Single Domain Generalization for Crowd Counting

Supplementary Material

6. More Implementation Details

This section presents more details about our implementations of MPCount and other DG methods adapted to crowd counting.

6.1. MPCount

Fig. 8 illustrates the detailed structure of our model backbone adapted from VGG-16BN. Skip connections are added to integrate multi-level information, enhancing the expressive power of crowd features. Note that we only utilize the deepest level of features for PC prediction, as shallow features usually encode more style information [23] and thus are more vulnerable to domain shift.

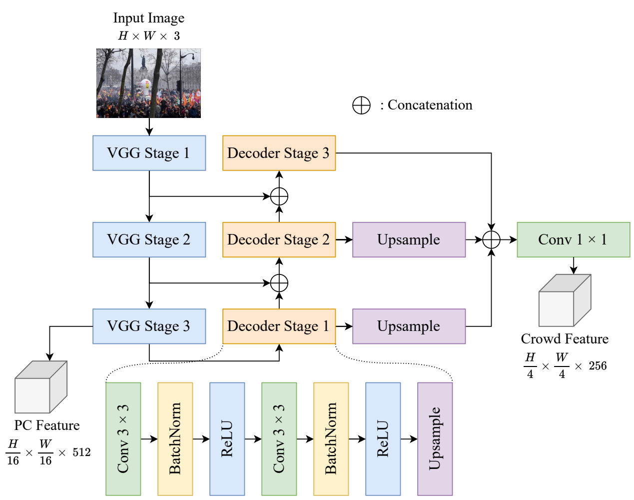


Figure 8. Structure of the model backbone of MPCount

For data augmentation, we use three types of photometric transformations: (1) *Color jittering* with probability 0.8 (brightness of 0.5, contrast of 0.2, saturation of 0.2, and hue of 0.1); (2) *Gaussian Blurring* with probability 0.5 (kernel size of 3 and sigma of 1); (3) *Sharpening* with probability 0.5 (sharpness factor of 5).

All experiments on MPCount are conducted on four NVIDIA GeForce 3090Ti GPUs, and implementations are based on Python 3.10 and PyTorch 2.0. During training, the batch size and number of workers are both set to 16. The random seed is set to 2023. Following [8], the ground-truth density maps are multiplied by 1,000.

6.2. Adapted Methods

IBN [23], SW [24] and ISW [5] are three DG methods originally designed for image classification or semantic segmentation. We select ResNet-50 [9] truncated after conv4_6 as their common backbone and connect it with a density regression head consisting of three convolutional layers, as illustrated in Fig. 9.

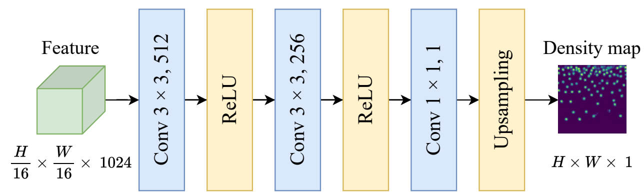


Figure 9. Regression head used for the adapted methods

7. More Quantitative Results

This section includes more quantitative results of MPCount.

7.1. Quantitative Results of PCM prediction

Tab. 6 shows the quantitative results of PCM prediction from MPCount on SHA and SHB. Three common metrics for image segmentation are used, namely mAcc, mIoU and mDice. Since PC is only an auxiliary task to assist crowd density prediction, and no attention is specifically paid to improve the accuracy of PCMs, we do not require and expect the performance to be perfect. However, as shown in Tab. 6, satisfactory results are still achieved to support the function of PCMs to filter out areas without human head.

Metric	mAcc(%)	mIoU(%)	mDice(%)
A → B	96.9	82.3	90.3
B → A	87.7	73.7	85.1

Table 6. Quantitative Results of PCM prediction on SHA (A) and SHB (B)

8. More Visualization Results

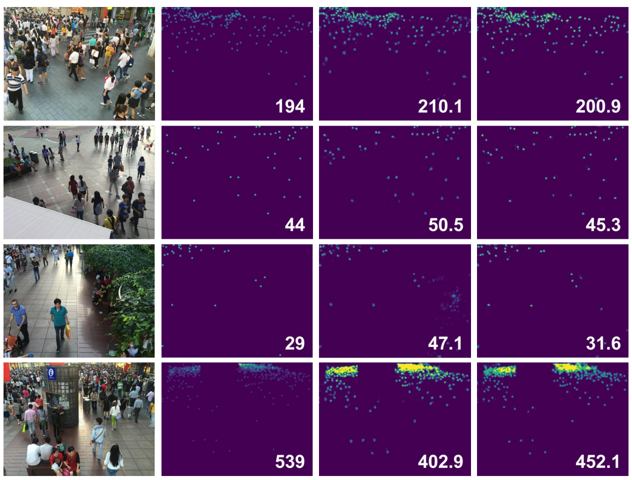
This section shows more visualization results of predicted density maps and PCMs from our MPCount. We use DC-CUS [6] as the baseline for comparison. Visualization results of density maps are shown in Fig. 10 ∼ Fig. 15, and PCMs are displayed in Fig. 16.

8.1. Visualization of Density Maps

The visualization result on various datasets demonstrate that MPCount predict more accurate density maps compared with DCCUS under different scenarios. The performance gap is even more significant on data with narrow distribution, as shown in Fig. 14 and Fig. 15. In many cases like the first row in Fig. 11, DCCUS wrongly outputs high density values from areas containing no human head. MPCount alleviates this problem with PC maps effectively filtering out areas without head based on accurate patch-level information.

8.2. Visualization of PCMs

Fig. 16 displays the ground-truth PCMs and the predicted PCMs before and after binarization from different datasets. The results indicate that, the predicted PCMs can accurately capture the information of crowd location, and noisy predictions are successfully filtered out by binarization.

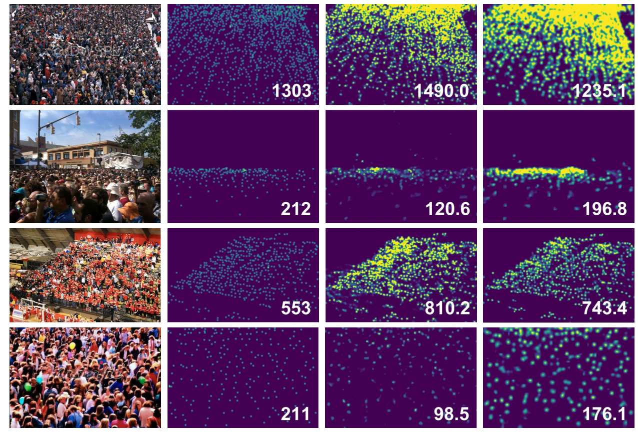


(a) Image (b) GT (c) DCCUS (d) MPCount

Figure 10. Visualization results on A → B

(a) Image (b) GT (c) DCCUS (d) MPCount

Figure 11. Visualization results on A → Q

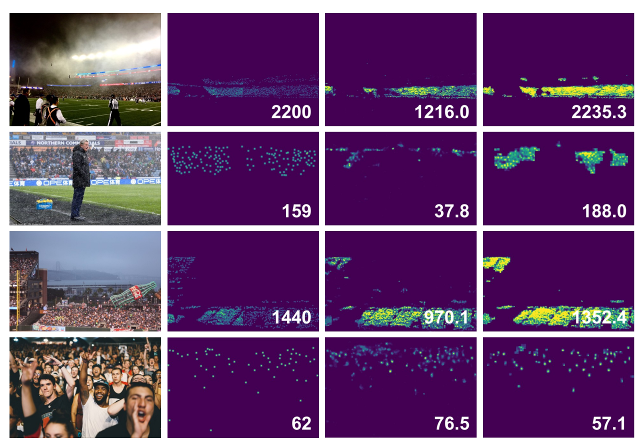


(a) Image (b) GT (c) DCCUS (d) MPCount

Figure 12. Visualization results on B → A

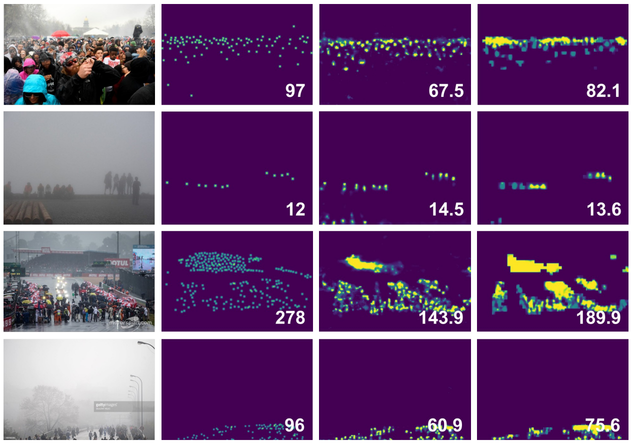
(a) Image (b) GT (c) DCCUS (d) MPCount

Figure 13. Visualization results on B → Q



(a) Image (b) GT (c) DCCUS (d) MPCount

Figure 14. Visualization results on SR → SD



(a) Image (b) GT (c) DCCUS (d) MPCount

Figure 15. Visualization results on SN $\rightarrow$ FH

(a) Image (b) GT (c) Predicted (d) Binarized

Figure 16. Visualization results of PCMs under various settings