Counting Passengers in Railway Compartment Surveillance Video

YUANZHI LIANG, LI ZHU and XUEMING QIAN, Xi'an Jiaotong University

Considering the convenience and expansibility of visual-based methods, the crowd counting via deep learning in computer vision raises more attention currently. The railway compartment, as a universal scene, also has the actual demands in passengers counting. However, the common counting methods in deep learning are unable to solve the problem due to the severe scale changing, high accurate requirement, and limited embedded devices in the actual scene. In this paper, we proposed a novel counting solution to handle the passengers counting. The solution contains a motion supervised multi-scale representation method which provides proposals against scale variation, a spatially-temporally enhanced counting which provide precise counting numbers, and a partial proposal training method which conducts methods to be utilized in reality. With the proposed solution, the passengers counting task is solved in the higher accuracy and practicable in compartment environment. In experiments, the results show that all the modules in our solution are useful and efficient in passengers counting, and the performances of our method are better than others.

Categories and Subject Descriptors: I.4.9 [**Image Processing and Computer Vision**]: —*Applications*

General Terms: Design, Algorithms, Experimentation

Additional Key Words and Phrases: crowd counting, video analysis

ACM Reference Format:

Yuanzhi Liang, Li Zhu, and Xueming Qian. 2019. Counting Passengers in Railway Compartment Surveillance Video. ACM Trans. Appl. Percept. 2, 3, Article 1 (July 2019), 18 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

To achieve the higher efficiency in checking tickets, get a more flexible allocation to attendants and offer more thoughtful services in compartments, passengers counting becomes a general requirement in the railway industry. While, in counting, many approaches are available, like with infrared sensors, but those nonvisual based methods need additional equipment and professional installation. Especially in the insufficient space of compartments, any additional installation may bring some new problems in management, even disturb the original operations. Besides, if taking counting task as an individual system, rather than integrating with other demands, the cost of maintaining such a set of system is also unignorable. All these facts lead to the need of counting passengers via computer vision, because of the convenience of installation, utilization, and multi-task expansion of the visual system in applications [Korshunov and Ooi 2011; Yang et al. 2016; Grant and Flynn 2017; Li et al. 2016; Zhang et al. 2017; Hu et al. 2015].

Author's address: Y. Liang, L. Zhu and X. Qian, Xi'an Jiaotong University, Xi'an 710049; Shaanxi, P. R. China; email: liangyzh13@stu.xjtu.edu.cn, zhuli@mail.xjtu.edu.cn, qianxm@mail.xjtu.edu.cn

DOI:http://dx.doi.org/10.1145/0000000.0000000

Fig. 1. Examples for common crowd counting datasets (Shanghaitech datasets [Zhang et al. 2016a], UCF_CC_50 [Idrees et al. 2013] datasets) and passengers counting datasets. There are 5 situations in passengers counting datasets. In every situations, the scale changings are apparently larger than common datasets.

Currently, among diverse counting approaches in computer vision, deep learning based methods [Kang et al. 2017; Kang and Chan 2018; Zou et al. 2018], become the mainstream due to their splendid performances. However, considering some special properties in the surveillance videos from the compartments, the crowd counting methods show poor performance in these problems. In experiments, the method like MCNN [Zhang et al. 2016b] has count error about 30 when during the journey, but the maximum number of passengers in a compartment is 50. The present deep learning based counting methods is not practicable in this scene. Furthermore, the limitation in equipment also a critical issue to transfer common counting methods to passengers counting. In our work, three factors are summarized in the dramatic domain gap as follow:

1. Extreme scale changing: Compaed with the accustomed crowding counting dataset [Idrees et al. 2013; Zhang et al. 2016a], the scale changing is much more intense, as shown in Fig. 1, which makes the catching and representing head position by heatmaps become much more difficult. Since the ordinary regression thought of crowd density heatmaps is inadaptable, the transfer and application of crowd counting methods are impracticable in passengers.

2. Precise number in single digits: Since the upper bound of numbers in a compartment is limited in about 50, the missing in ten digits is not acceptable. Comparatively, in crowd counting, the open areas in the images enable to contain hundreds even thousands of people. In that case, the error about 10 in number is trivial. For example, in UCF_CC_50 [Idrees et al. 2013] the average of ground truth is 1280. The miscounting in 10 only takes up 0.78.

3. Limited hardware requirement: Considering the unique electric system, narrow space for equipment, and the cost of widespread deployment, machines with powerful GPUs and radiator fans are not available. Meanwhile, unfortunately, the camera for surveillance videos need to have a high definition to support the regular needs of watching by a human being. Thus, the imbalance between high-quality inputs and limited computational resource should be noticed as well. That also restricts detection based counting methods because of the costly bounding box regression and the inadaptability to restricted memory.

In this paper, to solve the defeats above and obtain better performance in passengers counting task, we focus on the surveillance video in railway compartments and propose a novel framework with an apposite training method. Our framework contains two parts: the motion supervised multi-scale representation method and the spatially-temporally enhanced counting method. The representation method provides proposals for passengers heads against scale variation. Then, the counting method counts number by given proposals to achieve better performances. Moreover, to overcome equipment limitation, we also propose a partial proposal training method to adapt the tiny memory in an embedded device. The method supports our framework to train with a small memory cost. The experiments show the efficiency of our methods in compartment videos.

The main contributions of this paper are summarized as follows:

1. The motion supervised multi-scale representation method is proposed. The method solves the extreme scale changing by a designed multi-scale network and introducing inter-frame motion knowledge in videos to help the network learning. The network with motion knowledge enables to offer a solid proposal for counting method and have better performance in perceiving passengers in images against scale disturbance.

2. We also proposed a novel spatially-temporally enhanced counting method to achieve more precise counting numbers from proposals. In this method, a special module is designed to learn the spatial and temporal information in videos, which further boost performances in counting.

3. The partial proposal training method is proposed to provide a training solution in limited hardware condition in railway compartments. Following this training, our counting methods are trained with smaller memory cost, which adapts to the actual application scenes. The proposed counting framework combined with the training method jointly provide an effective solution for the passengers counting problem in surveillance videos. In experiments, the proposed methods show better performances than other counting methods as well.

We also provide our surveillance video dataset and annotation to support further researches in passengers counting. The dataset will be availiable soon.

The remainder of this paper is organized as follows. In section II, we review the related work on counting. Section III demonstrates the details of our method. Experiments of different methods and analysis of the proposed method are set up in section IV. Finally, the conclusion is given in section V.

2. RELATED WORK

Crowd counting is a challenging task because of the complete background in different scenes and various crowd distribution [Zhang et al. 2015; Dai et al. 2019; Sam et al. 2019]. Many researches explore this topic in different views. The end-to-end counting gives counting number directly by processing single images like in [Shang et al. 2016]. Density map based counting provides a representation of the crowd distribution first, and then count number through the density map like in [Zhang et al. 2015; Boominathan et al. 2016; Zhang et al. 2016b; Sam et al. 2017; Kumagai et al. 2017; Marsden et al. 2017; Sindagi and Patel 2017]. Compared with the end-to-end method, the density map gives more details and more convenient to apply in multi-task, like crowd velocity estimation and crowd motion analysis.

Moreover, methods depend on object detection models, like faster RCNN [Ren et al. 2015], YOLO [Redmon et al. 2016], SSD [Liu et al. 2016]. These methods have excellent performance in target detection, which provides new ideas and a great improvement to various counting tasks.

2.1 Counting based on density maps

Zhang et al. gave a cross-scene crowd counting solution via a deep convolutional neural network for the first time [Zhang et al. 2015]. The proposed structure produces a density map for crowd distribution and also contains a fully connected layer for counting number regression. In the crowd counting task, scale problem is one of the most important factors that affect the performance of the algorithm directly. The method in [Boominathan et al. 2016] uses a combination with several shallow networks which correspond to multiple scales in the crowd. The thought of applying various networks to capture different scales of crowd develops in many studies. A CNN structure called multiple-column network (MCNN) [Zhang et al. 2016b] is proposed for handling crowd image in different scales. Each column of network applies to a particular scale of the crowd, and all of them are merged by 1*1 convolution to get a density map. Switchable network [Sam et al. 2017] provides a flexible way to change sub-networks for different scales. It has a switch layer to make the decision for which sub-network can be used for the current image area, and then work out a density map for the crowd. Sindagi et al. [Sindagi and Patel 2017] showed a Cascaded Multi-task CNN model for a similar but better way. This model can be divided into two stages, which are high-level prior stage and density estimation stage.

The prior stage classifies crowd density level and sends prior density information to the estimation stage for boosting performance.

In recent, more interesting crowd counting researches are proposed, Rather than regressing oridinary heatmaps, Sheng et al. [Sheng et al. 2018] propose a attribute map based counting method and provide more information for network in learning the scene. To better handle the scale variation, the SaCNN [Zhang et al. 2018] provide a novel structure for counting and a geometry-adaptive Gaussian kernel used for better representation crowd densities. Moreover, in fusing more domain knowledge, Huang et al. [Huang et al. 2018] proposed a method to utilize the body part in images, which is more robust for occlusion. Issam et al. [Laradji et al. 2018] give a solution to solve the problem with point annotation. Viresh et al. [Ferrari et al. 2018] propose a iterative thought to count with two branch CNN for both low and high resolution density maps. To better fuse multi-scale information, the adaptive fusing module is proposed in [Kang and Chan 2018] and achieve better predictions. Rather than using multi-column networks with heavy computational costs, Ze el al. [Wang et al. 2018] proposed SCNet to handle the task in limited network width. Liu el al. [Liu et al. 2019] focus on the self-supervised problem in crowd counting and propose a ranking strategy combined with siamese network to solve the problem. PCC Net [Gao et al. 2019] is proposed to deal with the problems of high appearance similarity, perspective changes and severe congestion and obtain better performance. Zheng el al. [Zheng et al. 2018] further explore the problem of diverse densities in the same scene and propose a method to solve cross-line pedestrian counting.

2.2 Detection Based Counting

In low-density crowd estimation, as the targets can be recognized respectively, the counting task can be converted to a detection task. Many detection methods are also practicable for counting.

In tradition solutions of detection [Uijlings et al. 2013], potential regions proposals give probable regions for targets first. Then, the regression or classification process is adopted for proposal areas and produce detection output. Moreover, some binary classifiers like Bayes method [Chan and Vasconcelos 2009], random forest [Pham et al. 2015] also broaden the thought of detection and counting tasks. With the increasing research of deep learning, many approaches [Girshick 2015; Redmon et al. 2016; Ren et al. 2015; Liu et al. 2016] seeking the multiple target detection solutions are proposed. The region-based convolutional neural networks (R-CNNs) [Girshick et al. 2013] greatly improve the performance in detection. It uses features extracted from CNN models rather than traditional hand-craft features like Haar [Haar 1933], etc. Then, in Faster-RCNN [Ren et al. 2015], region proposal network (RPN) is designed and to reduce calculation in the proposal and also provide faster detection in speed.

Above RCNN related methods can be summarized to the same thought of two stages processing: proposal and classification. This kind of methods have high precision in recognition, but hard to be accelerated. YOLO [Redmon et al. 2016] provides a new solution for detection and efficiently improves the speed. It converts the thought of classification to regression in sub-areas. The input images are divided into several areas; YOLO computes bounding boxes and probabilities in each area. Then, compared with given thresholds, the method decides whether the output contains targets. SSD [Liu et al. 2016] further improves YOLO and combines the idea of an anchor box. It adds convolutional layers after baseline network, which makes SSD be able to calculate in multiple scales and has better performance in the small object than YOLO.

Though detection methods have good performance in multiple detections, these methods have problem in dealing with tiny target and severe scale changing. Since the primary purpose of these methods is detection, they would reject some imperfect targets, which may make detection based methods inaccuracy for counting.

3. METHODOLOGY

In our work, our solution contains a counting framework and a unique training method. In the counting framework, we proposed a motion supervised multi-scale representation method to offer proposals for passengers head. Then, to get accurate counting results, a spatially-temporally enhanced counting method is proposed to deal with proposals

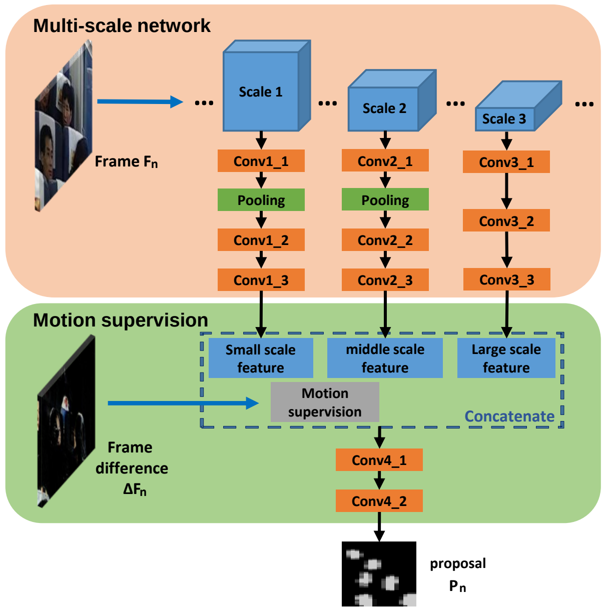


Fig. 2. Overview of our motion supervised multi-scale representation method. With the design of multi-scale network and utilization of motion knowledge, the method offer high-quality proposals for counting.

from representation methods and give counting numbers. Holistically, we train the counting framework with our novel training method and solve the passengers counting in actual scenes. More details would be discussed in this section.

3.1 Motion supervised multi-scale representation

To overcome the severe changing of scale, we proposed the motion supervised multi-scale representation method to get proposals of passengers heads. In this method, we design a multi-scale network to capture and learn targets in images. We also introduce motion information to lead the network attention on some person related area and conduct better head proposals. More details are given as follows.

3.1.1 *multi-scale network.* We propose multi-scale network to adapt the server changing in head sizes. As shown in Fig. 2, we first apply Resnet50 [He et al. 2016] as feature extractor and provide basic feature for the n-th frame images F_n. Then, three sets of CNN networks are designed for three scales of features extracted from the backbone network.

For features selection, generally, the sallow layers are available to express detailed texture, which corresponds to local features. The deep layers suit semantic information, which corresponds to global features. The coalition of shallow and deep layers features capture information in diverse scales in images [He et al. 2016]. Thus, we fuse three feature maps in a different scale from Resnet50 as shown in Fig.2. We also provide further discussion in experiments.

For the network design, rather than small kernel size used in [He et al. 2016], we applied larger kernel size to capture information from features and given the scale robustness to the networks. In detail, our multi-scale CNN extracts three scales features from Resnet correspondingly, which are 150*150 with 64 channels, 75*75 with 256 channels, and 38*38 with 512 channels. We denote them as $Scale1$, $Scale2$, and $Scale3$, respectively. Concretely, the convolution operation is applied to this $Scale1$ feature in $Conv1\_1$ with 7*7 kernel size and 32 channels. Then, max pooling layer is utilized for reducing size. The $Scale1$ feature should be reduced four times to fit the size of concatenation. Next, another two convolutional layers: $Conv1\_2$ and $Conv1\_3$ are applied to produce a small scale feature map which are 7*7 kernel size with 8 channels and 5*5 kernel size with 1 channel. Similar to the $Scale1$, $Scale2$ and $Scale3$ are used to generate middle and large scale feature maps respectively, for which $Conv2\_1$, $Conv2\_2$, $Conv2\_3$, $Con3\_1$, $Conv3\_2$ and $Conv3\_3$ are 3*3, 5*5, 5*5, 3*3, 5*5 and 5*5 kernel size and 64, 8, 1, 64, 8, 1 channels correspondingly. The $Conv1\_3$, $Conv2\_3$ and $Conv3\_3$ output multi-scale feature maps (with same sizes) individually and generate small, middle and large scale features.

3.1.2 *motion supervised method.* In surveillance videos, the frames are not isolated. Considering the movement of passengers, the relationships between frames contain valuable information about passengers locations and states.

In our method, we fully utilized the information in videos and introduce the motion knowledge into multi-scale network training. As shown in Fig. 2, we take the frame difference $\Delta F_n = F_n - F_{n-1}$ into account. An area with higher value in ΔF_n stands for a more significant probability of containing passengers. Thus, ΔF_n can be viewed as prior knowledge for passenger location.

In proposed method, we concatenate frame difference ΔF_n with the features of the multi-scale network. With motion information in ΔF_n, the network enables attention on some area with passengers movements easily, instead of attention on the background. Meanwhile, we loosen the supervision to prevent overfitting and forbid the network over-relying on some conspicuous motion. We model the motion supervision $S_n(t)$ can be defined as follows:

$$S_n(t) = \mu(t) * \Delta F_n \tag{1}$$

$$\mu(t) = \begin{cases} 1 - \frac{t}{T} & t < T \\ 0 & t \geq T \end{cases} \tag{2}$$

where t is the training epoch, $\mu(t)$ is a variable changing with training procedure, and T means the decreasing threshold. In this paper, we set the $T = 30$. $\mu(t)$ declines from one to zero linearly during the training procedure and sustains zero in the last several epochs.

In details, as shown in the Fig.2, we concatenate the motion supervision $S_n(t)$ with features from the multi-scale network. Then, another two convolution layer $Conv4\_1$ and $Conv4\_2$ are used to generate the proposal according to frame F_n.

3.2 Spatially-temporally enhanced counting method

To get a precise count number for passengers, we proposed a novel method to process proposals from the representation methods. Our counting method not only utilizes the information in the proposals but also exploit the spatial knowledge in videos and temporal knowledge in previous proposals. Specifically, the whole counting methods can be divided into three parts, the spatially enhanced module, the temporally enhanced module, and the proposal counting module. More details are given in following parts.

3.2.1 *Spatially enhanced module.* Since the motion information in ΔF_n always provide the instantaneous spatial information and are valuable in counting, we utilize the ΔF_n to enhance the proposals in the spatial side further.

We divide ΔF_n into three scaled sub-areas: 32*32, 16*16, 4*4, which corresponding to the condition of $k = 1, 2, 3$, and then we compute their mean value m_k.

$$m_k(i,j) = \frac{\sum_{\lfloor \frac{i*sub_k}{w} \rfloor}^{\lfloor \frac{i*sub_k}{w} \rfloor + sub_k} \sum_{\lfloor \frac{j*sub_k}{h} \rfloor}^{\lfloor \frac{j*sub_k}{h} \rfloor + sub_k} \Delta F_n(i,j)}{w * h / (sub_k)^2} \tag{3}$$

where i and j are the position value of the horizontal and vertical coordinate. The sub_k indicates sub-area scale in the k-th partitioning pattern (e.g., $sub_1 = 32$). $\lfloor * \rfloor$, w, and h indicate rounding down operation, width and height of ΔF_n. On each scale, we get the corresponding spatial proposal candidates of pattern k as follows:

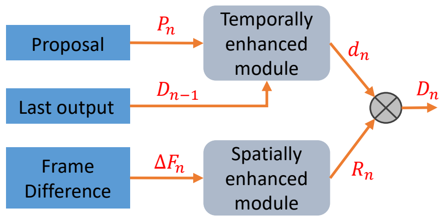


Fig. 3. Pipeline of our spatially-temporally enhanced counting method. The enhanced proposals D_n use the spatial information in ΔF_n and temporal knowledge in D_{n-1} jointly.

$$U_n(i,j,k) = \begin{cases} 1 & if \Delta F_n(i.j) \geq m_k(i,j) \\ 0 & else \end{cases} \tag{4}$$

Then we fuse the three scale spatial proposal candidates to get the jointly spatial proposal C_n of frame n as follows:

$$C_n(i,j) = \frac{\sum_{k=1}^{3} U_n(i,j,k)}{3} \tag{5}$$

Next, we get a smoothed response R_n by filtering C_n with a 5*5 Gaussian kernel as follows:

$$R_n(i,j) = Conv(C_n(i,j), G_5) \tag{6}$$

where $Conv$ is convolutional operation and R_n are calculated by the convolution of C_n with a 5*5 Gaussian kernel G_5.

3.2.2 *Temporally enhanced module.* With the video data, continuous proposal outputs are available in the output side. The proposals are not isolated and also contains useful temporal information to improve proposals. In this part, we fully explore the temporal connection between proposals and enhanced this information to get better proposals.

Based on Hebb learning rule [Sammut and Webb 2017], we fuse current raw proposal P_n from motion supervised multi-scale network with the final proposal of previous frame D_{n-1} to get the temporal enhancement. The Hebb learning rule can be explained to self-adapt network weights: increasing weight when neurons have high response simultaneously and vice versa [Sammut and Webb 2017]. As shown in Fig. 3, d_n is obtained by a weighted sum of D_{n-1} and P_n as follows:

$$d_n(i,j) = sigmoid(W_n^1(i,j) * P_n(i,j) + W_n^2(i,j) D_{n-2}(i,j)) \tag{7}$$

where W_n^1 and W_n^2 are the weight matrix for each pixel, which have the same sizes as the proposals.

$$\begin{aligned} W_n^1(i,j) &= (1-\gamma_1) * W_{n-1}^1(i,j) + \alpha_1 * D_{n-1}(i,j) * P_{n-1}(i,j), \\ W_n^2(i,j) &= (1-\gamma_2) * W_{n-1}^2(i,j) + \alpha_2 * D_{n-1}(i,j)^2 \end{aligned} \tag{8}$$

where γ_1, γ_2, α_1 and α_2 are predetermined parameters [Sammut and Webb 2017], for example, $\gamma_1 = \gamma_2 = 0.5$, $\alpha_1 = \alpha_2 = 0.5$. In initial setting, W_0^1 is all-one matrix and W_0^2 is all-zero matrix.

At last, we get the spatially-temporally enhanced proposal D_n by the certainty based weighting as follow:

$$D_n(i,j) = R_n(i,j) * d_n(i,j) \tag{9}$$

D_n take both enhancements into account, which is more robust and accurate than original proposals for counting.

3.2.3 *Proposal counting module.* In proposal counting, we first define a series of counting kernels. Each kernel is a two-dimensional matrix with the two factors: $\{width, height\}$. Concretely, the quadrate kernel is $Q_{\{S_q,S_q\}}$ and the rectangular kernel is $R_{\{2S_r,S_r\}}$, where S_q means the width of quadrate kernel and S_r means the height of rectangular kernels. The kernels can be defined as follows:

$$\begin{aligned} Q_{\{S_q,S_q\}}(i,j) &= \frac{1}{\sqrt{10\pi S_q}} e^{-\frac{(i-\frac{S_q}{2})^2+(j-\frac{S_q}{2})^2}{10S_q}}, \\ R_{\{2S_r,S_r\}}(i,j) &= \frac{1}{\sqrt{20\pi S_r}} e^{-\frac{(i-S_r)^2+j^2}{20S_r}} \end{aligned} \tag{10}$$

where S_q , $S_r \in \{30 + n * k | k = 10, n = 0, ..., 7\}$. We operate convolution to the proposal with generated kernels respectively and get the result by counting the peak of convoluted maps. The pseudo code description of the method is given as Algorithm 1.

ALGORITHM 1: Proposal counting algorithm

Input: proposal P_n, quadrate kernels Q, rectangular kernels R, proposal width w_p, proposal height h_p
Output: counting numbers Num_n

ξ = 1;
for kernel T_k in $\{Q, R\}$ **do**
  $E_q = Conv(P_n, T_k)$;
  for i = 1 to w_p **do**
    for j = 1 to h_p **do**
      if $E_q(i,j)$ is the local maximum in $E_q(i^*, j^*)$, where $i^* \in [i-10, i+10]$ and $j^* \in [j-10, j+10]$ **then**
        $\Theta_\xi = (i,j)$;
        $\xi++$;
      end
    end
  end
end
$Num_n = 0$;
for μ = 1 to ξ **do**
  for ν = 1 to ξ **do**
    $\eta = \|\theta_\mu - \theta_\nu\|_2$;
    if $\eta \geq 30$ **then**
      Num_n++;
    end
  end
end

Finally, to get a more smooth and reasonable outputs, the Kalman filter [Bianchi and Tinnirello 2003] is applied to the final counting result.

3.3 Partial proposal training method

To train and apply counting method in the embedded devices of the compartments, we focus on two aspects: 1. reduce the memory cost and 2. reduce the training time. We proposed a scene-aware partial input method to train in a limited image scale and provide high-quality data for training. For memory reduction, we keep the inputs in a small size cropping from the raw images and design a special scene-aware partial method for cropping. For more time reduction, we intend to decrease the number of inputs that participate in the training procedure and reduce the iteration of training, so we select a unique annotation compare to the usual counting tasks. Details are given as follows.

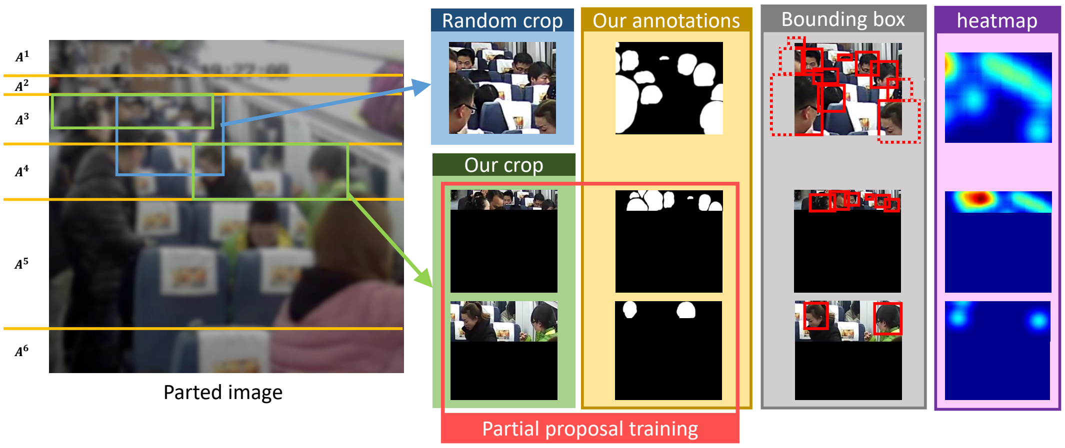


Fig. 4. In our partial proposal training method, according the prior knowledge of scene, the images are parted into 6 sub-areas. Rather than random crop in the overall images, the partial inputs are cropped in the sub-areas. Then, to better represent the targets, a pixel-level annotations are given for network training.

3.3.1 *Scene-aware partition method.* Due to the limited memory, the images must be cropped into the same small size for inputs. As the specialty of compartment scene, regular crop method always produces broken head images, which conduct the network hard to localize. Because of the diversity of passengers heads images which contain faces, hairs, necks, headwears, etc., the incomplete image data further increase the difficulty of network learning. Besides, the random crop in the images bring more uncertainty of the scale distribution.

To avoid getting too many incomplete samples in training data, in our work, we propose the scene-aware partial input method. As shown in Fig. 4, our method offers more complete head images after cropping. The details of method is given as follows.

We determine the partition according to the seats in the compartment based on images without passengers, e.g., Fig.1. According to [Gentle 2007], the connection of the real world distance and the pixels of the image is shown as follows:

$$y = f_y \frac{Y}{Z} + C_y \tag{11}$$

where y and Y denote the object height in images and the real world, Z is the distance between object and camera, f_y and c_y are the coefficients for mapping the coordinate system from the real world to images. Eq. 11 indicates that the height of an object in images is inversely proportional to the distance between object and camera in the real world. Considering that every row of seats in the compartment has the same distance between the front and back, meanwhile, every seat has the same height Y in the real world coordinate. Hence, the position of a ϵ-th row of seat $y(\epsilon)$ can be rewritten as follows:

$$y(k) = \alpha/\epsilon + \beta \tag{12}$$

where α and β are coefficients for mapping. $\epsilon = 1, 2, ..., 10$

These coefficients, as the basis of partition images, would be estimated by following three steps:

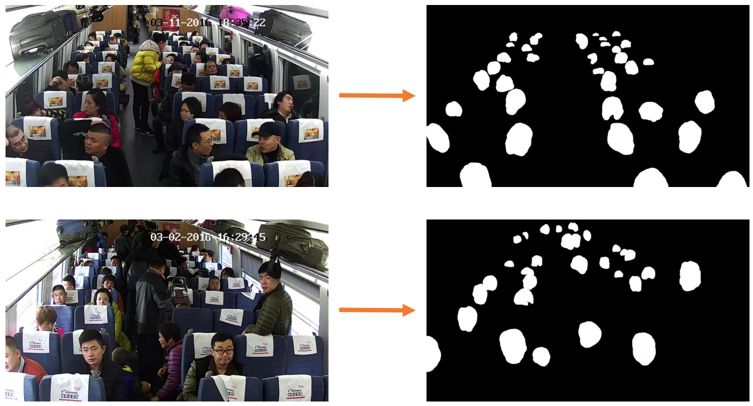


Fig. 5. Examples for image annotation. Examples in the left column are raw images and in the right are correspondent annotations.

(a) We used Hough transform [Mukhopadhyay and Chaudhuri 2015] to extract horizontal lines via the edge image obtained by Sobel edge detector [Kanopoulos et al. 1988] in an image without passengers. We choose the lines by constraining the absolute value of slope less than 0.1. K-medoids [Jin and Han 2010] is adopted to find 10 clustering centers in lines intercepts from Hough transfer. The 10 clustering centers would be used to fitting a $\hat{y}$ and estimate the coefficients $\hat{\alpha}$ and $\hat{\beta}$, which is inversely proportional to ϵ. (b) We take $\hat{y}(\epsilon)$ $\epsilon = 1, 2, ..., 10$ as the position of 10 rows. Then, we classify every two rows into a sub-area and generate five sub-areas. The farthest part to the camera (without seats) would be divided into an independent sub-area. Thus, we partition each images into 6 horizontal sub-areas: $A^1, A^2, , A^6$ as shown in Fig. 4. (c) The input data are cropped according to the partitions edges. As shown in Fig. 4, by cropping in sub-areas, we avoid to generate too many incomplete head images for training.

3.3.2 *Proposal annotation.* As shown in Fig. 4 and 5, Rather than common counting problem, the intense variation of scale in passenger counting make the heat map hard to represent targets. The bounding box annotation in detection is also impracticable due to the universal phenomenon of incomplete head images and occlusion in images. As shown in Fig.4, the bounding boxes are dense and have too many overlaps in annotations.

In this case, we apply the mask annotation to give 0 or 1 labels for every pixel. Through this measure, even a small part of an incomplete target can be noticed and provided for the network to learn. Furthermore, attributing to the mask annotation, we convert the common regression based counting thought (regression for the heatmaps, or bounding box position values) to a classification problem (whether the pixel belongs to a head or not). That reduces the difficulty for network learning and also avoid the considerable workload in annotation, which leads our methods more general and economical to embedded devices in compartments.

4. EXPERIMENT

In this section, we evaluate our method for passengers counting on the compartment surveillance video. We compare the proposed passenger counting approach with the detection based approaches, such as Faster RCNN [Ren et al. 2015], YOLO [Redmon et al. 2016], and SSD [Liu et al. 2016]. We also test the common crowd counting method like MCNN [Zhang et al. 2016b] to prove the impracticable of this kind of methods. To train the detectors with passengers data, we use ground truth proposal to generate a bounding box. Utilizing our counting method, we get the peak points as the center of bounding boxes and take the kernel size as box size. Then, we train the data together with VOC2007-2012 [Everingham et al. 2010] to avoid overfitting. For counting methods, the heatmaps are annotated according methods in [Zhang et al. 2015; Zhang et al. 2016b]. In the setting of our experiment, with the partial training method, the inputs to our network are 300*300. The network train with SGD optimizer in the learning rate of 0.01 and the decay step is 0.1 for every ten epoch. We finally train 75 epochs in our experiments.

4.1 Dataset

Table I. Ratio of Different Situation in Compartment Video

situations	DJ	NP	DB	BE	DE
ratio	0.616	0.21	0.062	0.074	0.038

The data we have contained 500 video sequences, each video is 8-10 minutes, and frame size is 1280*720*3. The video data can be repartitioned into five situations, which are: during journey (DJ), during boarding (DB), during exiting (DE), boarding and exiting simultaneously (BE) and no passengers (NP). The examples of situations are shown as Fig. 1. The ratio of every situation in compartment video data shown as TABLE I.

In the training stage, we choose 120 videos as the training set. We randomly capture two adjacent frames from each video to avoid repetition (one as the training input, the other for calculating ΔF_n). We annotate the 120 frames with the pixel-wise annotation as mentioned last section and shown in Fig. 4 and 5. In the test stage, we take 1000 frames from the other 380 videos. All training and test images are annotated with counting numbers.

4.2 Metric

We used the mean squared error (MSE) and the absolute error (MAE) as in [Zhang et al. 2015; Zhang et al. 2016b; Sam et al. 2017], which are defined as follows:

$$\begin{aligned} MSE &= \sqrt{\frac{1}{N}\sum_{i}^{N}(x_i - \hat{x_i})^2} \\ MAE &= \frac{1}{N}\sum_{i}^{N}\|x_i - \hat{x_i}\| \end{aligned} \tag{13}$$

where N is the number of test frames, x_i is the actual number of passengers, and $\hat{x_i}$ is the counting result by the algorithm. Generally, MAE and MSE manifest the accuracy and stability of estimation correspondingly.

4.3 Objective Performances

Table II. MSE and MAE for various situations in different counting method.

situations	MSE				MAE			
	Faster-RCNN	YOLO	SSD	Ours	Faster-RCNN	YOLO	SSD	Ours
DJ	15.16	18.78	9.24	**3.71**	14.85	18.60	8.65	**2.90**
DB	5.65	7.13	5.72	**3.42**	5.18	5.86	5.18	**2.74**
DE	7.68	13.33	4.60	**3.03**	6.77	11.56	3.91	**2.49**
BE	15.79	16.56	11.18	**4.40**	15.54	14.14	10.56	**3.68**
NP	1.73	0.89	0.93	**0.73**	1.48	0.52	0.48	**0.42**
average	12.24	15.15	8/04	**3.54**	11.86	14.47	7.46	**2.80**

Comparisons of different methods under different testing situations are shown in TABLE II, respectively. From TABLE II, all methods have low MSE and MAE values in Np, since the numbers in this situation are usually less than 5 and occlusion is not severe. The detectors can distinguish people in the scene. But the actions of people may cause deformation and occlusion, which may disturb the detector, our method is more robust to these disturbances and obtain better performances.

Meanwhile, we find that the DJ situation is the hardest but is the major of the dataset. Due to the severe occlusion for passengers behind the seats, detection based methods are hard to find convincible targets. In YOLO, because the YOLO is not good at small objects detection, in this case, the method performs worse than in other situations. Comparing with other methods, our method still achieves good performance in accuracy (MSE) and stability (MAE). Additionally, the DJ takes the majority of the dataset, so the results of the DJ are significantly dominant in average values.

Besides, the improvements of our method are not all from DJ situations. We also overperform in other situations in all the metrics. In BE, DE and BE, the large-scale movements of passengers induce more severe occlusion and scale changing, which are hard for other methods. In our method, the motion information and historical information are all considered, which lead to better representations in proposals. The results show that our method is more efficient in situations with massive movements.

In the view of generality, all the MSE values in our method are about 4.0, and all the MAE are about 3.0. While, the other methods show a large gap between all the situations, and have higher volatile. In the application of reality, the values of metrics should be stable, and methods should be useful in all the possibilities. Thus, our method has a significant advantage in practical application as well.

Moreover, we also test ordinary crowd counting method like MCNN. In experiments, MAE = 72.7447 and MSE = 73.4575 in NP, MAE = 45.7848 and MSE = 47.4236 in BE, MAE = 41.7374 and MSE = 42.2374 in DJ, which are much worse than detection based methods.

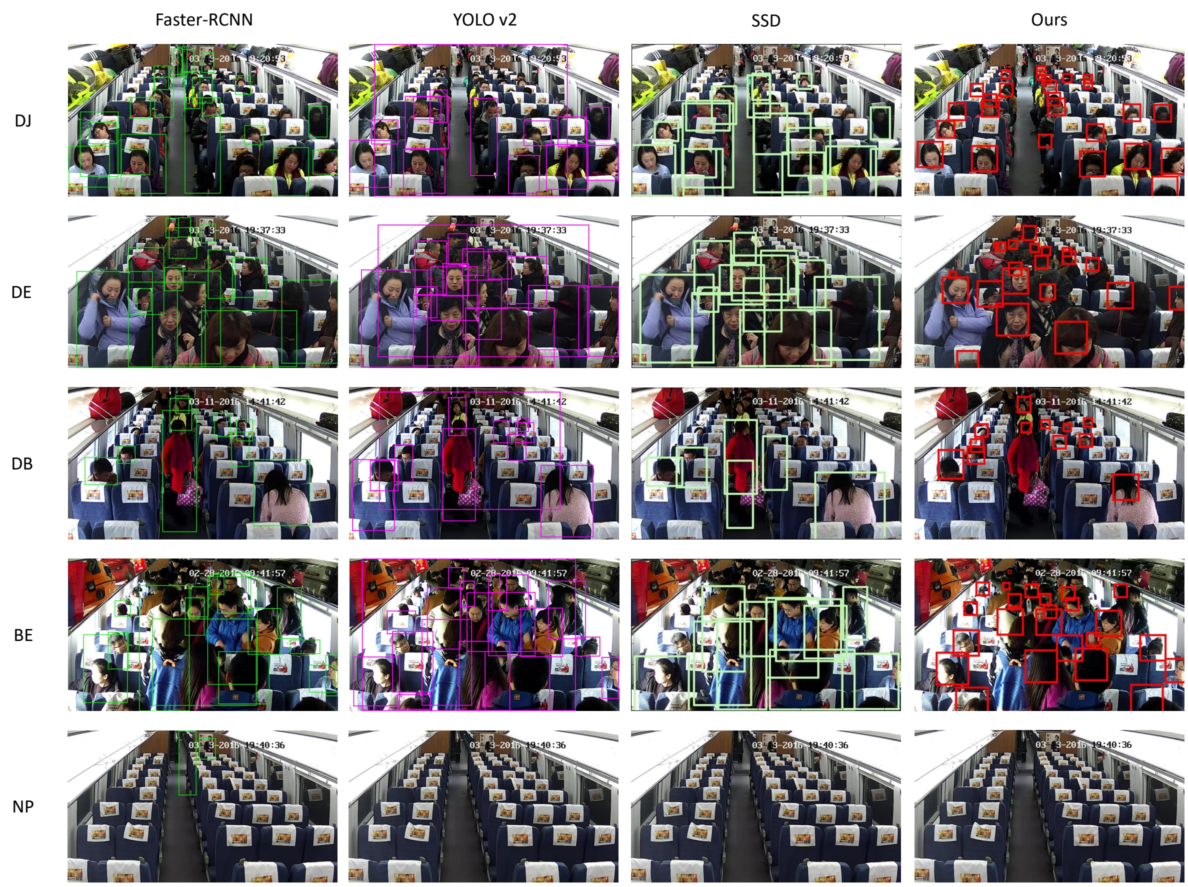


Fig. 6. Results of detection based methods and our methods. Our method is more robust for scale changing in every situations and more accurate in counting task.

4.4 Subjective Performances

The subjective results of Faster-RCNN, YOLO, SSD, and ours for the DJ, DB, DE, and BE situations are shown in Fig. 6. To visualize our proposal for comparison, we take the peaks position in counting module as the centers of boxes and the kernel scales as the box sizes.

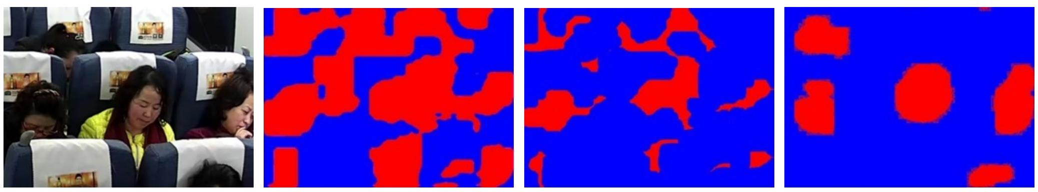
In experiments, YOLO is not good at the small object, so the method always tends to give a large bounding box output, as shown in the second column of Fig. 6. Sometimes, even the counting number is correct, the passengers' locations are wrong, and the bounding boxes contain multiple people at the same time.

Moreover, due to the region proposal module, Faster-RCNN is sensitive to potential head images, which cause the method more likely to give false detection, as shown in the second column of Fig. 6. Some areas may similar to the occluded head and recognized by Faster-RCNN as targets. That makes the counting numbers via Faster-RCNN larger, which also cause errors in the NP situation.

SSD also has more detection result and counting value in the same frame, but SSD is better than Faster-RCNN in passengers counting in most situations. In some cases which contain severe occlusion disturbance and scale problems like in BE situation in the fourth raw of Fig. 6, SSD method also has many duplicate results and does not locate the correct head position.

Our method is more acccurate in counting passengers for given surveillance video. The proposed method can count heads in the various scale against scale variation, and also shows better performance than other methods.

4.5 Discussions in Motion Supervised Multi-scale Network



(a) (b) (c) (d)

Fig. 7. Proposal generated from different features.(a) is the original frame. (b) is proposal applied Scale1 features. (c) is proposal applied Scale1 and Scale2 features. (d) is proposal applied features in all the scales. Only using features in all scales can generate the best proposal for counting.

4.5.1 *Multi-scale network comparison.* We fuse the features from $Scale1$, $Scale2$, and $Scale3$ for generating a proposal. In the combinations of features in different scales, the proposals generated from low-level layers, such as $Scale1$, $Scale2$, and their combinations are not practicable for counting. Since $Scale3$ contain a more semantic feature which is useful to identify human heads. The proposals generated from three scale levels features are best among all the combinations. Moreover, as shown in Fig. 7, with the addition of different scales, the proposal tend to be refined gradually. The proposals generated by $Scale1$ is raw and large. With the $Scale2$ added, the proposal become smaller and finer. Finally, with all scales utilized, the proposal is accurate and better to present targets.

To choose the best combination of feature maps, we test the counting performance in the situation of DJ with different scale features. As shown in Fig. 8, we compare the performances the average MSE and MAE values of our method using various features extracted from CNN. In Fig., $S1$, $S2$, and $S3$ means using only use $Scale1$ feature, $Scale2$ feature and $Scale3$ feature separately for proposal generation. $S1+2$, $S2+3$, $S1+3$, and All means using the combination of $Scale1$ and $Scale2$ features, $Scale2$ and $Scale3$ features, $Scale1$ and $Scale3$ features, and all the three scale features respectively.

$Scale1$ and $Scale2$ features contain some fine-texture information of head images, which are useful in small head detection. $Scale3$ has more semantic information, which is essential in detecting head and generate a proposal. In our experiment, the $Scale2$ features are useful when combining with semantic information in $Scale3$, but its hard to

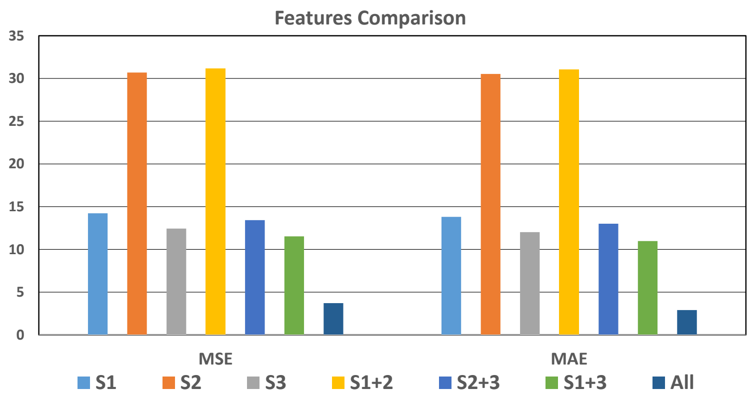
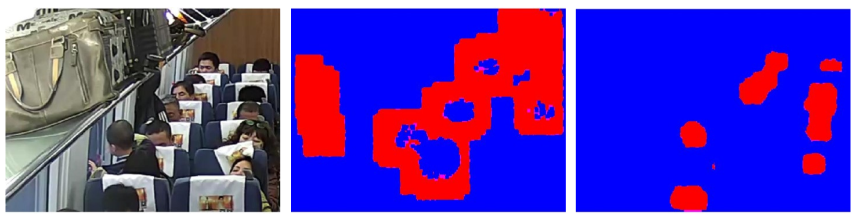


Fig. 8. Different scale features comparison. S1, S2 and S3 means using only use Scale1 feature, Scale2 feature and Scale3 feature separately for proposal.

produce a correct proposal only with $Scale2$ features individually or the combination of $Scale1$ and $Scale2$. When we use $Scale1$ features for proposal, with the detailed texture information, the proposals tend to be coarse and large, which could be captured by kernels in the counting method and produce counting results primarily. While, when with $Scale2$ features, the proposals tend to be finer and smaller, which is hard for counting. As shown in Fig. 7(c), the finer proposal may bring more noise for counting method, but better in representing targets. In the other side, in the experiment, $Scale2$ features offer some sorts of complement to $Scale1$. The over-fitting always appears when we try to generate a proposal only with $S2$ or $S1 + S2$ features. Moreover, when combining all the feature of three scales, the proposals become stable and reliable. All these facts indicate that texture information in $Scale1$ and $Scale2$ and semantic information in $Scale3$ are all necessary for counting.



(a) (b) (c)

Fig. 9. Comparison for with or without motion supervision. (a) is the raw images. (b) and (c) are from the method training without and with motion supervision.

4.5.2 *Motion supervision.* As shown in Fig. 9, considering the occlusion and scale changing in images, proposals with motion supervised focus on the passengers head better. In contrary, without motion information, the network tends to misrecognize the objects as passengers and give much larger proposals.

Numerically, in MSE and MAE metric, the inaccurate proposals bring the values to more than 20 in DJ and BE situations, which is unusable in passengers counting.

4.6 Discussion in Spatially-temporally Enhanced Counting Method

To verify the efficiency of the proposed module, experiments of utilizing spatial and temporal enhancement are operated and discussed in this part.

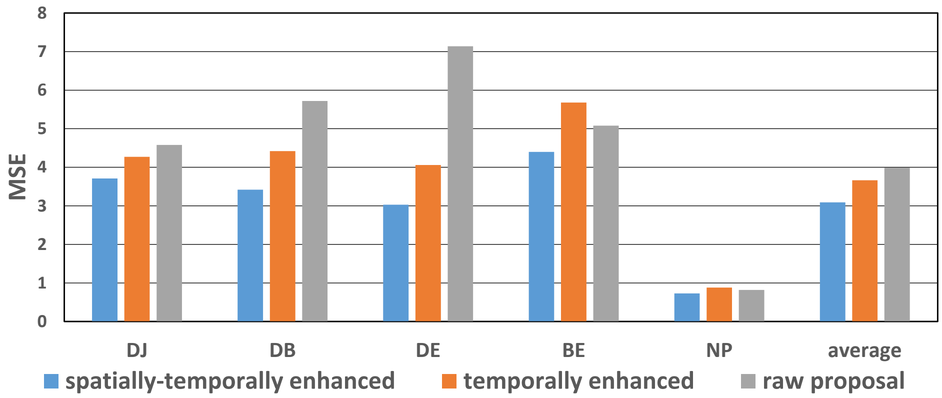


Fig. 10. MSE comparison of the passenger counting with spatial and temporal enhancements. MSE can reflect stability of the system. The enhancements effectively keep the proposals more stable and reliable for counting.

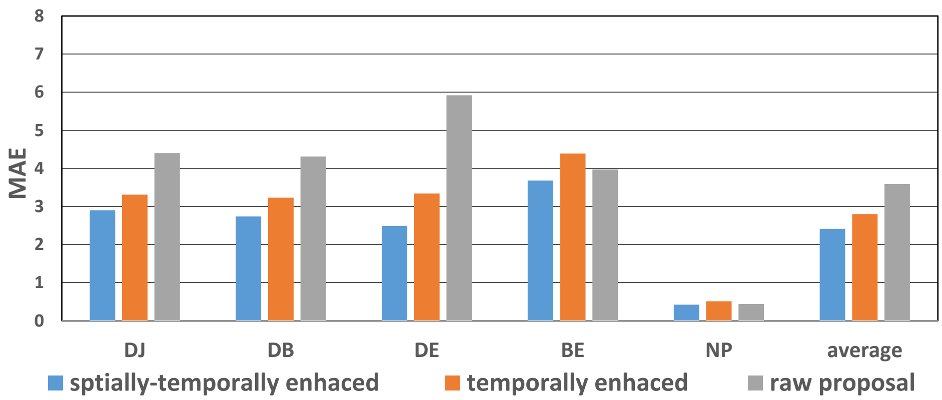


Fig. 11. MAE comparison of the passenger counting with spatial and temporal enhancement. MAE measures the absolute accuracy in counting. Though proposed enhanced methods, the proposals are better to represent targets and boost performances in counting.

As shown in Fig. 10 and 11, our motion supervised multi-scale network is already able to offer counting proposal. With the enhancement, the proposal can be better in all the situations. Since the more intense movement of passengers brings more severe scale changing, in the situations of DB, DE, and BE, the improvements become more apparent. If there is excessive movement in a single frame, the faulty proposals may connect to or adhere with others and form a broader range of mistakes. That makes the counting number plunges and produces larger MSE and MAE.

Further, the performance improved a lot when utilizing temporal information. The average MAE and MSE values decrease about 22.03% and 8.30% respectively compared to the primary proposal. Moreover, with the spatial information, the MAE and MSE values decrease about 32.81% and 22.63% respectively.

Moreover, in BE situation, the single temporal enhancement may conduct worse performance as shown in experiments. Because the BE situation has the most intense activity of passengers movement, the proposal from the last frame is far different from the current proposal. Addition of the historical knowledge would disturb the prediction of a new proposal in this case. However, benefiting from the spatial enhancement, the instantaneous information of passengers states are given. The influence of dramatic movements can be modified and reduced by spatial prior, which lead to a better performance in final proposals.

Table III. MSE and MAE for all the situations with or without Kalman filter.

situations	MSE		MAE	
	without	with	without	with
DJ	3.99	3.70	3.01	2.90
DB	3.87	3.42	2.98	2.74
DE	3.34	3.03	2.66	2.50
BE	4.97	4.40	4.08	3.68
NP	0.84	0.73	0.45	0.42
Average	3.88	3.54	2.96	2.81

We also provide some discussion of the application of Kalman filtering to prove that the main improvements are from proposed methods rather than some post-processing. The improvement from Kalman filtering The performance of adding Kalman filtering or not is shown in TABLE III. We find that both MSE and MAE improved by Kalman filtering slightly.

In sequential frames, the number fluctuation is ineluctable. Kalman filter alleviates waviness in counting number and modified output in a small computational complexity. Specifically, in BE, because the passengers concurrently in and out, the total amount of passengers wouldn't change dramatically. The counting number needs to be more stable and smooth. So, Kalman is useful in this situation and provides the largest improvement in decreasing 9.84% of MAE. The influence in alleviating waviness reflected more in MSE. The DB, BE, and NP situations have improvements for 11.60%, 11.56% and 12.62% in MSE value. However, all the improvements are in the single digits which are far smaller than the basic improvement from the methods.

4.7 Discussion in Partial Proposal Training

The proposed training method leads the tiny multi-scale network to train and evaluate in the embedded devices. In the specific value of the method, the overall network without the backbone is 6.6M, and enable to train in 680M of GPU memory with batch size 24. The total training epoch is 75, and the duration is about 10 hours. To be noticed, with the designed training method, the amount of annotations is only 120 images, which is far less general detection or counting datasets. All this fact make our methods practicable and feasible in limited equipment in compartment environment.

5. CONCOLUSION

In this paper, we present a complete pipeline for giving head proposals, counting numbers, and training in passengers counting problem. The proposed motion supervised representation provide solid and robust proposals against scale changing. The spatial and temporally enhanced counting offers accurate counting result from proposals. With the given partial proposal training method, the methods are able to operated and deal solve problem in actual scene of railway compartments. Advantages of our approach is particularly significant in the scene and our method achieve better performance than the other method in all situations in datasets.

REFERENCES

Giuseppe Bianchi and Ilenia Tinnirello. 2003. Kalman Filter Estimation of the Number of Competing Terminals in an IEEE 802.11 network. In Proceedings IEEE INFOCOM 2003, The 22nd Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, CA, USA, March 30 - April 3, 2003. 844–852. DOI:http://dx.doi.org/10.1109/INFCOM.2003.1208922

Lokesh Boominathan, Srinivas S. S. Kruthiventi, and R. Venkatesh Babu. 2016. CrowdNet: A Deep Convolutional Network for Dense Crowd Counting. In Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016. 640–644. DOI:http://dx.doi.org/10.1145/2964284.2967300

Antoni B. Chan and Nuno Vasconcelos. 2009. Bayesian Poisson regression for crowd counting. In IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009. 545–551. DOI:http://dx.doi.org/10.1109/ICCV.2009.5459191

Feng Dai, Hao Liu, Yike Ma, Juan Cao, Qiang Zhao, and Yongdong Zhang. 2019. Dense Scale Network for Crowd Counting. CoRR abs/1906.09707 (2019). http://arxiv.org/abs/1906.09707

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. 2010. The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision 88, 2 (01 Jun 2010), 303–338. DOI:http://dx.doi.org/10.1007/s11263-009-0275-4

Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (Eds.). 2018. Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VII. Lecture Notes in Computer Science, Vol. 11211. Springer. DOI:http://dx.doi.org/10.1007/978-3-030-01234-2

Junyu Gao, Qi Wang, and Xuelong Li. 2019. PCC Net: Perspective Crowd Counting via Spatial Convolutional Network. IEEE Transactions on Circuits and Systems for Video Technology (2019).

James E. Gentle. 2007. Matrix Algebra: Theory, Computations, and Applications in Statistics (1st ed.). Springer Publishing Company, Incorporated.

Ross B. Girshick. 2015. Fast R-CNN. In 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015. 1440–1448. DOI:http://dx.doi.org/10.1109/ICCV.2015.169

Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2013. Rich feature hierarchies for accurate object detection and semantic segmentation. CoRR abs/1311.2524 (2013). http://arxiv.org/abs/1311.2524

Jason M. Grant and Patrick J. Flynn. 2017. Crowd Scene Understanding from Video: A Survey. ACM Trans. Multimedia Comput. Commun. Appl. 13, 2, Article 19 (March 2017), 23 pages. DOI:http://dx.doi.org/10.1145/3052930

Alfred Haar. 1933. Der Massbegriff in der Theorie der Kontinuierlichen Gruppen. The Annals of Mathematics 34, 1 (Jan. 1933), 147. DOI:http://dx.doi.org/10.2307/1968346

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016), 770–778.

Xiping Hu, Junqi Deng, Jidi Zhao, Wenyan Hu, Edith C.-H. Ngai, Renfei Wang, Johnny Shen, Min Liang, Xitong Li, Victor C. M. Leung, and Yu-Kwong Kwok. 2015. SAfeDJ: A Crowd-Cloud Codesign Approach to Situation-Aware Music Delivery for Drivers. ACM Trans. Multimedia Comput. Commun. Appl. 12, 1s, Article 21 (Oct. 2015), 24 pages. DOI:http://dx.doi.org/10.1145/2808201

Siyu Huang, Xi Li, Zhongfei Zhang, Fei Wu, Shenghua Gao, Rongrong Ji, and Junwei Han. 2018. Body Structure Aware Deep Crowd Counting. IEEE Transactions on Image Processing 27 (2018), 1049–1059.

Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. 2013. Multi-source Multi-scale Counting in Extremely Dense Crowd Images. In 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013. 2547–2554. DOI:http://dx.doi.org/10.1109/CVPR.2013.329

Xin Jin and Jiawei Han. 2010. K-Medoids Clustering. Springer US, Boston, MA, 564–565. DOI:http://dx.doi.org/10.1007/978-0-387-30164-8_426

Di Kang and Antoni B. Chan. 2018. Crowd Counting by Adaptively Fusing Predictions from an Image Pyramid. In British Machine Vision Conference 2018, BMVC 2018, Northumbria University, Newcastle, UK, September 3-6, 2018. 89. http://bmvc2018.org/contents/papers/0283.pdf

Di Kang, Zheng Ma, and Antoni B. Chan. 2017. Beyond Counting: Comparisons of Density Maps for Crowd Analysis Tasks - Counting, Detection, and Tracking. CoRR abs/1705.10118 (2017). http://arxiv.org/abs/1705.10118

Nick Kanopoulos, Nagesh Vasanthavada, and Robert L Baker. 1988. Design of an image edge detection filter using the Sobel operator. IEEE Journal of solid-state circuits 23, 2 (1988), 358–367.

Pavel Korshunov and Wei Tsang Ooi. 2011. Video Quality for Face Detection, Recognition, and Tracking. ACM Trans. Multimedia Comput. Commun. Appl. 7, 3, Article 14 (Sept. 2011), 21 pages. DOI:http://dx.doi.org/10.1145/2000486.2000488

Shohei Kumagai, Kazuhiro Hotta, and Takio Kurita. 2017. Mixture of Counting CNNs: Adaptive Integration of CNNs Specialized to Specific Appearance for Crowd Counting. CoRR abs/1703.09393 (2017). http://arxiv.org/abs/1703.09393

Issam H. Laradji, Negar Rostamzadeh, Pedro O. Pinheiro, David Vázquez, and Mark W. Schmidt. 2018. Where Are the Blobs: Counting by Localization with Point Supervision. In Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part II. 560–576. DOI:http://dx.doi.org/10.1007/978-3-030-01216-8_34

Xuelong Li, Mulin Chen, and Qi Wang. 2016. Measuring Collectiveness via Refined Topological Similarity. ACM Trans. Multimedia Comput. Commun. Appl. 12, 2, Article 34 (March 2016), 22 pages. DOI:http://dx.doi.org/10.1145/2854000

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. 2016. SSD: Single Shot MultiBox Detector. In Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I. 21–37. DOI:http://dx.doi.org/10.1007/978-3-319-46448-0_2

Xialei Liu, Joost van de Weijer, and Andrew D. Bagdanov. 2019. Exploiting Unlabeled Data in CNNs by Self-Supervised Learning to Rank. IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (2019), 1862–1878.

Mark Marsden, Kevin McGuinness, Suzanne Little, and Noel E. O'Connor. 2017. Fully Convolutional Crowd Counting on Highly Congested Scenes. In Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017) - Volume 5: VISAPP, Porto, Portugal, February 27 - March 1, 2017. 27–33. DOI:http://dx.doi.org/10.5220/0006097300270033

Priyanka Mukhopadhyay and Bidyut B. Chaudhuri. 2015. A survey of Hough Transform. Pattern Recognition 48, 3 (2015), 993–1010. DOI:http://dx.doi.org/10.1016/j.patcog.2014.08.027

Viet-Quoc Pham, Tatsuo Kozakaya, Osamu Yamaguchi, and Ryuzo Okada. 2015. COUNT Forest: CO-Voting Uncertain Number of Targets Using Random Forest for Crowd Density Estimation. In 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015. 3253–3261. DOI:http://dx.doi.org/10.1109/ICCV.2015.372

Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. 2016. You Only Look Once: Unified, Real-Time Object Detection. In 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. 779–788. DOI:http://dx.doi.org/10.1109/CVPR.2016.91

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. 91–99. http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks

Deepak Babu Sam, Skand Vishwanath Peri, Mukuntha N. S., Amogh Kamath, and R. Venkatesh Babu. 2019. Locate, Size and Count: Accurately Resolving People in Dense Crowds via Detection. CoRR abs/1906.07538 (2019). http://arxiv.org/abs/1906.07538

Deepak Babu Sam, Shiv Surya, and R. Venkatesh Babu. 2017. Switching Convolutional Neural Network for Crowd Counting. In 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. 4031–4039. DOI:http://dx.doi.org/10.1109/CVPR.2017.429

Claude Sammut and Geoffrey I. Webb (Eds.). 2017. Encyclopedia of Machine Learning and Data Mining. Springer. DOI:http://dx.doi.org/10.1007/978-1-4899-7687-1

Chong Shang, Haizhou Ai, and Bo Bai. 2016. End-to-end crowd counting via joint learning local and global count. In 2016 IEEE International Conference on Image Processing, ICIP 2016, Phoenix, AZ, USA, September 25-28, 2016. 1215–1219. DOI:http://dx.doi.org/10.1109/ICIP.2016.7532551

Biyun Sheng, Chunhua Shen, Guosheng Lin, Jun Li, Wankou Yang, and Changyin Sun. 2018. Crowd Counting via Weighted VLAD on a Dense Attribute Feature Map. IEEE Transactions on Circuits and Systems for Video Technology 28 (2018), 1788–1797.

Vishwanath A. Sindagi and Vishal M. Patel. 2017. CNN-Based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In 14th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2017, Lecce, Italy, August 29 - September 1, 2017. 1–6. DOI:http://dx.doi.org/10.1109/AVSS.2017.8078491

Jasper R. R. Uijlings, Koen E. A. van de Sande, Theo Gevers, and Arnold W. M. Smeulders. 2013. Selective Search for Object Recognition. International Journal of Computer Vision 104, 2 (2013), 154–171. DOI:http://dx.doi.org/10.1007/s11263-013-0620-5

Ze Wang, Zehao Xiao, Kai Xie, Qiang Qiu, Xiantong Zhen, and Xianbin Cao. 2018. In Defense of Single-column Networks for Crowd Counting. CoRR abs/1808.06133 (2018). http://arxiv.org/abs/1808.06133

Xiaoshan Yang, Tianzhu Zhang, and Changsheng Xu. 2016. Semantic Feature Mining for Video Event Understanding. ACM Trans. Multimedia Comput. Commun. Appl. 12, 4, Article 55 (Aug. 2016), 22 pages. DOI:http://dx.doi.org/10.1145/2962719

Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. 2015. Cross-scene crowd counting via deep convolutional neural networks. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. 833–841. DOI:http://dx.doi.org/10.1109/CVPR.2015.7298684

Hong-Bo Zhang, Bineng Zhong, Qing Lei, Ji-Xiang Du, Jialin Peng, Duansheng Chen, and Xiao Ke. 2017. Sparse Representation-Based Semi-Supervised Regression for People Counting. ACM Trans. Multimedia Comput. Commun. Appl. 13, 4, Article 47 (Aug. 2017), 17 pages. DOI:http://dx.doi.org/10.1145/3106156

Lu Zhang, Miaojing Shi, and Qiaobo Chen. 2018. Crowd Counting via Scale-Adaptive Convolutional Neural Network. 2018 IEEE Winter Conference on Applications of Computer Vision (WACV) (2018), 1113–1121.

Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. 2016a. Single-Image Crowd Counting via Multi-Column Convolutional Neural Network. In 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. 589–597. DOI:http://dx.doi.org/10.1109/CVPR.2016.70

Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. 2016b. Single-Image Crowd Counting via Multi-Column Convolutional Neural Network. In 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. 589–597. DOI:http://dx.doi.org/10.1109/CVPR.2016.70

Huicheng Zheng, Zijian Lin, Jiepeng Cen, Zeyu Wu, and Yadan Zhao. 2018. Cross-Line Pedestrian Counting Based on Spatially-Consistent Two-Stage Local Crowd Density Estimation and Accumulation. IEEE Transactions on Circuits and Systems for Video Technology 29 (2018), 787–799.

Zhikang Zou, Xinxing Su, Xiaoye Qu, and Pan Zhou. 2018. DA-Net: Learning the Fine-Grained Density Distribution With Deformation Aggregation Network. IEEE Access 6 (2018), 60745–60756. DOI:http://dx.doi.org/10.1109/ACCESS.2018.2875495