Recent Trends and Study on Perspective Crowd Counting in Smart Environments

Vasupalli Jaswanth, Arun Reddy Yeduguru, Vura Seetha Manoj, K. Deepak, and S. Chandrakala

1 Introduction

Crowd counting has become an innovative idea in smart environments. It has gained serious attention in recent years as there is rapid development in private and public places due to the increase in global population. With an increase in population, there is a large amount of hustle and bustle almost everywhere which can lead to massive chaos that can be life threatening and hence should always be prevented in critical environments. Single image-based crowd counting is still gaining attention and is one of the difficult topics due to the complex distribution of people, non-uniform illumination, low image resolution, and dense crowds that have excessive overlaps and occlusions within each other. Moreover, perspective effects can cause a huge contrast in human appearance. For example, in the regions of people close to the camera, the people heads are big and their respective density values are accordingly low, and in the regions of people farther from a camera, the heads are small and the density values are high.

In recent times, the crowd counting problem has been addressed by a huge number of methods such as SFANet [1] and SegNet [1], NAS [2], compact [3] convolutional neural network, and HYGNN [4]. The prevalent crowd counting methods can be broadly categorized into: Detection then counting, direct count regression, CNN-based methods, perspective-based methods. Detection then counting-based methods involve more computations, and these types of methods are only suitable for fewer crowd densities and fail if the density of crowd is high. Direct count regression methods reduce the computations, and it produces more accurate results compared to detection then counting methods but they are not efficient when there are excessive overlaps in an image. CNN-based methods pay attention to multi-scale and multi-column architecture that integrates features in

V. Jaswanth · A. R. Yeduguru · V. S. Manoj · K. Deepak · S. Chandrakala (✉)
School of Computing, SASTRA University, Thanjavur, Tamil Nadu, India

R. R. Raje et al. (eds.), *Artificial Intelligence and Technologies*,
Lecture Notes in Electrical Engineering 806,
https://doi.org/10.1007/978-981-16-6448-9_7

variable sizes of the respective fields and can detect the people in the case of excessive overlaps. But, CNN-based methods have not focused on perspective changes. Perspective-based methods focus on the continuous scale variations [5] of every single person and perspective information played a major role in the prediction. So far, perspective and CNN-based methods have achieved higher performance than other methods in terms of accuracy and robustness.

2 Categories of Approaches Perspective-Based Crowd Counting

As shown in Fig. 1, a generic crowd counting model learns the spatial and perspective features from input images.

2.1 Detection Then Counting

In most of the early approaches [6, 7], crowd counting estimation is done by first detecting and segmenting individual objects in the scene then followed by counting. Some of the challenges faced by these kinds of methods are, they are computationally expensive as they produce more accurate results than the overall count and are mostly suitable for scenes in which the crowd density is low, and they do not perform well on scenes having high crowd density. Another challenge is that a large amount of work is needed in scenes having high crowd density, which includes bounding box or instance mask to train the object detectors.

2.2 Regression-Based Methods

In this kind of method, the detection problem which is faced by detection-based methods is avoided, and image features are used to estimate crowd counts. Earlier methods [8, 9] gave poor performance because count prediction is done based on the information from the features, and the spatial awareness is completely ignored. In the later methods [10, 11], crowd count is obtained by first generating the density

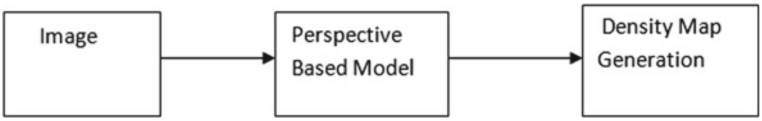

Fig. 1 Overview of perspective-based crowd counting

map and then combining all the information (pixel values) over the density map. Though spatial information is provided up to an extent by learning the density map, these methods lack in maintaining the high-frequency dissimilarities in the density map.

2.3 Graphical Methods

The concept of graphical neural network (GNN) was first introduced by Scarselli et al. in 2008 [12]. It extended recursive neural networks for processing graphical structure data. In 2016, Li et al. [13] proposed gated recurrent units to improve the representation capacity of GNN. To generalize the GNN, Gilmer et al. [14] in 2017 used message passing neural networks. The essential idea of GNN is to enhance the node representations by propagating information between nodes. Recently, GNN has been successfully applied in various applications like human-object interactions, attribute recognition, and crowd count estimation.

2.4 CNN-Based Methods

In recent years, there has been a rapid growth in CNN-based methods. Multi-scale, multi-task, and other techniques are usually carried out with the help of CNN-based approaches. Recently, there is an increase in the methods which incorporate handling of scale variation issues. Some of these include MCNN, which is a multi-column architecture proposed by Zhang et al. To obtain features with various scales, this architecture makes use of different filters on separate columns [15]. SANet is a novel encoder-decoder network proposed by Xinkhun et al. where multi-scale features are extracted by the encoder by using scale aggregation modules, and the high-resolution density maps are generated by decoder [16]. Also, there exist studies which focus on perspective maps [17] and region of interest (ROI) [18] to enhance the robustness and accuracy of the model.

3 Review on Recent Methods of PCC

3.1 Perspective Crowd Counting (PCC Net)

The entire PCC Net consists of three modules, which include density map estimation (DME), random high-level density classification (R-HDC), and fore-/background segmentation (FBS) [5]. Along with it, the Down Up Left Right (DULR) module is also present which takes the input from the full convolution network (FCN) and passes the features maps with encoded perspective changes to the DME and FBS module. Density map estimation (DME) module helps to generate density maps for crowd images. This module uses the FCN which can accept input of any dimension. In addition to that, upsampling is also done on the feature maps using the deconvolution layer present inside the full convolution network.

Random High-Level Density Classification (R-HDC). To learn global contextual information, the R-HDC module is used. In this, the entire density is divided into ten types of high-level labels. To perform this, an entire image is broken into many patches to cover the entire image. Then for each part, a random region of interest (ROI) is generated. This ROI generated must be as large as to cover more than 1/16 part of the image [5]. Then the pooling layer generates the feature maps for the ROI. Then the FCN layer classifies the feature maps as one among the ten high-level labels. In simple terms, the R-HDC model estimates the density of the image and then divides it into ten patches with labels.

Fore-/Background Segmentation (FBS). DME + R-HDC module neglects the contextual information in congested crowd scenes. To consider this, FBS is used. In this module, they generate a head segmentation map. This map helps us to cover the face region, its structure, and the semantic features. The last feature map in the FBS module is added with the last feature map of the DME module to obtain density map estimation [5].

Down Up Left Right (DULR) Module. By this point, the model can learn contextual features, global features, and local features. To translate perspective changes from four directions, this DULR module is used. The DULR module consists of four convolution layers, each handling four directions, namely top, down, left to right, and right to left, respectively. In each layer, the entire feature map is divided into h parts where h represents the height of the feature map [5]. Then each part is fed into the respective layer, and the output is then concatenated with the next part. For every convolution layer, this process is repeated iteratively for h parts. The output feature map of each layer is passed as the input feature map to the next layer. Note that the input feature map of the DULR module is of same shape as that of the output feature map of the DULR module.

3.2 Spatial Divide-and-Conquer (S-DC) Net

From Quantity to Interval. Rather than using regression to count the values in an open set, the local counts and classified count intervals are discretized. The interval partition of $[0, +\infty)$ is discretized as $\{0\}$, $(0, C1]$, $(C2, C3]$, …, $(\mathrm{CM} - 1, \mathrm{CM}]$ and $(\mathrm{CM}, +\infty)$. Here, $M + 1$ sub-intervals are present. The count value in $(C2, C3]$ is labeled as first class. This should not exceed the maximum local count present in the training set, which is obvious. The mid-value of every sub-interval is calculated dynamically when counting each interval; CM will be the last count value as the last sub-interval is (CM, + ∞]. But this leads to error, and this error is reduced by S-DC Net.

S-DC Net consists of VGG16 [19] feature encoder, Unet [20] decoder, a count interval classifier, and a division decider [21]. In the classifier, the first average pooling layer will have a stride size 2 and the final prediction will be of stride size 64. The fully connected layers are removed by the feature encoder. Assume the input size of the as 64 × 64. The feature map $F0$ is obtained from the convolution layer 5. From, 1/32nd resolution of the input image and extracted feature map $F0$, the classifier predicts the class label of count interval CLS0. The local count $C0$ can be obtained from CLS0.

In the first stage of execution, the shared classifier gets the input from the fused feature map F1. The division count $C1$ is obtained from the shared classifier. Precisely, $F0$ is upsampled by ×2 and attached to $F1$. The classifier extracts local features that related to spatially partitioned sub-regions. $C1$ is obtained from $F1$ and the classifier. Every 2 × 2 elements in $C1$ indicate the sub-count of the proportionate 32 × 32 sub-region. The division decider is used to divide among the obtained local counts $C0$ and $C1$. In the first stage of S-DC, the division decider produces a soft division mask $W1$ of similar size as $C1$ on F1 like for any w $W1$, w [0, 1]. No division is necessary at that position when w equals to zero. The division count $C1$ should be substituted in place of initial prediction when w equals to one. As $W1$ and $C1$ are double the count of $C0$, $C0$ is upsampled by ×2. Initial stage division count is calculated as,

$$\mathrm{DIV1} = (1 - W1) \text{ o avg } (C0) + W1 \text{ o } C1 \tag{1}$$

Here, 1 represents the matrix packed with ones and has the same size as $W1$. “o” represents Hadamark product, and avg represents averaging redistribution operator. S-DC Net can also be implemented by dividing the feature map till the first convolution block output is obtained.

3.3 Spatial/Channel-Wise Attention Regression Networks (SCAR)

Overview. The SAM and CAM attention models are the two important modules of the SCAR [20] network. First, the image is fed into a local feature extractor [22], which consists of the VGG-16 as backbone (first ten convolutional layers) followed by the dilation module. Even though this output contains some spatial contextual information, it is not large enough, and also it does not encode attention features. To get rid of these drawbacks, two stream architectures (SAM and CAM) are designed to translate spatial attention features as well as channel-wise attention features. At last, the predicted density map is obtained by concatenating the two types of features maps (one from SAM and other from CAM) via convolution operation.

Spatial-Wise Attention Model (SAM). For global images, it can be observed that there is a certain uniformity in the density distribution locally and globally because of perspectives changes of crowded scenes. Also, there is a consistent gradual trend of density change. To encode these two observations, SAM is designed. SAM considers a large range of contextual information and identifies the density distribution change. The output from the VGG-16 backbone layer which is of size $C \times H \times W$ is fed into three different convolutional layers of kernel size 1×1 [22]. Then by applying reshape or transformations, three features maps $S1$, $S2$, and $S3$ are attained. The spatial attention map Sa [22] of size HW × HW is generated by performing matrix multiplication of $S1$ and $S2$ followed by applying softmax operation. The obtained Sa then undergoes matrix multiplication with $S3$, and then output is reshaped to $C \times H \times W$. Then the output is scaled by a learnable factor and undergoes sum operation with F (output from VGG-16 backbone) to give the final output of SAM.

Channel-Wise Attention Model (CAM). Channel-wise attention model (CAM) is similar in structure with SAM. The purpose of CAM is to translate large-range dependencies on channel dimension [22]. The similarity between the foreground and background textures can be addressed by using CAM. There are two main differences between SAM and CAM. SAM has three convolution layers of size 1×1, whereas CAM has only one, and the intermediate feature maps are of different dimensions in SAM and CAM.

3.4 S-DC Net + DULR

We explore S-DC Net + DULR architecture by integrating the S-DC Net and DULR module. The detailed architecture is given in Fig. 2.

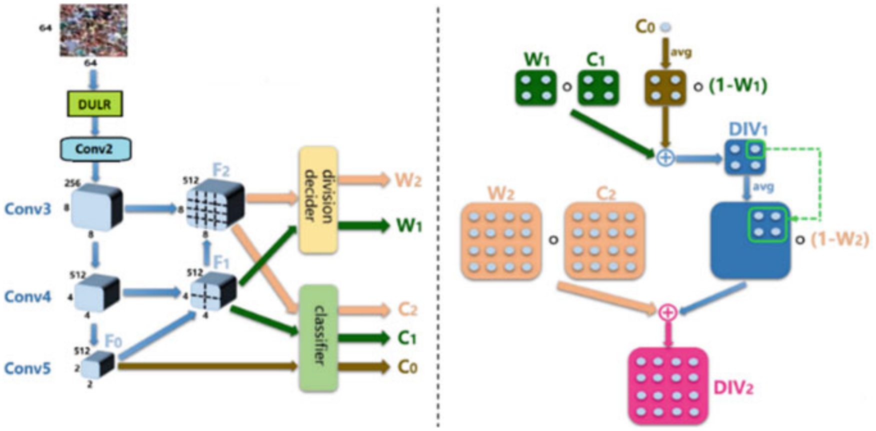

Fig. 2 Detailed architecture of S-DC Net with DULR module

Both the DULR module and the S-DC Net model are pre-trained with VGG16 NET. In the actual S-DC Net model, the image goes into the conv block which has two convolution layers. So, the first convolution layer from the conv block is replaced with the DULR module which encodes the perspective changes and passes the feature map to the second convolution layer in the conv block. From here, the whole functioning is the same as the S-DC Net model. The number of channels input to the DULR module is 3 and the number of output channels is 64 which is passed to the next convolution layer.

4 Experimental Studies

4.1 Dataset Description

The datasets used for the study are Shanghai part A and Shanghai part B. The Part A dataset has a total of 300 training images and a total of 182 testing images. The part B dataset consists of a total of 400 training images and 316 testing images. The datasets are trained and tested for the models, namely S-DC Net, PCC Net, SCAR, and S-DC Net + DULR, and the results were verified and recorded.

4.2 Experimental Results

The studied methods were evaluated on the Shanghai dataset, and the results are presented in Tables 1 and 2. The results clearly show that S-DC Net outperforms the other three models, namely SCAR, PCC Net, and S-DC Net + PCC Net.

Among the recent methods present, the S-DC Net is the best model for crowd counting, and it is proven by the results.

4.3 Analysis of the Studied Models

S-DC Net, PCCNet, and SCAR: The **S-DC Net** model gives the best accuracy as it is not affected by the perspective changes of the images. A divide-and-conquer

Table 1 Comparison of the four approaches over Shanghai Part A dataset

Methods	MAE	MSE
S-DC Net [21]	58.3	95.0
PCC Net [5]	73.5	102.7
SCAR [22]	66.3	114.1
S-DC Net + DULR	432.8	558.9

Table 2 Comparison of the four approaches over Shanghai Part B dataset

Methods	MAE	MSE
S-DC Net [21]	6.71	10.7
PCC Net [5]	11	19
SCAR [22]	9.5	15
S-DC Net + DULR	28.3	82.9

technique is used which causes the entire image to be compartmentalized. Thus, the whole process is repeated on the entire image by initially fixing the number of subparts. Each subpart contains the information of all its previous divided parts, thus making cumulative information to be available at each stage. In **PCC Net**, to prevent any loss from perspective changes, a DULR module is used. This module helps to get the spatial and contextual information from all the four directions. The information at a particular block of an image contains all the information of its previous blocks in all the four directions. Thus, the concatenated feature maps from the four directions form a resultant feature map that contains the perspective changes of the entire image. This model's accuracy is affected by occulted images. Hence, this architecture falls a little back of S-DC Net. **SCAR** has two modules that play a major role in the counting process. The spatial-wise attention module models the large contextual information and captures the changes in the density maps. The channel-wise attention model captures the contextual information between the three channels and also obtains the dependencies between the channels. This information helps to distinguish between the foreground and background. This model fills all possible gaps that are encountered in object counting and makes it a robust model for crowd counting.

Study on S-DC Net + DULR (Variant) Approach: Coming to the last model (S-DC Net + DULR), DULR module when added to S-DC Net made the model to be over-fit due to which the performance of the model was not good enough. Though conceptually the model looks perfect, the S-DC Net captures perspective changes along with spatial and contextual information. To this again, adding a DULR module which provides a feature map with encoded perspective changes makes the model to be over-fit due to which the model performance was moderate. As a future work, we are planning to propose a novel module to be integrated with S-DC Net for a better performance.

5 Direction for Further Research

- Focusing more on the spatial and contextual information to construct a more informative density maps that pushes the envelope further.

- Though the perspectives changes had been encoded well in the recent work, a little more focus toward the occlusion and background segmentation would yield better results.
- Integrating the recent work done on counting in images and applying it in real-time videos would have a good scope of exploring something innovative.

6 Conclusion

Crowd counting has become one of the most demanding tasks to be performed whether it be in security surveillance or in the advertisement sector. However, developing a model that fits the real-world environment is required. Among all the recent architectures that have been developed, S-DC Net appears to be the best model for crowd counting. The divide-and-conquer method used helps in the long run for better performance. Also, the PCC Net and SCAR works well as they have their perks. Based on the recent trends, new models are being developed to push the work done on crowd counting a little further.

Acknowledgements The authors would like to acknowledge Cognitive Science Research Initiative (CSRI)–(NO.DST/CSRI/2017/131 (G)) funding sanctioned by the Department of Science and Technology, Government of India and Council of Scientific and Industrial Research (CSIR) (09/1095(0043)/19-EMR-I).

References

1. Thanasutives P, Fukui K, Numao M, Kijsirikul B (2020) Encoder-decoder based convolutional neural networks with multi-scale-aware modules for crowd counting. arxiv: 2003.05586
2. Hu Y, Jiang X, Liu X, Zhang B, Han J, Cao X, Doermann D (2020) NAS-count: counting-by-density with neural architecture search. arxiv: 2003.00217
3. Shi X, Li X, Wu C, Kong S, Yang J, He L (2020) A real-time deep network for crowd counting. arxiv: 2002.06515
4. Luo A, Yang F, Li X, Nie D, Jiao Z, Zhou S, Cheng H (2020) Hybrid graph neural networks for crowd counting. arxiv: 2002.00092
5. Gao J, Wang Q, Li X (2019) PCC net: perspective crowd counting via spatial convolutional network. arXiv preprint arXiv: 1905.10085
6. Li M, Zhang Z, Huang K, Tan T (2008) Estimating the number of people in crowded scenes by MID based foreground segmentation and head-shoulder detection. In: ICPR, pp 1–4
7. Ge W, Collins RT (2009) Marked point processes for crowd counting. In: CVPR
8. Chan AB, Vasconcelos N (2012) Counting people with low-level features and bayesian regression. IEEE Trans Image Proces 21(4):2160–2177
9. Idrees H, Saleemi I, Seibert C, Shah M (2013) Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of IEEE conference on computer vision and pattern recognition, pp 2547–2554
10. Lempitsky VS, Zisserman A (2010) Learning to count objects in images. In: Proceedings of conference on neural information processing systems, pp 1324–1332

11. Pham V-Q, Kozakaya T, Yamaguchi O, Okada R (2015) COUNT forest: co-voting uncertain number of targets using random forest for crowd density estimation. In: Proceedings of international conference on computer vision, pp 3253–3261
12. Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G (2008) The graph neural network model. TNNLS 20(1):61–80
13. Li Y, Tarlow D, Brockschmidt M, Zemel R (2016) Gated graph sequence neural networks. In: ICLR
14. Gilmer J, Schoenholz SS, Riley PF, Vinyals O, Dahl GE (2017) Neural message passing for quantum chemistry. CoRR abs/1704.01212
15. Zhang Y, Zhou D, Chen S, Gao S, Ma Y (2016) Single-image crowd counting via multi-column convolutional neural network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 589–597
16. Cao X, Wang Z, Zhao Y, Su F (2018) Scale aggregation network for accurate and efficient crowd counting. In Proceedings of the European conference on computer vision (ECCV), pp 734–750
17. Shi M, Yang Z, Xu C, Chen Q (2018) Perspective-aware CNN for crowd counting. CoRR, abs/1807.01989
18. Liu W, Lis K, Salzmann M, Fua P (2018) Geometric and physical constraints for head plane crowd density estimation in videos. CoRR, abs/1803.08805
19. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. Comput Sci
20. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention, pp 234–241
21. Xiong H, Lu H, Liu C, Liang L, Cao Z, Shen C (2019) From open set to closed set: counting objects by spatial divide-and-conquer. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV)
22. Gao J, Wang Q, Yuan Y (2019). SCAR: spatial-/channel-wise attention regression networks for crowd counting. arXiv preprint arXiv: 1908.03716