# Spatiotemporal Modeling for Crowd Counting in Videos

Feng Xiong Xingjian Shi Dit-Yan Yeung
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
{fxiongab,xshiab,dyyeung}@cse.ust.hk

## Abstract

*Region of Interest (ROI) crowd counting can be formulated as a regression problem of learning a mapping from an image or a video frame to a crowd density map. Recently, convolutional neural network (CNN) models have achieved promising results for crowd counting. However, even when dealing with video data, CNN-based methods still consider each video frame independently, ignoring the strong temporal correlation between neighboring frames. To exploit the otherwise very useful temporal information in video sequences, we propose a variant of a recent deep learning model called convolutional LSTM (ConvLSTM) for crowd counting. Unlike the previous CNN-based methods, our method fully captures both spatial and temporal dependencies. Furthermore, we extend the ConvLSTM model to a bidirectional ConvLSTM model which can access long-range information in both directions. Extensive experiments using four publicly available datasets demonstrate the reliability of our approach and the effectiveness of incorporating temporal information to boost the accuracy of crowd counting. In addition, we also conduct some transfer learning experiments to show that once our model is trained on one dataset, its learning experience can be transferred easily to a new dataset which consists of only very few video frames for model adaptation.*

## 1. Introduction

Crowd counting is the problem of estimating the number of people in a still image or a video. It has drawn a lot of attention due to the need for solving this problem in many real-world applications such as video surveillance, traffic control, and emergency management. Proper use of crowd counting techniques can help to prevent some serious accidents such as the massive stampede that happened in Shanghai, China during the 2015 New Year's Eve festivities, killing 35 people. Moreover, some crowd counting methods can also be applied to other object counting applications such as cell counting in microscopic images [15, 29], vehicle counting in public areas [18, 34], and animal counting in the wild [3].

The methods for crowd counting in videos fall into two broad categories: (a) Region of Interest (ROI) counting, which estimates the total number of people in some region at a certain time; and (b) Line of Interest (LOI) counting, which counts the number of people crossing a detecting line in a video during a certain period of time. Since LOI counting is more restrictive in its applications and is much less studied than ROI counting, we focus on ROI counting in this paper.

Many methods have been proposed in the past for crowd counting. Some methods take the approach of tackling the crowd counting problem in an unsupervised manner through grouping based on self-similarities [1] or motion similarities [21]. However, the accuracy of such fully unsupervised counting methods is limited. Thus more attention has been paid to the supervised learning approach. Supervised crowd counting methods generally fall into two categories: detection-based methods and regression-based methods. In detection-based methods, some given object detectors [12, 37, 16, 9] are used to detect people individually. They usually operate in two stages by first producing a real-valued confidence map and then locating from the map those peaks that correspond to individual people. Once the locations of all individuals have been estimated, the counting problem becomes trivial. However, object detection could be a challenging problem especially under severe occlusion in highly crowded scenes.

In recent years, regression-based methods have achieved promising counting results in crowded scenes. Regression-based methods avoid solving the difficult detection problem. Instead, they regard crowd counting as a regression problem by learning a regression function or mapping from some holistic or local visual features to a crowd count or a crowd density map. Linear regression, Gaussian process regression, and neural networks are often used as the regression models. Currently, most methods which achieve state-of-the-art performance are regression-based methods [4, 7, 2, 20, 6, 32, 36, 29].

With the recent resurgence of interest in convolutional neural network (CNN) models which have reported promising results for many computer vision tasks [14], in the recent two years some CNN-based methods [32, 36, 29, 26] have also been proposed for crowd counting, giving state-of-the-art results on the existing crowd counting datasets such as UCSD [4] and UCF [11]. Unlike traditional regression-based methods [4, 7], CNN-based methods do not need handcrafted features but can learn powerful features in an end-to-end manner. However, even when dealing with video datasets, these CNN-based methods still regard the data as individual still images and ignore the strong temporal correlation between neighboring video frames.

In this paper, we propose a variant of a recent deep learning model called convolutional LSTM (ConvLSTM) [24] for crowd counting. While CNN-based methods exploit only spatial features by ignoring the otherwise very useful temporal information in video sequences, our method fully captures both spatial and temporal dependencies. Incorporating the temporal dimension is important as it is well known that motion information can help to boost the counting accuracy in complex scenes. Thorough experimental validation using four publicly available datasets will be reported later in this paper to demonstrate the effectiveness of incorporating temporal information to boost the accuracy of crowd counting.

## 2. Related Work

### 2.1. Deep learning methods for crowd counting

C. Zhang *et al.* [32] proposed the first CNN-based method for crowd counting. Following this work, Y. Zhang *et al.* [36] proposed a multi-column CNN architecture which allows the input image to be of arbitrary size or resolution. The multi-column CNN architecture also uses a different method for computing the crowd density. Walach and Wolf [29] proposed a stage-wise approach by carrying out model training in stages. In the spirit of the gradient boosting approach, CNNs are added one at a time such that every new CNN is trained to estimate the residual error of the earlier prediction. After the first CNN has been trained, the second CNN is trained on the difference between the current estimate and the learning target. The third CNN is then added and the process continues. Rubio *et al.* [19] proposed a framework called Hydra CNN which uses a pyramid of image patches extracted at multiple scales to perform the final density prediction. All these methods have reported good results for the UCSD dataset. However, to the best of our knowledge, temporal dependencies have not been explicitly exploited by deep learning models for crowd counting. These CNN-based methods simply treat the video sequences in the UCSD dataset as a set of still images without considering their temporal dependencies.

[1]

### 2.2. Density map regression for crowd counting

Lempitskey and Zisserman [15] proposed a method to change the target of regression from a single crowd count to a crowd density map. We note that crowd density is more informative than crowd count, since the former also includes location information of the crowd. With a crowd density map, the crowd count of any given region can be estimated easily. The crowd count of the whole image is simply the integral of the density function over the entire image. All CNN-based methods mentioned above have used the crowd density map as the regression target.

### 2.3. ConvLSTM for spatiotemporal modeling

Recurrent neural networks (RNNs) have been applied successfully to various sequence learning tasks [27]. The incorporation of long short-term memory (LSTM) cells enables RNNs to exploit longer-term temporal dependencies. By extending the fully connected LSTM (FC-LSTM) to have convolutional structures in both the input-to-state and state-to-state connections, Shi *et al.* [24] proposed the ConvLSTM model for precipitation nowcasting which is a spatiotemporal forecasting problem. The ConvLSTM layer not only preserves the advantages of FC-LSTM but is also suitable for spatiotemporal data due to its inherent convolutional structures.

ConvLSTM models have also proven effective for some other spatiotemporal tasks. Finn *et al.* [8] employed stacked ConvLSTMs to generate motion predictions. Villegas *et al.* [28] proposed a ConvLSTM-based method to model the spatiotemporal dynamics for pixel-level prediction in natural videos. Also, Y. Zhang *et al.* [35] applied network-in-network principles, batch normalization, residual connections, and ConvLSTMs to build very deep recurrent and convolutional structures for speech recognition.

## 3. Our Crowd Counting Method

### 3.1. Crowd density map

Following the previous work [15] as reviewed above, we also formulate crowd counting as a density map estimation problem. Compared to methods that give an estimated crowd count of the whole image as output, methods that give a crowd density map also provide location information about the crowd distribution which is useful for many applications.

We assume that each training image $I_i$ is annotated with a set of 2D points $\mathcal{P}_i = \{P_1, \ldots, P_{C(i)}\}$, where $C(i)$ is the

[1] While preparing the camera-ready paper, it was brought to our attention that some related papers had been accepted by the same conference, e.g.,[25, 33].

total number of people annotated. We define the ground-truth density map for supervised learning as a sum of Gaussian kernels each of which is centered at the location of one person. The ground-truth density map $F_i(p)$ for image $I_i$ can be defined as follows:

$$\forall p \in I_i,\ F_i(p) = \sum_{P \in \mathcal{P}_i} \mathcal{N}(p; P, \sigma^2 I_{2\times 2}), \qquad (1)$$

where $p$ denotes a pixel in image $I_i$, $\mathcal{P}_i$ is the set of annotated points (usually corresponding to the positions of the human heads), $\mathcal{N}(p; P, \sigma^2 I_{2\times 2})$ represents a normalized 2D Gaussian kernel evaluated at the pixel position $p$ with its mean at the head position $P$ and an isotropic $2 \times 2$ covariance matrix $I_{2\times 2}$ with variance $\sigma^2$.

For annotated points which are close to the image boundary, part of their probability mass will reside outside the image. Consequently, integrating the ground-truth density map over the entire image will not match the crowd count exactly. Fortunately, this effect can be neglected for most applications because the differences are generally small. Moreover, in many cases, a pedestrian who lies partially outside the image boundary should not be counted as a whole person.

Another subtlety that is worth noticing is that the images are often not captured with a bird's-eye view and hence leads to perspective distortion. As a result, the pixels associated with different annotated points correspond to regions of different scales in the actual 3D scene. To overcome the effects due to perspective distortion, we need to normalize the crowd density map with the perspective map $M(p)$. The pixel value in the perspective map represents the number of pixels in the image corresponding to one meter at that location in the real scene. In our experiments, we set $\sigma = 0.3M(p)$ and then normalize the whole distribution to ensure that the sum of ground-truth density is equal to the crowd count of the image.

### 3.2. ConvLSTM model

FC-LSTM has proven powerful for handling temporal correlations, but it fails to maintain structural locality. To exploit temporal correlations for video crowd counting, we propose a model based on ConvLSTM [24] to learn a density map. As an extension of FC-LSTM, ConvLSTM has convolutional structures in both the input-to-state and state-to-state connections. We can regard all the inputs, cell outputs, hidden states $\mathcal{H}_1, ..., \mathcal{H}_t$, and gates $i_t, f_t, o_t$ of the ConvLSTM as 3D tensors whose last two dimensions are spatial dimensions. The outputs of ConvLSTM cells depend on the inputs and past states of the local neighbors. The key equations of ConvLSTM are shown in (2) below, where '*' denotes the convolution operator, '$\circ$' denotes the Hadamard product, and $\sigma(\cdot)$ denotes the logistic sigmoid function:

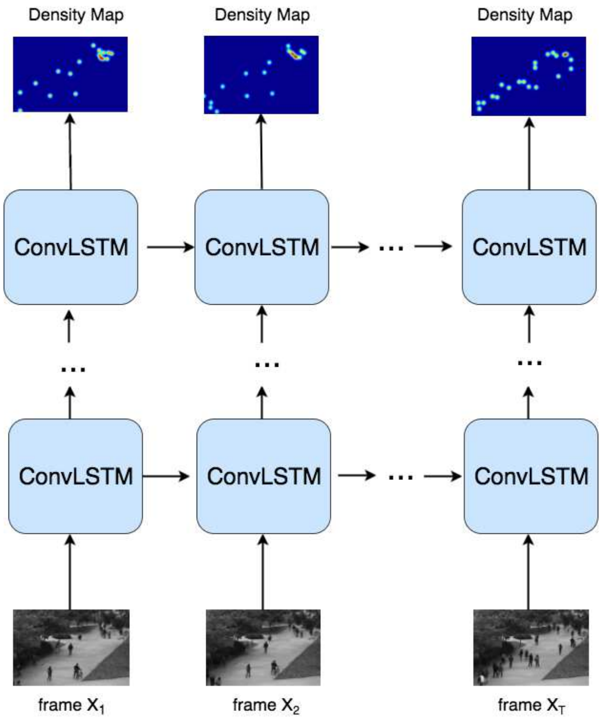

Figure 1. ConvLSTM model for crowd counting

$$\begin{aligned}
i_t &= \sigma(W_{xi} * \mathcal{X}_t + W_{hi} * \mathcal{H}_{t-1} + W_{ci} \circ \mathcal{C}_{t-1} + b_i),\\
f_t &= \sigma(W_{xf} * \mathcal{X}_t + W_{hf} * \mathcal{H}_{t-1} + W_{cf} \circ \mathcal{C}_{t-1} + b_f),\\
\mathcal{C}_t &= f_t \circ \mathcal{C}_{t-1} + i_t \circ \tanh(W_{xc} * \mathcal{X}_t + W_{hc} * \mathcal{H}_{t-1} + b_c),\\
o_t &= \sigma((W_{xo} * \mathcal{X}_t + W_{ho} * \mathcal{H}_{t-1} + W_{co} \circ \mathcal{C}_t + b_o),\\
\mathcal{H}_t &= o_t \circ \tanh(\mathcal{C}_t).
\end{aligned} \qquad (2)$$

Figure 1 shows our ConvLSTM model for crowd counting where each building block involves a ConvLSTM.

The inputs $\boldsymbol{\mathcal{X}}_{1:t} = \mathcal{X}_1, \ldots, \mathcal{X}_t$ are consecutive frames of a video and the cell outputs $\mathcal{C}_1, \ldots, \mathcal{C}_t$ are the estimated density maps of the corresponding frames. If we remove the connections between ConvLSTM cells, we can regard each ConvLSTM cell as a CNN model with gates. We set all the input-to-state and state-to-state kernels to size $5 \times 5$ and the number of layers to 4. To relate the feature maps to the density map, we adopt filters all of size $1 \times 1$. We use the Euclidean distance to measure the difference between the estimated and ground-truth density maps. So we define the loss function $L(\theta)$ between the estimated density map $F(\boldsymbol{\mathcal{X}}_{1:t}; \theta)$ and the ground-truth density map $D_t$ as follows:

$$L(\theta) = \frac{1}{2T} \sum_{t=1}^{T} \left\| F(\boldsymbol{\mathcal{X}}_{1:t}; \theta) - D_t \right\|_2^2, \qquad (3)$$

where $T$ is the length of the video clip and $\theta$ denotes the

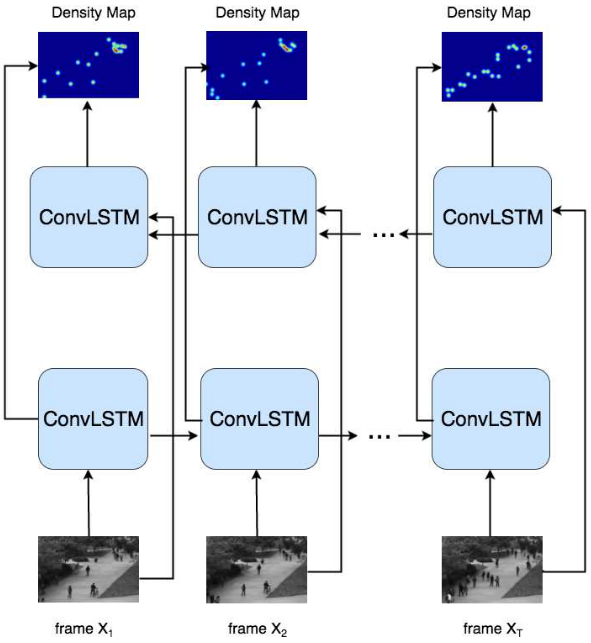

Figure 2. Bidirectional ConvLSTM model for crowd counting

parameter vector.

### 3.3. From ConvLSTM to bidirectional ConvLSTM

Inspired by [10, 35], we further extend the ConvLSTM model to a bidirectional ConvLSTM model which can access long-range information in both directions.

Figure 2 depicts the bidirectional ConvLSTM model for crowd counting. Its inputs and outputs are the same as those in the ConvLSTM model. It works by computing the forward hidden sequence $\vec{h}$, backward hidden sequence $\overleftarrow{h}$, and output sequence by iterating backward from $t = T$ to $t = 1$, iterating forward from $t = 1$ to $t = T$, and then updating the output layer. If we denote the state updating function in (2) as $\mathcal{H}_t, \mathcal{C}_t = \text{ConvLSTM}(\mathcal{X}_t, \mathcal{H}_{t-1}, \mathcal{C}_{t-1})$, the equation of bidirectional ConvLSTM can be written as follows:

$$\begin{aligned}
\vec{\mathcal{H}}_t, \vec{\mathcal{C}_t} &= \text{ConvLSTM}(\mathcal{X}_t, \vec{\mathcal{H}}_{t-1}, \vec{\mathcal{C}}_{t-1}), \\
\overleftarrow{\mathcal{H}}_t, \overleftarrow{\mathcal{C}_t} &= \text{ConvLSTM}(\mathcal{X}_t, \overleftarrow{\mathcal{H}}_{t+1}, \overleftarrow{\mathcal{C}}_{t+1}), \\
\mathcal{Y}_t &= \text{concat}(\vec{\mathcal{H}}_t, \overleftarrow{\mathcal{H}}_t),
\end{aligned} \tag{4}$$

where $\mathcal{Y}_t$ is the output at timestamp $t$.

Y. Zhang *et al.* [35] found that bidirectional ConvLSTM consistently outperforms its unidirectional counterpart in speech recognition. In the next section, we also compare bidirectional ConvLSTM with the original ConvLSTM for crowd counting using different datasets.

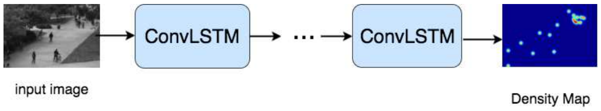

Figure 3. ConvLSTM-nt model for crowd counting

### 3.4. ConvLSTM-nt: a degenerate variant of ConvLSTM for comparison

To better understand the effectiveness of exploiting temporal information, we propose a degenerate variant of ConvLSTM, called ConvLSTM with no temporal information (ConvLSTM-nt), by removing all connections between the ConvLSTM cells. ConvLSTM-nt can be seen as a CNN model with gates. The parameters of ConvLSTM-nt are the same as those of ConvLSTM introduced above. The structure of ConvLSTM-nt is shown in Figure 3.

All our three models have 4 layers, with $128, 64, 64$ and $64$ hidden states respectively in the four ConvLSTM layers. For the training scheme, we train all models using the TensorFlow library, optimizing to convergence using ADAM [13] with the suggested hyperparameters in TensorFlow.

In the experiments to be reported in the next section, whenever the dataset consists of still images not forming video sequences, both the original ConvLSTM and our bidirectional extension cannot be used but only ConvLSTM-nt will be used.

## 4. Experiments

We conduct comparative study using four annotated datasets which include the UCF_CC_50 dataset [11], UCSD dataset [4], Mall dataset [7], and WorldExpo'10 dataset [32, 31]. Some statistics of these datasets are summarized in Table 1. We also conduct experiments in the transfer learning setting by using one of the UCSD and Mall datasets as the source domain and the other one as the target domain.

### 4.1. Evaluation metric

For crowd counting, the mean absolute error (MAE) and mean squared error (MSE) are the two most commonly used evaluation metrics. They are defined as follows:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|p_i - \hat{p}_i|\,, \ \text{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(p_i - \hat{p}_i)^2}, \tag{5}$$

where $N$ is the total number of frames used for testing, $p_i$ and $\hat{p}_i$ are the true number and estimated number of people in frame $i$ respectively. As discussed above, $\hat{p}_i$ is calculated by summing over the estimated density map over the entire image.

Table 1. Statistics of the four datasets

| Dataset | Resolution | Color | Num | FPS | Max | Min | Average | Total |
|---|---|---|---|---|---|---|---|---|
| UCF_CC_50 | different | Grey | 50 | Images | 4543 | 94 | 1279.5 | 63974 |
| UCSD | 158 × 238 | Grey | 2000 | 10 | 46 | 11 | 24.9 | 49885 |
| Mall | 640 × 480 | RGB | 2000 | - | 53 | 11 | 31.2 | 62315 |
| WorldExpo | 576 × 720 | RGB | 3980 | 50 | 253 | 1 | 50.2 | 199923 |

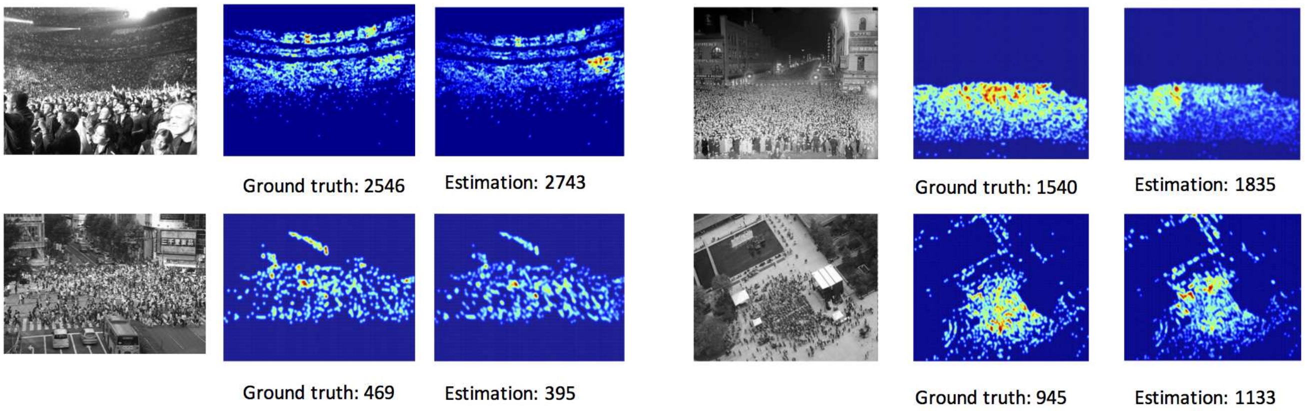

Figure 4. Results for four test images from the UCF_CC_50 dataset. For each example, we show the input image (left), ground-truth density map (middle), and density map obtained by ConvLSTM-nt (right).

### 4.2. UCF_CC_50 dataset

The UCF_CC_50 dataset was first introduced by Idress *et al.* [11]. It is a very challenging dataset because it contains only 50 images of different resolutions, different scenes, and extremely high crowd density. In particular, the number of pedestrians ranges between 94 and 4,543 with an average of 1,280. Annotations of all the 63,794 people in all 50 images are available in the dataset. Since the 50 images have no temporal correlation between them, we cannot demonstrate the advantage of exploiting temporal information. So only the ConvLSTM-nt variant is applied on this dataset. The goal here is to show that our model can still give very good results for such extremely dense crowd images even though temporal information is not available.

Following the setting in [11], we split the dataset randomly and perform 5-fold cross validation. To handle different resolutions, we randomly crop patches of size 72 × 72 from each image for training and testing. As for the overlapping patches in the test set, we calculate the density at each pixel by averaging the overlapping patches.

We compare our method with six existing methods on the UCF_CC_50 dataset. The results are shown in Table 2. Rodriguez *et al.* [22] adopted the density map estimation in detection-based methods. Lempitsky *et al.* [15] extracted 800 dense SIFT features from the input image and learned a density map with the proposed MESA distance (where MESA stands for Maximum Excess over SubArrays). Idress *et al.* [11] estimated the crowd count by multi-source features which include SIFT and head detection. The methods proposed by C. Zhang *et al.* [32], Y. Zhang *et al.* [36], and Shang *et al.* [23] are all CNN-based methods. Shang *et al.* [23] used a model pre-trained on the WorldExpo dataset as initial weights and yielded the best MAE. However, when considering only methods that do not use additional data for training, our ConvLSTM-nt model achieves the lowest MAE and MSE.

Table 2. Results of different methods on the UCF_CC_50 dataset. It should be noticed that Shang *et al.* [23] used additional data for training, so it is not fair to compare its result with the others directly.

| Method | MAE | MSE |
|---|---|---|
| Head detection [22] | 655.7 | 697.8 |
| Density map + MESA [15] | 493.4 | 487.1 |
| Multi-source features [11] | 419.5 | 541.6 |
| Crowd CNN [32] | 467.0 | 498.5 |
| Multi-column CNN [36] | 377.6 | 509.1 |
| ConvLSTM-nt | **284.5** | **297.1** |
| Shang *et al.* [23] | **270.3** | - |

Some results obtained by ConvLSTM-nt are shown in Figure 4. Although the images have wide variations in the background and crowd density, ConvLSTM-nt is quite robust in producing reasonable density maps and hence the overall crowd counts.

### 4.3. UCSD dataset

The UCSD dataset [4] contains a 2,000-frame video of pedestrians on a walkway of the UCSD campus captured by a stationary camera. The video was recorded at 10 fps with

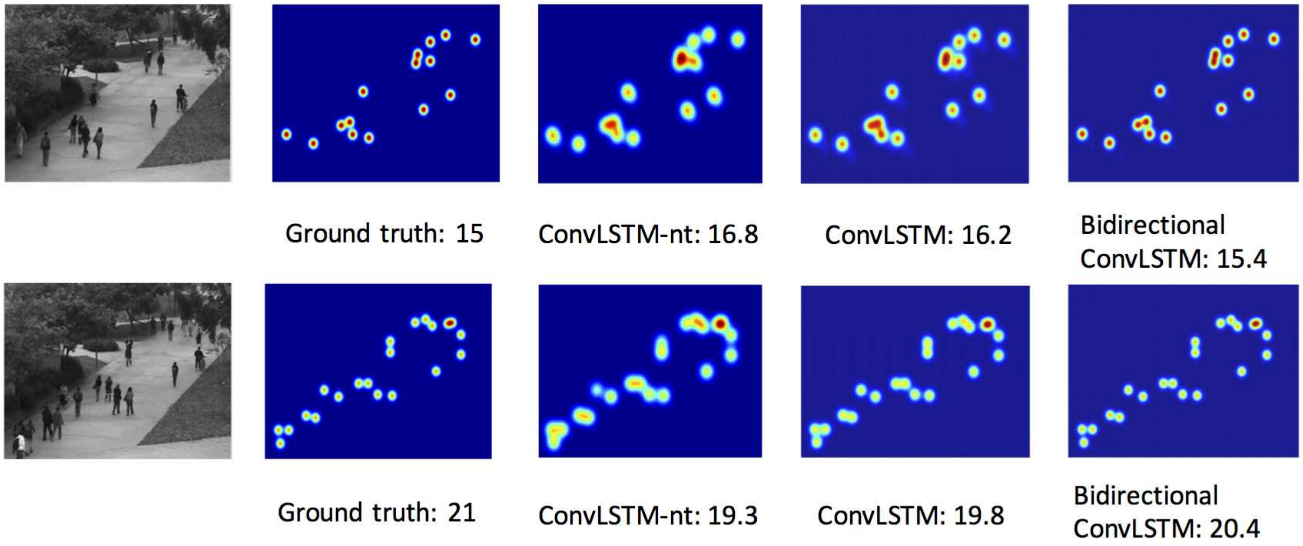

Figure 5. Results for two test video frames from the UCSD dataset. For each example, we show the input video frame, ground-truth density map, and density maps obtained by the three variants of our method.

dimension $238 \times 158$. The labeled ground truth marks the center of each pedestrian. The ROI and the perspective map are provided in the dataset.

Using the same setting as in [4], we use frames 601–1,400 as the training data and the remaining 1,200 frames as test data. The provided perspective map is used to adjust the ground-truth density map by setting $\sigma = 0.3M(p)$. The values of the pixels outside the ROI are set to zero.

The results of different methods are shown in Table 3. [4, 7, 6] are traditional methods which give the crowd count for the whole image. [15, 20] are density map regression methods using handcrafted features and regression algorithms such as linear regression and random forest regression. Most state-of-the-art methods are based on CNNs [29, 32, 19, 36]. Bidirectional ConvLSTM achieves comparable MAE and MSE with these methods. From the results of ConvLSTM-nt, unidirectional ConvLSTM, and bidirectional ConvLSTM , we can draw the conclusion that temporal information can boost the performance for this dataset.

Figure 5 shows two illustrative examples. We can see that bidirectional ConvLSTM produces density maps that are closest to the ground truth. While ConvLSTM-nt can give a rough estimation, ConvLSTM and bidirectional ConvLSTM are more accurate in the fine details.

## 4.4. Mall dataset

The Mall dataset was provided by Chen *et al.* [7] for crowd counting. It was captured in a shopping mall using a publicly accessible surveillance camera. This video contains 2,000 annotated frames of moving and stationary pedestrians with more challenging lighting conditions and glass surface reflections. The ROI and the perspective map are also provided in the dataset.

Table 3. Results of different methods on the UCSD dataset

| Method | MAE | MSE |
|---|---|---|
| Gaussian process regression [4] | 2.24 | 7.97 |
| Ridge regression [7] | 2.25 | 7.82 |
| Cumulative attribute regression [6] | 2.07 | 6.90 |
| Density map + MESA [15] | 1.70 | - |
| Count forest [20] | 1.60 | 4.40 |
| Crowd CNN [32] | 1.60 | 3.31 |
| Multi-column CNN [36] | **1.07** | **1.35** |
| Hydra CNN [19] | 1.65 | - |
| CNN boosting [29] | 1.10 | - |
| ConvLSTM-nt | 1.73 | 3.52 |
| ConvLSTM | 1.30 | 1.79 |
| Bidirectional ConvLSTM | 1.13 | 1.43 |

Following the same setting as [7], we use the first 800 frames for training and the remaining 1,200 frames for testing. We perform comparison against Gaussian process regression [4], ridge regression [7], cumulative attribute ridge regression [6], and random forest regression [20]. Bidirectional ConvLSTM achieves state-of-the-art performance with respect to both MAE and MSE. The results are shown in Table 4, which also demonstrates the effectiveness of exploiting temporal information.

## 4.5. WorldExpo dataset

The WorldExpo dataset was introduced by C. Zhang *et al.* [32, 31]. This dataset contains 1,132 annotated video sequences captured by 108 surveillance cameras, all from the 2010 Shanghai World Expo. The annotations of 199,923

Table 4. Results of different methods on the Mall dataset

| Method | MAE | MSE |
|---|---|---|
| Gaussian process regression [4] | 3.72 | 20.1 |
| Ridge regression [7] | 3.59 | 19.0 |
| Cumulative attribute regression [6] | 3.43 | 17.7 |
| Count forest [20] | 2.50 | 10.0 |
| ConvLSTM-nt | 2.53 | 11.2 |
| ConvLSTM | 2.24 | 8.5 |
| Bidirectional ConvLSTM | **2.10** | **7.6** |

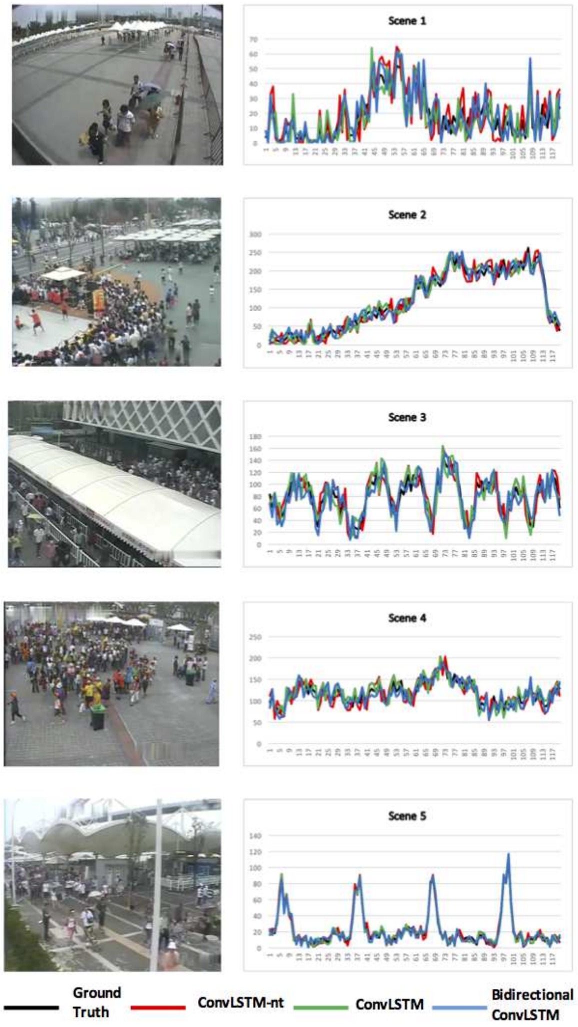

Figure 6. Density map estimation examples from the WorldExpo dataset (best viewed in color). In each row, the left one shows one video frame from the test scene and the right one shows the estimation results of that scene, where the $x$-axis represents the frame index and the $y$-axis represents the crowd count.

pedestrians in 3,980 frames include the location of the center of each human head. The test set contains five separate video sequences each of which has 120 annotated frames. The regions of interest (ROIs) are provided for these five test scenes. The perspective maps are also provided.

For fair comparison, we follow the work of the multi-column CNN to generate the density map according to the perspective map with the relation $\delta = 0.2M(x)$, where $M(x)$ denotes the number of pixels in the image representing one square meter at location $x$. Table 5 compares our model and its variants with the state-of-the-art methods. We use MAE as the evaluation metric, as suggested by the author of [32]. On average, bidirectional ConvLSTM achieves the lowest MAE. It also gives the best result for scene 5.

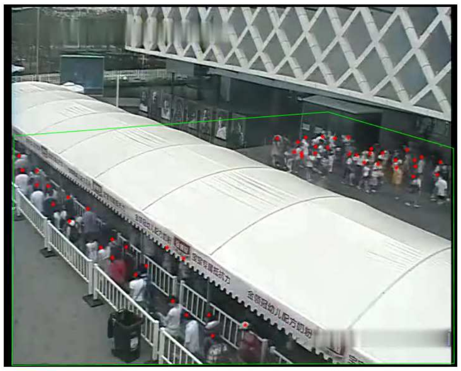

Figure 7. A video frame from test scene 3 of the WorldExpo dataset. The region outlined in green indicates the ROI and the red dots mark the positions of the heads.

We show the estimation results for the five test scenes obtained by our models in Figure 6. The crowd count curves are shown in different colors for the ground truth (black) and the estimation results of ConvLSTM-nt (red), ConvLSTM (green), and bidirectional ConvLSTM (blue). We note that the five scenes differ significantly in the scene type, crowd density, and change in crowd count over time.

From Table 5 and Figure 6, we can see that bidirectional ConvLSTM outperforms ConvLSTM and ConvLSTM outperforms ConvLSTM-nt in most cases (scene 1,2,4,5), which gives evidence to the effectiveness of incorporating temporal information for crowd counting. As for scene 3, a closer investigation reveals a potential problem with the labels provided in this test scene. Figure 7 illustrates the problem. There are in fact many people walking under the white ceiling of the covered walkway as we can see their moving legs clearly when playing the video, but only two red dots are provided in the frame because the heads of most of the people there are hidden. Spatiotemporal models tend to count them since motion is detected when exploiting the temporal information, but unfortunately they are not annotated in the provided labels.

## 4.6. Transfer learning experiments

To demonstrate the generalization capability of our model, we conduct some experiments in the transfer learning setting. Specifically, we compare with some previous methods that have also been evaluated in the transfer learn-

Table 5. Results of different methods on the WorldExpo dataset

| Method | Scene 1 | Scene 2 | Scene 3 | Scene 4 | Scene 5 | Average |
|---|---|---|---|---|---|---|
| LBP features + ridge regression | 13.6 | 59.8 | 37.1 | 21.8 | 23.4 | 31.0 |
| Deep CNN [32] | 9.8 | **14.1** | 14.3 | 22.2 | 3.7 | 12.9 |
| Multi-column CNN [36] | **3.4** | 20.6 | **12.9** | **13.0** | 8.1 | 11.6 |
| ConvLSTM-nt | 8.6 | 16.9 | 14.6 | 15.4 | 4.0 | 11.9 |
| ConvLSTM | 7.1 | 15.2 | 15.2 | 13.9 | 3.5 | 10.9 |
| Bidirectional ConvLSTM | 6.8 | 14.5 | 14.9 | 13.5 | **3.1** | **10.6** |

ing setting using the UCSD and Mall datasets, which were both captured using stationary cameras. As shown in Figure 8, the two datasets are quite different in terms of the scene type (outdoor for UCSD but indoor for Mall), crowd density, frame rate, and camera angle, among others.

We consider two transfer learning tasks by using one dataset as the source domain and the other one as the target domain. For each task, 800 frames are used for training the model and 50 frames of the other dataset are used as the adaptation set. Following the same setting as [5, 30, 17], we use MAE as the evaluation metric. Table 6 presents the results for different methods on the two transfer learning tasks. Bidirectional ConvLSTM achieves state-of-the-art performance in both transfer learning tasks. We note that the performance of our method in the transfer learning setting is even better than many approaches tested on the standard, non-transfer-learning setting. For instance, with 800 frames of the UCSD dataset for training and 50 frames of the Mall dataset for adaptation, bidirectional ConvLSTM can achieve an MAE of 2.63, which outperforms many algorithms using 800 frames of the Mall dataset for training, according to Table 4. We can draw the conclusion that bidirectional ConvLSTM has good generalization capability. Once trained on one dataset, the learning experience can be transferred easily to a new dataset which consists of only very few video frames for adaptation.

Table 6. Results of transfer learning across datasets with MAE as evaluation metric. FA: feature alignment; HGP: hierarchical Gaussian process; GPA: Gaussian process adaptation; GPTL: Gaussian process with transfer learning.

| Method | UCSD to Mall | Mall to UCSD |
|---|---|---|
| FA [5] | 7.47 | 4.44 |
| HGP [30] | 4.36 | 3.32 |
| GPA [17] | 4.18 | 2.79 |
| GPTL [17] | 3.55 | 2.91 |
| Bidirectional ConvLSTM | **2.63** | **1.82** |

## 5. Conclusion

In this paper, we have pursued the direction of spatiotemporal modeling for improving crowd counting in videos. By jointly capturing both spatial and temporal dependencies, we overcome a major limitation of the recent CNN-based crowd counting methods and advance the state of the art. Specifically, our models outperform existing crowd counting methods on the UCF_CC_50 dataset, Mall dataset, and WorldExpo dataset, and achieve comparable results on the UCSD dataset. The superior result on the UCF_CC_50 dataset shows that our model can still perform well on extremely dense crowd images even when temporal information is not available. As for the other three datasets, the results show that explicitly exploiting temporal information has a clear advantage. Finally, the last set of experiments shows that our model is robust under the transfer learning setting to generalize from previous learning experience.

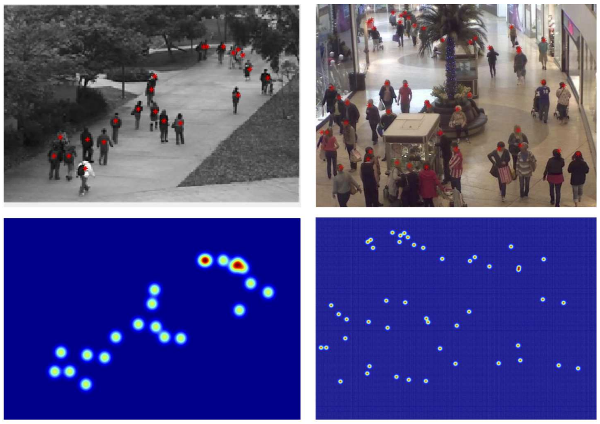

Figure 8. The UCSD and Mall datasets used for transfer learning experiments. Left column: UCSD dataset; right column: Mall dataset. Upper row: input images with annotations; lower row: density maps.

In the future, we are going to extend our model to deal with the active learning setting for crowd counting. We will output an additional confidence map and actively query the labeler to label only the less confident regions, which would greatly alleviate the expensive labeling effort for crowd counting in videos.

## 6. Acknowledgement

This research has been supported by General Research Fund 16207316 from the Research Grants Council of Hong Kong and Innovation and Technology Fund ITS/170/15FP from the Innovation and Technology Commission of Hong Kong.

## References

[1] N. Ahuja and S. Todorovic. Extracting texels in 2.1 D natural textures. In *ICCV*, pages 1–8, 2007.

[2] C. Arteta, V. Lempitsky, J. A. Noble, and A. Zisserman. Interactive object counting. In *ECCV*, pages 504–518, 2014.

[3] C. Arteta, V. Lempitsky, and A. Zisserman. Counting in the wild. In *ECCV*, pages 483–498, 2016.

[4] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *CVPR*, pages 1–7, 2008.

[5] C. L. Chen, S. Gong, and T. Xiang. From semi-supervised to transfer counting of crowds. In *ICCV*, pages 2256–2263, 2013.

[6] K. Chen, S. Gong, T. Xiang, and C. L. Chen. Cumulative attribute space for age and crowd density estimation. In *CVPR*, pages 2467–2474, 2013.

[7] K. Chen, C. C. Loy, S. Gong, and T. Xiang. Feature mining for localised crowd counting. In *BMVC*, page 3, 2012.

[8] C. Finn, I. Goodfellow, and S. Levine. Unsupervised learning for physical interaction through video prediction. In *NIPS*, pages 64–72, 2016.

[9] W. Ge and R. T. Collins. Marked point processes for crowd counting. In *CVPR*, pages 2913–2920, 2009.

[10] Y. Huang, W. Wang, and L. Wang. Bidirectional recurrent convolutional networks for multi-frame super-resolution. In *NIPS*, pages 235–243, 2015.

[11] H. Idrees, I. Saleemi, C. Seibert, and M. Shah. Multi-source multi-scale counting in extremely dense crowd images. In *CVPR*, pages 2547–2554, 2013.

[12] H. Idrees, K. Soomro, and M. Shah. Detecting humans in dense crowds using locally-consistent scale prior and global occlusion reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(10):1986–1998, 2015.

[13] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[15] V. Lempitsky and A. Zisserman. Learning to count objects in images. In *NIPS*, pages 1324–1332, 2010.

[16] M. Li, Z. Zhang, K. Huang, and T. Tan. Estimating the number of people in crowded scenes by MID based foreground segmentation and head-shoulder detection. In *ICPR*, pages 1–4, 2008.

[17] B. Liu and N. Vasconcelos. Bayesian model adaptation for crowd counts. In *ICCV*, pages 4175–4183, 2015.

[18] T. N. Mundhenk, G. Konjevod, W. A. Sakla, and K. Boakye. A large contextual dataset for classification, detection and counting of cars with deep learning. In *ECCV*, pages 785–800, 2016.

[19] D. Oñoro Rubio and R. J. López-Sastre. Towards perspective-free object counting with deep learning. In *ECCV*, pages 615–629, 2016.

[20] V.-Q. Pham, T. Kozakaya, O. Yamaguchi, and R. Okada. Count forest: Co-voting uncertain number of targets using random forest for crowd density estimation. In *ICCV*, pages 3253–3261, 2015.

[21] V. Rabaud and S. Belongie. Counting crowded moving objects. In *CVPR*, pages 705–711, 2006.

[22] M. Rodriguez, I. Laptev, J. Sivic, and J.-Y. Audibert. Density-aware person detection and tracking in crowds. In *ICCV*, pages 2423–2430, 2011.

[23] C. Shang, H. Ai, and B. Bai. End-to-end crowd counting via joint learning local and global count. In *ICIP*, pages 1215–1219, 2016.

[24] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *NIPS*, pages 802–810, 2015.

[25] V. A. Sindagi and V. M. Patel. Generating high-quality crowd density maps using contextual pyramid cnns. In *ICCV*, 2017.

[26] P. Sourtzinos, S. A. Velastin, M. Jara, P. Zegers, and D. Makris. People counting in videos by fusing temporal cues from spatial context-aware convolutional neural networks. In *ECCV Workshop*, pages 655–667, 2016.

[27] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[28] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee. Decomposing motion and content for natural video sequence prediction. In *ICLR*, 2017.

[29] E. Walach and L. Wolf. Learning to count with CNN boosting. In *ECCV*, pages 660–676, 2016.

[30] K. Yu, V. Tresp, and A. Schwaighofer. Learning Gaussian processes from multiple tasks. In *ICML*, pages 1012–1019, 2005.

[31] C. Zhang, K. Kang, H. Li, X. Wang, R. Xie, and X. Yang. Data-driven crowd understanding: A baseline for a large-scale crowd dataset. *IEEE Transactions on Multimedia*, 18(6):1048–1061, 2016.

[32] C. Zhang, H. Li, X. Wang, and X. Yang. Cross-scene crowd counting via deep convolutional neural networks. In *CVPR*, pages 833–841, 2015.

[33] S. Zhang, G. Wu, J. P. Costeira, and J. M. Moura. Fcn-rlstm: Deep spatio-temporal neural networks for vehicle counting in city cameras. In *ICCV*, 2017.

[34] S. Zhang, G. Wu, J. P. Costeira, and J. M. Moura. Understanding traffic density from large-scale web camera data. In *CVPR*, 2017.

[35] Y. Zhang, W. Chan, and N. Jaitly. Very deep convolutional networks for end-to-end speech recognition. *arXiv preprint arXiv:1610.03022*, 2016.

[36] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma. Single-image crowd counting via multi-column convolutional neural network. In *CVPR*, pages 589–597, 2016.

[37] T. Zhao, R. Nevatia, and B. Wu. Segmentation and tracking of multiple humans in crowded environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1198–1211, 2008.