sampled frames with long time intervals.
- **PDFN-ST** (2020) [19] predicts crowd density by decomposing the density heatmap into spatially overlapping patches and learning their latent representation.
- **LMC** (2021) [33] introduces the long-term motion context memory with memory alignment learning. It performs well in predicting video frames of human actions such as walking and running. We adapt this work to predict

more complex crowd density with optical flow rather than frame difference as input in our MSCDP.

To further evaluate whether the key components used in our model are useful to the studied problem, we also compare the full version of MSCDP with the following variants.

- **MSCDP w/o Fra** removes the frame sequence branch, this is to evaluate whether the frame sequence branch can help extract useful features for crowd density prediction.
- **MSCDP w/o Den** removes the density heatmap sequence branch. This is to evaluate whether density heatmap sequence branch can help extract useful features for crowd density prediction, and verify the contribution of video frames and their corresponding density heatmaps to the density heatmap prediction by comparing with MSCDP w/o Fra.
- **MSCDP w/o D_m** removes the density long-term optical flow context memory module. This is to evaluate whether the proposed density optical flow memory module can help store long-term dynamic features and thus improve the multi-step crowd density prediction.
- **MSCDP w/o F_m** removes the frame long-term optical flow context memory module. This is to evaluate whether the proposed frame optical flow memory module can help store long-term dynamic features and thus improve the prediction performance.
- **MSCDP w/o C-Att** removes C-Att unit. This is to evaluate whether the C-Att unit can help refine the density or frame optical flow memory features from two sequences and thus improve the model performance.
- **MSCDP w/o S-Att** removes S-Att unit. This is to evaluate whether the S-Att unit can help refine the concatenated features from two sequences and thus improve the model performance.

4.4 Evaluation Metrics

We evaluate the performance of the prediction methods with Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [61], and LPIPS [62]. MSE and PSNR are calculated by the pixel-wise difference between the ground truth and the predicted density heatmaps. We also evaluate the performance using SSIM which considers the structural similarity between ground truth and prediction. We utilize LPIPS as a perceptual metric, which tends to be similar to the human recognition system [62]. Higher values are better for PSNR and SSIM while lower values are better for MSE and LPIPS.

4.5 Implementation and Hyperparameter Settings

Experiments were conducted on a Linux server with one Intel(R) Core(TM) i9-10940X CPU @ 3.30GHZ and NVIDIA GeForce RTX3090 GPU 24GB card. We split each of the two datasets into a training set with the first 80%, testing

sets with trips in the last 10%, and a validation set with the rest. In the network training, Adam [63] is used for optimizing trainable parameters with a batch size of 8. The learning rate is set to 0.0002, and the training epoch is 10000. A validation-based early stopping mechanism [64] is applied to avoid over-fitting by stopping the training process when the MAE of validation does not decrease in ten successive epochs.

In experiments, we set the short-term length as 9 frames in matching phase for all the prediction tasks, and the long-term length as 10, 14, and 19 frames in storing phase for one-step, five-step, and ten-step prediction tasks, respectively. The optical flow memory size $M1 = M2 = 50$. The parameters of baseline methods are set based on the original papers. To compare the prediction performance, we also set input sequence length $n = 9$ in the baseline methods.

4.6 Experiment Results

We discuss performance of our MSCDP compared with that of baseline techniques as well as MSCDP variants. We also qualitatively discuss prediction performance by visualizing prediction outcomes.

1) MSCDP versus Baseline Methods. Table 4 and 5 summaries crowd prediction performances of comparison methods on both GC and Mall datasets. The best results are highlighted in bold font. We observe that our MSCDP achieves the best performance among all the methods on both tasks. ConvLSTM, SAM, PDFN-ST, and LMC showed degraded performances on both datasets. These results demonstrate that frame sequences can provide useful dynamic spatial-temporal information for crowd prediction. The inputs of these four baseline models are only historical density heatmap sequences. Compared to Alert model, the input sequence has both density heatmap sequences and frame sequences, the model is only applicable to one-step prediction. And the one-step prediction results illustrate our MSCDP can better extract crowd flow dynamic information and further realize multi-step prediction. ConvLSTM, SAM and LMC have the ability to extract spatio-temporal features. With the same configuration, the performance of LMC method is more prominent.

2) MSCDP versus its variants. To understand how each component in MSCDP is helpful to the prediction task, we compare MSCDP with its variants MSCDP w/o Fra, MSCDP w/o Den, MSCDP w/o D_m, MSCDP w/o F_m, MSCDP w/o C-Att and MSCDP w/o S-Att. The result is in Table 6. We can observe that density optical flow memory, frame optical flow memory, frame sequences, density heatmap sequences, channel-wise attention and spatial attention are all useful to the model as removing any one of them will increase the prediction error. In both datasets, the density heatmap sequences seem more important for density heatmap prediction because the prediction error increases remarkably when the density heatmap sequences are ignored. Combining these components achieves the lowest MSE, demonstrating that all of them are useful to the studied problem. MSCDP w/o Fra and MSCDP

Table 4 Comparison Between MSCDP and Baseline Methods on GC Dataset

	Metric	ConvLSTM	SAM	Alert	PDFN-ST	LMC	MSCDP
One step	MSE	190.864	200.538	184.074	151.504	128.677	**98.572**
	PSNR	18.916	13.590	20.602	21.536	22.266	**24.058**
	SSIM	0.783	0.744	0.818	0.832	0.860	**0.890**
	LPIPS	0.213	0.351	0.222	0.299	0.236	**0.150**
Five steps	MSE	195.695	275.024	-	158.445	130.490	**124.595**
	PSNR	20.375	9.550	-	21.346	22.216	**22.414**
	SSIM	0.795	0.550	-	0.833	0.846	**0.849**
	LPIPS	0.271	0.357	-	0.308	0.266	**0.213**
Ten steps	MSE	204.349	276.660	-	169.978	146.124	**141.498**
	PSNR	20.207	9.570	-	21.074	21.712	**21.887**
	SSIM	0.796	0.550	-	0.830	0.842	**0.844**
	LPIPS	0.283	0.358	-	0.309	0.284	**0.265**

Table 5 Comparison Between MSCDP and Baseline Methods on MALL Dataset

	Metric	ConvLSTM	SAM	Alert	PDFN-ST	LMC	MSCDP
One step	MSE	223.741	260.189	184.528	161.472	159.805	**146.889**
	PSNR	18.979	11.939	20.567	21.186	21.221	**21.593**
	SSIM	0.821	0.444	0.866	0.852	0.868	**0.869**
	LPIPS	0.222	0.289	0.209	0.332	0.290	**0.176**
Five steps	MSE	236.454	279.364	-	191.406	184.900	**174.472**
	PSNR	19.507	12.160	-	20.481	20.628	**20.876**
	SSIM	0.818	0.439	-	0.847	0.857	**0.860**
	LPIPS	0.251	0.290	-	0.330	0.302	**0.223**
Ten steps	MSE	242.006	288.102	-	197.810	185.022	**176.184**
	PSNR	19.817	11.974	-	20.832	20.859	**20.859**
	SSIM	0.804	0.416	-	0.854	0.861	**0.861**
	LPIPS	0.259	0.292	-	0.340	0.302	**0.256**

w/o Den further prove that both density heatmap and frame sequences can extract useful spatio-temporal dynamic information for crowd prediction. The variants MSCDP w/o D_m and MSCDP w/o F_m showed degraded performances compared to the whole MSCDP. These results illustrate long-term optical flow context memory w/o LOFCM units are powerful in multi-step crowd prediction. MSCDP w/o C-Att illustrates C-Att module can refine optical flow memory features to remove redundant information. MSCDP w/o S-Att illustrates concatenated features often bring in redundant information, S-Att module can refine concatenate features to remove redundant information.

4) Visual analysis on prediction results. To further understand prediction quality, we visualize the predicted density maps on one step and five steps from both GC and Mall datasets.

Table 6 Comparison Between MSCDP and Variants with Five Steps Prediction

	Variants	MSE	PSNR	SSIM	LPIPS
GC	MSCDP w/o Den	188.266	20.625	0.838	0.298
	MSCDP w/o Fra	128.076	22.281	0.845	0.269
	MSCDP w/o D_m	131.199	22.199	0.846	0.264
	MSCDP w/o F_m	128.346	22.284	0.846	0.262
	MSCDP w/o C-Att	129.490	22.390	0.846	0.274
	MSCDP w/o S-Att	130.940	22.179	0.845	0.275
	MSCDP	**124.595**	**22.414**	**0.849**	**0.213**
MALL	MSCDP w/o Den	197.742	20.330	0.851	0.296
	MSCDP w/o Fra	202.079	20.241	0.844	0.311
	MSCDP w/o D_m	203.203	20.240	0.846	0.307
	MSCDP w/o F_m	204.819	20.204	0.846	0.307
	MSCDP w/o C-Att	174.916	20.873	0.860	0.299
	MSCDP w/o S-Att	175.401	20.866	0.853	0.305
	MSCDP	**174.472**	**20.876**	**0.860**	**0.223**

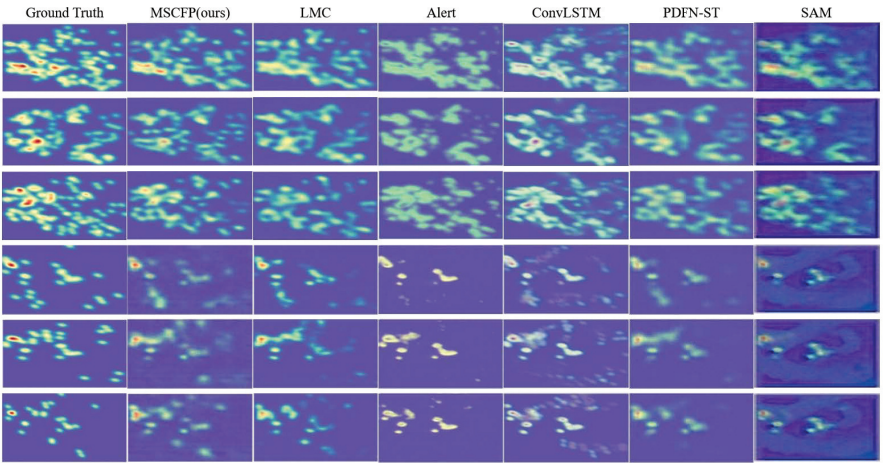

Fig. 5 Qualitative results with given 9 steps historical sequences on the GC (first three-row) and Mall (second three-row) datasets. The result is a future one-step density map. MSCDP is our model and others are baselines from recent years. The first column is the true value, and the next six columns represent experimental results for different models.

One-step prediction. Fig. 5 visualizes prediction results of the GC dataset (first three rows) and the Mall dataset (the second three rows). By comparing the highest density of ground truth to predictions of MSCDP and baseline models, we can see MSCDP model can better predict the location and peak of the highest density crowd, however, the peak prediction of the baseline models is not as good as ours. For low-density crowds, the predictions of baseline models are inaccurate but MSCDP still predicted accurately (see top right corner of the first row maps in Fig. 5).

Five-step prediction. Fig. 6 and Fig. 7 visualize five-step prediction results, in which, the first row is the ground truth density heatmap on the future time step, the second row is what our model predicts and the third to sixth rows show the predicted results of the baseline models. The results show that our method can better predict the location and peak of high-density crowds. We can observe that a high-density crowd appears in the upper left area from $T = 1$ to $T = 5$ (ground truth row in Fig. 7), and our MSCDP clearly and accurately predicted the location and peak of high-density crowds. Although the baseline methods are able to predict the location of these crowds but fail to predict the peak density values accurately. We also observe that a pedestrian appears in the upper right corner at time $T = 4$ (ground truth row in Fig. 6). Although our method does not predict this pedestrian at $T = 4$, we predict it at $T = 5$. However, all other methods fail to predict the appearance of this pedestrian.

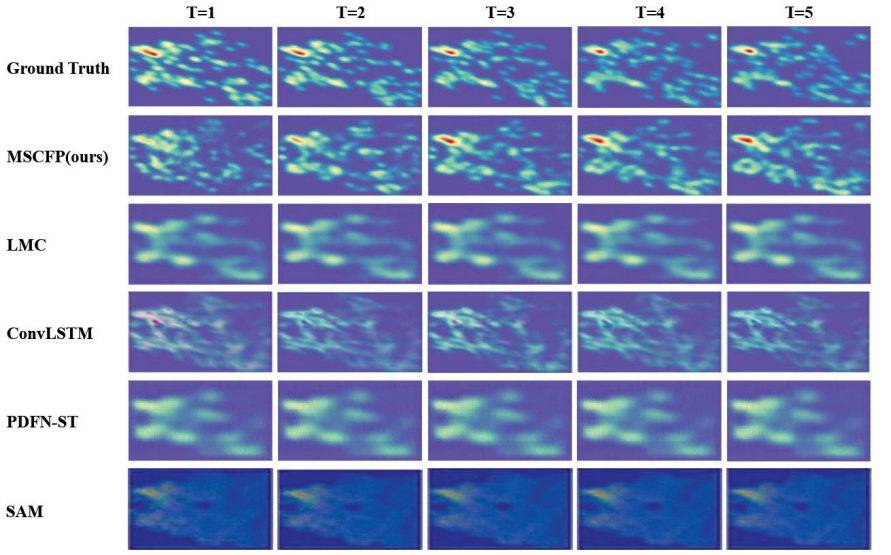

Fig. 6 Crowd density prediction visualizations on the GC dataset. Each row is predicted density heatmaps in five-time steps of different methods, except the first row visualizes ground truth.

4) Impact of the parameters. In our MSCDP model, there are two important hyperparameters $M1$ and $M2$, which represent the size of storage space for long-term optical flow patterns. We set them equal because of the symmetry of the input data. $M1$ and $M2$ are of great significance for multi-step prediction tasks. The experimental results of quantitative analysis are shown in Fig. 8. and Fig. 9. We can see that with the increase of $M1$ and $M2$ parameters, the four indicators are improved, but too large $M1$ and $M2$ cannot continue to improve the four indicators. This demonstrates that too

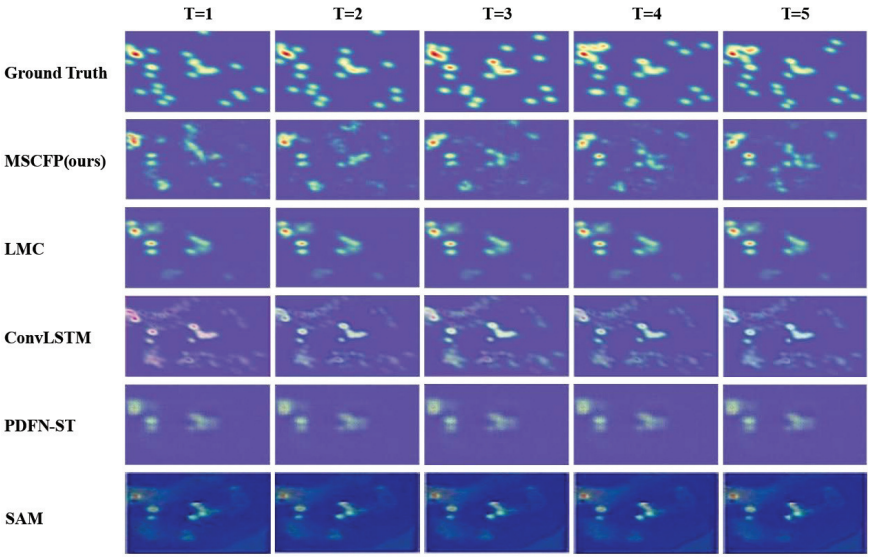

Fig. 7 Crowd density prediction visualizations on the Mall dataset. Each row is predicted density heatmaps in five-time steps of different methods, except the first row visualizes ground truth.

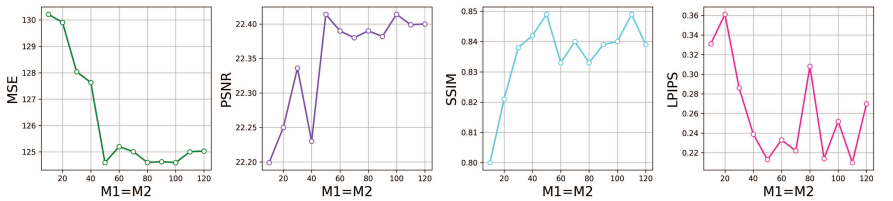

Fig. 8 The influence of M1 and M2 on GC.

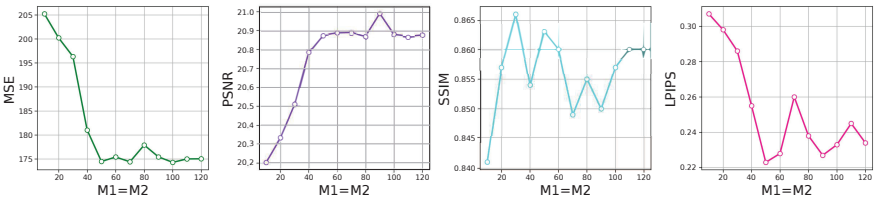

Fig. 9 The influence of M1 and M2 on MALL.

small $M1$ and $M2$ are not enough to store the common long-term optical flow patterns, and too large $M1$ and $M2$ may not be able to effectively capture the long-term pattern and fail to improve the prediction accuracy.

5 Conclusion

We propose a Multi-Step Crowd Density Predictor (MSCDP) to predict crowd density distribution in future multiple-time steps. MSCDP takes that both video frames and their corresponding density heatmaps as input, and encodes them with 3D convolution networks and a long-term optical flow context memory (LOFCM) module to capture motion dynamics in both long-term and short-term from their optical flows. Gating mechanism fuses two-stream features and spatial attention module refines concatenated features. Evaluation results have shown our proposed method outperforms the state-of-the-art crowd density prediction techniques on two real world datasets. Our work can be further improved by 1) modeling social interaction of each crowd and 2) introducing inflow and outflow of the indoor environment.

Acknowledgments The work described in this paper was partially sponsored by the National Key Research and Development Program of China under grant No. 2019YFB2102200, Natural Science Foundation of China under Grant No.62102082, 62263028, 62262062, the Project 11771365 supported by NSFC, the Natural Science Foundation of Jiangsu Province of China (BK20210203), the Postgraduate Research & Practice Innovation Program of Jiangsu Province of China (KYCX19_0089), partially sponsored by the Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, and partially sponsored by the Natural Science Foundation of Xizang of China (XZ202001ZR0046G, XZ202001ZR0065G).

6 Declarations

Ethical Approval This declaration is not applicable to our research.

Competing interests All authors declare that No conflict of interest exists.

Authors' contributions Shuyu Wang designed the network, conducted main experiments, and wrote the first original draft preparation. Yan Lyu revised the original manuscript and put forward the idea of the subject research. Yuhang Xu conducted part of the data processing and experimental design. Weiwei Wu proposed the research questions and supervised the whole research process.

Funding The National Key Research and Development Program of China (Grant No.2019YFB2102200), the Natural Science Foundation of China (No.62102082), and the Natural Science Foundation of Jiangsu Province of China (BK20210203), and provide the research direction, as well as resources such as highly configured servers needed to carry out experiments. The Natural Science Foundation of China (No.62263028), the Natural Science Foundation of China (No.62262062), the Natural Science Foundation of Xizang of China (XZ202001ZR0065G), and the Natural Science Foundation of Xizang of China (XZ202001ZR0046G) provide the first author of this research paper, as the

principal project participant, with partial hardware resources to carry out the research.

Availability of data and materials Two publicly available datasets are used in our study. Datasets are published in the literature [34] and [54].

References

[1] Gu, J., Jiang, Z., Fan, W.D., Wu, J., Chen, J.: Realtime passenger flow anomaly detection considering typical time series clustered characteristics at metro stations. Journal of Transportation Engineering, Part A: Systems **146**(4), 04020015 (2020)

[2] King, D., Srikukenthiran, S., Shalaby, A.: Using simulation to analyze crowd congestion and mitigation at canadian subway interchanges: case of bloor-yonge station, toronto, ontario. Transportation Research Record **2417**(1), 27–36 (2014)

[3] Zhou, Q., Zhang, J., Che, L., Shan, H., Wang, J.Z.: Crowd counting with limited labeling through submodular frame selection. IEEE Transactions on Intelligent Transportation Systems **20**(5), 1728–1738 (2018)

[4] Liu, M., Jiang, J., Guo, Z., Wang, Z., Liu, Y.: Crowd counting with fully convolutional neural network. In: IEEE International Conference on Image Processing, pp. 953–957 (2018)

[5] Liu, J., Gao, C., Meng, D., Hauptmann, A.G.: Decidenet: Counting varying density crowds through attention guided detection and density estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5197–5206 (2018)

[6] Yan, L., Tong, W., Hui, D., Zongzhi, W.: Research and application on risk assessment dea model of crowd crushing and trampling accidents in subway stations. Procedia engineering **43**, 494–498 (2012)

[7] Lu, L., Chan, C.-Y., Wang, J., Wang, W.: A study of pedestrian group behaviors in crowd evacuation based on an extended floor field cellular automaton model. Transportation Research Part C: Emerging Technologies **81**, 317–329 (2017)

[8] Sohn, S.S., Zhou, H., Moon, S., Yoon, S., Pavlovic, V., Kapadia, M.: Laying the foundations of deep long-term crowd flow prediction. In: European Conference on Computer Vision, pp. 711–728 (2020)

[9] Xiong, F., Shi, X., Yeung, D.-Y.: Spatiotemporal modeling for crowd counting in videos. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5151–5159 (2017)

[10] Liu, Y., Wen, Q., Chen, H., Liu, W., Qin, J., Han, G., He, S.: Crowd counting via cross-stage refinement networks. IEEE Transactions on Image Processing **29**, 6800–6812 (2020)

[11] Wang, C., Zhang, H., Yang, L., Liu, S., Cao, X.: Deep people counting in extremely dense crowds. In: Proceedings of the 23rd ACM International Conference on Multimedia, pp. 1299–1302 (2015)

[12] Fu, M., Xu, P., Li, X., Liu, Q., Ye, M., Zhu, C.: Fast crowd density estimation with convolutional neural networks. Engineering Applications of Artificial Intelligence **43**, 81–88 (2015)

[13] Zhang, C., Li, H., Wang, X., Yang, X.: Cross-scene crowd counting via deep convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 833–841 (2015)

[14] Li, Y., Zhang, X., Chen, D.: Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1091–1100 (2018)

[15] Liu, X., Li, G., Han, Z., Zhang, W., Yang, Y., Huang, Q., Sebe, N.: Exploiting sample correlation for crowd counting with multi-expert network. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3215–3224 (2021)

[16] Abdelghany, A., Abdelghany, K., Mahmassani, H., Alhalabi, W.: Modeling framework for optimal evacuation of large-scale crowded pedestrian facilities. European Journal of Operational Research **237**(3), 1105–1118 (2014)

[17] Dong, H., Zhou, M., Wang, Q., Yang, X., Wang, F.-Y.: State-of-the-art pedestrian and evacuation dynamics. IEEE Transactions on Intelligent Transportation Systems **21**(5), 1849–1866 (2019)

[18] Niu, Y., Shi, W., Liu, W., He, S., Pan, J., Chan, A.B.: Over-crowdedness alert! forecasting the future crowd distribution. ArXiv preprint (2020)

[19] Minoura, H., Yonetani, R., Nishimura, M., Ushiku, Y.: Crowd density forecasting by modeling patch-based dynamics. IEEE Robotics and Automation Letters **6**(2), 287–294 (2020)

[20] Geng, Y., Du, J., Liang, M.: Abnormal event detection in tourism video based on salient spatio-temporal features and sparse combination learning. World Wide Web **22**(2), 689–715 (2019)

[21] Pawar, K., Attar, V.: Deep learning approaches for video-based anomalous activity detection. World Wide Web **22**(2), 571–601 (2019)

[22] Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S.: Social lstm: Human trajectory prediction in crowded spaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 961–971 (2016)

[23] Vemula, A., Muelling, K., Oh, J.: Social attention: Modeling attention in human crowds. In: IEEE International Conference on Robotics and Automation, pp. 4601–4607 (2018)

[24] Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social gan: Socially acceptable trajectories with generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2255–2264 (2018)

[25] Huang, Y., Bi, H., Li, Z., Mao, T., Wang, Z.: Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 6272–6281 (2019)

[26] Kothari, P., Kreiss, S., Alahi, A.: Human trajectory forecasting in crowds: A deep learning perspective. IEEE Transactions on Intelligent Transportation Systems **23**(7), 7386–7400 (2021)

[27] Helbing, D., Molnar, P.: Social force model for pedestrian dynamics. Physical review E **51**(5), 4282 (1995)

[28] Zhou, R., Cui, Y., Wang, Y., Jiang, J.: A modified social force model with different categories of pedestrians for subway station evacuation. Tunnelling and Underground Space Technology **110**, 103837 (2021)

[29] Jiang, X., Xiao, Z., Zhang, B., Zhen, X., Cao, X., Doermann, D., Shao, L.: Crowd counting and density estimation by trellis encoder-decoder networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6133–6142 (2019)

[30] Cao, X., Wang, Z., Zhao, Y., Su, F.: Scale aggregation network for accurate and efficient crowd counting. In: Proceedings of the European Conference on Computer Vision, pp. 734–750 (2018)

[31] Wan, J., Liu, Z., Chan, A.B.: A generalized loss function for crowd counting and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1974–1983 (2021)

[32] Gao, J., Lin, W., Zhao, B., Wang, D., Gao, C., Wen, J.: c^3 framework:

An open-source pytorch code for crowd counting. ArXiv preprint (2019)

[33] Lee, S., Kim, H.G., Choi, D.H., Kim, H.-I., Ro, Y.M.: Video prediction recalling long-term motion context via memory alignment learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3054–3063 (2021)

[34] Yi, S., Li, H., Wang, X.: Understanding pedestrian behaviors from stationary crowd groups. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3488–3496 (2015)

[35] Ranzato, M., Szlam, A., Bruna, J., Mathieu, M., Collobert, R., Chopra, S.: Video (language) modeling: a baseline for generative models of natural videos. arXiv preprint:1412.6604 (2014)

[36] Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using lstms. In: International Conference on Machine Learning, pp. 843–852 (2015)

[37] Wang, Y., Long, M., Wang, J., Gao, Z., Yu, P.S.: Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. Advances in Neural Information Processing Systems **30** (2017)

[38] Wang, Y., Gao, Z., Long, M., Wang, J., Philip, S.Y.: Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In: International Conference on Machine Learning, pp. 5123–5132 (2018)

[39] Wang, Y., Wu, H., Zhang, J., Gao, Z., Wang, J., Yu, P., Long, M.: Predrnn: A recurrent neural network for spatiotemporal predictive learning. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)

[40] Chang, Z., Zhang, X., Wang, S., Ma, S., Ye, Y., Xinguang, X., Gao, W.: Mau: A motion-aware unit for video prediction and beyond. Advances in Neural Information Processing Systems **34**, 26950–26962 (2021)

[41] Sadeghian, A., Kosaraju, V., Sadeghian, A., Hirose, N., Rezatofighi, H., Savarese, S.: Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1349–1358 (2019)

[42] Zhu, J.-Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. Advances in Neural Information Processing Systems **30** (2017)

[43] Kosaraju, V., Sadeghian, A., Martín-Martín, R., Reid, I., Rezatofighi, H.,

Savarese, S.: Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. Advances in Neural Information Processing Systems **32** (2019)

[44] Mangalam, K., An, Y., Girase, H., Malik, J.: From goals, waypoints & paths to long term human trajectory forecasting. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 15233–15242 (2021)

[45] Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: An evaluation of the state of the art. IEEE transactions on Pattern Analysis and Machine Intelligence **34**(4), 743–761 (2011)

[46] Wang, M., Wang, X.: Automatic adaptation of a generic pedestrian detector to a specific traffic scene. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3401–3408 (2011)

[47] Idrees, H., Saleemi, I., Seibert, C., Shah, M.: Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2547–2554 (2013)

[48] Kumagai, S., Hotta, K., Kurita, T.: Mixture of counting cnns: Adaptive integration of cnns specialized to specific appearance for crowd counting. arXiv preprint:1703.09393 (2017)

[49] Lucas, B.D., Kanade, T., *et al.*: An iterative image registration technique with an application to stereo vision. In: Proceedings of DARPA Image Understanding Workshop (1981)

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

[51] Zagoruyko, S., Komodakis, N.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. arXiv preprint:1612.03928 (2016)

[52] Denby, L., Martin, R.D.: Robust estimation of the first-order autoregressive parameter. Journal of the American Statistical Association **74**(365), 140–146 (1979)

[53] Zhou, B., Wang, X., Tang, X.: Random field topic model for semantic region analysis in crowded scenes from tracklets. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.

3441–3448 (2011)

[54] Ke, C., Chen, C.L., Gong, S., Tao, X.: Feature mining for localised crowd counting. In: British Machine Vision Conference (2012)

[55] Zhang, Y., Zhou, D., Chen, S., Gao, S., Yi, M.: Single-image crowd counting via multi-column convolutional neural network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2016)

[56] Plyer, A., Le Besnerais, G., Champagnat, F.: Massively parallel lucas kanade optical flow for real-time video processing applications. Journal of Real-Time Image Processing **11**(4), 713–730 (2016)

[57] Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional lstm network: A machine learning approach for precipitation nowcasting. Advances in Neural Information Processing Systems **28** (2015)

[58] Liu, B., Chen, Y., Liu, S., Kim, H.-S.: Deep learning in latent space for video prediction and compression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 701–710 (2021)

[59] Zheng, H., Lin, F., Feng, X., Chen, Y.: A hybrid deep learning model with attention-based conv-lstm networks for short-term traffic flow prediction. IEEE Transactions on Intelligent Transportation Systems **22**(11), 6910–6920 (2020)

[60] Lin, Z., Li, M., Zheng, Z., Cheng, Y., Yuan, C.: Self-attention convlstm for spatiotemporal prediction. Proceedings of the AAAI Conference on Artificial Intelligence **34**(7), 11531–11538 (2020)

[61] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)

[62] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 586–595 (2018)

[63] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. Computer Science (2014)

[64] Li, M., Soltanolkotabi, M., Oymak, S.: Gradient descent with early

stopping is provably robust to label noise for overparameterized neural networks. In: International Conference on Artificial Intelligence and Statistics, pp. 4313–4324 (2020)