This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Video Individual Counting for Moving Drones

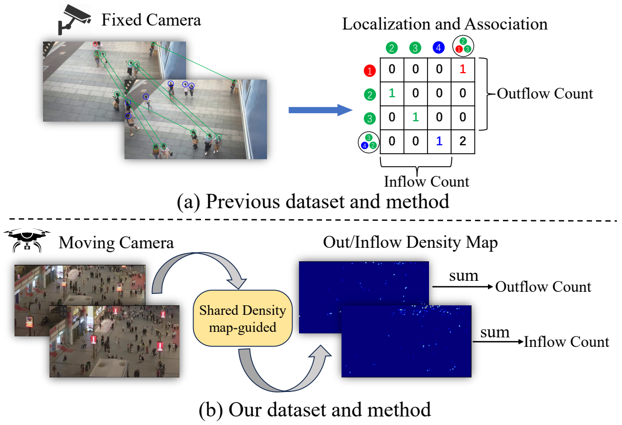
Yaowu Fan[1] Jia Wan[2] Tao Han[3] Antoni B. Chan[4] Andy J. Ma[1] * ✉
[1]Sun Yat-sen University [2]Harbin Institute of Technology (Shenzhen)
[3]Hong Kong University of Science and Technology [4]City University of Hong Kong
{fywyukee, jiawan1998, hantao10200}@gmail.com, abchan@cityu.edu.hk, majh8@mail.sysu.edu.cn

Abstract

Video Individual Counting (VIC) has received increasing attention for its importance in intelligent video surveillance. Existing works are limited in two aspects, i.e., ***dataset*** *and* ***method****. Previous datasets are captured with fixed or rarely moving cameras with relatively sparse individuals, restricting evaluation for a highly varying view and time in crowded scenes. Existing methods rely on localization followed by association or classification, which struggle under dense and dynamic conditions due to inaccurate localization of small targets. To address these issues, we introduce the MovingDroneCrowd Dataset, featuring videos captured by fast-moving drones in crowded scenes under diverse illuminations, shooting heights and angles. We further propose a* ***S****hared* ***D****ensity map-guided* ***Net****work (****SDNet****) using a Depth-wise Cross-Frame Attention (****DCFA****) module to directly estimate shared density maps between consecutive frames, from which the inflow and outflow density maps are derived by subtracting the shared density maps from the global density maps. The inflow density maps across frames are summed up to obtain the number of unique pedestrians in a video. Experiments on our datasets and publicly available ones show the superiority of our method over the state of the arts in highly dynamic and complex crowded scenes. Our dataset and codes have been released publicly*[1].

1. Introduction

Crowd counting is a fundamental task in crowd analysis to estimate the pedestrian density and quantity in images or videos. This task plays an important role in safety monitoring and early warning of stampedes to prevent crowd disasters caused by abnormal congestion [44].



(a) Previous dataset and method

(b) Our dataset and method

Figure 1. Comparison between our dataset/method and existing ones. **Dataset:** Existing datasets are captured by fixed or hardly moving cameras with sparse targets, while our data is collected from high-speed moving drones in crowded scenes. **Method:** Existing methods first localize pedestrians and then perform cross-frame association or classification. They fail on challenging datasets like ours due to the difficulty in accurately localizing pedestrians under crowded and complex scenes. Instead, our shared density map-guided method adopts a more learnable and optimizable approach by first estimating shared density maps via cross-frame attention and then inferring inflow and outflow density maps, leading to better performance under challenging scenarios.

Previous works primarily focus on crowd counting in images from handheld cameras, smartphones, and fixed surveillance cameras [9, 12, 17, 21, 23, 32, 45]. While achieving remarkable progress, these methods are gradually failing to meet the demands of complex and dynamic real-world scenarios. On the one hand, these images are often captured at low heights and cover only limited regions. As a result, the perspective effect causes heads in regions that are far away from the cameras occlude each other, leading to inaccuracies in counting. On the other hand, counting in images provides only the number of pedestrians in a specific location at a given moment. It fails to meet the real-world needs for estimating the number and density of pedestrians over large areas and periods of time, such as in pedestrian streets or crowded squares.

To address the issues caused by ground-based cameras,

* A.J. Ma is also with the Guangdong Province Key Laboratory of Information Security Technology, China, and the Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China.

✉ Corresponding author.

[1] https://github.com/fyw1999/MovingDroneCrowd

existing works [2, 26, 30, 40, 48] collect a series of drone-based datasets. Nevertheless, most of them are image-level or captured from a fixed drone viewpoint, restricting the monitoring of crowdedness within a limited view and time. Although a drone video dataset is introduced in [25], it includes both vehicles and pedestrians, resulting in a relatively low pedestrian density. Moreover, since their videos were collected by drones in suburbs with uniform shooting heights, angles, and lighting, they may not be able to represent complex and crowded real-world scenes.

Besides dataset limitations, accurately counting pedestrians with different identities in a video (a.k.a. video individual counting [14]) remains challenging. The most straightforward idea is to apply multi-object tracking (MOT) techniques [3, 29, 33, 38, 46] and count the tracklets. Since MOT-based methods are typically designed for sparse scenes with large targets, they fail in crowded scenes with low-resolution targets. Recently, several methods [14, 20, 25] have been proposed specifically for this task, which localize persons in each frame and then associate or classify them between two consecutive frames to infer inflow count. Despite these efforts, all methods heavily depend on accurate pedestrian localization, which is unreliable in dense crowds. Poor localization leads to degraded association or classification, resulting in significant counting deviations across videos. Hence, the localization-then-association or localization-then-classification paradigm is fragile in complex environments with dense crowds, particularly when captured by a fast-moving drone. The most related method to ours is [35], which directly predicts inflow and outflow masks and then multiplies them with global density maps to obtain inflow and outflow density maps. However, we argue that directly predicting frame-specific pedestrians from two frames is more difficult. In contrast, our method first estimates the shared density maps between frames and then infers the inflow and outflow density maps.

The dataset and method limitations in existing works are illustrated in Fig 1. To overcome these limitations, we collect a MovingDroneCrowd Dataset and propose a shared density map-guided method for video individual counting. Unlike existing datasets, our dataset specifically focuses on crowded scenes captured by moving drones under diverse and complex conditions, including pedestrian streets, tourist attractions, and squares. It features complex camera motion patterns and a wider variety of light conditions, shooting angles, and shooting heights, making the task of video individual counting highly challenging and existing methods less effective. For methodology, the proposed method is inspired by the observation in image-level crowd counting that density map-based methods yield lower counting errors than localization-based ones in crowded scenes, and by the intuition that identifying shared objects between two sets is easier and more learnable than detecting set-specific ones.

Specifically, we design a Depth-wise Cross-Frame Attention (DCFA) module to learn the respective shared density maps for two adjacent frames, where each shared density map includes the density of pedestrians that appear in both the current and the adjacent frame. The proposed DCFA takes multi-scale features from two consecutive frames as input and computes cross-frame attention across features with different scales. The features of each frame output by the DCFA module are decoded by the shared density map decoder to obtain their respective shared density maps. Finally, outflow and inflow density maps are estimated by subtracting the shared density maps from the global density maps. During testing, unique pedestrians in a video clip are counted by summing the inflow density maps across frames. Our method is weakly supervised, which requires only inflow and outflow labels indicating whether pedestrians enter or exit the view. The contributions of this paper are summarized as follows:

- We collect a video-level individual counting dataset captured by fast-moving drones in various crowded scenes. Compared to prior datasets, our one is with higher crowd density, more complex camera motions, and greater variations in lighting, shooting angles and heights.
- We propose a shared density map-guided VIC method that bypasses the challenging localization step and instead adopts a more learnable manner by first learning shared pedestrian density maps between consecutive frames.
- We design a Depth-wise Cross-Frame Attention (DCFA) module to extract shared density maps, which are then subtracted from the global density maps to obtain accurate inflow density maps.
- Experiments on our dataset and publicly available ones show that the proposed method outperforms the state of the arts in highly dynamic, dense, and complex scenes.

2. Related Works

2.1. Image-level crowd counting

In early works of crowd counting [5, 16, 27], handcrafted features were utilized to regress the number of persons in images. Spatial information is leveraged to improve performance in [18] by learning a mapping between image features and density maps. Nowadays, CNNs or Transformers are used to map the image features to density maps. These works tackle challenges such as perspective effects [31, 42, 43], domain differences [7, 11, 13, 24, 37, 41], or scale variations [8, 15, 36]. Though density map-based methods can provide more accurate counts, they cannot determine the exact coordinates of individuals, especially in regions far away from the camera. To this end, crowd localization is proposed to directly regress the coordinates of each person using neural networks [22, 32]. [6, 10, 19] leverage adjacent frames to enhance counting and localization performance in the target frame. They still count the same person multiple times across different frames, so they are still categorized as image-level crowd counting. Traditional image-level methods can only perform counting within a fixed region at a single time point, whereas our method enables counting over dynamically changing views.

2.2. Video-level crowd counting

Counting pedestrians with different identities over a period of time is more meaningful. We classify this task as video-level crowd counting, and in work [14], it is also defined as Video Individual Counting. Intuitively, MOT techniques [1, 34, 46] offer a potential solution. However, these methods struggle in highly crowded scenes with several occlusions and are ineffective in handling rapid camera movements. Han *et al.* [14] decomposes this task as a pedestrian association problem between two consecutive frames. Liu *et al.* [25] further proposed a weakly-supervised group-level matching method. [20] regress the coordinates of person and then classify them into shared, inflow, and outflow person. However, these methods require localizing individuals in each frame, followed by association or classification, where localization errors can severely affect accuracy. Wan *et al.* [35] proposed a density map-based method that predicts inflow and outflow masks and then multiplies the masks with global density maps to obtain inflow and outflow density maps, but this process is difficult to learn and optimize. In contrast, our method formulates this task in a more learnable manner by first estimating the shared density maps and then inferring inflow and outflow density maps.

2.3. Drone-based crowd counting datasets

Currently, datasets for crowd counting from a drone perspective remain relatively scarce. Bahmanyar *et al.*[2] collected an aerial crowd dataset using DSLR cameras mounted on a helicopter. The datasets proposed in [26, 30] are formed in RGB and thermal pairs captured by drones. However, these datasets are all image-level, meaning they only allow counting the number of persons at a specific moment within a fixed view. The multi-object tracking dataset [48] for drone perspectives contains video clips with dense crowds. However, during annotation, these crowded regions were entirely ignored. Luo *et al.* [39, 40] released a video-level drone crowd dataset, but the video clips were captured by hovering drones, with each clip covering only a fixed field of view, similar to image-level datasets. The dataset UAVVIC [25] collects video clips captured by drones in relatively simple and uniform conditions. It includes not only pedestrians but also a large number of vehicles, leading to a lower pedestrian density. Compared to them, our dataset is captured by fast-moving drones under more complex conditions, including denser crowds, more challenging lighting, and more diverse flying altitudes and camera angles.

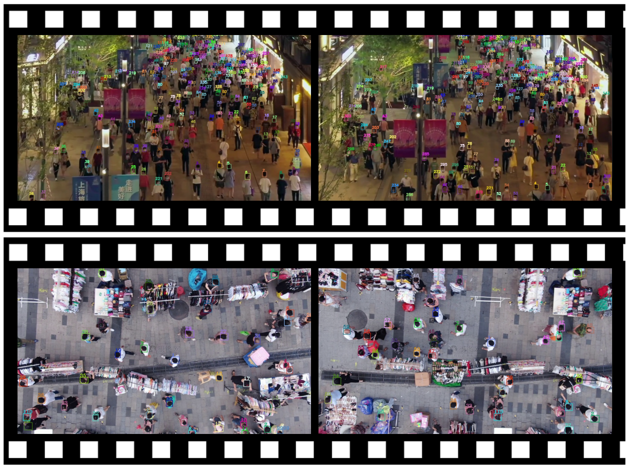


Figure 2. Two example clips from our dataset. The head bounding boxes and ID annotations are presented in each frame. The diverse light conditions, shooting angles, heights and densely packed pedestrians make it a highly challenging dataset. Only two frames per clip are shown to save space and provide a clearer presentation. Zoom in to see more details.

3. MovingDroneCrowd Dataset

To promote practical crowd counting, we introduce MovingDroneCrowd — a video-level dataset specifically designed for dense pedestrian scenes captured by moving drones under complex conditions. Notably, our dataset provides precise bounding box and ID labels for each person across frames, making it suitable for multiple pedestrian tracking from drone perspective in complex scenarios. We detail the dataset and compare it with existing ones below.

Data Processing and Scale: Due to strict regulations on drone flights, we obtained raw drone videos from the internet using keywords like “aerial”, “drone”, “pedestrian flow”, and “pedestrian street”. The raw videos were first segmented into clips covering entire locations. To reduce redundancy, each clip was downsampled to 1fps, 3fps, or 6fps based on drone speed. Some drone videos have very narrow shooting angles, making pedestrians farther from the camera appear extremely blurry. To alleviate the difficulty of annotation, these clips are cropped until the pedestrians within the shooting range can be identified by annotators. Finally, 89 clips (4940 frames) with resolutions of 720p, 1080p, 2K, and 4K are obtained.

Annotation: The annotation process was carried out by 10 well-trained annotators using the labeling tool DarkLabel [2] and took a month to complete. Each annotator was asked to label bounding boxes that tightly enclose pedestrians’ heads and assign unique IDs to different individuals in an entire video. Once the annotations were completed, the clips were reassigned to different annotators for error checking and revision. Finally, **325541** head bounding boxes and

[2] https://github.com/darkpgmr/DarkLabel

Dataset	Perspective	Moving	MFR	MPR	MPPF	Light	Height	Angle	IDs
CroHD	Surveillance	✗	0	0	0	day&night	Fixed	Fixed	✓
VSCrowd	Surveillance	✗	0	0	0	day&night	Fixed	Fixed	✓
DroneCrowd	Drone	✗	0	0	0	day&night	Fixed	Fixed	✓
UAVVIC	Drone	✓✗	51%	39%	32	day	∼ 20m	∼ 90°	✗
MovingDroneCrowd	Drone	✓	100%	100%	66	day&night	∼ 3-20m	∼ 45-90°	✓

Table 1. Comparison of recent video datasets. MFR represents the proportion of moving frames to all frames, MPR denots the proportion of pedestrians in moving frames to the total number of pedestrians, and MPPF is the average number of pedestrians per frame in moving frames. Our dataset is captured in highly dynamic and complex scenarios, making it the most challenging.

16153 tracklets were obtained. Fig. 2 displays two video clips from our dataset, with head bounding boxes and ID labels, illustrating their diverse lighting conditions, shooting angles, and heights, as well as higher crowd density. These attributes make our dataset more challenging and distinguish it from previous datasets.

Dataset Partition: The dataset is split into training (70%), testing (20%), and validation (10%) sets at the **scene level**, ensuring no overlapping scenes. This setup places higher demands on the algorithm's generalization ability. In addition, the data split process ensures that each set contains diverse data.

Comparison: As shown in Table 1, we compare our dataset against recent video datasets. Compared with the previous drone dataset [25], ours specifically focuses on dense pedestrians and has diverse light conditions, shooting angles, and shooting heights, as well as more complex motion patterns. Fig. 3 shows the pedestrian count distribution per frame of moving data between our dataset and UAVVIC. Because UAVVIC's test set is unavailable, we only include the comparative results of the training set. Based on the statistical results, most moving frames in UAVVIC contain fewer than 50 pedestrians, whereas our dataset exhibits a higher proportion of frames in the ranges of $50-99$ and $100-149$, which correspond to typical crowded scenarios. Additionally, our training set has frames distributed in the more crowded range of $250-349$, and our test set includes some extremely crowded moving frames with pedestrian count in the range of $350-549$, whereas UAVVIC lacks. In summary, our dataset offers a more diverse and challenging pedestrian count distribution.

4. Methodology

4.1. Problem Formulation

Formally, the training set $\mathcal{V}_t = \{\mathbf{V}_i, \mathbf{L}_i\}_{i=1}^{N_t}$ consists of N_t video clips and annotations, where the i^{th} video $\mathbf{V}_i = \{V_j\}_{j=1}^{n_i}$ has n_i frames, and $\mathbf{L}_i = \{P_j, ID_j\}_{j=1}^{n_i}$ provides the coordinates and identities of the person in each frame of video $\mathbf{V}_i$. Notably, our method is weakly supervised and does not require ID labels, making it applicable even when only inflow I_j and outflow labels O_j that indicate pedestrian entries and exits are provided.

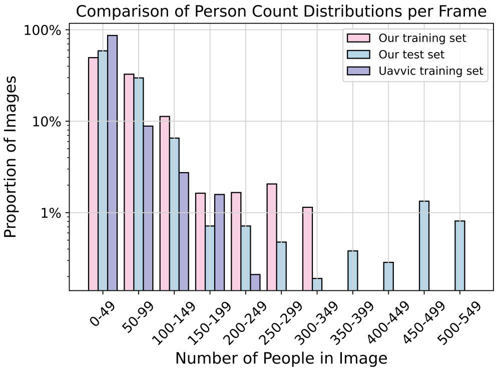


Figure 3. Comparison of pedestrian count distribution per frame between our dataset and UAVVIC.

For consecutive frames V_j and $V_{j+\delta}$ (with a fixed interval δ), our method estimates the outflow density map $\hat{\mathbf{D}}_j^{out}$ for V_j and inflow density map $\hat{\mathbf{D}}_{j+\delta}^{in}$ for $V_{j+\delta}$. The sum of $\hat{\mathbf{D}}_j^{out}$ gives the number of pedestrians in V_j who exit the view of $V_{j+\delta}$, while the sum of $\hat{\mathbf{D}}_{j+\delta}^{in}$ represents the number of pedestrians entering the view of $V_{j+\delta}$. Consequently, the total number of unique pedestrians in video $\mathbf{V}_i$ can be computed as:

$$M(\mathbf{V}_i) \approx M(V_1) + \sum_{k=1}^{(n_i/\delta)-1} \text{sum}(\hat{\mathbf{D}}_{1+k\times\delta}^{in}), \quad (1)$$

where $M(V_1)$ represents the number of persons in the first frame, and $\hat{\mathbf{D}}_{1+k\times\delta}^{in}$ is the inflow density map of frame $V_{1+k\times\delta}$ relative to frame $V_{1+(k-1)\times\delta}$.

4.2. Overall Framework

To achieve the goal mentioned above, *i.e.*, estimating the inflow density map for each frame, we first estimate the shared density map, as illustrated in Fig. 4. Specifically, given two consecutive frames V_j and $V_{j+\delta}$, we first extract their multi-scale features $\mathcal{F}_j$ and $\mathcal{F}_{j+\delta}$. Then, the extracted multi-scale features pass through our proposed Depth-wise Cross-Frame Attention module to obtain shared features $\mathbf{F}_j^s$ and $\mathbf{F}_{j+\delta}^s$ for each frame. The shared density map decoder $\mathcal{D}_s$ maps the shared features to shared density maps $\hat{\mathbf{D}}_j^s$ and $\hat{\mathbf{D}}_{j+\delta}^s$. Meanwhile, the multi-scale features of each frame are fused and then mapped to global density maps $\hat{\mathbf{D}}_j^g$ and $\hat{\mathbf{D}}_{j+\delta}^g$ through the global density map decoder $\mathcal{D}_g$. Finally, the differences between the global and shared density maps are used to derive the outflow density map $\hat{\mathbf{D}}_j^{out}$ for V_j and inflow density map $\hat{\mathbf{D}}_{j+\delta}^{in}$ for $V_{j+\delta}$.

4.3. Depth-wise Cross-frame Attention

To learn the shared and global features, we first extract multi-scale features. Given sampled consecutive frames V_j and $V_{j+\delta}$, a shared-weight backbone network and a Feature Pyramid Network extract multi-scale features $\mathcal{F}_j$ and $\mathcal{F}_{j+\delta}$,

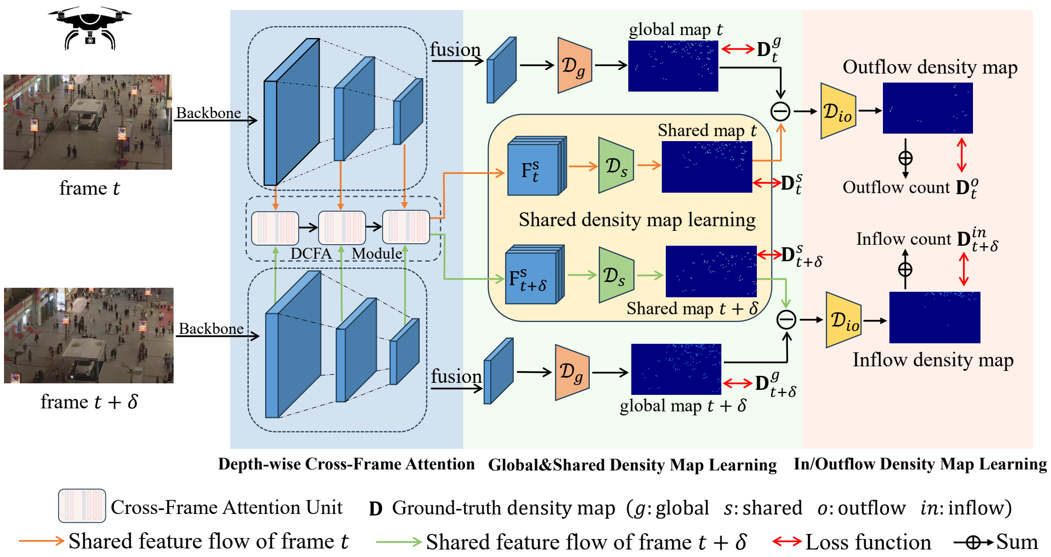

Backbone
frame t
Backbone
frame $t+\delta$
fusion
fusion
DCFA Module
global map t
$\mathbf{D}_t^g$
$\mathcal{D}_g$
Shared map t
$\mathbf{F}_t^s$
$\mathcal{D}_s$
$\mathbf{D}_t^s$
Shared density map learning
$\mathbf{F}_{t+\delta}^s$
$\mathcal{D}_s$
$\mathbf{D}_{t+\delta}^s$
Shared map $t+\delta$
$\mathcal{D}_g$
global map $t+\delta$
$\mathbf{D}_{t+\delta}^g$
Outflow density map
$\mathcal{D}_{io}$
Outflow count $\mathbf{D}_t^o$
Inflow count $\mathbf{D}_{t+\delta}^{in}$
$\mathcal{D}_{io}$
Inflow density map
Depth-wise Cross-Frame Attention
Global&Shared Density Map Learning
In/Outflow Density Map Learning
Cross-Frame Attention Unit
D Ground-truth density map (g: global s: shared o: outflow in: inflow)
Shared feature flow of frame t
Shared feature flow of frame $t+\delta$
Loss function
Sum

Figure 4. The pipeline of our shared density map-guided VIC method. First, multi-scale features are extracted using a shared-weight CNN and FPN. The DCFA module computes cross-frame attention across features at all scales to obtain the shared features, while global features are obtained by fusing the multi-scale features. Then, a global decoder and a shared decoder generate global and shared density maps for each frame. Finally, the inflow-outflow decoder processes the difference between global and shared density maps to produce the outflow density map for the first frame and the inflow density map for the second frame. During testing, simply accumulating the sum of the inflow density maps across all frames yields the total number of unique pedestrians in the entire video.

where $\mathcal{F}_j = \{\mathbf{F}_j^i\}_{i=1}^{N_f}$, and N_f is the number of multi-scale feature levels. The dimension of the i-th scale feature $\mathbf{F}_j^i$ is $C \times H/2^{(i+1)} \times W/2^{(i+1)}$. Here, H and W are the height and width of input image, respectively, and C is the number of feature channels.

With the extracted multi-scale features, our designed Depth-wise Cross-Frame Attention (DCFA) module is used to learn shared features for each frame. The details of our DCFA are illustrated in Fig. 5. DCFA consists of N_u cross-frame attention units, each containing N_b cross-frame attention blocks. The number of units in DCFA corresponds to the number of scale levels in the multi-scale features. When computing the shared feature of frame V_j, the first cross-frame attention unit directly takes $\mathbf{F}_j^1$ as input, while for i^{th} unit ($i > 1$), the i^{th} scale feature $\mathbf{F}_j^i$ of frame V_j is first fused with the output $\hat{\mathbf{F}}_j^{i-1}$ of the $(i-1)^{\text{th}}$ unit:

$$\tilde{\mathbf{F}}_j^i = \text{Fusion}(\hat{\mathbf{F}}_j^{i-1}, \mathbf{F}_j^i). \tag{2}$$

The process of computing the output of the i^{th} unit is then performed as follows:

$$\begin{aligned} \mathbf{F}_j^{i\prime} &= \text{MSA}(\text{LN}(\tilde{\mathbf{F}}_j^i)) + \tilde{\mathbf{F}}_j^i, \\ \mathbf{F}_j^{i\prime\prime} &= \text{MCA}(\text{LN}(\mathbf{F}_j^{i\prime}), \mathbf{F}_{j+\delta}^i) + \mathbf{F}_j^{i\prime}, \\ \hat{\mathbf{F}}_j^i &= \text{MLP}(\text{LN}(\mathbf{F}_j^{i\prime\prime})) + \mathbf{F}_j^{i\prime\prime}, \end{aligned} \tag{3}$$

where LN denotes layer normalization, MSA represents multi-head self-attention layer, and MCA refers to multi-head cross-attention layer. The computation of the MCA layer in Eq. 3 indicates that the multi-scale features from frames V_j and $V_{j+\delta}$ are set as the *query* and *key*, respectively. This process can be formulated as follows :

$$\begin{aligned} Q_h = \mathbf{F}_j^{i\prime} W_h^Q,\ \ K_h &= \mathbf{F}_{j+\delta}^i W_h^K,\ \ V_h = \mathbf{F}_{j+\delta}^i W_h^V, \\ Head_h &= \text{Softmax}(\frac{Q_h K_h^T}{\sqrt{D}}) V_h, \\ \mathbf{F}_j^{i\prime\prime} &= \text{Concat}(Head_1, ..., Head_H) + \mathbf{F}_j^{i\prime}, \end{aligned} \tag{4}$$

where W_h^Q, W_h^K and W_h^V are learnable projection matrices. Here, h represents the h^{th} dependent head, and the final output is obtained by concatenating the outputs of all heads.

This process is repeated iteratively until the final cross-frame attention unit outputs $\hat{\mathbf{F}}_j^{N_u}$, serving as the shared feature $\mathbf{F}_j^s$ of V_j. Similarly, swapping the roles of $\mathbf{F}_j^i$ and $\mathbf{F}_{j+\delta}^i$, i.e. setting $\mathbf{F}_{j+\delta}^i$ as the *query* and $\mathbf{F}_j^i$ as the *key* and *value*, yields the shared feature $\mathbf{F}_{j+\delta}^s$ for frame $V_{j+\delta}$. The DCFA module effectively integrates multi-scale features and captures rich cross-frame information, thereby learning features that retain only shared pedestrian information between the consecutive frames.

4.4. Inflow/Outflow Density Map Learning

To derive the inflow and outflow density maps, shared and global density maps for frames V_j and $V_{j+\delta}$ are first decoded:

$$\hat{\mathbf{D}}_j^g = \mathcal{D}_g(\mathbf{F}_j^g), \quad \hat{\mathbf{D}}_j^s = \mathcal{D}_s(\mathbf{F}_j^s), \tag{5}$$

where $\mathcal{D}_g$ and $\mathcal{D}_s$ denote global and shared density map decoders, respectively. They have identical architectures comprising of alternating convolutional layers and upsampling operations to progressively restore the resolution to match

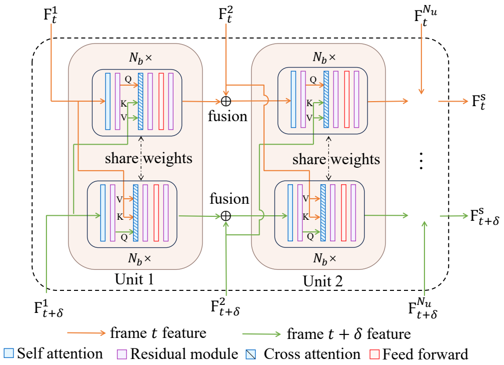


Figure 5. The details of our proposed DCFA module. It contains N_u cross-frame attention units, each comprising N_b cross-frame blocks. The number of units matches the multi-scale feature levels. For the i^{th} unit, cross-frame attention is computed using the fused feature of the first frame's feature at i^{th} scale and the output of the $(i-1)^{\text{th}}$ unit as the *query* and the second frame's feature at i^{th} scale level as *key* and *value*. The final unit's output is the shared feature of the first frame. Swapping the roles of the two frames, yields the shared feature of the second frame.

the input image size. Here, $\mathbf{F}_j^g$ is the global feature of V_j, obtained by directly fusing the multi-scale features in $\mathcal{F}_j$.

The global density maps contain the densities of all pedestrians in each frame, while the shared density maps only include densities for pedestrians appearing in both frames. Consequently, the outflow and inflow density maps can be obtained from the difference between the global and shared density maps:

$$\begin{aligned}\hat{\mathbf{D}}_j^o &= \mathcal{D}_{io}(\hat{\mathbf{D}}_j^g - \hat{\mathbf{D}}_j^s),\\ \hat{\mathbf{D}}_{j+\delta}^{in} &= \mathcal{D}_{io}(\hat{\mathbf{D}}_{j+\delta}^g - \hat{\mathbf{D}}_{j+\delta}^s),\end{aligned} \tag{6}$$

where $\mathcal{D}_{io}$ is the inflow-outflow decoder that is composed of convolutional layers. Obviously, the outflow density map contains the density of pedestrians appearing only in frame V_j, while the inflow density map contains the density of those appearing only in frame $V_{j+\delta}$. During testing, summing the inflow density maps of all frames yields the total number of pedestrians in the video.

Our framework is trained with four MAE losses: global $\mathcal{L}_g$, shared $\mathcal{L}_s$, outflow $\mathcal{L}_o$, and inflow $\mathcal{L}_{in}$ density map loss. These losses are computed as follows:

$$\begin{aligned}\mathcal{L}_g &= \frac{1}{2N}\sum_{i=1}^{2N}||\hat{\mathbf{D}}_i^g - \mathbf{D}_i^g||,\ \mathcal{L}_s = \frac{1}{2N}\sum_{i=1}^{2N}||\hat{\mathbf{D}}_i^s - \mathbf{D}_i^s||,\\ \mathcal{L}_o &= \frac{1}{N}\sum_{i=1}^{N}||\hat{\mathbf{D}}_{2i-1}^o - \mathbf{D}_{2i-1}^o||,\ \mathcal{L}_{in} = \frac{1}{N}\sum_{i=1}^{N}||\hat{\mathbf{D}}_{2i}^{in} - \mathbf{D}_{2i}^{in}||,\end{aligned} \tag{7}$$

where N is the number of image pairs in the training batch. $\mathbf{D}^g$, $\mathbf{D}^s$, $\mathbf{D}^o$, and $\mathbf{D}^{in}$ are ground-truth global, shared, outflow, and inflow density maps, respectively. Note that the ground-truth density maps can be generated using either fully supervised labels (IDs) or weakly supervised labels (inflow and outflow annotations).

5. Experiments

Due to space limitations, please refer to the supplementary materials for more details on implementation details.

5.1. Datasets

Datasets UAVVIC and our MovingDroneCrowd are used for evaluation. A detailed description and comparison of these two datasets have been introduced above.

5.2. Evaluation Metrics

Similar to image-level crowd counting, MAE and RMSE are used for evaluation, but they are computed at the video level. Additionally, we also adopt the metric WRAE, MIAE, and MOAE defined in [14]. WRAE (Weighted Relative Absolute Errors) accounts for the impact of frame counts in different videos when computing relative errors. MIAE and MIOE measure the prediction quality of inflow and outflow, respectively. Please refer to [14] and its Supplementary for details.

5.3. Comparison with State of the Arts

Comparison Methods: To demonstrate the superiority of our method, we compare it against a diverse range of related works. In addition to algorithms specifically designed for VIC, we also include other relevant approaches, such as multiple object tracking and cross-line crowd counting.

Results on MovingDroneCrowd: Table 2 compares our method with other approaches on our dataset MovingDroneCrowd. Our approach significantly outperforms others, reducing MAE and RMSE by 37% and 47%, respectively, compared to the latest approach CGNet. For a more in-depth and detailed analysis, we divide the test scenes by pedestrian density and evaluate MAE under different density levels. As pedestrian density increases, other methods degrade sharply, while our method consistently maintains reasonable performance. The MOT-based methods completely fail in high-density scenes due to their reliance on individual detection and global identity association, which becomes infeasible in our dataset, including complex scenes with severe occlusion and rapid camera movements. VIC methods alleviate some issues but still rely on localization and cross-frame association, leading to unsatisfactory performance in highly crowded scenes. Density-based method FMDC [35] performs poorly despite avoiding localization and association, as directly predicting inflow and outflow masks is highly challenging. In contrast, our method first infers the more learnable shared density maps, and then derives the inflow and outflow maps, allowing it to achieve satisfying results even in complex and crowded scenes.

Results on UAVVIC: We also conduct comparative experiments on the drone video dataset UAVVIC. Since its test set has not been released, comparisons are performed on the validation set. The results in Table 3 show that our

Method	Venue	ID	MAE↓	RMSE↓	WRAE↓	MIAE↓	MIOE↓	MAE on four different density levels			
								D0	D1	D2	D3
ByteTrack[46]	ECCV'22	✗	153.17	227.62	63.82	13.25	11.22	83.38	24.00	325.00	441.33
BoT-SORT[1]	arxiv'22	✓	150.61	223.46	62.53	13.11	11.22	82.46	**22.00**	327.00	430.00
OC-SORT[4]	CVPR'23	✗	203.56	276.84	87.75	10.90	13.63	101.46	232.00	405.00	569.33
DiffMOT[28]	CVPR'24	✓	229.17	450.86	71.27	23.01	21.41	45.85	292.00	952.00	761.67
DRNet[14]	CVPR'22	✓	81.14	126.34	33.36	5.64	**5.09**	28.73	129.88	217.13	246.69
CGNet[25]	CVPR'24	✗	66.06	110.36	29.16	-	-	25.92	111.00	144.00	199.00
LOI[47]	ECCV'16	✓	241.77	337.90	99.63	-	-	110.13	294.46	467.57	719.33
FMDC[35]	WACV'24	✗	120.31	183.57	48.82	8.21	6.40	61.66	75.71	54.92	411.09
Ours	-	✗	**41.00** ↓ 37.8%	**58.34** ↓ 47.1%	**19.32** ↓ 33.7%	**5.50**	6.39	**23.71**	79.77	**41.21** ↓ 24.9%	**102.88** ↓ 48.3%

Table 2. Performance comparison on the MovingDroneCrowd dataset. D0 – D3 respectively denote four pedestrian density ranges: [0, 150), [150, 300), [300, 450), ≥ 450. **Bold** indicates the best result, underline denotes the second-best, and red shows the improvement of our method over the second-best. The performance advantage of our method becomes even more pronounced as crowd density increases.

Method	Venue	Overall					Static		Dynamic	
		MAE↓	RMSE↓	WRAE↓	MIAE↓	MOAE↓	MAE↓	RMSE↓	MSE↓	RMSE ↓
ByteTrack[46]	ECCV'22	14.19	21.51	68.92	**1.77**	**2.09**	9.40	10.21	15.69	23.98
OC-SORT[4]	CVPR'23	18.81	35.42	71.01	2.42	3.06	7.20	7.77	22.44	40.34
LOI [47]	ECCV'16	21.70	38.21	99.00	-	-	11.12	11.59	25.01	43.29
CGNet[25]	CVPR'24	24.95	52.57	83.82	-	-	6.80	8.22	30.62	60.05
Ours	-	**6.37** ↓ 55%	**11.01** ↓ 48.8%	**46.01** ↓ 33.2%	1.81	2.18	**3.30** ↓ 51.5%	**4.12** ↓ 47%	**7.33** ↓ 53.3%	**12.40** ↓ 48.3%

Table 3. Performance comparison on validation set of UAVVIC. The results shows that our method consistently achieves the best results across overall, static, and dynamic scenes, demonstrating its effectiveness in both dynamic and sparse scenarios.

Ablation	Setting		MAE↓	RMSE↓	WRAE↓	MIAE↓	MIOE↓
Backbone	VGG	w/o PE	**41.00**	**58.34**	**19.32**	**5.50**	**6.39**
		w/ PE	66.64	102.66	37.73	7.80	9.37
	ViT	w/o PE	98.56	142.83	48.50	7.84	8.07
		w/ PE	51.76	66.99	24.90	9.21	8.10
Cross-Frame	DCFA		**41.00**	**58.34**	**19.32**	**5.50**	6.39
	SCFA		70.42	90.13	44.71	5.80	**6.02**
Inflow Learning	Direct		65.64	99.34	47.12	6.41	7.63
	SDNet		**41.00**	**58.34**	**19.32**	**5.50**	**6.39**

Table 4. Ablation study for our method. "Direct" represents directly learning the inflow density map rather than first learning shared density map.

method achieves the best overall performance, demonstrating that our method not only handles dense scenes effectively but also performs well in sparse scenes. UAVVIC contains both static and dynamic drone videos, so we conduct separate tests in both scenarios to ensure a more comprehensive analysis. As shown in Table 3, the performance of other methods declines significantly in dynamic scenes compared to their performance in static scenes, whereas our method achieves consistently strong results in both settings. This indicates that other methods struggle to handle dynamic scenes with complex motion patterns, while our method performs effectively.

5.4. Ablation Studies

Effect of Backbone: In our method, image features can be extracted either by CNN or Transformer. Therefore, we first investigate the impact of the backbone. As shown in the first row of Table 4, using the VGG-16 backbone yields the best performance. This suggests that CNN can provide richer local details for pixel-level tasks such as counting.

Effect of Depth-wise Cross-Frame Attention: To invalidate the effectiveness of our proposed DCFA module, we directly use the global features $\mathbf{F}_j^g$ and $\mathbf{F}_{j+\delta}^g$ to compute the cross-frame attention, which we refer to as Shallow-wise Cross-Frame Attention (SCFA). To ensure a fairer comparison, we adjust the hyperparameters in SCFA to ensure its number of parameters is equal to that of DCFA. The results in Table 4 show that DCFA achieves superior performance, as it effectively integrates multi-scale features while learning shared pedestrian information across adjacent frames.

Effect of Position Embedding in DCFA: The experimental results in Table 4 show that positional encoding has distinct effects when using different backbones. Specifically, when using CNN as the backbone, incorporating positional encoding in DCFA leads to a decrease in final performance.

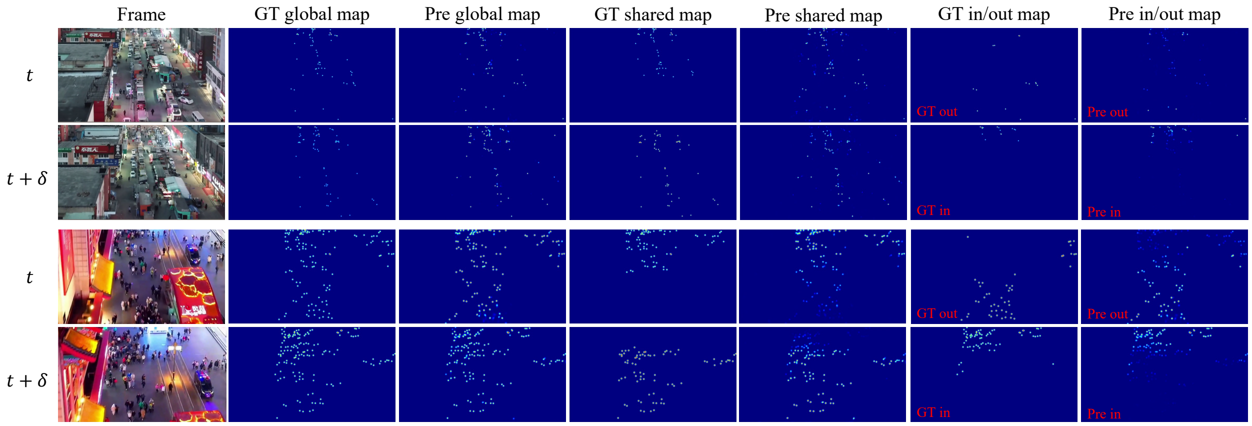


Figure 6. The visualization results of our method on MovingDroneCrowd. It presents the results of two consecutive frames. In addition to the global density map for each frame, the first frame includes its shared density map and outflow density map relative to the second frame, while the second frame includes its shared density map and inflow density map relative to the first frame.

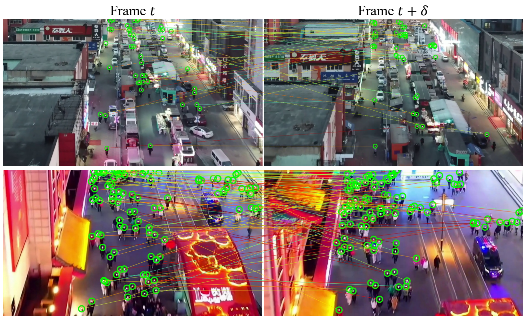


Figure 7. The visualization of CGNet on MovingDroneCrowd. There are numerous localization errors in dense scenes, and the cross-frame association are almost entirely incorrect.

However, with a Transformer backbone, adding positional encoding significantly enhances the counting performance. This is because CNN inherently encodes positional information, and adding extra positional encoding may disrupt the semantic integrity of CNN features. In contrast, Transformer features rely on positional encoding to specify the location of each pixel.

Effect of Learning Strategy: Our method first predicts the shared density maps, from which the outflow and inflow maps are derived by subtraction from the global density map. To validate the effectiveness of this strategy, we conduct an ablation study where the output of DCFA is decoded and then directly supervised by the ground-truth outflow and inflow density maps, i.e. learning them directly instead of first predicting the shared density map. As shown in the seventh row of Table 4, directly learning the inflow density map leads to a significant drop in final performance. This suggests that learning shared information between two frames is easier than learning the private information of each frame, further validating the rationality behind our approach.

5.5. Qualitative Results

Fig. 6 illustrates the visual results of our method on examples of MovingDroneCrowd. The inflow and outflow density maps reflect pedestrian entries and exits within the field of view. Although some erroneous responses exist, their values are effectively suppressed. Fig. 7 presents the visual results of CGNet on the same image pairs. Significant errors are observed in both localization and association, with the association being almost entirely incorrect. This suggests that previous localization and association-based methods struggle to handle dynamic and dense scenes effectively.

6. Conclusion

This paper explores a flexible approach to counting unique individuals over a large area in a period of time, specifically in videos captured by moving drones. Due to the lack of relevant datasets and effective algorithms, we introduce MovingDroneCrowd, a challenging video-level dataset captured by moving drones in crowded scenes with diverse lighting, altitudes, angles, and complex motion patterns. These factors make previous methods ineffective. Therefore, we propose a shared density map-guided algorithm for video individual counting that first estimates the shared density maps. Then, inflow and outflow density maps are obtained from the differences between global and shared density maps. Experiments on both our and previous benchmarks demonstrate that our method effectively handles high-density and dynamic scenes while also achieving excellent results in static and sparse scenarios.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (U22A2095, 62276281, 62406090), the Guangdong Basic and Applied Basic Research Foundation (2024A1515011882), and the Jiangsu Science and Technology Major Program (BG2024041).

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 3, 7

[2] Reza Bahmanyar, Elenora Vig, and Peter Reinartz. Mrcnet: Crowd counting and density map estimation in aerial and ground imagery. *arXiv preprint arXiv:1909.12743*, 2019. 2, 3

[3] Jiarui Cai, Mingze Xu, Wei Li, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Memot: Multi-object tracking with memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8090–8100, 2022. 2

[4] Jinkun Cao, Jiangmiao Pang, Xinshuo Weng, Rawal Khirodkar, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9686–9696, 2023. 7

[5] Antoni B. Chan and Nuno Vasconcelos. Bayesian poisson regression for crowd counting. In *2009 IEEE 12th International Conference on Computer Vision*, pages 545–551, 2009. 2

[6] Li Dong, Haijun Zhang, Jianghong Ma, Xiaofei Xu, Yimin Yang, and Q. M. Jonathan Wu. Clrnet: A cross locality relation network for crowd counting in videos. *IEEE Transactions on Neural Networks and Learning Systems*, 35(5): 6408–6422, 2024. 2

[7] Zhipeng Du, Jiankang Deng, and Miaojing Shi. Domain-general crowd counting in unseen scenarios. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(1):561–570, 2023. 2

[8] Zhipeng Du, Miaojing Shi, Jiankang Deng, and Stefanos Zafeiriou. Redesigning multi-scale neural network for crowd counting. *IEEE Transactions on Image Processing*, 32: 3664–3678, 2023. 2

[9] Yaowu Fan, Jia Wan, and Andy J. Ma. Learning crowd scale and distribution for weakly supervised crowd counting and localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(1):713–727, 2025. 1

[10] Yanyan Fang, Biyun Zhan, Wandi Cai, Shenghua Gao, and Bo Hu. Locality-constrained spatial transformer network for video crowd counting. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 814–819, 2019. 2

[11] Junyu Gao, Tao Han, Yuan Yuan, and Qi Wang. Domain-adaptive crowd counting via high-quality image translation and density reconstruction. *IEEE Transactions on Neural Networks and Learning Systems*, 34(8):4803–4815, 2023. 2

[12] Mingyue Guo, Li Yuan, Zhaoyi Yan, Binghui Chen, Yaowei Wang, and Qixiang Ye. Regressor-segmenter mutual prompt learning for crowd counting. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28380–28389, 2024. 1

[13] Tao Han, Junyu Gao, Yuan Yuan, and Qi Wang. Focus on semantic consistency for cross-domain crowd understanding. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1848–1852, 2020. 2

[14] Tao Han, Lei Bai, Junyu Gao, Qi Wang, and Wanli Ouyang. Dr.vic: Decomposition and reasoning for video individual counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3083–3092, 2022. 2, 3, 6, 7

[15] Tao Han, Lei Bai, Lingbo Liu, and Wanli Ouyang. Steerer: Resolving scale variations for counting and localization via selective inheritance learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 21848–21859, 2023. 2

[16] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[17] Xiaoheng Jiang, Li Zhang, Mingliang Xu, Tianzhu Zhang, Pei Lv, Bing Zhou, Xin Yang, and Yanwei Pang. Attention scaling for crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[18] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2010. 2

[19] Haopeng Li, Lingbo Liu, Kunlin Yang, Shinan Liu, Junyu Gao, Bin Zhao, Rui Zhang, and Jun Hou. Video crowd localization with multifocus gaussian neighborhood attention and a large-scale benchmark. *IEEE Transactions on Image Processing*, 31:6032–6047, 2022. 2

[20] Rui Li, Yishu Liu, Huafeng Li, Jinxing Li, and Guangming Lu. Prototype-guided dual-transformer reasoning for video individual counting. page 10258–10267, 2024. 2, 3

[21] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[22] Dingkang Liang, Wei Xu, and Xiang Bai. An end-to-end transformer model for crowd localization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 38–54, 2022. 2

[23] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[24] Weizhe Liu, Nikita Durasov, and Pascal Fua. Leveraging self-supervision for cross-domain crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5341–5352, 2022. 2

[25] Xinyan Liu, Guorong Li, Yuankai Qi, Ziheng Yan, Zhenjun Han, Anton van den Hengel, Ming-Hsuan Yang, and Qingming Huang. Weakly supervised video individual counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19228–19237, 2024. 2, 3, 4, 7

[26] Zhihao Liu, Zhijian He, Lujia Wang, Wenguan Wang, Yixuan Yuan, Dingwen Zhang, Jinglin Zhang, Pengfei Zhu,

Luc Van Gool, Junwei Han, Steven Hoi, Qinghua Hu, Ming Liu, Junwen Pan, Baoqun Yin, Binyu Zhang, Chengxin Liu, Ding Ding, Dingkang Liang, Guanchen Ding, Hao Lu, Hui Lin, Jingyuan Chen, Jiong Li, Liang Liu, Lin Zhou, Min Shi, Qianqian Yang, Qing He, Sifan Peng, Wei Xu, Wenwei Han, Xiang Bai, Xiwu Chen, Yabin Wang, Yinfeng Xia, Yiran Tao, Zhenzhong Chen, and Zhiguo Cao. Visdrone-cc2021: The vision meets drone crowd counting challenge results. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 2830–2838, 2021. 2, 3

[27] D.G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150–1157, 1999. 2

[28] Weiyi Lv, Yuhang Huang, Ning Zhang, Ruei-Sung Lin, Mei Han, and Dan Zeng. Diffmot: A real-time diffusion-based multiple object tracker with non-linear prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19321–19330, 2024. 7

[29] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixé, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8844–8854, 2022. 2

[30] Tao Peng, Qing Li, and Pengfei Zhu. Rgb-t crowd counting from drone: A benchmark and mmccn network. In *Computer Vision – ACCV 2020*, pages 497–513, 2021. 2, 3

[31] Miaojing Shi, Zhaohui Yang, Chao Xu, and Qijun Chen. Revisiting perspective information for efficient crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[32] Qingyu Song, Changan Wang, Zhengkai Jiang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yang Wu. Rethinking counting and localization in crowds: A purely point-based framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3365–3374, 2021. 1, 2

[33] Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, and Ping Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20993–21002, 2022. 2

[34] Ramana Sundararaman, Cedric De Almeida Braga, Eric Marchand, and Julien Pettre. Tracking pedestrian heads in dense crowd. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3865–3875, 2021. 3

[35] Chang-Lin Wan, Feng-Kai Huang, and Hong-Han Shuai. Density-based flow mask integration via deformable convolution for video people flux estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6573–6582, 2024. 2, 3, 6, 7

[36] Mingjie Wang, Hao Cai, Xian-Feng Han, Jun Zhou, and Minglun Gong. Stnet: Scale tree network with multi-level auxiliator for crowd counting. *IEEE Transactions on Multimedia*, 25:2074–2084, 2023. 2

[37] Qi Wang, Tao Han, Junyu Gao, and Yuan Yuan. Neuron linear transformation: Modeling the domain shift for crowd counting. *IEEE Transactions on Neural Networks and Learning Systems*, 33(8):3238–3250, 2022. 2

[38] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking. In *Computer Vision – ECCV 2020*, pages 107–122, 2020. 2

[39] Longyin Wen, Dawei Du, Pengfei Zhu, Qinghua Hu, Qilong Wang, Liefeng Bo, and Siwei Lyu. Drone-based joint density map estimation, localization and tracking with space-time multi-scale attention network. *arXiv preprint arXiv:1912.01811*, 2019. 3

[40] Longyin Wen, Dawei Du, Pengfei Zhu, Qinghua Hu, Qilong Wang, Liefeng Bo, and Siwei Lyu. Detection, tracking, and counting meets drones in crowds: A benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7812–7821, 2021. 2, 3

[41] Haiyang Xie, Zhengwei Yang, Huilin Zhu, and Zheng Wang. Striking a balance: Unsupervised cross-domain crowd counting via knowledge diffusion. In *Proceedings of the 31st ACM International Conference on Multimedia*, page 6520–6529, 2023. 2

[42] Zhaoyi Yan, Yuchen Yuan, Wangmeng Zuo, Xiao Tan, Yezhen Wang, Shilei Wen, and Errui Ding. Perspective-guided convolution networks for crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[43] Yifan Yang, Guorong Li, Zhe Wu, Li Su, Qingming Huang, and Nicu Sebe. Reverse perspective network for perspective-aware object counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[44] Biao Zhan, Dorothy N Monekosso, Paolo Remagnino, Sergio A Velastin, and Li-Qun Xu. Crowd analysis: a survey. *Machine Vision and Applications*, 19(5):345–357, 2008. 1

[45] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[46] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Computer Vision – ECCV 2022*, pages 1–21, 2022. 2, 3, 7

[47] Zhuoyi Zhao, Hongsheng Li, Rui Zhao, and Xiaogang Wang. Crossing-line crowd counting with two-phase deep neural networks. In *Computer Vision – ECCV 2016*, pages 712–726, 2016. 7

[48] Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Heng Fan, Qinghua Hu, and Haibin Ling. Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7380–7399, 2021. 2, 3