# Calibration-Free Multi-view Crowd Counting

Qi Zhang[1,2(✉)] and Antoni B. Chan[2]

[1] College of Computer Science & Software Engineering, Shenzhen University, Shenzhen, China
[2] Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China
qzhang364-c@my.cityu.edu.hk, abchan@cityu.edu.hk

**Abstract.** Deep learning based multi-view crowd counting (MVCC) has been proposed to handle scenes with large size, in irregular shape or with severe occlusions. The current MVCC methods require camera calibrations in both training and testing, limiting the real application scenarios of MVCC. To extend and apply MVCC to more practical situations, in this paper we propose *calibration-free* multi-view crowd counting (CF-MVCC), which obtains the scene-level count directly from the density map predictions for each camera view without needing the camera calibrations in the test. Specifically, the proposed CF-MVCC method first estimates the homography matrix to align each pair of camera-views, and then estimates a matching probability map for each camera-view pair. Based on the matching maps of all camera-view pairs, a weight map for each camera view is predicted, which represents how many cameras can reliably see a given pixel in the camera view. Finally, using the weight maps, the total scene-level count is obtained as a simple weighted sum of the density maps for the camera views. Experiments are conducted on several multi-view counting datasets, and promising performance is achieved compared to calibrated MVCC methods that require camera calibrations as input and use scene-level density maps as supervision.

## 1 Introduction

Crowd counting has many applications in real life, such as crowd control, traffic scheduling or retail shop management, etc. In the past decade, with the strong learning ability of deep learning models, single-view image counting methods based on density map prediction have achieved good performance. However, these single-view image methods may not perform well when the scene is too large or too wide, in irregular shape, or with severe occlusions. Therefore, multi-view crowd counting (MVCC) has been proposed to fuse multiple camera views to mitigate these shortcomings of single-view image counting.

**Supplementary Information** The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-20077-9_14.


S. Avidan et al. (Eds.): ECCV 2022, LNCS 13669, pp. 227–244, 2022.
https://doi.org/10.1007/978-3-031-20077-9_14

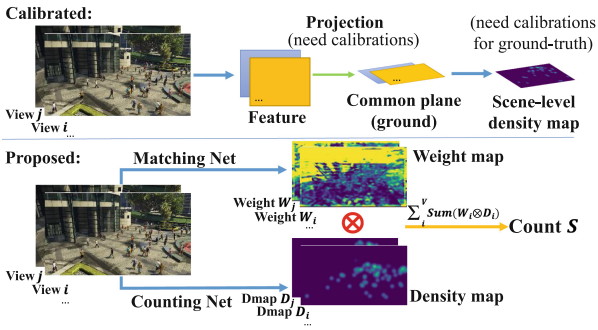


**Fig. 1.** The proposed calibration-free multi-view crowd counting (CF-MVCC) combines single-view predictions with learned weight maps to obtain the scene-level count.

The current MVCC methods rely on camera calibrations (both intrinsic and extrinsic camera parameters) to project features or density map predictions from the single camera views to the common ground-plane for fusion (see Fig. 1 top). The camera calibration is also required to obtain the ground-truth people locations on the ground-plane to build scene-level density maps for supervision. Although the latest MVCC method [55] handles the cross-view cross-scene (CVCS) setting, it still requires the camera calibrations during training and testing, which limits its real application scenarios. Therefore, it is important to explore *calibration-free* multi-view counting methods.

For calibration-free MVCC, the key issue is to align the camera views without pre-provided camera calibrations. However, it is difficult to calibrate the cameras online from the multi-view images in MVCC, since there are a relatively small number of cameras (less than 5) that are typically on opposite sides of the scene (i.e., large change in camera angle). It may also be inconvenient to perform multi-view counting by calibrating the camera views first if the model is tested on many different scenes. Besides, extra priors about the scenes are required to estimate camera intrinsic or extrinsic, such as in [1,2,4]. We observe that the people' heads are approximately on a plane in the 3D world, and thus the same person's image coordinates in different camera views can be roughly modeled with a homography transformation matrix. Thus, instead of using a common ground-plane for aligning all the camera views together like previous methods [53,55], we propose to align pairs of camera views by estimating pairwise homography transformations.

To extend and apply MVCC to more practical situations, in this paper, we propose a calibration-free multi-view crowd counting (CF-MVCC) method, which obtains the scene-level count as a weighted summation over the predicted density maps from the camera-views (see Fig. 1). The weight maps applied to each density map consider the number of cameras in which the given pixel is visible (to avoid double counting) and the confidence of each pixel (to avoid poorly predicted regions such as those far from the camera). The weight maps are generated using estimated pairwise homographies in the testing stage, and thus CF-MVCC can be applied to a novel scene without camera calibrations.

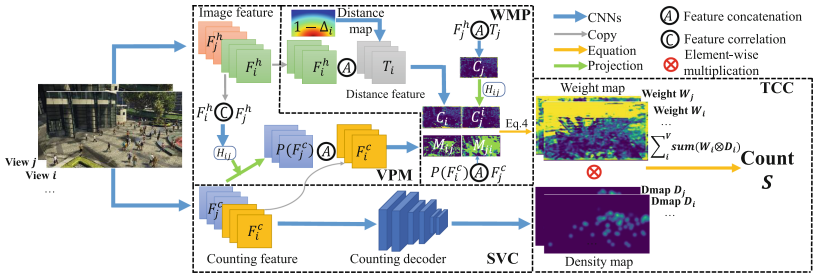


**Fig. 2.** Pipeline of CF-MVCC. The single-view counting (SVC) module predicts density maps $D_i$ for each camera-view. Given a pair of camera-views $(i, j)$, the view-pair matching (VPM) module estimates the homography $H_{ij}$ and a matching probability map $M_{ij}$ between them. The weight map prediction (WMP) module calculates the weight map $W_i$ for each camera using the matching probability maps $M_{ij}$ and confidence maps $C_i$, where the confidence maps are estimated from image features $F_i^h$ and distance features $T_i$. Finally, the total count calculation (TCC) is obtained as a weighted sum between the density maps $D_i$ and the weight maps $W_i$.

Specifically, the proposed CF-MVCC method estimates the total crowd count in the scene via 4 modules. 1) *Single-view counting module* (SVC) consists of feature extraction and density map prediction submodules. 2) *View-pair matching module* (VPM) estimates the homography between pairs of camera views. For each camera pair, the features from one camera view are then projected to the other view, concatenated, and used to estimate a matching probability map between the two camera view. 3) *Weight map prediction module* (WMP) calculates a weight map for each view using all the matching probability maps. In addition, image content and distance information are used when calculating the weight maps to adjust for the confidence from each camera view. 4) *Total count calculation module* (TCC) obtains the total count as a weighted sum of the predicted single-view density maps using the estimated weight maps. In summary, the contributions of the paper are three-fold:

1. We propose a calibration-free multi-view counting model (CF-MVCC) to further extend the application of MVCC methods to more unconstrained scenarios, which can be applied to new scenes without camera calibrations. As far as we know, this is the first work to extend multi-view counting to the calibration-free camera setting.
2. The proposed method uses single-view density map predictions to directly estimate the scene crowd count without pixel-level supervision, via a weighting map with confidence score that is guided by camera-view content and distance information.
3. We conduct extensive experiments on multi-view counting datasets and achieve better performance than calibration-free baselines, and promising performance compared to well-calibrated MVCC methods. Furthermore, our model trained on a large synthetic dataset can be applied to real novel scenes with domain adaptation.

## 2 Related Work

In this section, we review single-image and multi-view counting, followed by DNN-based homography estimation.

**Single-Image Counting.** Early research works on single-image counting rely on hand-crafted features [13,41], including detection-based [35], regression-based [6] or density map based methods [16]. Deep-learning based methods have been proposed for single image counting via estimating density maps [3,29,37,51]. Among them, many have focused on handling the scale variation and perspective change issues [12,14,17,20,40]. Unlike [38] and [47], [49] corrected the perspective distortions by uniformly warping the input images guided by a predicted perspective factor. Recent research explore different forms of supervision (e.g., regression methods or loss functions) [43,45]. [22] introduced local counting maps and an adaptive mixture regression framework to improve the crowd estimation precision in a coarse-to-fine manner. [25] proposed Bayesian loss, which adopts a more reliable supervision on the count expectation at each annotated point.

To extend the application scenarios of crowd counting, weakly supervised [5,21,50,57] or semi-supervised methods [23,36,42] have also been proposed. Synthetic data and domain adaptation have been incorporated for better performance [46]. Other modalities are also fused with RGB images for improving the counting performance under certain conditions, such as RGBD [18] or RGBT [19]. In contrast to category-specific counting methods (e.g., people), general object counting has also been proposed recently [24,31,48]. [31] proposed a general object counting dataset and a model that predicts counting maps from the similarity of the reference patches and the testing image.

Generally, all these methods aim at counting objects in single views, while seldom have targeted at the counting for whole scenes where a single camera view is not enough to cover a large or a wide scene with severe occlusions. Therefore, multi-view counting is required to enhance the counting performance for large and wide scenes.

**Multi-view Counting.** Multi-view counting fuses multiple camera views for better counting performance for the whole scene. Traditional multi-view counting methods consist of detection-based [8,26], regression-based [34,44] and 3D cylinder-based methods [10]. These methods are frequently trained on a small dataset like PETS2009 [9]. Since they rely on hand-crafted features and foreground extraction techniques, their performance is limited.

Recently, deep-learning multi-view counting methods have been proposed to better fuse single views and improve the counting performance. A multi-view multi-scale (MVMS) model [53] is the first DNNs based multi-view counting method. MVMS is based on 2D projection of the camera-view feature maps to a common ground-plane for predicting ground-plane scene-level density maps. However, the projection operation requires that camera calibrations are provided for training and testing. Follow-up work [54] proposed to use 3D density maps and 3D projection to improve counting performance. [55] proposed a cross-view cross-scene (CSCV) multi-view counting model by camera selection and noise

injection training. [58] enhanced the performance of the late fusion model in MVMS by modeling the correlation between each pair of views.

For previous works, the single camera views (feature maps or density maps) are projected on the ground plane for fusion to predict the scene-level density maps, and thus camera calibrations are needed in the testing stage, which limits their applicability on novel scenes where camera calibrations are unavailable. In contrast, we propose a calibration-free multi-view counting method that does not require camera calibrations during testing. Our calibration-free setting is more difficult compared to previous multi-view counting methods.

**Deep Homography Estimation.** Our work is also related to homography estimation works [27,30], especially DNNs-based methods [7,52]. [7] proposed to estimate the 8°C-of-freedom homography from an image pair with CNNs. [28] proposed an unsupervised method that minimizes the pixel-wise intensity error between the corresponding regions, but their unsupervised loss is not applicable when the change in camera view angle is large. [52] proposed to learn an outlier mask to select reliable regions for homography estimation. [15] proposed a multi-task learning framework for dynamic scenes by jointly estimating dynamics masks and homographies.

Our proposed model estimates the homography matrix between the people head locations in the two views of each camera pair. Note that the change in view angle for camera-view pairs in the multi-view counting datasets (e.g., CityStreet) is quite large, which is in contrast to the typical setting for previous DNN-based homography estimation works where the change in angle is small. Therefore, the priors for unsupervised methods (e.g., [28]) are not applicable. Furthermore, the homography matrix in the proposed model is constructed based on the correspondence of people heads in the camera view pair, which are more difficult to observe compared to the objects in typical homography estimation datasets. Instead, we use a supervised approach to predict the homography matrix.

## 3 Calibration-Free Multi-view Crowd Counting

In this section we propose our model for calibration-free multi-view crowd counting (CF-MVCC). In order to avoid using the projection operation, which requires camera calibration, we could obtain the total count by summing the density maps predicted from each camera view. However, just summing all the single-view density maps would cause double counting on pixels that are also visible from other cameras. Therefore, we apply a weight map to discount the contribution of pixels that are visible from other camera views (see Fig. 1). The weight map is computed from a matching score map, which estimates the pixel-to-pixel correspondence between a pair of camera-views, and a confidence score map, which estimates the reliability of a given pixel (e.g., since predictions on faraway regions are less reliable). Specifically, our proposed CF-MVCC model consists of following 4 modules: single-image counting, view-pair matching, weight map prediction, and total count calculation. The pipeline is illustrated in Fig. 2. Furthermore, to validate the proposed method's effectiveness on novel scenes, we

also train our model on a large synthetic dataset, and then apply it to real scenes via domain adaptation.

### 3.1 Single-View Counting Module (SVC)

The SVC module predicts the counting density map $D_i$ for each camera-view $i$, based on an extracted feature map $F_i^c$. For fair comparison with the SOTA calibrated MVCC method CVCS [55], in our implementation, we follow CVCS [55] and use the first 7 layers of VGG-net [39] as the feature extraction subnet, and the remaining layers of CSR-net [17] as the decoder for predicting $D_i$. Other single-view counting models are also tested in the ablation study of the experiments and Supp. The loss used for training SVC is $l_d = \sum_{i=1}^{V} ||D_i - D_i^{gt}||_2^2$, where $D_i$ and $D_i^{gt}$ are the predicted and ground-truth density maps, the summation $i$ is over cameras, and $V$ is the number of camera-views.

### 3.2 View-Pair Matching Module (VPM)

The VPM module estimates the matching score $M_{ij}$ between any 2 camera views $i$ and $j$. First, we use a CNN to estimate the homography transformation matrix from camera view $i$ to $j$, denoted as $H_{ij}$. This CNN extracts the 2 camera views' feature maps $F_i^h$ and $F_j^h$. Next, the correlation map is computed between $F_i^h$ and $F_j^h$, and a decoder is applied to predict the homography transformation matrix $H_{ij}$. For supervision, the homography matrix ground-truth $H_{ij}^{gt}$ is calculated based on the corresponding people head locations in the 2 camera views. In the case that the camera view pair have no overlapping field-of-view, then a dummy homography matrix is used as ground-truth to indicate the 2 camera views are non-overlapped. The loss used to train the homography estimation CNN is $l_h = \sum_{i=1}^{V} \sum_{j \neq i} ||H_{ij} - H_{ij}^{gt}||_2^2$.

Next a subnetwork is used to predict the matching score map $M_{ij}$, whose elements indicate the probability of whether the given pixel in view $i$ has a match *anywhere* in view $j$. The input into the subnet is the concatenation of features $F_i^c$ from view $i$, and the aligned features from view $j$, $P(F_j^c, H_{ij})$, where $P$ is the projection layer adopted from STN [11].

### 3.3 Weight Map Prediction Module (WMP)

The WMP module calculates the weight $W_i$ for each view $i$ based on the matching score maps $\{M_{ij}\}_{j \neq i}$ with other camera views. Specifically, the weight map $W_i$ is:

$$W_i = 1/(1 + \sum_{j \neq i} M_{ij}). \tag{1}$$

Note that for pixel $p$, the denominator $1 + \sum_{j \neq i} M_{ij}(p)$ is the number of camera-views that see pixel $p$ in camera-view $i$ (including camera-view $i$ itself). Thus the weight $W_i(p)$ will average the density map values of corresponding pixels across

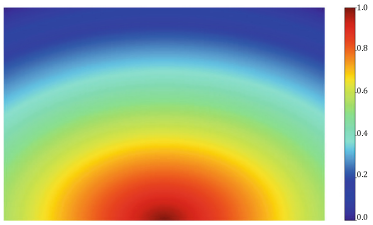


**Fig. 3.** Example of distance map $(1 - \Delta_i)$. Usually, in surveillance cameras, the top and side areas on the image plane are faraway regions and the bottom areas are the nearer regions.

visible views, thus preventing double-counting of camera-view density maps with overlapping fields-of-view.

In (1), the contribution of each camera-view is equal. However, single-view density map prediction may not always be reliable. Generally, the confidence (reliability) for regions with occlusions is lower than regions without occlusions, and the confidence of regions far from the camera is lower than near-camera regions. Therefore, to factor in these issues, we estimate a confidence score map $C_i$ for each camera view $i$, based on the image content features and pixel-wise distance information. The confidence maps are then incorporated into (1),

$$W_i = C_i / (C_i + \sum_{j \neq i} C_j^i \odot M_{ij}), \tag{2}$$

where $C_j^i = P(C_j, H_{ij})$ is the projection of confidence map $C_j$ to camera view $i$. Note that in (2), the views with higher confidence will have higher contribution to the count of a given pixel.

The confidence map $C_i$ is estimated with a CNN whose inputs are the image feature map $F_i^h$ and distance feature map $T_i$. Ideally, $T_i$ should be computed by feeding a distance map $\Delta_i$, where each pixel is the distance-to-camera in the 3D scene, into a small CNN. We note the surveillance cameras are usually angled downward to cover the scene, where the top and side areas on the image plane are faraway regions and the bottom areas are the nearer regions. Since we do not have camera calibration to compute 3D distances, we use a simple approximation for $\Delta_i$ where the bottom-middle pixel is considered as the pixel nearest to the current camera (the value is 0), and values of other pixels are the Euclidean distance to the bottom-middle pixel (See Fig. 3). The distance map $\Delta_i$ is then normalized to $[0, 1]$, and $(1 - \Delta_i)$ is fed into a CNN to output the distance feature $T_i$.

**Related Work.** The weight map of our proposed method is different from the comparison method Dmap_weighted from [53]. Specifically, Dmap_weighted uses the camera calibrations and assumes each image pixel's height in 3D world is the average person height to calculate how many cameras can see a given pixel. Dmap_weighted also does not consider occlusion handling and prediction

confidence. In contrast, our method does not use camera calibrations, but instead estimates matching scores based on estimated homographies between camera views, image contents and geometry constraints (see Eq. 1). Furthermore, we incorporate confidence scores to adjust each view's contribution, due to occlusion and distance (see Eq. 2).

### 3.4 Total Count Calculation Module (TCC)

With the estimated weight map $W_i$ for each camera view $i$, the final count $S$ is the weighted summation of the density map predictions $D_i$: $S = \sum_{i=1}^{V} \text{sum}(W_i \odot D_i)$, where $\odot$ is element-wise multiplication, and sum is the summation over the map. For training, the total count loss is the MSE of the count prediction: $l_s = ||S - S^{gt}||_2^2$ , where $S^{gt}$ is the ground-truth count. Finally, the loss for training the whole model is $l = l_s + l_d + l_h$.

### 3.5 Adaptation to Novel Real Scenes

To apply our model to new scenes with novel camera views, we need a large number of multi-view counting scenes for training. Therefore, we train the proposed model on a large multi-view counting dataset [55]. However, directly applying the trained model to real scenes might not achieve satisfying performance due to the domain gap between the synthetic and real data in terms of single-view counting, view-pair homography estimation and matching. To reduce the domain gap, we first fine-tune the model trained on synthetic data on each real test scene with an unsupervised domain adaptation (UDA) technique [55], where only the test images are used without counting annotations or camera calibrations. To further improve the performance, we use one image with density map annotations from the training set of the target scenes, and only fine-tune the SVC module of the proposed model with the one labeled frame. Compared to [46], we only use synthetic labels and one labeled frame from the target scene, and do not require large amounts of target scene annotations; while compared to [55], we do not need calibrations of the real scenes. Therefore, ours is a more difficult and practical setting for applying the trained multi-view counting model to real scenes.

## 4 Experiment

### 4.1 Experiment Setting

**Ground-Truth.** We use the single-view density maps, homography transformation matrix, and scene crowd count as ground-truth for training. The ground-truth for the single-view density maps are constructed as in typical single-image counting methods [56]. The ground-truth homography transformation matrix of a camera-view pair is calculated with the corresponded people head coordinates (normalized to $[-1, 1]$). If there are no common people in the 2 camera views (no overlapped region), a "dummy" homography matrix is used as the ground-truth: $H = [0, 0, -10; 0, 0, -10; 0, 0, 1]$. As for the ground-truth people count, we only

require the total scene-level count, which is in contrast to [53], which requires scene-level people annotations on the ground-plane. Thus our setting is more difficult compared to the previous multi-view counting methods that use camera calibration and pixel-level supervision.

**Training and Evaluation.** The training is stage by stage: we train the SVC and homography estimation CNNs, then fix both of them and train the remaining modules. On the large synthetic dataset, we use learning rates of $10^{-3}$. On the real scene datasets, the learning rate is $10^{-4}$. Network settings are in the supplemental. Mean absolute error (MAE) and mean normalized absolute error (NAE) of the predicted counts are used as the evaluation metrics.

**Datasets.** We validate the proposed calibration-free multi-view counting on both a synthetic dataset CVCS [55] and real datasets, CityStreet [53] and PETS2009 [9]. Furthermore, we also apply the proposed model trained on CVCS dataset to real datasets CityStreet, PETS2009 and DukeMTMC [32,53].

- **CVCS** is synthetic dataset for multi-view counting task, which contains 31 scenes. Each scene contains 100 frames and about 100 camera views (280k total images). 5 camera views are randomly selected for 5 times for each scene in the training, and 5 camera views are randomly selected for 21 times for each test scene during testing. No camera calibrations are used in the training or testing. The input image resolution is $640 \times 360$.
- **CityStreet, PETS2009 and DukeMTMC** are 3 real scene datasets for multi-view counting. CityStreet contains 3 camera views and 300 multi-view frames ($676 \times 380$ resolution) for training and 200 for testing. PETS2009 contains 3 camera views and 1105 multi-view frames ($384 \times 288$) for training and 794 for testing. DukeMTMC contains 4 camera views and 700 multi-view frames ($640 \times 360$) for training and 289 for testing. Among these 3 datasets, CityStreet is the most complicated dataset as it contains more severe occlusions and larger angle changes between camera views.

**Comparison Methods.** We denote our method using the weight maps in Eq. 1 as CF-MVCC, and the weight maps with confidence scores in Eq. 2 as CF-MVCC-C. As there are no previous calibration-free methods proposed, we adapt existing approaches to be calibration-free:

- **Dmap_weightedH**: This is the calibration-free version of Dmap_weighted in [53]. With Dmap_weighted, the density maps are weighted by how many times an image pixel can be seen by other camera views, based on the camera calibrations. Since camera calibrations are not available in our setting, the estimated homography $H_{ij}$ is used to calculate the weight maps. Note that this method only considers the camera geometry, and not other factors (e.g., image contents, occlusion, and distance) when computing the weights.
- **Dmap_weightedA**: The camera-view features are concatenated and used to estimate the weight maps for summing single-view predictions, which is a self-attention operation. Compared to Dmap_weightH and our method, Dmap_weightedA only considers image contents, and no geometry constraints.

**Table 1.** Scene-level counting performance on synthetic multi-scene dataset CVCS.

| | Method | MAE | NAE |
|---|---|---|---|
| Calibrated | CVCS_backbone | 14.13 | 0.115 |
| | CVCS (MVMS) | 9.30 | 0.080 |
| | CVCS | **7.22** | **0.062** |
| Calibration-free | Dmap_weightedH | 28.28 | 0.239 |
| | Dmap_weightedA | 19.85 | 0.165 |
| | Total_count | 18.89 | 0.157 |
| | 4D_corr | 17.76 | 0.149 |
| | CF-MVCC (ours) | 16.46 | 0.140 |
| | CF-MVCC-C (ours) | **13.90** | **0.118** |

- **Total_count**: Since scene-level density maps are not available in our setting, we replace scene-level density maps with total count loss in CVCS [55].
- **4D_corr**: Replacing the VPM module in CF-MVCC with a 4D correlation [33] method for estimating the matching score $M_{ij}$ of the camera-view pair.

Finally, we compare with multi-view counting methods that use camera calibrations: MVMS [53], 3D [54], CVCS_backbone and CVCS [55], and CVF [58].

### 4.2 Experiment Results

**Scene-Level Counting Performance.** We show the scene-level counting performance of the proposed models and comparison methods on CVCS, CityStreet and PETS2009 in Tables 1 and 2. On CVCS dataset, the proposed CF-MVCC-C achieves the best performance among the calibration-free methods. The comparison methods Dmap_weightedH and Dmap_weightedA only consider the camera geometry or the image contents, and thus their performance is worse than CF-MVCC, which considers both. Including confidence score maps into the weights (CF-MVCC-C) will further improve the performance. Total_count replaces the pixel-level supervision in CVCS with the total count loss, but directly regressing the scene-level count is not accurate since the projection to the ground stretches the features and makes it difficult to learn to fuse the multi-view features without pixel-level supervision. The 4D_corr method also performs poorly because the supervision from the total-counting loss is too weak to guide the learning of the matching maps from the 4D correlation maps. Finally, our CF-MVCC-C performs worse than calibrated methods CVCS and CVCS (MVMS), but still better than CVCS_backbone, which is reasonable since our method does not use any calibrations and no pixel-wise loss is available for the scene-level prediction.

In Table 2, on both real single-scene datasets, our proposed calibration-free methods perform better than the other calibration-free methods. Furthermore, CF-MVCC-C is better than CF-MVCC, indicating the effectiveness of the confidence score in the weight map estimation. Compared to calibrated methods,

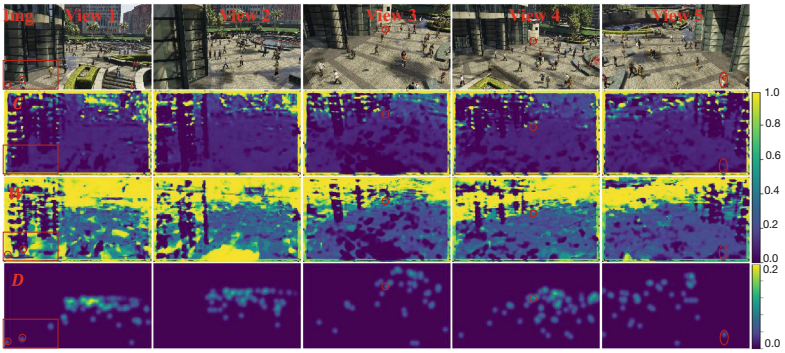


**Fig. 4.** Example of confidence maps $C$, weight maps $W$ and density maps $D$.

**Table 2.** Scene-level counting performance on real single-scene datasets.

| | Method | CityStreet | | PETS2009 | |
|---|---|---|---|---|---|
| | | MAE | NAE | MAE | NAE |
| Calibrated | MVMS | 8.01 | 0.096 | 3.49 | 0.124 |
| | 3D_counting | 7.54 | **0.091** | 3.15 | **0.113** |
| | CVF | **7.08** | - | **3.08** | - |
| Calibration-free | Dmap_weightedH | 9.84 | 0.107 | 4.23 | 0.136 |
| | Dmap_weightedA | 9.40 | 0.123 | 6.25 | 0.252 |
| | Total_count | 11.28 | 0.152 | 6.95 | 0.265 |
| | 4D_corr | 8.82 | 0.102 | 4.55 | 0.147 |
| | CF-MVCC (ours) | 8.24 | 0.103 | 3.84 | 0.125 |
| | CF-MVCC-C (ours) | **8.06** | **0.102** | **3.46** | **0.116** |

CF-MVCC-C is comparable to MVMS [53], and slightly worse than 3D [54] and CVF [58]. The reason might be that the calibrated methods can implicitly learn some specific camera geometry in the fusion step, since the methods are trained and tested on the same scenes.

**Visualization Results.** We show the visualization results the predicted confidence, weight, and density maps in Fig. 4. The red boxes indicate regions that cannot be seen by other cameras, and thus their predicted weights are large regardless of the confidence scores. The red circles show a person that can be seen in 3 camera views (3, 4 and 5) – the weights are small since the person can be seen by multiple cameras. This shows that the proposed method is effective at estimating weight maps with confidence information. See the supplemental for more visualizations (*eg.* projection results with ground-truth and predicted homography matrix).

**Ablation Studies.** Various ablation studies are evaluated on the CVCS dataset.

**Table 3.** Ablation study on estimating the confidence map using image features and/or distance information.

| Method | Feat. | Dist. | MAE | NAE |
|---|---|---|---|---|
| CF-MVCC | | | 16.46 | 0.140 |
| CF-MVCC-F | ✓ | | 16.13 | 0.139 |
| CF-MVCC-D | | ✓ | 16.12 | 0.135 |
| CF-MVCC-C | ✓ | ✓ | **13.90** | **0.118** |

**Table 4.** Ablation study on single-view counting networks for SVC module.

| SVC | Method | MAE | NAE |
|---|---|---|---|
| CSR-Net [17] | CF-MVCC | 16.46 | 0.140 |
| | CF-MVCC-C | 13.90 | 0.118 |
| LCC [22] | CF-MVCC | 14.01 | 0.117 |
| | CF-MVCC-C | **12.79** | **0.109** |

**Ablation Study on Confidence Map.** We conduct an ablation study on the confidence score estimation: 1) without the confidence scores, i.e., CF-MVCC; 2) using only image features to estimate confidence scores, denoted as CF-MVCC-F; 3) using only distance information, denoted as CF-MVCC-D; 4) using both image features and distance, i.e., our full model CF-MVCC-C. The results are presented in Table 3. Using either image features (CF-MVCC-F) or distance information (CF-MVCC-D) can improve the performance compared to not using the confidence map (CF-MVCC). Furthermore, using both image features and distance information (CF-MVCC-C) further improves the performance. Thus, the confidence map effectively adjusts the reliability of the each camera view's prediction, in order to handle occlusion and/or low resolution.

**Ablation Study on Single-View Counting Network.** We implement and test our proposed model with another recent single-view counting network LCC [22], which uses a larger feature backbone than CSRnet, and is trained with traditional counting density maps as in our model. The results presented in Table 4 show that the proposed CF-MVCC-C achieves better performance than CVCS when using different single-view counting networks in the SVC module.

**Ablation Study on the Homography Prediction Module.** We also conduct experiments to show how the homography prediction module affects the performance of the model. Here the ground-truth homography matrix is used for training the proposed model. The performance of the proposed model trained with homography prediction $H_{pred}$ or ground-truth $H_{gt}$ is presented in Table 5. The model with ground-truth homography achieves better performance, and CF-MVCC-C performs better than CF-MVCC.

**Ablation Study on Variable Numbers of Camera-Views.** The modules of the proposed models are shared across camera-views and camera-view pairs,

**Table 5.** Ablation study on the homography matrix input.

| Homography | Method | MAE | NAE |
|---|---|---|---|
| $H_{pred}$ | CF-MVCC | 16.64 | 0.140 |
| | CF-MVCC-C | 13.90 | 0.118 |
| $H_{gt}$ | CF-MVCC | 12.04 | 0.101 |
| | CF-MVCC-C | **11.69** | **0.098** |

**Table 6.** Ablation study on testing with different numbers of input camera-views. The model is trained on CVCS dataset with 5 camera-views as input.

| No. Views | CVCS_backbone | | CVCS | | CF-MVCC-C | |
|---|---|---|---|---|---|---|
| | MAE | NAE | MAE | NAE | MAE | NAE |
| 3 | 14.28 | 0.130 | 7.24 | 0.071 | 11.01 | 0.107 |
| 5 | 14.13 | 0.115 | 7.22 | 0.062 | 13.90 | 0.118 |
| 7 | 14.35 | 0.113 | 7.07 | 0.058 | 18.45 | 0.147 |
| 9 | 14.56 | 0.112 | 7.04 | 0.056 | 22.23 | 0.174 |

so our method can be applied to different numbers of camera views at test time. In Table 6, the proposed models are trained on the CVCS dataset [55] with 5 input camera views and tested on different number of views. Note that the ground-truth count is the people count covered by the multi-camera views. The performance of the proposed method CF-MVCC-C is worse than the calibrated method CVCS [55], but better than the calibrated method CVCS_backbone [55] when the number of test camera views are close to the number of training views (3 and 5). Unlike CVCS method, the performance of CF-MVCC-C degrades as the number of cameras increases. The reason is that the error in weight map prediction might increase when the number of camera views changes.

**Adaptation to Novel Real Scenes.** In this part, we use domain adaption to apply the proposed model CF-MVCC-C pre-trained on the synthetic CVCS dataset to the real scene datasets CityStreet, PETS2009 and DukeMTMC. We consider 3 training methods: 1) **Synth**, where pre-trained model is directly tested on the real scenes; 2) **Synth+UDA**, where unsupervised domain adaptation is applied to the pre-trained model. Specifically, 2 discriminators are added to distinguish the single-view density maps and weight maps of the source and target scenes. 3) **Synth+F**, where the models pre-trained on the synthetic dataset are fine-tuned with one labeled image set. Specifically, our pre-trained proposed model's SVC module is fine-tuned with only 1 labeled camera-view image (V) from the training set of the real dataset, denoted as "+F(V)". For comparison, the calibrated CVCS_backbone and CVCS are fine-tuned with one set of multi-view images (V) and one labeled scene-level density map (S), denoted "+F(V+S)".

The results are presented in Table 7. The first 7 methods are calibrated methods that train and test on the *same* single scene (denoted as 'RealSame'). This

**Table 7.** Results on real testing datasets. "Training" column indicates different training methods: "RealSame" means training and testing on the same single real scene; "Synth" means cross-scene training on synthetic dataset and directly testing on the real scenes; "+UDA" means adding unsupervised domain adaptation; "+F(V+S) means finetune the calibrated methods on a set of multi-view images (V) and one corresponding scene-level density map (S); "+F(V)" means finetuning the single-view counting with one labeled camera view image (V) from the training set of real scenes.

| | Model | Training | PETS2009 | | DukeMTMC | | CityStreet | |
|---|---|---|---|---|---|---|---|---|
| | | | MAE | NAE | MAE | NAE | MAE | NAE |
| Calibrated | Dmap_weighted [34] | RealSame | 7.51 | 0.261 | 2.12 | 0.255 | 11.10 | 0.121 |
| | Dect+ReID [53] | RealSame | 9.41 | 0.289 | 2.20 | 0.342 | 27.60 | 0.385 |
| | LateFusion [53] | RealSame | 3.92 | 0.138 | 1.27 | 0.198 | 8.12 | 0.097 |
| | EarlyFusion [53] | RealSame | 5.43 | 0.199 | 1.25 | 0.220 | 8.10 | 0.096 |
| | MVMS [53] | RealSame | 3.49 | 0.124 | 1.03 | **0.170** | 8.01 | 0.096 |
| | 3D [54] | RealSame | 3.15 | **0.113** | 1.37 | 0.244 | 7.54 | **0.091** |
| | CVF [58] | RealSame | **3.08** | - | **0.87** | - | **7.08** | - |
| Calibrated | CVCS_backbone [55] | Synth | 8.05 | 0.257 | 4.19 | 0.913 | 11.57 | 0.156 |
| | CVCS_backbone [55] | Synth+UDA | 5.91 | 0.200 | 3.11 | 0.551 | 10.09 | 0.117 |
| | CVCS_backbone [55] | Synth+F(V+S) | 5.78 | 0.186 | 2.92 | 0.597 | 9.71 | 0.111 |
| | CVCS [55] | Synth | 5.33 | 0.174 | 2.85 | 0.546 | 11.09 | 0.124 |
| | CVCS [55] | Synth+UDA | 5.17 | 0.165 | 2.83 | **0.525** | 9.58 | 0.117 |
| | CVCS [55] | Synth+F(V+S) | **5.06** | **0.164** | **2.81** | 0.567 | **9.13** | **0.108** |
| Calib-free | CF-MVCC-C (ours) | Synth | 14.63 | 0.458 | 5.16 | 0.984 | 48.58 | 0.602 |
| | CF-MVCC-C (ours) | Synth+UDA | 12.76 | 0.398 | 2.65 | 0.498 | 14.89 | 0.176 |
| | CF-MVCC-C (ours) | Synth+F(V) | **4.85** | **0.162** | **1.80** | **0.293** | **8.13** | **0.095** |

can be considered as the upper-bound performance for this experiment. The remaining 9 methods are calibrated and calibration-free methods using domain adaptation. The proposed method trained with Synth+F(V) achieves better performance than other training methods or CVCS [55] with domain adaptation or Synth+F(V+S). Compared to calibrated single-scene models [34,53,54,58], the CF-MVCC-C training with Synth+F(V) still achieves promising performance, and is slightly worse than MVMS and 3D. Note that Synth+F(V) only uses one frame annotated with people during fine-tuning, and does not require camera calibrations during test time. Thus, our method has practical advantage over the calibrated single-scene methods, which require much more annotations and the camera calibrations.

## 5 Conclusion

In this paper, we propose a calibration-free multi-view counting method that fuses the single-view predictions with learned weight maps, which consider both similarity between camera-view pairs and confidence guided by image content

and distance information. The experiments show the proposed method can achieve better performance than other calibration-free baselines. Compared to previous calibrated multi-view methods, our proposed method is more practical for real applications, since our method does not need camera calibrations in the testing stage. The performance can be further improved by pre-training on a synthetic dataset, and applying domain adaptation with a single annotated image. In this case, our fine-tuned calibration-free method outperforms fine-tuned calibrated methods. Our work provides a promising step towards practical multi-view crowd counting, which requires no camera calibrations from the test scene and only one image for fine-tuning the single-view density map regressor.

**Acknowledgements.** This work was supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11212518, CityU 11215820), and by a Strategic Research Grant from City University of Hong Kong (Project No. 7005665).

## References

1. Agarwal, S., et al.: Building Rome in a day. Commun. ACM **54**(10), 105–112 (2011)
2. Ammar Abbas, S., Zisserman, A.: A geometric approach to obtain a bird's eye view from an image. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (2019)
3. Bai, S., He, Z., Qiao, Y., Hu, H., Wu, W., Yan, J.: Adaptive dilated network with self-correction supervision for counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4594–4603 (2020)
4. Bhardwaj, R., Tummala, G.K., Ramalingam, G., Ramjee, R., Sinha, P.: Autocalib: automatic traffic camera calibration at scale. ACM Trans. Sensor Netw. (TOSN) **14**(3–4), 1–27 (2018)
5. von Borstel, M., Kandemir, M., Schmidt, P., Rao, M.K., Rajamani, K., Hamprecht, F.A.: Gaussian process density counting from weak supervision. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 365–380. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_22
6. Chan, A.B., Vasconcelos, N.: Counting people with low-level features and Bayesian regression. IEEE Trans. Image Process. **21**(4), 2160–2177 (2012)
7. DeTone, D., Malisiewicz, T., Rabinovich, A.: Deep image homography estimation. arXiv preprint arXiv:1606.03798 (2016)
8. Dittrich, F., de Oliveira, L.E., Britto Jr, A.S., Koerich, A.L.: People counting in crowded and outdoor scenes using a hybrid multi-camera approach. arXiv preprint arXiv:1704.00326 (2017)
9. Ferryman, J., Shahrokni, A.: Pets 2009: dataset and challenge. In: 2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, pp. 1–6. IEEE (2009)
10. Ge, W., Collins, R.T.: Crowd detection with a multiview sampler. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6315, pp. 324–337. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15555-0_24
11. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. In: Advances in Neural Information Processing Systems (NIPS), pp. 2017–2025 (2015)

12. Jiang, X., et al.: Attention scaling for crowd counting. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020
13. Junior, J.C.S.J., Musse, S.R., Jung, C.R.: Crowd analysis using computer vision techniques. IEEE Signal Process. Mag. **27**(5), 66–77 (2010)
14. Kang, D., Chan, A.: Crowd counting by adaptively fusing predictions from an image pyramid. In: BMVC (2018)
15. Le, H., Liu, F., Zhang, S., Agarwala, A.: Deep homography estimation for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7652–7661 (2020)
16. Lempitsky, V., Zisserman, A.: Learning to count objects in images. In: Advances in Neural Information Processing Systems, pp. 1324–1332 (2010)
17. Li, Y., Zhang, X., Chen, D.: CSRNET: dilated convolutional neural networks for understanding the highly congested scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1091–1100 (2018)
18. Lian, D., Li, J., Zheng, J., Luo, W., Gao, S.: Density map regression guided detection network for RGB-D crowd counting and localization. In: CVPR, pp. 1821–1830 (2019)
19. Liu, L., Chen, J., Wu, H., Li, G., Li, C., Lin, L.: Cross-modal collaborative representation learning and a large-scale RGBT benchmark for crowd counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4823–4833, June 2021
20. Liu, W., Salzmann, M., Fua, P.: Context-aware crowd counting. In: CVPR, pp. 5099–5108 (2019)
21. Liu, X., van de Weijer, J., Bagdanov, A.D.: Exploiting unlabeled data in CNNs by self-supervised learning to rank. IEEE Trans. Pattern Anal. Mach. Intell. **41**(8), 1862–1878 (2019)
22. Liu, X., Yang, J., Ding, W., Wang, T., Wang, Z., Xiong, J.: Adaptive mixture regression network with local counting map for crowd counting. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12369, pp. 241–257. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58586-0_15
23. Liu, Y., Liu, L., Wang, P., Zhang, P., Lei, Y.: Semi-supervised crowd counting via self-training on surrogate tasks. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12360, pp. 242–259. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58555-6_15
24. Lu, E., Xie, W., Zisserman, A.: Class-agnostic counting. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) ACCV 2018. LNCS, vol. 11363, pp. 669–684. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20893-6_42
25. Ma, Z., Wei, X., Hong, X., Gong, Y.: Bayesian loss for crowd count estimation with point supervision, pp. 6141–6150 (2019)
26. Maddalena, L., Petrosino, A., Russo, F.: People counting by learning their appearance in a multi-view camera environment. Pattern Recogn. Lett. **36**, 125–134 (2014)
27. Mishkin, D., Matas, J., Perdoch, M., Lenc, K.: WXBS: wide baseline stereo generalizations. In: British Machine Vision Conference (2015)
28. Nguyen, T., Chen, S.W., Shivakumar, S.S., Taylor, C.J., Kumar, V.: Unsupervised deep homography: a fast and robust homography estimation model. IEEE Robot. Autom. Lett. **3**(3), 2346–2353 (2018)
29. Oñoro-Rubio, D., López-Sastre, R.J.: Towards perspective-free object counting with deep learning. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 615–629. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_38

30. Pritchett, P., Zisserman, A.: Wide baseline stereo matching. In: International Conference on Computer Vision (1998)
31. Ranjan, V., Sharma, U., Nguyen, T., Hoai, M.: Learning to count everything. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3394–3403, June 2021
32. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 17–35. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_2
33. Rocco, I., Cimpoi, M., Arandjelović, R., Torii, A., Pajdla, T., Sivic, J.: Neighbourhood consensus networks. arXiv preprint arXiv:1810.10510 (2018)
34. Ryan, D., Denman, S., Fookes, C., Sridharan, S.: Scene invariant multi camera crowd counting. Pattern Recogn. Lett. **44**(8), 98–112 (2014)
35. Sabzmeydani, P., Mori, G.: Detecting pedestrians by learning shapelet features. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2007)
36. Sam, D.B., Sajjan, N.N., Maurya, H., Radhakrishnan, V.B.: Almost unsupervised learning for dense crowd counting. In: Thirty-Third AAAI Conference on Artificial Intelligence, vol. 33(1), pp. 8868–8875 (2019)
37. Sam, D.B., Surya, S., Babu, R.V.: Switching convolutional neural network for crowd counting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, p. 6 (2017)
38. Shi, M., Yang, Z., Xu, C., Chen, Q.: Revisiting perspective information for efficient crowd counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7279–7288 (2019)
39. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
40. Sindagi, V.A., Patel, V.M.: Generating high-quality crowd density maps using contextual pyramid CNNs. In: IEEE International Conference on Computer Vision (ICCV), pp. 1879–1888. IEEE (2017)
41. Sindagi, V.A., Patel, V.M.: A survey of recent advances in CNN-based single image crowd counting and density estimation. Pattern Recogn. Lett. **107**, 3–16 (2018)
42. Sindagi, V.A., Yasarla, R., Babu, D.S., Babu, R.V., Patel, V.M.: Learning to count in the crowd from limited labeled data. arXiv preprint arXiv:2007.03195 (2020)
43. Song, Q., et al.: Rethinking counting and localization in crowds: a purely point-based framework. arXiv preprint arXiv:2107.12746 (2021)
44. Tang, N., Lin, Y.Y., Weng, M.F., Liao, H.Y.: Cross-camera knowledge transfer for multiview people counting. IEEE Trans. Image Process. **24**(1), 80–93 (2014)
45. Wan, J., Liu, Z., Chan, A.B.: A generalized loss function for crowd counting and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1974–1983, June 2021
46. Wang, Q., Gao, J., et al.: Learning from synthetic data for crowd counting in the wild. In: CVPR, pp. 8198–8207 (2019)
47. Yan, Z., et al.: Perspective-guided convolution networks for crowd counting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 952–961 (2019)
48. Yang, S.D., Su, H.T., Hsu, W.H., Chen, W.C.: Class-agnostic few-shot object counting. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 870–878 (2021)

49. Yang, Y., Li, G., Wu, Z., Su, L., Huang, Q., Sebe, N.: Reverse perspective network for perspective-aware object counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4374–4383 (2020)
50. Yang, Y., Li, G., Wu, Z., Su, L., Huang, Q., Sebe, N.: Weakly-supervised crowd counting learns from sorting rather than locations. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12353, pp. 1–17. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58598-3_1
51. Zhang, C., Li, H., Wang, X., Yang, X.: Cross-scene crowd counting via deep convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 833–841 (2015)
52. Zhang, J., Wang, C., Liu, S., Jia, L., Ye, N., Wang, J., Zhou, J., Sun, J.: Content-aware unsupervised deep homography estimation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12346, pp. 653–669. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_38
53. Zhang, Q., Chan, A.B.: Wide-area crowd counting via ground-plane density maps and multi-view fusion CNNs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8297–8306 (2019)
54. Zhang, Q., Chan, A.B.: 3d crowd counting via multi-view fusion with 3d gaussian kernels. In: AAAI Conference on Artificial Intelligence, pp. 12837–12844 (2020)
55. Zhang, Q., Lin, W., Chan, A.B.: Cross-view cross-scene multi-view crowd counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 557–567 (2021)
56. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: 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 (2016)
57. Zhao, Z., Shi, M., Zhao, X., Li, L.: Active crowd counting with limited supervision. arXiv preprint arXiv:2007.06334 (2020)
58. Zheng, L., Li, Y., Mu, Y.: Learning factorized cross-view fusion for multi-view crowd counting. In: 2021 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE (2021)