Multimedia Tools and Applications (2021) 80:6121–6141
https://doi.org/10.1007/s11042-020-09888-1

Crowd aware summarization of surveillance videos by deep reinforcement learning

Junfeng Xu[1] · Zhengxing Sun[1] · Chen Ma[1]

Received: 23 December 2019 / Revised: 17 August 2020 / Accepted: 16 September 2020 /
Published online: 12 October 2020

Abstract
Surveillance videos which record crowd behaviors have dramatically increased due to the wide applications. A quick view of such crowd surveillance video in a constrained time is an increasing demand because it always contain a huge number of redundancy frames. In this paper, we focus on summarization of crowd surveillance videos. But it is not easy due to two reasons. First, how to make the decision to keep or discard a subshot from the input surveillance video stream so that the summary can outline the main behaviors of the crowd over a limited frames sequence. Second, how to maintain performance of summarization model for long surveillance videos. To tackle these challenges, we formulate surveillance video summarization as a sequential decision-making process and train the summarization network with reinforcement learning-based framework. A novel crowd location-density reward is proposed to teach summarization network to produce high-quality summaries. In addition, a summarization network with three layers LSTM is designed to maintain performance across longer time spans. Extensive experiments on three public crowd surveillance videos datasets show that the proposed method achieves state-of-the-art performance.

Keywords Surveillance video summarization · Crowd behaviors · Deep reinforcement learning · Unsupervised video summarization

Junfeng Xu and Chen Ma are Co-First Authors

✉ Zhengxing Sun
szx@nju.edu.cn

Junfeng Xu
njumagic@nju.edu.cn

Chen Ma
njumagic@nju.edu.cn

[1] State Key Lab for Novel Software Technology, Nanjing University, Nanjing 210023, China

1 Introduction

In recent days, surveillance videos, especially the ones that record crowds have dramatically increased due to the wide applications, such as crowd surveillance in the square, railway station, shopping malls, schools etc. These surveillance videos contain a huge number of frames (about 3000 frames a minute) that is a barrier to many practical usages. Video summarization is used to shorten an input video in the form of key shots or frames while still preserving the important information it contains. The shortened video provides an efficient way to browse large amounts of video data. In previous works of surveillance videos summarization, [9, 29] selected frames with moving targets as summarization according to frame-level dissimilarity measure. But it is sensitive to the minor changes in video stream, so they are not suitable for crowd surveillance which contain a large number of moving targets. [26, 43] proposed event-based surveillance video summarization, they selected key frames highly dependent on complicated abnormal event detection results. Obviously, as discussed in [25], the performance may decline significantly when there are no predefined abnormal event in the video stream.

In this paper, a novel unsupervised learning-based video summarization approach is proposed. Our goal is to select key shots to summarize crowd surveillance videos. Our approach is motivated by the following two facts. First, crowd location and density are two main contents in surveillance video, which are widely concerned in the field of video analysis [15, 30, 31, 39]. Second, high-quality video summary should keep the main contents of input video [6, 23, 50]. Inspired by the two reasons, we try to learn a crowd surveillance video summarization model that selects shots according to crowd location and density while meet high-quality video summary requirements.

Recently, sequence-to-sequence learning techniques which can be categorized as supervised [11, 12, 40, 46] and unsupervised [14, 18, 19, 21, 32, 33, 36, 38] have introduced several promising models. Supervised approaches learn from human-created summary ground truths. But there are few public crowd surveillance video data sets with labels. The demand of time-consuming and labor-intensive annotation procedures, which has been a limiting factor of existing datasets [48]. Thus unsupervised techniques are more applicable to our tasks where the annotated data is scarce.

More specifically, we develop a long short-term memory (LSTM) cell [13] based network that has been exploited to model the sequential patterns in video shots to summarize crowd surveillance videos. To train our model, reinforcement learning (RL) is used due to the following two reasons. First, unsupervised setting is focused on in our work. As mentioned in [53], RL can provide supervision from a reward as input signals to LSTM. Second, crowd location and density rewards are computed over the whole video sequence, which can only be made at the end of video streams. RL teaches the model to select better shots by the rewards iteratively. The reward function that consists of crowd location and density measures how well the generated summary can represent the main contents (can be taken as a set of different crowd behaviors) in original video according to the count of people and where they are. It is designed in terms of high-quality video summary requirement [6, 23, 50] that summary should be key shots whose contents was similar to contents of original videos, while different from shots already selected. Therefore, the novel reward function for calculating similarity and difference in crowd location and density is designed to encourage summarization network to produce high-quality summaries.

Although LSTM-based summarization network trained by RL has obtained significant results in different video summarization tasks [52, 53], the ideal length of video for LSTM

modeling is less than 100 frames. Unfortunately, most of surveillance videos contain thousands of frames. Apply LSTM to surveillance videos summarization directly may restrict the quality of summary results. Due to this reason, a hierarchical LSTM instead of single lay bidirectional LSTM/GRU [52, 53] is part of our video summarization model to capture dependencies across longer time spans. As shown in Fig. 1, three layers of LSTM units are used for modeling frames and shots. The first layer LSTM is used to obtain a representation for shots which generated by cutting original video evenly, and the final hidden state of each shot is input to the next layer. The output of last layer is treated as the embedding for the all shots and determine whether shots is key shots. Experiments show that two layers LSTM can get high performance summary videos than those from single layer LSTM, but surveillance videos are longer than ordinary videos (such as videos in standard datasets SumMe [11] and TVSum [42]). To this end, the network with three layers LSTM is designed to maintain the performance of our surveillance video summarization model.

To conclude the introduction, we summarize the main contributions of this paper as follows: (1) a RL-based unsupervised framework for crowd surveillance videos summarization is proposed. A novel crowd location and density reward function is designed to encourage summarization network to produce high-quality summaries. (2) A hierarchical LSTM is introduced as the summarization network in our framework to maintain the model performance for long crowd surveillance videos. (3) To show the effectiveness of the proposed approach, an extensive study on three crowd surveillance video public datasets has demonstrated that our method outperforms the state-of-the-art methods.

2 Related work

2.1 Supervised video summarization

Although there may be overfitting, learning from manually labeled video summary ground truth can achieve remarkable results and has been widely concerned. Gong et al. [3] proposed a two-pronged approach for learning a determinantal point process (DPP) from labeled data for modeling diversity. Intuitively, a DPP defines a probability distribution which makes subsets of higher diversity more likely to be selected. Inspired by this, Zhang et al. [48] first proposed a LSTM-based model for video summarization. Bidirectional LSTM layers were used for modeling better long-rang dependency in both the past and the future directions. Then, it

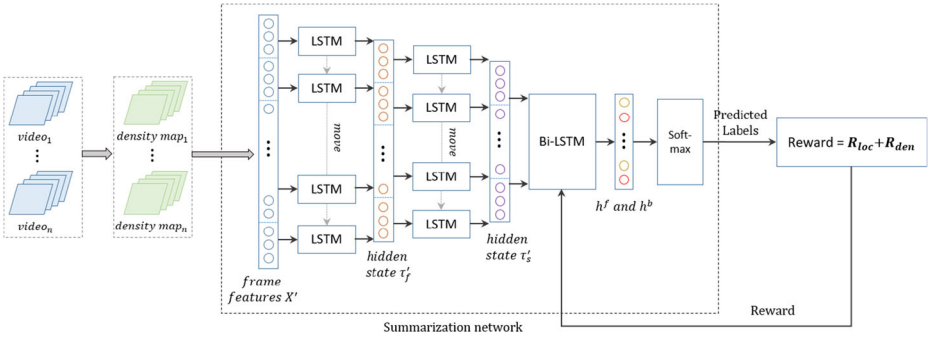


Fig. 1 Training summarization network with reinforcement learning

was enhanced by a DPP to increase the diversity in the selected frames. The concern of Zhang et al. [49] was to leverage non-parametric learning from exemplar videos to transfer summary structures to novel input videos. The two points of interest are how similar the new video is to annotated ones and how the training videos are summarized. The similarity was inferred by comparing visual features at each frame. And key frames were selected from the human created training summaries. Vasudevan et al. [46] proposed a method that generated video summaries adapted to a text query. The technical key is the relevance model to rank frames of a video according to their relevance given a text query. A learned visual-semantic embedding space and a query-independent term help to compute the relevance, while summary frames were selected in terms of relevance, representativeness and diversity using a submodular mixture of objectives. Feng et al. [6] considered that better summary come from the understanding of whole video. Hence, an external memory was utilized to record the whole video, then it was understood by a global attention mechanism.

Although various supervised approaches have achieved remarkable results on benchmark data sets, such as SumMe [11] and TVSum [42], the supervised techniques have limited applicability when the annotated data is scarce. Tagging surveillance videos recorded by a camera is a time-consuming task. In addition, there are differences between surveillance videos recorded by different cameras. Hence, learning from manually labeled video summary ground truth is not suitable for our application.

2.2 Unsupervised video summarization

Unsupervised approaches select key frames/shots without the guidance of human-created ground truths but rely on manually designed criteria, web images or video categories. Song et al. [42] observed that the title (title-based images) always serves as a prior on expected summary. To select frames from input videos, they learned a joint factorial representational of images and video data sets. To summarize user-generated videos which consist of long, poorly-filmed and unedited contents, Lei et al. [22] developed a graph-based method to rank the frame segments (clustering frames of the original video). Kang et al. [18] analyzed that space-time were informative in some videos. And salient portions in videos is determined by sptio-temporal contrast. Lee et al. [40] proposed methods that learns category-independent importance cues to target key objects and people to summarize egocentric videos which captured from a wearable camera. The goal of Lu et al. [28] was to create story-driven summaries for long, unedited videos. The basic ideal of [23] was that summary should be key frames whose visual content was similar to contents of original videos, while different from the frames already selected in the summary. Subsequent works were affected by this criteria. Zhou et al. [53] proposed a deep reinforcement learning framework to train deep summarization network. The reward function is inspired by this general criteria. They used a dissimilarity function to measure the different between the selected frames and a set of medoids to measure the similarity between selected frames and contents of original videos. Another criteria is the machine-generated summary should be similar to the original video in an abstract semantic space. Inspired by this criteria, Zhang et al. [50] used regression loss for matching summaries, the summary and the original, mismatched summary and original to measure the amount of information conveyed in the original sequence and the summary. With the same criteria, Mahasseni et al. [33] built a Generative Adversarial Network (GAN) by the selector LSTM, the encoder LSTM, the decoder LSTM and discriminator LSTM. The summarization performance of these unsupervised methods [33, 50, 53] is superior than contemporaneous supervised methods on benchmark data sets [11, 42].

Obviously, the criteria for training summary model is designed manually according to the application. And the changes in the crowd reflects main contents of crowd surveillance video. Therefore, crowd aware rewards are used as the criteria to evaluate whether the summary results capture the main content of crowd surveillance video. On the other hand, hierarchal network which proposed by a supervised method [51] is adopted to capture long-span dependencies because surveillance video always across longer time spans.

2.3 Reinforcement learning

Reinforcement learning (RL) has been popular and successful in many areas. Seijen et al. [45] decomposed the reward function into a number of different reward functions for constructing an easy-to-learn value function. Dong et al. [5] train an attention agent for action recognition because the attention model cannot be trained end-to-end with the whole network. Janisch et al. [16] formalized the problem of classification with costly features as a Markov decision process. Hence, RL was a natural choice. In our previous work [24], we modeled the dynamic selection of nodes in camera network as a Markov decision process to obtain the most informative camera node while simultaneously reducing camera switching. Jay et al. [17] utilized RL to tackle the crucial and timely challenge of internet congestion control. Zhou et al. [53] first used RL in the domain of video summarization. The two main technical differences between our and their approaches are hierarchical network and crowd aware rewards which make our approach more suitable for long crowd surveillance videos summarization.

3 Problem formulation and background

3.1 Problem formulation

An input video can be represented as a series of consecutive frames:

$$F = \{f_1, f_2, \cdots, f_t, \cdots, f_T\}, \tag{1}$$

where f_t is the frame at time t. There are two forms of output summarization. The first is selected key frames [10, 27, 34] as the output:

$$F' = \{f_{r1}, f_{r2}, \cdots, f_{rn}, \cdots, f_{rN}\}, \tag{2}$$

where $F' \in F$ is the selected frames with a size of N ($N < T$), $r_n \in \{1, 2, \cdots, N\}$ and $r_n < r_{n+1}$. The second is selected interval-based key shots [11, 12, 37] as the output:

$$F'' = \{\mathcal{F}_1, \cdots, \mathcal{F}_k\}, \tag{3}$$

where $F'' \in F$ is the selected shots, $\forall \mathcal{F}_i \cap \forall \mathcal{F}_j = \varnothing$, $i \neq j$.

Essentially, our approach falls into the second category. We try to select a smaller set of interval-based key shots for video summarization. But there are two problems need to be solved. First, most of surveillance videos contain thousands of frames, and how to capture dependencies across longer time spans. Second, how to select shots to summary main contents in crowd surveillance videos. For the former problem, as shown in Fig. 1, a hierarchical LSTM [51] is used for our summarization model. The input video is divided into some subsequences evenly. Then the first layer LSTM is utilized to exploit the sequential information by performing convolutional operations on each

subsequence which typically contains up to 80 consecutive frames according to the performance of RNN. The output of each subsequence is a hidden state of LSTM that can capture short-range temporal dependency. These hidden states are treated as the input of next layer. Hence, the second layer LSTM can capture the long-rang temporal dependency. This kind hierarchical RNN can reduce the information loss in long sequence modeling to improve summary performance. For the second problem, a popular criteria of high-quality video summary is that summary should be key frames/shots whose content was similar to content of original videos, while different from the frames already selected [23]. Hence, we use the distance between selected shots to cluster centers of the original video frames in terms of crowd location and density to measure the similarity. The intuition behind it is that the clustering centers can represent videos contents [53], and the closer the selected shots to clustering centers, the more similar they are to videos content. The dissimilarity of crowd location and density between selected shots is used to measure the difference. The novel crowd location-density based measurement is utilized as the penalty term in our RL framework to teach the summarization model to select better shots. And the main content of a crowd surveillance video can be taken as a set of different crowd behaviors in our experiments.

3.2 Background: Long short-term memory (LSTM)

LSTM is a popular variant of standard Recurrent Neural Network (RNN) which constructed by feedforward network and an extra feedback connection. LSTM is designed to address the issue of hard to train for the gradient vanishing problem [1] and suitable for modeling long-range dependencies. The most significant difference between LSTM and stand RNN is the external memory cell which encodes the knowledge of inputs that have been observed up to that step. There are three gates to control the calculation of hidden state h_t and memory cell c_t. Specifically, this process can be described as follows:

$$i_t = \sigma(W_{ix}x_t + U_{ih}h_{t-1} + b_i), \tag{4}$$

$$f_t = \sigma\left(W_{fx}x_t + U_{fh}h_{t-1} + b_f\right), \tag{5}$$

$$o_t = \sigma(W_{ox}x_t + U_{oh}h_{t-1} + b_o), \tag{6}$$

$$g_t = \phi\left(W_{gx}x_t + U_{gh}h_{t-1} + b_g\right), \tag{7}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t, \tag{8}$$

$$h_t = o_t \odot \phi(c_t), \tag{9}$$

where the input gate i_t controls whether to consider current input x_t, the forget gate f_t allows to forget previous memory c_t, and the output gate o_t decides how much of the memory to transfer to the hidden states h_t. σ denotes the sigmoid function and all the W_s, U_s, b_s are the training weights and bias. $\odot$ denotes element-wise products.

4 Approach

In this section, we describe our methods for summarization crowd surveillance videos. Because of we want to learn a model to predict probabilities for video shots in terms of crowd location and density by RL to solve the two problems described in section 3.1 (i.e., how to capture dependencies across longer time spans and how to select shots that summary main content), the location and density aware frame representation is discussed in section 4.1 first. Then, hierarchical based modeling for long time spans is described in section 4.2. Finally, we introduce new reward functions related to crowd location, density and summary criteria [23] in section 4.3.

4.1 Frame feature representation

The input of the neural network model is a set of features corresponding to the original video frames F. It has been confirmed that deep convolutional features consistently improved performance over the hand-crafted features in video summarization [48], and has been used in many works [6, 50, 52, 53]. Specifically, in our work, the visual feature vector are extracted from the penultimate layer of the GoogLeNet [44] for each frame.

However, there are many noises in the background of crowd surveillance videos, such as building, vehicle, and natural environment. The noises are all embedded in feature space if we extract the feature from each frame directly. Actually, the information of background is useless in the surveillance video summarization task. The differences of background may interfere with the practical application of learning-based video summarization methods. Because surveillance videos are always obtained from different cameras, and backgrounds in the videos may be different from each other. This can lead to an extreme situation that we need to retrain the summarization model for each surveillance video if they are all obtained from different cameras, which is obviously unacceptable.

For the reasons above, we first calculate a crowd density maps [41] set $\mathbb{M} = \{\omega_1, \ldots, \omega_T\}$ for original video frames F (Eq. 1), where ω_t is the crowd density map of the frame f_t in F. Then the deep convolutional feature vectors set

$$X = \{x_1, x_2, \cdots, x_t, \cdots, x_T\} \tag{10}$$

is extracted from the crowd density maps set $\mathbb{M}$ instead of frames F themselves as the input of our model, where x_t is the deep feature vector of the density map ω_t at time t.

As shown in Fig. 2, the crowd density map records the location and relative density of crowds and filters out the background (such as buildings and lawns) in the form of heat maps. And the feature vectors set X highlights visual information of crowd location and relative density. As discussed in the experiments, it brings another benefit for cross-scene surveillance videos summarization task. We use the vectors set as the input of our video summary model in both training and testing processes, and the output of the model is a probability value set used to evaluate each subshot.

4.2 Hierarchical deep summarization network

In this section, we describe the hierarchical summarization model in details. As discussed in [36], the ideal length of video for LSTM modeling is less than 100 frames. Thus, it is challenging to model surveillance videos that are usually with long durations. Zhao et al. [51] trained a two-layer LSTM

model with supervised framework to capture dependencies across long time spans. But surveillance videos are always longer than ones in standard datasets. To this end, we improve the summarization model with three layers made of LSTM units for modeling frames and shots. And we train it with an unsupervised RL framework. As shown in Fig.1, the first and second layer are two LSTMs and responsible for modeling at the frame level and shot level respectively. The third layer is a bi-directional LSTM and employed to predict the confidence of certain shot to be selected into the video summary.

Specifically, the input of the first layer is

$$X^{'} = \{x_1, \cdots, x_n\} \cup \{x_{n+1}, \cdots, x_{2n}\} \cup \cdots \cup \{x_{mn+1}, \cdots x_{(m+1)n}\}, \tag{11}$$

which means the feature vectors X (Eq. 10) is separated into m consecutive and disjoint subsequences. If the $T < (m+1)n$ in Eq. 11, the finally subsequence is padded with zeros. One subsequence in $X^{'}$ can be calculated as $LSTM(\{x_{i+1}, \cdots, x_{2i}\})$, where $LSTM(\cdot)$ is short for Eqs. (4)–(9). The output of first layer is

$$\tau_f = \{\tau_{f\_1}, \cdots, \tau_{f\_m}\}, \tag{12}$$

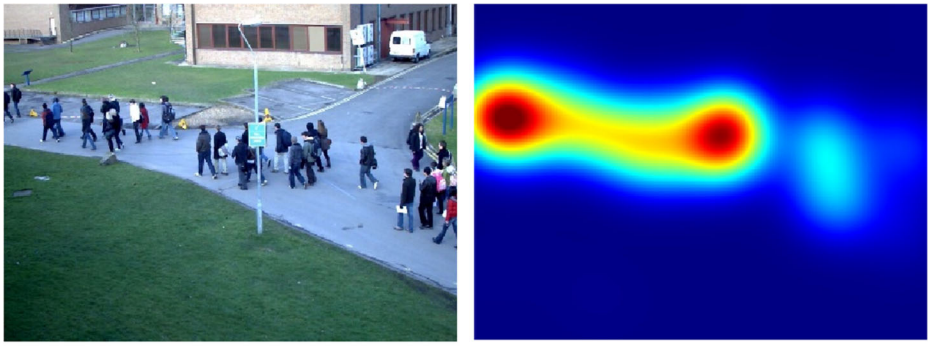
where τ_{f_i} denotes the final hidden state of the i_th subsequences in $X^{'}$, which can be treated as the representation of the i^{th} subsequences. High quality summary results can be obtained by two layers LSTM [51], however, surveillance videos are always longer than standard videos (such as videos in SumMe [11]). Thus, three layers LSMT is used in our work and τ_f is further divided into shots

$$\tau_f^{'} = \{\tau_{f\_1}, \cdots, \tau_{f\_n}\} \cup \{\tau_{f\_(n+1)}, \cdots, \tau_{f\_2n}\} \cup \cdots \cup \{\tau_{f\_(kn+1)}, \cdots, \tau_{f\_(k+1)n}\}, \tag{13}$$

as input of the second layer ($k < m$). It means features vectors τ_f of subsequences is separated into k consecutive and disjoint representation of shots (The finally shots is padded with zeros if $m < (k+1)n$). The similar with Eq. (12), the output of second layer is

$$\tau_s = \{\tau_{s\_1}, \cdots, \tau_{s\_k}\}, \tag{14}$$

where τ_{s_i} denotes the final hidden state of the i^{th} subsequences in $\tau_f^{'}$, which can be treated as the representation of the i^{th} shots. Then, similar with $\tau_f^{'}$, τ_s is divided into subshots



(a) (b)

Fig. 2 An example of crowd density map. (**a**) A frame in the crowd surveillance video dataset PETS [7]; (**b**) the corresponding crowd density map calculated by [41]

$$\tau_s^{'} = \{\tau_{s\_1}, \cdots, \tau_{s\_n}\} \cup \{\tau_{s\_(n+1)}, \cdots, \tau_{s\_2n}\} \cup \cdots \cup \{\tau_{s\_(qn+1)}, \cdots, \tau_{s\_(q+1)n}\}, \quad (15)$$

as input of the third layer which is composed of a bi-directional LSTM. The output of last layer are $h^f = \{h_1^f, \cdots, h_q^f\}$ and $h^b = \{h_1^b, \cdots, h_q^b\}$, where h^f and h^b are output hidden state of forward LSTM and backward LSTM respectively. Then, a softmax layer is used to predict a probability

$$p_t = softmax\left(\tanh\left(W_p\left[h_t^f, h_t^b, \tau_{s\_t}\right] + b_p\right)\right) \quad (16)$$

to indicate whether the t^{th} shot is select or not. And W_p and b_p are the parameters to be learned. The softmax function is utilized to constrain the sum of the elements in p_t to be 1. Actually, p_t is a two dimensional vector, each element of which indicates the possibility of the i^{th} subshot is key or non-key.

4.3 Reward function

During the training process, the reward function will send a signal to the summarization model in each iteration to evaluate the result of generated summaries. RL ensures the summarization model to select high-quality summaries when the expected rewards is maximized. As stated in the criteria [23], high-quality summary should keep the contents of original videos, while different from the frames/shots already selected in the summary. And the main contents of crowd surveillance videos are crowd density and location. To this end, we design a reward function to evaluate the quality of summaries according to crowd density and location.

Crowd location reward. As discussed in section 4.1, we extract visual features from crowd density map [41] as the input of our summarization model. Although the density map keeps information of crowd density, it is also sensitive to crowd location and emphasizes where the highest density of crowd is on the map. Therefore, visual features extracted from crowd density map are used to measure the quality of the selected shots, and this term is named the crowd location reward. Inspired by [53], unsupervised diversity-representativeness reward is employed,

$$R_{loc} = \frac{1}{|\mathcal{Y}||\mathcal{Y}-1|}\sum_{t\in\mathcal{Y}}\sum_{\substack{t^{'}\in\mathcal{Y} \\ t^{'}\neq t}} d\left(s_t, s_{t^{'}}\right) + exp\left(-\frac{1}{T}\sum_{t=1}^{T} \min_{t^{'}\in\mathcal{Y}} \left\|s_t - s_{t^{'}}\right\|_2\right), \quad (17)$$

where $d(\cdot)$ is cosine dissimilarity, $\mathcal{Y} = \{y_i | a_{y_i} = 1, i = 1, \cdots, |\mathcal{Y}|\}$ contains indices of selected T shots, s is shot feature. Similar to [6], s is calculated by average deep features of all frames within the shot. According to the criteria [23], during the training process, the first term computes dissimilarity between selected shots and the second term measures how much information of original video do the shots contain.

Crowd density reward. There are two preference about crowd density when users browse surveillance videos. First, the shots are useless if they do not involve characters, which should be filtered out as redundant information. Second, the rate of change in the number of personnel (i.e., the rapid appearance or disappearance of crowds) is a point of concern. Hence, for the first preference, we penalize the summary video with −5 if no characters are shown in a shot of the summary video. Otherwise we reward the summary video with +1. The reward R_{den_1} is calculated as:

$$R_{den\_1} = \begin{cases} R_{den\_1} + 1, & if\,(\rho_t > 0) \\ R_{den\_1} - 5, & otherwise \end{cases}, \tag{18}$$

where ρ_t is the estimated count of personnel for each selected video shot, which calculated by [41], $t \in \mathcal{Y}$. Eq. 18 indicates that the summary video will get a high R_{den_1} score if characters are shown in each shot. In other words, R_{den_1} is used to prevent the shots which contains no characters in the summary video. For the second preference, we introduce a reward R_{den_2} based on the classical definition of rate of change. The intuition behind this reward is: in the same time span, more difference between ρ_{t-1} and ρ_t, the higher reward that the model can receive. We compute R_{den_2} as the mean of the pairwise rate of change between adjacent two selected shots:

$$R_{den\_2} = \frac{1}{|\mathcal{Y}-1|} \sum_{t \in \mathcal{Y}} r(s_{t-1}, s_t), \tag{19}$$

where $r(\cdot, \cdot)$ is the rate of change function calculated by:

$$r(s_{t-1}, s_t) = \frac{|\rho_t - \rho_{t-1}|}{(\rho_t + \rho_{t-1})}. \tag{20}$$

Hence, crowd density reward can be calculated as:

$$R_{den} = R_{den\_1} + R_{den\_2}, \tag{21}$$

Finally, R_{loc} and R_{den} complement to each other and work jointly to guide the learning of our summarization model.

The feature vectors set X defined in Eq. 10 is used as the input of our video summary network in both training and testing processes, and the output of the network is a set of probability values corresponding to each subshot. The set of probability values is used to evaluate each subshot in both training and testing processes, and we select top 15% subshots as the summary result from the shots sequence according to the descending order of probability values. During the training process, Eq. 21 is used to evaluate the summary result (i.e., a sequence of shots) and send the bi-directional LSTM a signal to train our summarization model with policy gradient, where the feature vector of a subshots is the mean of the features vectors of frames in it. And the features vectors of frames are recorded in the set X in Eq. 10. We will discuss models trained with different rewards in section 5.

4.4 Implementation details

We use GoogLeNet [44] trained on ImageNet [4] to extract frame features. We train our summarization model with policy gradient. Adam [20] with mini-batch size of 10 and initial learning rate $1e-4$ is implemented as the optimization algorithm. The dimension of embedding space and hidden units of all used LSTM are 256. The epoch for training LSTM is 40. The length $\mathcal{L}$ of LSTM varies from 25 to 60 in each layer can obtain stable performance [51]. The lengths $\mathcal{L}$ of three layers LSTM are set to 30 in our experiments. Hence, our model can handle the frame sequence less than 27,000 ($30 \times 30 \times 30$). There are two ways to deal with the problem of videos contain more than 27,000 frames. First, $\mathcal{L}$ can be set to a larger value. Second, videos can be sampled to meet the constraint because consecutive frames in a video share much redundant semantic information. Videos are padded with zeros if they contain fewer than 27,000 frames.

5 Experiments

To verify the effectiveness of the proposed approach, it is tested on three publicly surveillance video datasets [8, 9, 52] and web videos. We first compare our method with several baselines to demonstrate the contribution of different rewards and hierarchical summarization model to the final performance in section 5.4. Then we compare our method with several state-of-the-art methods on short and long crowd surveillance video respectively in section 5.5. In addition, the advantage of using density map is discussed in the experiments.

5.1 Datasets

We test our summarization model on three publicly crowd surveillance video datasets UMN [8], PETS [7] and WorldExpo'10 dataset [47]. The UMN [8] dataset consists of 11 videos. The content of these videos is consisted of several distinct crowd activities, such as wandering, being scattered in all directions and so on. PETS [7] dataset comprises multi-sensor sequences containing crowd scenarios with increasing scene complexity. The main crowd behaviors in the videos content include the crowd moving slowly or rapidly through the scene, the crowd standing in the scene, the crowd gathering or thinning. The two surveillance video datasets are very useful to illustrate the performance of our method, because crowd behaviors which form the main content of surveillance videos are very different. A high performance video summary is one that preserves these differences while filtering out redundant information. So we choose them to illustrate the differences among our method, baseline methods and state-of-the-art methods. But the videos in UMN dataset and PETS dataset are short (about 1–2 min). Hence, WorldExpo'10 dataset [47] is used to verify the effectiveness of our method on long videos. It has 1127 one-minute long video sequences out of 103 scenes and 5 one-hour long video sequences from 5 different scenes, all from Shanghai 2010 WorldExpo captured by 108 surveillance cameras. The 5 one-hour long video sequences which are separated into 15 consecutive and disjoint subsequences with about 20 min long are used in our experiments. A notable benefit is that, the same with the other two datasets [7, 8], the video content can be divided into several different behaviors of the crowd. It is useful to illustrate the performance of methods in retaining the main content of crowd surveillance videos.

Cross-scene surveillance videos summarization is important to actual applications because training a summary model for each scene is a time-consuming task. To discuss the advantages of using density map in our model for cross-scene summary, we download several surveillance videos from YouTube as a supplementary dataset to train video summary models. However, these downloaded videos are still very lengthy and noisy since they contain a proportion of frames that irrelevant to crowd scene. Therefore, we segment web videos using KTS [37] and filter out the noisy parts. Finally, 20 downloaded surveillance videos (less than 4 min) are served as a training dataset in our experiments.

5.2 Evaluation setup

For a fair comparison among our method, baseline methods and state-of-the-art methods, the keyshot-based metric proposed in [48] is used for evaluation. Let A be generated keyshots which to be less than 15% in duration of original video and B the user-annotated keyshots. The precision P and recall R can be calculated as:

$$P = \frac{\text{duration of overlap between } A \text{ and } B}{\text{duration of } A}, \tag{22}$$

$$R = \frac{\text{duration of overlap between } A \text{ and } B}{\text{duration of } B}, \tag{23}$$

then, the harmonic mean F-score:

$$F = 2P \times \frac{R}{P+R} \times 100\%, \tag{24}$$

is used as the evaluation metric. The output of our method is importance score p_t of keyshots in $\tau_s^{'}$. But several methods [26, 43, 53] only provide key frame scores. To generate keyshots for a fair comparison, the videos are initially temporally segmented into disjoint intervals evenly with the same length (the count of frames) as keyshots in $\tau_s^{'}$. Then, the importance score of an interval is calculated as the average score of the frames in that interval and the resulting intervals are ranked based on their importance score. Finally, the keyshots are selected from the ranked intervals, which are less than 15% of the duration of the original video.

Although ground truth labels evaluation is often carried out using human judgments, the standard approach is described in [12, 42]. We create the ground truth set according to the standard approach and surveillance video datasets. Before the task, each video is segmented into uniform-length shots for capturing local context with good visual coherence. The shot length is empirically two seconds. Then the shots are clustered using k-means (k = length of video in seconds/10) and presented the shots within each cluster in random order to prevent chronological bias [2] which indicates that humans have a tendency to assign higher scores to shots appear earlier in video. During the task, the participants were asked to provide an importance score of 1 to 5 to each of shots. The score of 5 indicates that the shot can represent the activity of crowds very well. The score of 1 indicates no crowd activity. In addition, the frequency of score 5, score 4, score 3, score 2 and score 1 in the ground truth of a shot were assigned between 1% and 5%, 5% and 10%, 10% and 20%, 20% and 40% and gets the rest respectively to ensure the score distribution is appropriate for generating summaries.

5.3 Baselines and comparison

To clarify the performance of our method, we set several baseline models. To investigate how much different rewards contribute to the hierarchical summarization network model, the baseline models as the ones trained with R_{loc} only and R_{den} only, which are denoted by L-HSN and D-HSN, respectively. The model trained with the two rewards are represented as LD-HSN. Furthermore, our hierarchical summarization model contains three layers LSTM is suitable for the summarization task that the length of input video is about 20–30 min. But we have to pad with lots of zeros at the finally subsequence of X' if the length of input video is less than 5 min (such as videos in dataset [7, 8]). An efficient way is to use hierarchical model with two layers LSTM for short videos summary, i.e. the parameter τ_{s_t} in formula (16) is replaced by τ_{f_t}. The three layers model is denoted by HSN_3 while two layers model is denoted by HSN_2. Baseline models L-HSN_2, D-HSN_2, LD-HSN_2 and LD-HSN_3 are discussed in experiments. To verify our hierarchical summarization network model, we use DSN [53] which was constructed by a bidirectional recurrent neural

network and a fully connected layer instead of our hierarchical summarization network. This baseline model is represented as LD-DSN.

To compare with other approaches, we retrieve results of other approaches including surveillance videos summary [26, 43], DPP-LSTM [42], DSN [53], GAN-based [33] methods.

5.4 Comparison with baselines

Qualitative Evaluation. We compare our method LD-HSN with two baselines L-HSN and D-HSN on datasets UMN [8], PETS [7] to investigate how much different rewards contribute to the model. Quantitative evaluation on the two datasets can make it easier to understand the difference between the two baselines and our method. The three models (denoted by L-HSN_2, D-HSN_2, LD-HSN_2) consist of two layers LSTM respectively because videos in UMN and PETS are short. We provide qualitative results for two example videos that from UMN (a1 ~ a3) and PETS (b1 ~ b3) in Fig. 3.

The main content in the example video from UMN (a1 ~ a3 in Fig. 3) consists of two parts. The crowd behaviors are working around in the temporal interval Part I while swarming from all directions in the temporal interval Part II. As shown in (a1), the summarized shots that obtained by the reward R_{loc} are always closer to ground truth in Part I than that in Part II. Besides, the shots do not contain any information about people in Part II may be selected as part of the result, because the reward R_{loc} is more sensitive to changes in crowd position.

For a more comprehensive and accurate summary of the crowd behaviors in the video, the crowd density reward R_{den} is used as a supplementary. As shown in (a2), most of selected frames fall into the temporal interval Part II because R_{den} is designed to capture changes in the number of people. In addition, R_{den_1} can effectively prevent the model from selecting shots that without any information of people as parts of the summary result. The summary result produced by LD-HSN_2 (a3 in Fig. 3) is much closer to ground truth in the two parts. It is because LD-HSN_2 benefits from that the changes in crowd position and density are captured simultaneously.

As discussed above, the purpose of our method is to summarize crowd behaviors resulting from changes in crowd position and density in video sequences. And the shots which do not contain the changes in crowd position and density are always be filtered out as the redundant information. As shown in Fig. 3 (b1) ~ (b3), the main content in the example video from UMN consists of three parts. The crowd behaviors are getting together or swarming from all directions in the temporal intervals Part I and Part III respectively, while keeping still in the temporal interval Part II. From results of (b1) ~ (b3), we can observe that almost all of selected key shots fall into Part I and Part III. The peak regions of ground truth are almost captured by LD-HSN_2. While shots that without significant changes in the crowd in Part II are filtered out as the redundant information.

Quantitative evaluation. We compare our method with several baselines to investigate the different hierarchical summarization network models and rewards on datasets PETS [7], UMN [8] and WorldExpo'10 [47]. Videos in UMN and PETS last 1–2 min, and one-hour long videos in WorldExpo'10 are separated into 15 sub-videos with about 20 min long. According to the hierarchical structure, summarization models in Table 1 can be divided into three types: single layer (LD-DSN), two layers (L-HSN_2, D-HSN_2 and LD-HSN_2) and three layers (L-HSN_3, D-HSN_3 and LD-HSN_3).

We can find that the performances of two-layer LSTM model LD-HSN_2 and three-layer LSTM model LD-HSN_3 are significantly better than those of single-layer LSTM model LD-

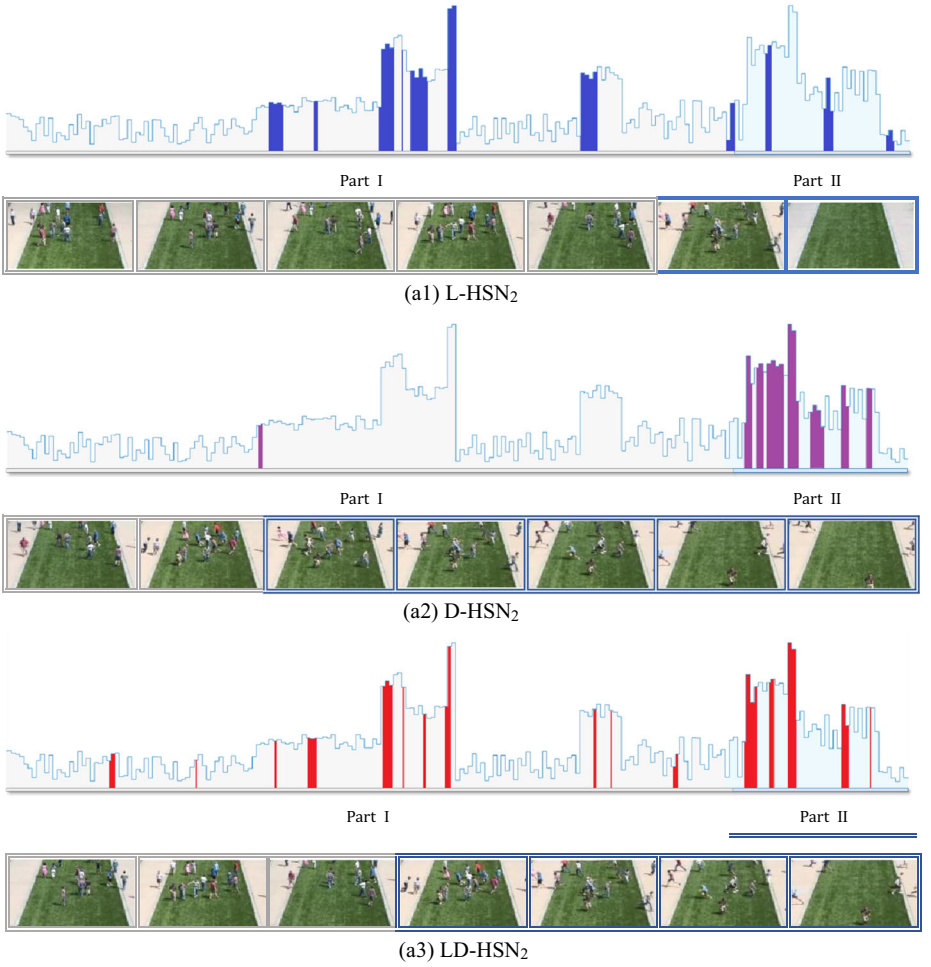
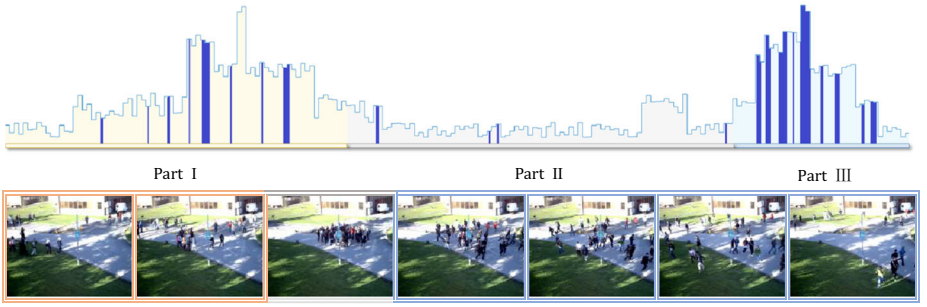
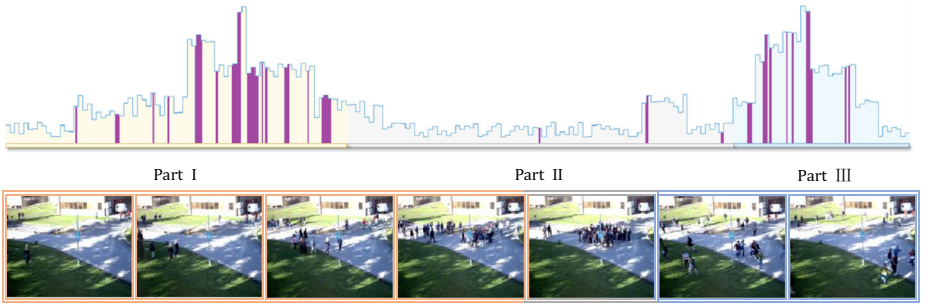


Fig. 3 Quantitative evaluation among our model LD-HSN_2 and two baselines L-HSN_2, D-HSN_2 on datasets UMN (a1 ~ a3) and PETS (b1 ~ b3), respectively. The light-gray, light-blue and light-yellow bars in (**a1**) to (**b3**) correspond to ground truth importance scores in different temporal intervals, while the colored areas correspond to the selected parts by different models

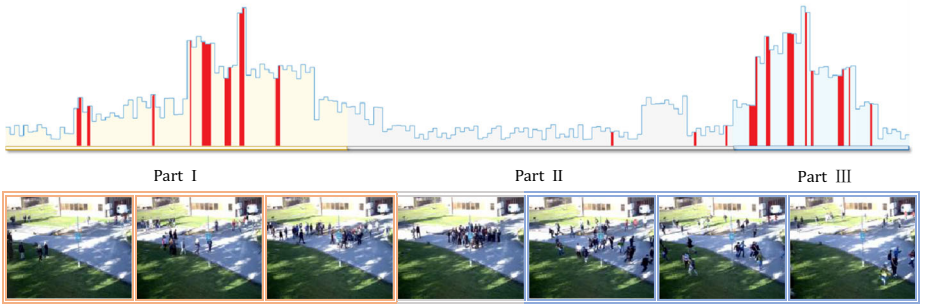
DSN on long video dataset WorldExpo'10 from Table 1, which indicates that the hierarchical summarization network can capture more crowd changes on long surveillance video sequences. On the other hand, the performances of LD-HSN_2 and LD-HSN_3 are still slightly better than those of LD-DSN on short video datasets UMN and PETS, which indicates multiply layers LSTM models also have advantages in short video summary. On the other hand, comparing LD-HSN_3 with LD-HSN_2 on short video datasets UMN and PETS, we can find that the two models perform similarly (48.4 vs. 48.3 on UMN and 47.6 vs 47.6 on PETS) on short surveillance video sequences. But the performance is difference on long surveillance video sequences (39.6 vs. 37.1 on WorldExpo'10). It indicates that the hierarchical network with three layers LSTM can improve summary performance on long crowd surveillance video (about 20 min in our experiments).



(b1) L-HSN$_2$



(b2) D-HSN$_2$

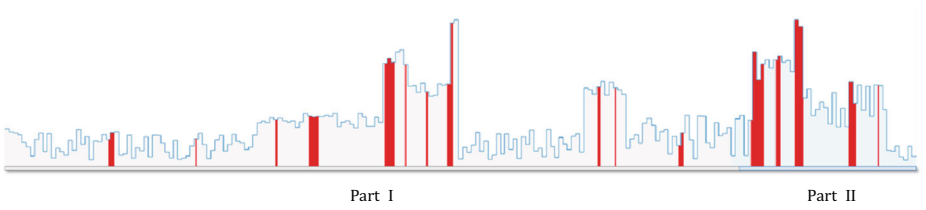


(b3) LD-HSN$_2$

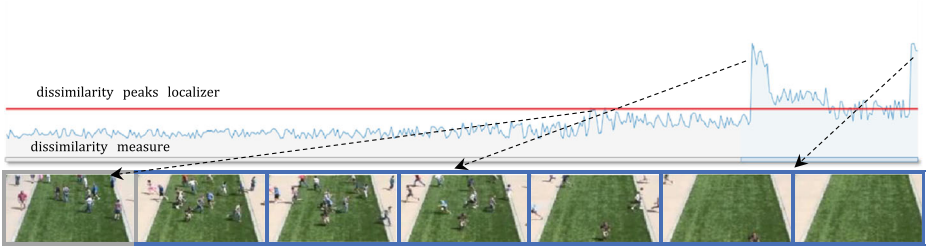
Fig. 3 (continued)

Table 1 F-scores of different variants of our method on PETS, UMN and WorldExpo'10

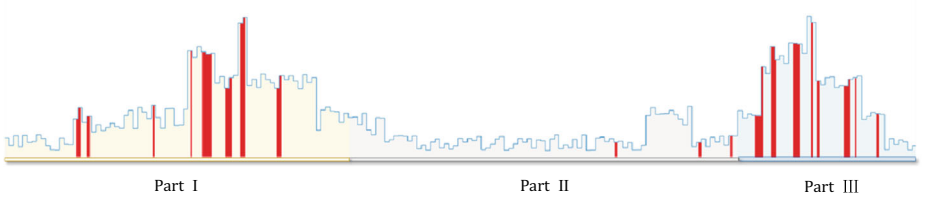
Method	PETS	UMN	WorldExpo'10
L-HSN$_2$	45.2	44.5	31.6
D-HSN$_2$	42.8	41.2	29.7
L-HSN$_3$	45.6	44.5	34.4
D-HSN$_3$	43.3	41.4	32.9
LD-DSN	47.1	46.7	33.8
LD-HSN$_2$	48.3	47.6	37.1
LD-HSN$_3$	**48.4**	**47.6**	**39.6**



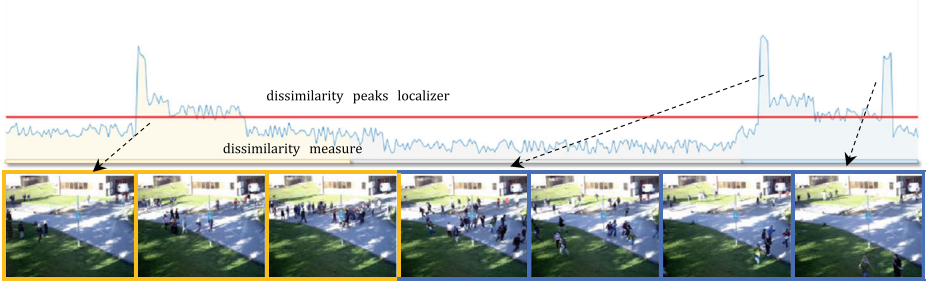
(a1) LD-HSN$_2$



(a2) Dissimilarity-based method [10]



(b1) LD-HSN$_2$



(b2) Dissimilarity-based method [10]

Fig. 4 Comparison between our method and dissimilarity-based method [10]. The blue curves in (**a2**) and (**b2**) are temporal dissimilarity, while the red straight lines are used to localize the local dissimilarity peaks

We also can find that video summary model LD-HSN$_n$ trained with crowd location reward and crowd density reward jointly outperforms video summary model trained with crowd location reward L-HSN$_n$ or density reward D-HSN$_n$ from Table 1. It demonstrates that we can better train our video summary model HSN to produce high-quality summaries by using crowd location reward R_{loc} and crowd density reward R_{den} jointly.

5.5 Comparison with stat-of-the-art

We compare our method with the current state-of-the-art which includes surveillance video summary methods [9, 29] and unsupervised deep learning based methods [33, 48, 53].

Table 2 F-scores of unsupervised deep learning based approaches and dissimilarity-based approach on PETS, UMN and WorldExpo'10

Method	PETS	UMN	WorldExpo'10
LSTM [48]	40.8	39.2	30.5
GAN [33]	42.7	40.2	31.4
DR-DSN [53]	44.6	43.3	32.1
Change point [29]	41.1	37.2	25.6
Dissimilarity measure [9]	43.9	40.4	25.3
LD-DSN	47.1	46.7	33.8
LD-HSN$_2$	48.3	47.6	37.1
LD-HSN$_3$	**48.4**	**47.6**	**39.6**

Comparison with surveillance video summary methods. The aim of [26, 43] is to summary predefined abnormal events. Therefore, we compare our method with the more general surveillance video summary approaches. Gao [9] and their previous work [29] proposed dissimilarity-based surveillance video summary methods. The intuitive behind their works is to detect the changes from a surveillance video sequence. To this end, their works are consisted of two key processes: dissimilarity measure and dissimilarity peaks localizer (Fig. 4 a2 and b2). The first step is to measure dissimilarity between frames, the second step is to localize the local dissimilarity peaks.

As shown in Fig. 4, we compare our method with dissimilarity-based method [9] for the same example videos in Fig. 3. Although their work can filter out frames which do not contain any changes well (Fig. 4 b2, frames in the temporal interval Part II are almost filtered out), low-level features, such as color histogram, are used to measure the dissimilarity between frames. It is suitable for summarization moving targets under the condition of the sparse target environment. But it cannot capture changes in crowd behavior accurately. For example, the background without moving targets was preserved in Fig. 4 (a2), while almost all frames in Part I were discarded. We can see that our method clearly outperforms [9, 29] on the three datasets from Table 2. It demonstrates that we can better capture surveillance video shots as the summary result, which outline the main behaviors of the crowd over a limited frames sequence.

Comparison with deep learning based methods. The quantitative comparison among our method (LD-DSN, LD-HSN$_2$, and LD-HSN$_3$), the state-of-the-art unsupervised deep learning based methods [33, 48, 53] and dissimilarity-based surveillance video summary methods [9, 29] on three datasets is illuminated in Table 2. Eighty percent of the data is used for training models and the rest of the data is used for testing. We can find that, benefit from deep features, unsupervised deep learning based methods (including our methods) clearly outperforms dissimilarity-based method [9] on datasets UMN and WorldExpo'10. But dissimilarity-based

Table 3 F-scores of unsupervised deep learning based approaches on PETS, UMN and WorldExpo'10 under the condition of cross-scene

Method	PETS	UMN	WorldExpo'10
LSTM [48]	32.4	30.7	21.4
GAN [33]	34.1	32.3	24.3
DR-DSN [53]	39.7	35.1	27.6
LD-HSN$_3$	**48.2**	**47.1**	**39.2**

methods [9, 29] score well on dataset PETS, because color differences between frames can reflect changes in crowd behavior to some extent.

Comparing the most competitive deep summary model (DR-DSN) which trained with diversity-representativeness (DR) reward with our baseline LD-DSN (44.6 vs. 47.1 on PETS, 43.3 vs. 46.7 on UMN and 32.1 vs. 33.8 on WorldExpo'10), it demonstrates that crow location reward and crow density reward can help us training the summary model to better capture the representative behaviors of the crowd.

Cross-scene surveillance videos summary. For better summary results in the case of different monitoring scenes, the most straightforward way is to retain the surveillance video summarization model for each monitor scene. But it is a time-consuming task. An ideal video summary model should generate better results while without repeated training. In this section, a quantitative comparison is used to illustrate that we do not need to repeat training for different monitoring scenes by using our method. Table 3 compares our method with the state-of-the-art unsupervised deep learning based methods under the condition that the training dataset is obtained from YouTube while the test datasets are PETS, UMN and WorldExpo'10, respectively. We can find that our method can maintain stable performance, while the performance of other deep learning based methods is obviously decreased. Because the density map used in our method has filtered out the background information.

6 Conclusion

In this paper, we present a RL-based unsupervised method for summarization crowd surveillance videos. Our goal is to maintain distinct crowd behaviors while filter out redundancy shots in the summary result. To this end, a crowd location-density reward is used to teach our model to produce high-quality summaries. Compared with dissimilarity-based surveillance videos summarization methods and deep learning based methods, our method can better capture surveillance video shots as the summary result, which outline the main behaviors of the crowd over a limited frames sequence. On the other hand, our hierarchical network model can maintain performance for long (20 min) crowd surveillance videos.

In the future, we will explore more crowd behavior patterns which could be used in surveillance videos summarization. It will expand the application scope of our method.

Acknowledgments This work was supported by National High Technology Research and Development Program of China (No.2007AA01Z334), National Natural Science Foundation of China (Nos.61321491 and 61272219), Innovation Fund of State Key Laboratory for Novel Software Technology (Nos. ZZKT2013A12, ZZKT2016A11 and ZZKT2018A09).

References

1. Bengio Y, Simard PY, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Netw 5(2):157–166
2. Berger V (2007) Selection bias and covariate imbalances in randomized clinical trials, vol 66. Sons, John Wiley & Sons
3. Chao W-L, Gong B, Grauman K, Sha F (2015) Large-margin Determinantal point processes. UAI:191–200
4. Deng J, Dong W, Socher R, Li L-J, Li K, Li F-F (2009) Imagenet: a large-scale hierarchical image database. CVPR:248–255

5. Dong W, Zhang Z, Tan T (2019) Attention-aware sampling via deep reinforcement learning for action recognition. AAAI 33:8247–8254
6. Feng L, Li Z, Kuang Z, Zhang W (2018) Extractive video summarizer with memory augmented neural networks. ACM Multiedia:976–983
7. Ferryman JM, Pets AE (2010) Dataset and challenge. AVSS 2010:143–150
8. Fradi H, Dugelay J-L (2015) Towards crowd density-aware video surveillance applications. Information Fusion 24:3–15
9. Gao Z, Lu G, Lyu C, Yan P (2018) Key-frame selection for automatic summarization of surveillance videos: a method of multiple change-point detection. Mach Vis Appl 29(7):1101–1117
10. Gong B, Chao W-L, Grauman K, Sha F (2014) Diverse sequential subset selection for supervised video summarization. NIPS:2069–2077
11. Gygli M, Grabner H, Riemenschneider H, Van Gool L (2014) Creating summaries from user videos. ECCV:505–520
12. Gygli M, Grabner H, Van Gool L (2015) Video summarization by learning submodular mixtures of objectives. CVPR:3090–3098
13. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
14. Hong R, Tang J, Tan H-K, Ngo C-W, Yan S, Chua T-S (2011) Event driven summarization for web videos. TOMCCAP 7(4):35:1–35:18
15. Idress H, Tayyab M, Athrey K, Dong Z, AI-Maadeed S, Rajpoot NM, Shah M (2018) Composition loss for counting, density map estimation and localization in dense crowds. ECCV:544–559
16. Janisch J, Pevny T, Lisy V (2019) Classification with costly features using deep reinforcement learning. AAAI 33:3959–3966
17. Jay N, Rotman NH, Godfrey B, Schapira M, Tamar A (2019) A deep reinforcement learning perspective on internet congestion control. ICML:3050–3059
18. Kang H-W, Matsushita Y, Tang X, Chen X-Q (2006) Space-time video montage. CVPR:1331–1338
19. Khosla A, Hamid R, Lin C-J, Sundaresan N (2013) Large-scale video summarization using web-image priors. CVPR:2698–2705
20. Kingma DP, Ba J (2015) Adam: a method for stochastic optimization. In: ICLR
21. Lee YJ, Ghosh J, Grauman K (2012) Discovering important people and objects for egocentric video summarization. CVPR:1346–1353
22. Lei Z, Zhang C, Zhang Q, Qiu G (2019) FrameRank: a text processing approach to video summarization. ICME:368–373
23. Li Y, Merialdo B (2011) Multi-video summarization based on OB-MMR. CBMI:163–168
24. Li Q, Sun Z, Chen S, S-m X (2016) Dynamic node selection in camera networks based on approximate reinforcement learning. Multimed Tools Appl 75(24):17393–17419
25. Li JZN, Kong W, Liu S, Li TH, Li G (2019) Graph convolutional label noise cleaner: train a plug-and-play action classifier for anomaly detection. CVPR:1237–1246
26. Lin W, Zhang Y, Lu J, Zhou B, Wang J, Yu Z (2015) Summarizing surveillance videos with local-patch-learning-based abnormality detection, blob sequence optimization, and type-based synopsis. Neurocomputing 155:84–98
27. Liu T, Kender JR (2002) Optimization algorithms for the selection of key frame sequences of variable length. ECCV:403–417
28. Lu Z, Grauman K (2013) Story-driven summarization for egocentric video. CVPR:2714–2721
29. Lu G, Zhou Y, Li X, Yan P (2017) Unsupervised, efficient and scalable key-frame selection for automatic summarization of surveillance videos. Multimed Tools Appl 76(5):6309–6331
30. Lu X, Wang W, Ma C, Shen J, Shao L, Porkli F (2019) See more, know more: unsupervised video object segmentation with co-attention siamese networks. CVPR:3623–3632
31. Lv P, Liu S, Mingliang X, Zhou B (2018) Abnormal Event Detection and Location for Dense Crowds using Repulsive Forces and Sparse Reconstruction. CoRR abs/1808.:06749
32. Ma Y-F, Lie L, Zhang HJ, Li M (2002) A user attention model for video summarization. ACM Multimedia: 533–542
33. Mahasseni B, Lam M, Todorovi S (2017) Unsupervised video summarization with adversarial LSTM networks. CVPR:2982–2991
34. Mundur P, Rao Y, Yesha Y (2006) Keyframe-based video summarization using delaunay clustering. Int J Digit Libr 6(2):219–232
35. Ng JY-H, Hausknecht MJ, Vijayanarasimhan S, Vinyals O, Monga R, Toderici G (2015) Beyond short snippets: deep networks for video classification. CVPR:4694–4702
36. Ngo C-W, Ma Y-F, Zhang HJ (2003) Automatic video summarization by graph modeling. ICCV:104–109
37. Potapov D, Douze M, Harchaoui Z, Schmid C (2014) Category-specific video summarization. ECCV:540–555

38. Pritch Y, Rav-Acha A, Gutman A, Peleg S (2007) Webcam synopsis: peeking around the world. ICCV:1–8
39. Saleh SAM, Suandi SA, Lbrahim H (2015) Recent survey on crowd density estimation and counting for visual surveillance. Eng Appl Artif Intell 41:103–114
40. Sharghi A, Lurel JS, Gong B (2017) Query-focused video summarization: dataset, evaluation, and a memory network based approach. CVPR:2127–2136
41. Sindagi VA, Patel VM (2017) CNN-based cascaded multi-task learning of high-level Priorand density estimation for crowd counting. AVSS:1–6
42. Song Y, Vallmitjana J, Stent A, Jaimes A (2015) Tvsum: summarizing web videos using titles. CVPR: 5179–5187
43. Song X, Sun L, Lei J, Tao D, Yuan G, Song M (2016) Event-based large scale surveillance video summarization. Neurocomputing 187:66–74
44. Szegedy C, Liu W, Jia Y, Sermanet P, Reed SE, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. CVPR:1–9
45. van Seijen H, Fatemi M, Laroche R, Romoff J, Barnes T, Tsang J (2017) Hybrid reward architecture for reinforcement learning. NIPS:5392–5402
46. Vasudevan AB, Gygli M, Volokitin A, Van Gool L (2017) Query-adaptive video summarization via quality-aware relevance estimation. ACM Multimedia:582–590
47. Zhang C, Li H, Wang X, Yang X (2015) Cross-scene crowd counting via deep convolutional neural networks. CVPR:833–841
48. Zhang K, Chao W-L, Sha F, Grauman K (2016) Video summarization with long short-term memory. ECCV:766–782
49. Zhang K, Chao W-L, Sha F, Grauman K (2016) Summary transfer: exemplar-based subset selection for video summarization. CVPR:1059–1067
50. Zhang K, Grauman K, Sha F (2018) Retrospective encoders for video summarization. ECCV:391–408
51. Zhao B, Li X, Xiaoqiang L (2017) Hierarchical recurrent neural network for video summarization. ACM Multimedia:863–871
52. Zhou K, Xiang T, Cavallaro A (2018) Video summarisation by classification with deep reinforcement learning. BMVC 298
53. Zhou K, Qiao Y, Xiang T (2018) Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. AAAI:7582–7589

Junfeng Xu is now a Ph.D. candidate in the Department of Computer Science and Technology in Nanjing University, China. He joined the State Key Laboratory for Novel Software Technology at Nanjing University in 2012. His research interests include Computer Vision, Multimedia Analysis and Processing.

Zhengxing Sun received the Ph.D. degree from NUAA in 1996 and finished his Post-doctoral researches in Nanjing University in 1999. He is now a full professor and academia committee man of the Department of Computer Science and Technology in Nanjing University. His research interests include Multimedia Computing, Computer Vision, and Perceptive Human-Computer Interaction.

Chen Ma is now a Ph.D. candidate in the Department of Computer Science and Technology in Nanjing University, China. He joined the State Key Laboratory for Novel Software Technology at Nanjing University in 2014. His research interests include Non-photorealistic rendering, Multimedia Analysis and Processing.