Open-World Object Counting in Videos

Niki Amini-Naieni, Andrew Zisserman

Visual Geometry Group, Dept. of Engineering Science, University of Oxford, UK
nikian@robots.ox.ac.uk, az@robots.ox.ac.uk

Abstract

We introduce a new task of open-world object counting in videos: given a text description, or an image example, that specifies the target object, the objective is to enumerate all the unique instances of the target objects in the video. This task is especially challenging in crowded scenes with occlusions and objects of similar appearance, where avoiding double counting and identifying reappearances is crucial. To this end, we make the following contributions: we introduce a model, COUNTVID, for this task. It leverages an image-based counting model, and a promptable video segmentation and tracking model, to enable automated open-world object counting across video frames. To evaluate its performance, we introduce VIDEOCOUNT, a new dataset for this novel task built from the TAO and MOT20 tracking datasets, as well as from videos of penguins and metal alloy crystallization captured by x-rays. Using this dataset, we demonstrate that COUNTVID provides accurate object counts, and significantly outperforms strong baselines.

Code, models, dataset —
https://www.robots.ox.ac.uk/vgg/research/countvid/

Extended version — https://arxiv.org/pdf/2506.15368

Introduction

Our objective in this paper is *open-world* object counting in videos – determining how many instances of an object class are present in a video, where the object class of interest is specified by a text description or an image exemplar and may not have been included in the training data. This is a time-dependent task, as both the current (visible count) can be reported at the frame level, or other temporal intervals, as well as the cumulative count over the entire video.

As fig. 1 illustrates, keeping count in videos is naturally a *correspondence or tracking task* – since we do not want to count the same instance multiple times, we must establish that instances in consecutive frames are the same. However, as the figure also illustrates, one of the fundamental challenges of counting in videos is *instance identification* – is an object appearing in a frame a new instance? Or, is it one that drifted out of frame earlier in the sequence or was temporarily occluded? This challenge is exacerbated as the objects become indistinguishable: discriminating individual fish is possible if they have different colors and markings, but discriminating insects or crows may be impossible.

Somewhat surprisingly, automated counting in *videos* is a relatively unexplored area. While there are a few automated methods for closed-vocabulary counting (Loy et al. 2013; Fang et al. 2019; Zhu et al. 2021a; Makhura and Woods 2019; Han et al. 2022; Wen et al. 2021; Liu et al. 2024b), there are no open-vocabulary methods that we are aware of. This in contrast to the extensive exploration of counting in *images*, where open-vocabulary methods are able to use both text and exemplars to specify the target object, and can count up to thousands of instances (Amini-Naieni, Han, and Zisserman 2024; Pelhan et al. 2024a; Ranjan et al. 2021). Even some large-scale Vision-Language Models, such as Molmo (Deitke et al. 2025), are now able to accurately count beyond ten instances in an image.

This lack of research in video counting is especially surprising given the wide variety of science applications and need. Conservationists need to count animals in video sequences captured by drones for population monitoring (Jones et al. 2023; Mustafa et al. 2019). This can take up to 30 hours for a trained human analyst to annotate manually for a single one-hour flight (Wich et al. 2021). Material scientists count crystals forming from liquid metal alloys to determine how cooling affects the speed of the formation process (Liotti et al. 2018). Epidemiologists use human and vehicle counts from videos captured on city streets to study causes of pedestrian exposure to air pollution and mitigate them (Schroeder et al. 2024, 2020). An 'open-world' method that can be quickly applied to all these problems out-of-the-box, with no manual counting or additional training needed, has the potential to catalyze these applications, eliminating the annotation time, and significantly benefiting their research.

In this paper, we introduce a model, COUNTVID, for open-world object counting in videos that accepts a video and a prompt that specifies the target object to count as inputs, and outputs how many unique instances of the object appear in the video. The prompt can consist of a free-form text description and/or any number of 'visual exemplars', where the visual exemplars indicate the object of interest by bounding boxes and can come from a video frame or an external image.

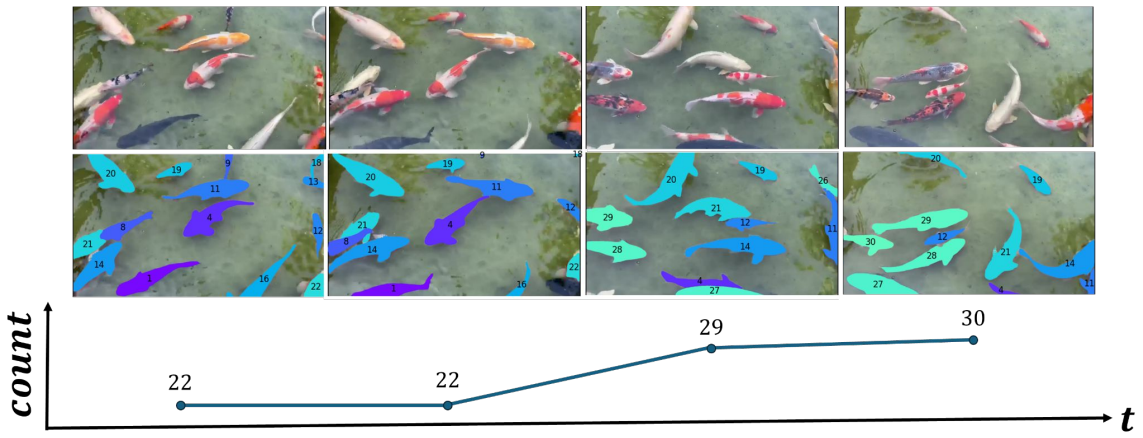

Figure 1: Object counting in a video. Given the video in the top row (shown by a few sample frames) and the text "fish", COUNTVID is able to accurately match instances across the video (indicated by the color and number assigned to each fish), identify new objects, and estimate the count. The video is available on the project website.

The COUNTVID model builds on advances for two distinct tasks: (i) powerful open-vocabulary image counting and detection models (Amini-Naieni, Han, and Zisserman 2024; Pelhan et al. 2024b; Zhizhong et al. 2024); and (ii) powerful class-agnostic video segmentation and tracking models (Ravi et al. 2025; Yang et al. 2024). However, naively combining an image detector to provide instances for a tracker is not sufficient since state-of-the-art object detectors struggle to count high numbers of objects in densely packed scenes with many occlusions and overlapping instances (Amini-Naieni, Han, and Zisserman 2024; Dai, Liu, and Cheung 2024). To overcome this detection problem, for our model we leverage accurate image-based *counters* that also output the bounding boxes required by the tracker (Pelhan et al. 2024b; Zhizhong et al. 2024).

One of the innovations of this paper is to extend the most flexible image-based counter, CountGD (Amini-Naieni, Han, and Zisserman 2024), to also produce bounding boxes, offering versatility in accepting text, visual exemplar, or combined prompts and detection capabilities within the same model. We show that its performance surpasses image detectors when there are many instances of similar objects in the image, and also prior models (Amini-Naieni, Han, and Zisserman 2024; Pelhan et al. 2024a; Nguyen et al. 2022).

A second innovation is to propose a temporal filter to remove false positive tracks resulting from erroneous detections. The extended CountGD model, dubbed COUNTGD-BOX, and other detection-based counters are used to provide box prompts over multiple frames in the video, and the tracker is used to *associate* the resultant segmentations and propagate them to other frames.

To assess the performance of video counting, we introduce VIDEOCOUNT, a new dataset with ground truth for this task. VIDEOCOUNT has two types of benchmarks: first, we re-purpose standard tracking datasets, TAO (Dave et al. 2020) and MOT20 (Dendorfer et al. 2020), by adding additional annotations to ensure all objects are counted (since tracking benchmarks typically only evaluate on a subset of the objects, e.g. not accounting for static objects); and second, we introduce two science applications of counting with new videos containing real footage from monitoring penguins in their natural habitats and x-ray images of the crystallization process for metal alloys. The number of objects in videos from our dataset ranges from one to over a thousand.

In summary, we make the following four contributions: *first* we present the novel task of open-world object counting in videos; *second* we propose a model for this task, COUNTVID, by re-purposing and combining open-vocabulary image counting and class-agnostic segmentation and tracking models; *third* we extend CountGD to produce bounding boxes as outputs, and introduce an automated method to remove false tracks; and *fourth* we release VIDEOCOUNT, a new video dataset for evaluating algorithms for this open-world object counting task.

Related Work

A note about terminology: in the object counting literature (Amini-Naieni et al. 2023; Amini-Naieni, Han, and Zisserman 2024; Liu et al. 2022), *open-world counting* refers to counting instances of an object class specified at test time via text or visual prompts. We adopt this definition, but also discuss how 'open-world' has been interpreted differently in the past in (Amini-Naieni and Zisserman 2025).

Open-World Object Counting in Images. Prior work on open-world object counting only focuses on images. The first image-based methods required the user to manually annotate a few example objects with 'visual exemplars' to count at inference time (Liu et al. 2022; Lu, Xie, and Zisserman 2018; Nguyen et al. 2022; Ranjan et al. 2021; Shi et al. 2022; Yang et al. 2021; You et al. 2023; Djukic et al. 2023; Lin et al. 2022). More recent works (Dai, Liu, and Cheung 2024; Amini-Naieni et al. 2023; Amini-Naieni, Han, and Zisserman 2024; Jiang, Liu, and Chen 2023; Kang et al.

2024) have leveraged pre-trained vision-language foundation models to enable the category to be specified by text. CountGD is a state-of-the-art open-world counting model that uses the joint vision-language embedding space of the Grounding DINO (Liu et al. 2024a) foundation model to allow the user to specify the object to count with text. In addition to this, unlike most prior approaches that either only accept text or only accept visual exemplars, CountGD allows for *both* inputs. By accepting only text, CountGD can adapt to novel classes without human intervention, and by accepting visual exemplars, it provides greater accuracy. We build on CountGD in this work.

Open-World Object Counting in Videos. While there is no prior work that explicitly focuses on open-world object counting in videos, there are open-world trackers (Li et al. 2023) that can be repurposed as counters. State-of-the-art open-world trackers rely on object detectors (Li et al. 2024b,a; Qian et al. 2024; Heigold et al. 2023). For example, the open-world tracker MASA (Li et al. 2024a) leverages detectors such as Grounding DINO and Detic (Zhou et al. 2022) to first detect any object using text and then associate it across video frames. The unique objects identified and tracked throughout the video can be enumerated to estimate the count. VOVTrack (Qian et al. 2024) uses region proposals from Faster R-CNN (Ren et al. 2017) for object localization. Because these approaches extend detectors, they inherit their limitations.

Trackers that do not rely on pre-trained image-based object detectors also have limitations. Trackformer (Meinhardt et al. 2022) cannot adapt to novel categories at inference, only tracking objects that it has been trained to track. On the other hand, OVTR (Li et al. 2025) is an end-to-end transformer-based tracking model that can track novel objects given text input. However, it does not accept visual prompts limiting its accuracy. Furthermore, it has not been trained or tested on scenes with hundreds to over a thousand objects. SAM 2, SAM 2.1 (Ravi et al. 2025) and SAMURAI (Yang et al. 2024) are recent state-of-the-art tracking and segmentation models that also adapt to novel objects without retraining. SAM 2.1 and SAMURAI extend SAM 2 with motion cues, longer video training, and occlusion handling, but both require manual prompting, which COUNTGD-BOX and COUNTVID automate. SAMURAI also focuses mainly on single-object tracking.

COUNTVID and COUNTGD-BOX Models

In this section, we first present COUNTVID, our method for open-world object counting in videos. We then present COUNTGD-BOX, a multi-modal counting model that outputs bounding boxes and extends CountGD.

COUNTVID

COUNTVID is a model that inputs a video and flexible prompts including text only, visual exemplars only, or both and outputs frame-level counts and a global count indicating the number of unique objects in the video that match the prompts. At inference, COUNTVID processes the video in three stages, at decreasing levels of granularity. The stages are illustrated in fig. 2.

Stage 1 – Processing at the Frame Level. The first stage is where the visual exemplar and text prompts are used, and the objective is to automatically obtain bounding boxes and segmentation masks for all instances of the target object in each frame. To achieve this, the visual exemplar and text prompts are fed to a counting and detection model that is applied to each video frame independently to obtain bounding boxes. The bounding boxes output by the counting model are used as box prompts for a segmentation model, which outputs masks for all the objects in the frame. We use exemplars from a single frame and apply them across the video to reduce user annotation effort. To improve efficiency, frames are subsampled before this stage begins.

Stage 2 – Processing in the Short Term. Although counting models are very accurate at counting, they can still produce false positive detections. This may occur due to motion blur. The objective of Stage 2 is to remove these false positive predictions with a temporal filter. This filter leverages the observation that false positives tend to be transient in independent (per-frame) predictions from Stage 1, disappearing almost immediately in subsequent frames. For each detection from Stage 1, the filter checks if the object exists within a temporal window of w frames. Using a segmentation and tracking model, the filter tracks $w-1$ frames backwards and $w-1$ frames forward from the frame the detection is in, resulting in a span of $2w-1$ frames. Objects are matched using the intersection over union (IoU) of the propagated masks from tracking and the masks from independent per-frame detections in Stage 1. An IoU greater than 0.5 is considered a match. If the object is matched in a sequence of at least w consecutive frames, it is kept. This ensures the filter tolerates a degree of occlusion or intermittent visibility. Otherwise, it is removed before Stage 3 begins. Note, it is necessary to track both forwards and backwards in time, as new objects can appear (and are verified by tracking forwards in time), and also objects can disappear (e.g. by occlusion) and are verified by tracking backwards in time.

Stage 3 – Processing in the Long Term. In the last stage, COUNTVID applies a segmentation and tracking model to the full video, keeping track of objects in the long term while checking for new objects in each frame. For each object, COUNTVID predicts a *masklet*, an object mask propagated over time. New objects are detected by comparing existing masklets with per-frame masks from Stage 2. Per-frame masks that do not overlap with the existing masklets are identified as new objects. The new objects are then also tracked going forward. Once all the frames have been checked for new objects, the masklets are enumerated to estimate the final global count. The New Object Detection Logic is visually explained in detail in (Amini-Naieni and Zisserman 2025).

COUNTVID Implementation. We implement COUNTVID with COUNTGD-BOX (described below) as the counting/detection model, and SAM 2.1 as the tracker. The boxes from COUNTGD-BOX are used as box prompts for SAM 2.1 in Stage 1, and then SAM 2.1 tracks the masks from the prompted objects, producing masklets. Though COUNTVID

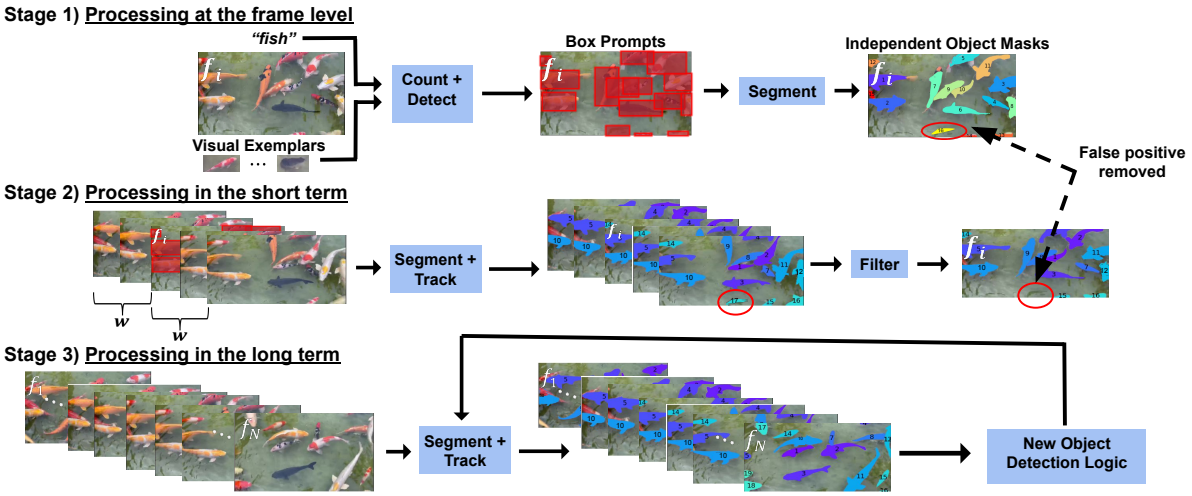

Figure 2: Inference with COUNTVID. Each stage processes the video at decreasing granularity. In Stage 1, a detection-based counter uses text and exemplars to generate box prompts for each frame, which are used by a segmentation model to produce object masks. Stage 2 applies a temporal filter over sequences of w frames to remove transient false positives (e.g., the red-circled object in f_i). Stage 3 propagates object masks from Stage 2 across the full video while checking for new objects in each frame. All the identified objects are enumerated to produce the final global count.

is agnostic to the choice of counter/detector and tracker. We compare to other choices in the experiments.

COUNTGD-BOX

To automatically obtain box prompts for the segmentation model, we need a detector that can handle dense scenes with many similar overlapping objects, because this will occur in the video for our challenging task. As shown by our results and prior work (Amini-Naieni, Han, and Zisserman 2024; Amini-Naieni et al. 2023), there are open-world detectors, but they do not perform well in this setting. On the other hand, there are open-world counters that do. Out of all these counters, CountGD is the most flexible, accepting either text only, visual exemplars only, or both simultaneously to specify the object. It also provides generally strong counting performance across all prompt settings as shown by our results. However, unlike other less flexible detection-based counting models, CountGD outputs points, not boxes.

Point prompts do not specify the object to count in a well-defined way. For example, a point on the window of a car could mean every window or every car should be counted. Given there is likely more than one window on a car, this problem could result in an erroneously high or low count. This ambiguity is resolved if an image region (a box) is specified instead of a point.

To obtain well-defined object prompts for the segmentation model, we train CountGD to output *boxes* in addition to *points*. CountGD originally lacks this capability due to the limited bounding box data available in object counting datasets such as FSC-147 (Ranjan et al. 2021). This scarcity exists because labeling hundreds to thousands of objects with bounding boxes is extremely tedious. Instead, these datasets only provide box annotations for a few objects per image. Drawing inspiration from DAVE (Pelhan et al. 2024a), we extend CountGD to take advantage of these weak training labels. Because CountGD is built on top of the Grounding DINO architecture, it already outputs four parameters per object. The first two are used as the center of the object, while the last two are discarded by CountGD. We add two new terms, $\mathcal{L}^e_{h,w}$ and $\mathcal{L}^e_{GIoU}$, to CountGD's loss, to train the last two parameters to be the height and width of the bounding box. The new loss is defined as $\mathcal{L} = \lambda_{loc}\left(\mathcal{L}^e_{h,w} + \mathcal{L}_{center}\right) + \lambda_{GIoU}\mathcal{L}^e_{GIoU} + \lambda_{cls}\mathcal{L}_{cls}$ where $\mathcal{L}^e_{h,w}$ and $\mathcal{L}^e_{GIoU}$ are based on the bounding box regression losses from Grounding DINO. The difference here is that these losses are only calculated for the exemplars, while in Grounding DINO, they are calculated for all the objects in the image. $\mathcal{L}^e_{h,w}$ is the sum of the absolute errors of the height and widths and $\mathcal{L}^e_{GIoU}$ is the generalized intersection over union between the predicted and ground truth exemplar boxes. By training on the exemplar boxes, CountGD learns to not only predict points, but to also predict bounding boxes. We name the extended CountGD as COUNTGD-BOX and use it at inference to produce box prompts for the segmentation model.

VIDEOCOUNT: A New Counting Dataset

Current benchmarks for object counting are not sufficient for the open-world object counting in videos task. This is because existing counting datasets either only support images (Ranjan et al. 2021; Hsieh, Lin, and Hsu 2017) or only include a limited number of categories (Loy et al. 2013; Zhu et al. 2021b; Dendorfer et al. 2020). Furthermore, existing tracking datasets such as TAO (Dave et al. 2020) only provide exhaustive annotations for a subset of objects and

	# Videos	# Classes	# Objects	Video Len
TAO-Count	357	139	1-10 2.7 Avg	8s-78s 34s Avg
MOT20-Count	3	1	80-1203 520.3 Avg	17s-133s 87s Avg
Science-Count	10	2	10-154 63 Avg	4s-30s 11s Avg

Table 1: VIDEOCOUNT details. Our dataset, composed of 3 benchmarks, covers a wide range of object categories (141 in total) and a wide range of object counts (1-1203 per video).

only label at most ten objects per video, which is far too low for practical counting use cases. Therefore, in this section, we present VIDEOCOUNT, a new dataset for open-world object counting in videos that overcomes these limitations. VIDEOCOUNT is made up of three benchmarks: TAO-Count, MOT20-Count, and Science-Count. It contains 370 videos covering a wide range of object categories and counts as shown in table 1. We include further details in (Amini-Naieni and Zisserman 2025).

Our dataset is built from diverse sources. For TAO-Count and MOT20-Count, we add metadata to subsets of the existing tracking datasets TAO (Dave et al. 2020) and MOT20 (Dendorfer et al. 2020) specifying the counts of target objects. For Science-Count, we release new videos and count annotations from monitoring penguin populations and videoing the formation of crystals from liquid metal alloys captured with x-ray radiography (Liotti et al. 2018). Example video frames from VIDEOCOUNT are shown in fig. 3 and (Amini-Naieni and Zisserman 2025).

VIDEOCOUNT tests COUNTVID's ability to adapt to a wide range of challenging scenarios. TAO-Count tests how well COUNTVID counts low numbers of objects in scenes with significant motion. MOT20-Count tests how well COUNTVID counts in heavily crowded scenes (e.g., > 1000 objects) with many overlapping instances. Science-Count evaluates COUNTVID on tricky real-world applications with many similar objects, some even rapidly changing structure over time in x-ray videos typically out-of-domain for foundation models.

Experiments

Implementation Details: COUNTGD-BOX is initialized with the pre-trained weights of CountGD. Its MLP box detection heads are then fine-tuned on the FSC-147 training set for 30 epochs with early stopping on the validation set. λ_{loc}, λ_{GIoU}, and λ_{cls} in the loss are set to 5, 2, and 2 respectively using a grid search on the validation set. For COUNTVID, in Stage 1, we sample frames at 3 fps. In Stage 2, the window size of the temporal filter, w, is set to 3 frames, corresponding to one second. The IoU threshold for matching is set to 0.5. Additional implementation details, including detailed analysis of the inference time and memory consumption for each stage with and without the temporal filter, are given in (Amini-Naieni and Zisserman 2025).

Datasets & Metrics

Images: To evaluate the counting and detection accuracy of state-of-the-art detection and counting models in crowded scenes, we use FSCD-147 (Nguyen et al. 2022), which provides bounding boxes for the validation and test sets of the widely established open-world image object counting dataset FSC-147. Exhaustive bounding boxes are not provided for the training set. Each image is annotated with three visual exemplars. To measure the detection accuracy, we follow (Nguyen et al. 2022) by providing the mean average precision over thresholds 0.5 to 0.95 (AP) and the average precision at the IoU threshold of 0.5 (AP50). We also report the image-based count MAE and RMSE used in (Ranjan et al. 2021). We take the count as the enumeration of the bounding boxes following (Pelhan et al. 2024b), and report results given text only, three exemplars only, or both, depending on what each method allows. In addition, we report zero-shot counting results for detectors and counters on the ShanghaiTech (Zhang et al. 2016) crowd counting dataset in (Amini-Naieni and Zisserman 2025).

Videos: To evaluate counting accuracy for videos, we report results on VIDEOCOUNT. Due to the class overlap of the training set of FSC-147 and TAO-Count, we also report results on the subset of videos in TAO-Count with the training classes in FSC-147 removed. To measure counting accuracy for videos, we draw on prior work on object counting in images (Ranjan et al. 2021) that uses the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). We define video analogues of these metrics for our new task. More specifically, we define the video MAE and RMSE as: $MAE = (1/N)\sum_{i=1}^{N}|\hat{y}_i - y_i|, RMSE = [(1/N)\sum_{i=1}^{N}(\hat{y}_i - y_i)^2]^{1/2}$, where N is the number of test videos, $\hat{y}_i$ is the predicted count for video X_i, and y_i is the ground truth count for video X_i. In more detail, y_i is the number of *unique* objects in the video that match the prompts. Importantly, the MAE and RMSE metrics for counting in videos differ from those used for images. In the video setting, the ground truth count reflects the number of *unique object identities*, not the number of detections. This requires both matching and re-identification: objects that reappear across frames must not be double-counted, so repeated detections of the same object must be correctly associated.

Assessing Processing at the Frame Level

In table 3, we evaluate different *image* counting and detection methods on FSCD-147 Test given different prompts including text only ('text'), exemplars only ('exemp'), or both together ('both'). Results for FSCD-147 Val are given in (Amini-Naieni and Zisserman 2025). Following (Pelhan et al. 2024b), the count is determined by enumerating the bounding boxes, not by summing density maps (Pelhan et al. 2024a) or applying test-time procedures (Amini-Naieni, Han, and Zisserman 2024) as these operations do not provide boxes. For exemplars, we use the three provided by FSC-147 for each sample. Text descriptions either come from the FSC-147 class names or FSC-147-D (Amini-Naieni et al. 2023). Note, we obtain box predictions from

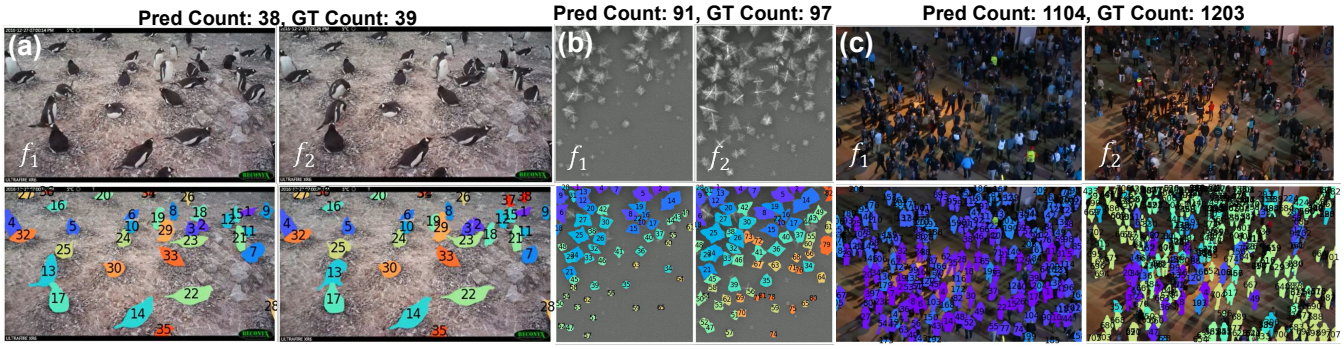

Figure 3: Qualitative results on VIDEOCOUNT. f_1 and f_2 are frames sampled consecutively in time. COUNTVID handles dense (b, c), deforming (b), and similar (a, b) objects.

	TAO-Count		TAO-Count-FSC		MOT20-Count		Penguins		Crystals	
Method	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓
DR. VIC (Baseline)	-	-	-	-	100.0	110.8	-	-	-	-
MASA (Baseline)	14.1	21.5	13.9	20.4	630.0	725.4	9.0	11.5	72.3	87.9
COUNTVID (Ours)	**2.6**	**6.0**	**2.5**	**5.2**	**50.0**	**61.3**	**4.0**	**5.3**	**69.1**	**86.0**

Table 2: Counting performance on VIDEOCOUNT. MASA and COUNTVID are given *only text* prompts. DR. VIC only counts humans and so can only be evaluated on MOT20-Count. TAO-Count-FSC is TAO-Count with the training classes in FSC-147 removed. The class overlap between TAO-Count and FSC-147 has no significant influence on COUNTVID's counting accuracy.

		FSCD-147 Test			
		Counting		Detection	
Method	Prompt	MAE ↓	RMSE ↓	AP ↑	AP50 ↑
GDINO	text	54.16	157.87	11.60	17.80
OWLv2	text	41.83	149.82	22.84	35.76
PSeCo	text	16.58	129.77	**41.14**	**69.03**
DAVE$_{prm}$	text	15.52	**114.10**	18.50	50.24
CountGD	text	15.19	119.40	18.10	52.90
CGD-B	text	**15.01**	118.16	30.44	61.56
C-DETR	exemp.	16.79	123.56	22.66	50.57
PSeCo	exemp.	13.05	112.86	**43.53**	74.64
DAVE	exemp.	10.45	74.51	26.81	62.82
GeCo	exemp.	**7.91**	**54.28**	43.42	**75.06**
CountGD	exemp.	10.77	99.51	19.76	57.24
CGD-B	exemp.	10.85	99.60	34.81	69.46
CountGD	both	**10.18**	**96.20**	20.50	59.40
CGD-B	both	10.29	96.33	**36.20**	**72.39**

Table 3: Results on FSCD-147 for image counting methods that output boxes (for Stage 1). The abbreviations are: COUNTGD-BOX (CGD-B); C-DETR (Counting-DETR (Nguyen et al. 2022)). SAM masks are not used here.

Temporal Filter	Prompt	MAE ↓	RMSE ↓
✗	text	6.6	17.5
✓	text	**2.6**	**6.0**

Table 4: Temporal filter ablation on TAO-Count (Stage 2).

the original CountGD (without the additional training losses added in this paper) by using its full bounding box outputs, rather than, as in the original model, only using the points as output.

From these results, we draw three conclusions: (i) as confirmed by prior work, SoTA detectors like OWLv2 (Minderer, Gritsenko, and Houlsby 2023) and Grounding DINO (Liu et al. 2024a) do not work well for the counting setting, where there are many similar and overlapping objects. The caveat here is that these detectors have not been trained on FSC-147, as the counters have. To address this, we also benchmark OWLv2 and Grounding DINO against COUNTGD-BOX and CLIP-Count (Jiang, Liu, and Chen 2023) on the held-out ShanghaiTech counting dataset without any training on ShanghaiTech and show that the counters still significantly beat the detectors in (Amini-Naieni and Zisserman 2025); (ii) Extending CountGD to COUNTGD-BOX, significantly improves its detection accuracy while preserving its counting accuracy; (iii) the SoTA model depends on the type of prompt (text/exemplar/both) used. While COUNTGD-BOX is 'a good all-rounder,' it is not the best for all cases. Both COUNTGD-BOX and PSeCo (Zhizhong et al. 2024) perform competitively in the text-only setting. GeCo (Pelhan et al. 2024b) is the superior model in the exemplar-only setting, although both COUNTGD-BOX and DAVE (Pelhan et al. 2024a) are strong contendors. For models that accept both exemplars and text, COUNTGD-BOX is superior over CountGD for detection. In some cases, the text does add information to the exemplar, by specifying location or color for example (see section 4.5 of (Amini-Naieni, Han, and Zisserman 2024)). However, in

CountVid Variation	Prompt	Penguins MAE ↓	Penguins RMSE ↓	Crystals MAE ↓	Crystals RMSE ↓
CGD-B/BT	text	4.3	5.5	71.6	88.1
CGD-B/S2	text	**3.3**	**4.8**	**69.1**	**86.0**
CGD-B/S2.1	text	4.0	5.3	**69.1**	**86.0**
CGD-B/BT	exemp	4.0	**4.2**	**31.1**	**52.8**
GeCo/S2	exemp.	11.7	13.1	59.6	104.2
GeCo/S2.1	exemp.	11.3	14.5	46.1	82.8
CGD-B/S2	exemp.	**3.3**	4.8	37.4	61.0
CGD-B/S2.1	exemp.	**3.3**	4.8	33.7	59.8
CGD-B/BT	both	7.0	8.3	12.7	16.6
CGD-B/S2	both	**0.3**	**0.6**	**12.0**	**13.5**
CGD-B/S2.1	both	0.7	0.8	13.4	14.8

Table 5: CountVid video counting results on Science-Count using various prompts, counters, and trackers. The abbreviations are: CountGD-Box (CGD-B); ByteTrack (BT); SAM 2 (S 2); and SAM 2.1 (S 2.1).

settings where this is not the case, like FSC-147 where the text and exemplar both represent the class, GeCo given only exemplars should be used.

Assessing Processing in the Short Term

In table 4, we assess the effectiveness of the temporal filter on TAO-Count. Specifically, we report the video-based MAE and RMSE from applying CountVid with and without the temporal filter given text only. For the counter, we use CountGD-Box, and for the tracker, we use SAM 2.1. The scenes in TAO-Count involve significant motion and blur, inducing false positives. The temporal filter effectively removes these false positives, reducing the MAE and RMSE by over 50%, improving the counting accuracy significantly.

Assessing Processing in the Long Term

In this section, we evaluate CountVid's overall video-based counting performance on VideoCount, and compare its performance to strong baselines. For the text descriptions for TAO-Count, we use the category synsets (Dave et al. 2020). We use the text 'human' for MOT20-Count. We use 'white crystal' and 'penguin' for Science-Count. When exemplars are used, 3–6 exemplars are provided for the first frame of the video and applied to all subsequent frames. The overall results are given in table 2 and table 5.

Baselines: We compare CountVid to strong baselines built from a Multi-Object Tracking (MOT) method and a Video Individual Counting (VIC) method. Specifically, for the first baseline, we repurpose the strong open-world tracker MASA (Li et al. 2024a) implemented with Grounding DINO. The unique tracks are enumerated to estimate the count. For the second baseline, we use the specialized crowd counting method DR. VIC (Han et al. 2022) trained on HT21 (Sundararaman et al. 2021) and only report results on MOT20-Count, as the other benchmarks in VideoCount contain objects other than humans. Note the HT21 training set contains two of the videos in MOT20-Count.

CountVid implemented with CountGD-Box and SAM 2.1 achieves significantly better counting accuracy than the MASA baseline, as shown in table 2. For a fair comparison with MASA, CountVid is given text only, even though CountVid can also accept exemplars. Remarkably, CountVid also significantly improves on the performance of DR. VIC, even though DR. VIC was trained specifically for counting humans and has been trained on one out of the three videos in MOT20-Count, so it has been trained on one third of the videos we are evaluating it on, while CountVid can count other objects in addition to humans and is applied to MOT20-Count zero-shot.

Note that none of the components of CountVid (including CountGD-Box and any hyperparameters) were fine-tuned on the data in VideoCount. We hypothesize the significantly superior performance of CountVid can be attributed to (i) leveraging the image-based *counting* model CountGD-Box rather than the Grounding DINO *detection* model to handle crowded scenes. CountGD-Box extends Grounding DINO by fine-tuning on counting data and adding modules to enable visual exemplar inputs, meaning it is more accurate and capable than Grounding DINO at counting in video frames; (ii) removing false positives with a temporal filter; and (iii) effectively leveraging video foundation models like SAM 2.1.

In table 5, we compare different variations of CountVid implemented with different combinations of counters/trackers and given different prompts. For the counters, we use CountGD-Box and GeCo, and for the trackers, we use ByteTrack (Zhang et al. 2022), SAM 2 and SAM 2.1. For the ByteTrack variant, we rely on the tracker to detect new objects and do not apply the temporal filter, since there are no segmentation masks. We note the exemplar-only performance is better than text-only, and providing both prompts is the best, showing CountVid effectively benefits from more information about the object. We find that while GeCo works well for images, it is not as accurate as CountGD-Box on videos. SAM 2.1 performs significantly better than SAM 2 in the exemplar-only setting for Crystals. However, it falls slightly behind SAM 2 for the other settings.

As shown in fig. 3, CountVid counts in dense scenes, detects new objects while retaining old ones, and counts deforming objects. Errors can occur due to false negatives from the counting model and tracker re-identification challenges. Scenes with many occlusions and similar instances can cause higher errors due to more of these cases.

Conclusion

We present the novel task of open-world object counting in videos together with a new model, CountVid, and a new dataset, VideoCount, to test the model. CountVid inputs flexible visual exemplar and text prompts and outputs both frame-level counts and a global count indicating the number of unique objects in the video that match the prompts. CountVid will continue to benefit from better trackers and class-agnostic detection-based counting models, as they can easily be plugged into the framework we have proposed.

Acknowledgments

The authors would like to thank Professor Tom Hart and Penguin Watch for the videos in the Penguins (ScienceCount) benchmark, Dr Enzo Liotti for the videos in the Crystals (ScienceCount) benchmark and Shun Yang for the curation and preparation of these videos, Jer Pelhan for his extensive support of GeCo, and Siyuan Li for his extensive support of MASA. This research is funded by an AWS Studentship, the Reuben Foundation, a Qualcomm Innovation Fellowship (mentors: Dr Farhad Zanjani and Dr Davide Abati), the AIMS CDT program at the University of Oxford, EPSRC Programme Grant VisualAI EP/T028572/1, and a Royal Society Research Professorship RSRP\R\241003.

References

Amini-Naieni, N.; Amini-Naieni, K.; Han, T.; and Zisserman, A. 2023. Open-world Text-specified Object Counting. In *Proceedings of the British Machine Vision Conference*.

Amini-Naieni, N.; Han, T.; and Zisserman, A. 2024. CountGD: Multi-Modal Open-World Counting. In *Advances in Neural Information Processing Systems*.

Amini-Naieni, N.; and Zisserman, A. 2025. Open-World Object Counting in Videos. *arXiv preprint arXiv:2506.15368*.

Dai, S.; Liu, J.; and Cheung, N.-M. 2024. Referring Expression Counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Dave, A.; Khurana, T.; Tokmakov, P.; Schmid, C.; and Ramanan, D. 2020. TAO: A Large-Scale Benchmark for Tracking Any Object. In *Proceedings of the European Conference on Computer Vision*.

Deitke, M.; Clark, C.; Lee, S.; Tripathi, R.; Yang, Y.; Park, J. S.; Salehi, M.; Muennighoff, N.; Lo, K.; Soldaini, L.; Lu, J.; Anderson, T.; Bransom, E.; Ehsani, K.; Ngo, H.; Chen, Y.; Patel, A.; Yatskar, M.; Callison-Burch, C.; Head, A.; Hendrix, R.; Bastani, F.; VanderBilt, E.; Lambert, N.; Chou, Y.; Chheda, A.; Sparks, J.; Skjonsberg, S.; Schmitz, M.; Sarnat, A.; Bischoff, B.; Walsh, P.; Newell, C.; Wolters, P.; Gupta, T.; Zeng, K.-H.; Borchardt, J.; Groeneveld, D.; Nam, C.; Lebrecht, S.; Wittlif, C.; Schoenick, C.; Michel, O.; Krishna, R.; Weihs, L.; Smith, N. A.; Hajishirzi, H.; Girshick, R.; Farhadi, A.; and Kembhavi, A. 2025. Molmo and PixMo: Open Weights and Open Data for State-of-the-Art Vision-Language Models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Reid, D. C. I.; Roth, S.; and Leal-Taixé, K. S. L. 2020. MOT20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*.

Djukic, N.; Lukezic, A.; Zavrtanik, V.; and Kristan, M. 2023. A Low-Shot Object Counting Network With Iterative Prototype Adaptation. In *Proceedings of the International Conference on Computer Vision*.

Fang, Y.; Zhan, B.; Cai, W.; Gao, S.; and Hu, B. 2019. Locality-constrained Spatial Transformer Network for Video Crowd Counting. *arXiv preprint arXiv:1907.07911*.

Han, T.; Bai, L.; Gao, J.; Wang, Q.; and Ouyang, W. 2022. DR.VIC: Decomposition and Reasoning for Video Individual Counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Heigold, G.; Keysers, D.; Minderer, M.; Lučić, M.; Gritsenko, A.; Yu, F.; Bewley, A.; and Kipf, T. 2023. Video OWL-ViT: Temporally-consistent open-world localization in video. In *Proceedings of the International Conference on Computer Vision*.

Hsieh, M.-R.; Lin, Y.-L.; and Hsu, W. H. 2017. Drone-Based Object Counting by Spatially Regularized Regional Proposal Network. In *Proceedings of the International Conference on Computer Vision*.

Jiang, R.; Liu, L.; and Chen, C. 2023. CLIP-Count: Towards Text-Guided Zero-Shot Object Counting. In *Proceedings of the ACM Multimedia Conference*.

Jones, L.; Elmore, J.; Boopalan, S. K.; Samiappan, S.; Evans, K.; Pfeiffer, M.; and Iglay, R. 2023. Controllable factors affecting accuracy and precision of human identification of animals from drone imagery. *Ecosphere*, 14.

Kang, S.; Moon, W.; Kim, E.; and Heo, J.-P. 2024. VLCounter: Text-Aware Visual Representation for Zero-Shot Object Counting. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Li, J.; Yu, E.; Chen, S.; and Tao, W. 2025. OVTR: End-to-End Open-Vocabulary Multiple Object Tracking with Transformer. In *Proceedings of the International Conference on Learning Representations*.

Li, S.; Fischer, T.; Ke, L.; Ding, H.; Danelljan, M.; and Yu, F. 2023. OVTrack: Open-Vocabulary Multiple Object Tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Li, S.; Ke, L.; Danelljan, M.; Piccinelli, L.; Segu, M.; Van Gool, L.; and Yu, F. 2024a. Matching Anything By Segmenting Anything. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Li, S.; Ke, L.; Yang, Y.-H.; Piccinelli, L.; Segu, M.; Danelljan, M.; and van Gool, L. 2024b. SLAck: Semantic, Location, and Appearance Aware Open-Vocabulary Tracking. In *Proceedings of the European Conference on Computer Vision*.

Lin, W.; Yang, K.; Ma, X.; Gao, J.; Liu, L.; Liu, S.; Hou, J.; Yi, S.; and Chan, A. 2022. Scale-Prior Deformable Convolution for Exemplar-Guided Class-Agnostic Counting. In *Proceedings of the British Machine Vision Conference*.

Liotti, E.; Arteta, C.; Zisserman, A.; Lui, A.; Lempitsky, V.; and Grant, P. S. 2018. Crystal nucleation in metallic alloys using x-ray radiography and machine learning. *Science advances*, 4(4).

Liu, C.; Zhong, Y.; Zisserman, A.; and Xie, W. 2022. CounTR: Transformer-based Generalised Visual Counting. In *Proceedings of the British Machine Vision Conference*.

Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Li, C.; Yang, J.; Su, H.; Zhu, J.; et al. 2024a. Grounding DINO: Marrying dino with grounded pre-training for open-set object detection. In *Proceedings of the European Conference on Computer Vision*.

Liu, X.; Li, G.; Qi, Y.; Yan, Z.; Han, Z.; van den Hengel, A.; Yang, M.-H.; and Huang, Q. 2024b. Weakly Supervised Video Individual Counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Loy, C. C.; Chen, K.; Gong, S.; and Xiang, T. 2013. Crowd Counting and Profiling: Methodology and Evaluation. In *Modeling, Simulation and Visual Analysis of Crowds*.

Lu, E.; Xie, W.; and Zisserman, A. 2018. Class-agnostic Counting. In *Proceedings of the Asian Conference on Computer Vision*.

Makhura, O. J.; and Woods, J. C. 2019. Video Object Counting Dataset. In *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*.

Meinhardt, T.; Kirillov, A.; Leal-Taixé, L.; and Feichtenhofer, C. 2022. TrackFormer: Multi-Object Tracking With Transformers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Minderer, M.; Gritsenko, A.; and Houlsby, N. 2023. Scaling Open-Vocabulary Object Detection. In *Advances in Neural Information Processing Systems*.

Mustafa, O.; Braun, C.; Esefeld, J.; Knetsch, S.; Maercker, J.; Pfeifer, C.; and Rümmler, M.-C. 2019. Detecting Antarctic Seals and Flying Seabirds by UAV. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*.

Nguyen, T.; Pham, C.; Nguyen, K.; and Hoai, M. 2022. Few-Shot Object Counting And Detection. In *Proceedings of the European Conference on Computer Vision*.

Pelhan, J.; Lukežič, A.; Zavrtanik, V.; and Kristan, M. 2024a. DAVE – A Detect-and-Verify Paradigm for Low-Shot Counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Pelhan, J.; Lukežič, A.; Zavrtanik, V.; and Kristan, M. 2024b. A Novel Unified Architecture for Low-Shot Counting by Detection and Segmentation. In *Advances in Neural Information Processing Systems*, volume 37. Curran Associates, Inc.

Qian, Z.; Han, R.; Hou, J.; Song, L.; and Feng, W. 2024. VOVTrack: Exploring the Potentiality in Videos for Open-Vocabulary Object Tracking. *arXiv preprint arXiv:2410.08529*.

Ranjan, V.; Sharma, U.; Nguyen, T.; and Hoai, M. 2021. Learning To Count Everything. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; Mintun, E.; Pan, J.; Alwala, K. V.; Carion, N.; Wu, C.-Y.; Girshick, R.; Dollár, P.; and Feichtenhofer, C. 2025. SAM 2: Segment Anything in Images and Videos. In *Proceedings of the International Conference on Learning Representations*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Schroeder, A. K.; Haugen, M. J.; Stettler, M. E. J.; and Boies, A. M. 2020. Using Computer Vision with Instantaneous Vehicle Emissions Modelling. In *2020 Forum on Integrated and Sustainable Transportation Systems (FISTS)*.

Schroeder, A. K.; Woodward, H.; Cornec, C. M. L.; Proust, T.; Benie, P. J.; Fan, S.; Aristodemou, E.; Jones, R. L.; Linden, P. F.; de Nazelle, A.; Boies, A. M.; and Stettler, M. E. 2024. Vehicle emission models alone are not sufficient to understand full impact of change in traffic signal timings. *Atmospheric Environment: X*.

Shi, M.; Lu, H.; Feng, C.; Liu, C.; and CAO, Z. 2022. Represent, Compare, and Learn: A Similarity-Aware Framework for Class-Agnostic Counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Sundararaman, R.; De Almeida Braga, C.; Marchand, E.; and Pettré, J. 2021. Tracking Pedestrian Heads in Dense Crowd. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Wen, L.; Du, D.; Zhu, P.; Hu, Q.; Wang, Q.; Bo, L.; and Lyu, S. 2021. Detection, Tracking, and Counting Meets Drones in Crowds: A Benchmark. In *CVPR*.

Wich, S. A.; Hudson, M.; Andrianandrasana, H.; and Longmore, S. N. 2021. Drones for Conservation. In *Conservation Technology*. Oxford University Press. ISBN 978-0-19-885024-3.

Yang, C.-Y.; Huang, H.-W.; Chai, W.; Jiang, Z.; and Hwang, J.-N. 2024. SAMURAI: Adapting Segment Anything Model for Zero-Shot Visual Tracking with Motion-Aware Memory. *arXiv preprint arXiv:2411.11922*.

Yang, S.-D.; Su, H.-T.; Hsu, W. H.; and Chen, W.-C. 2021. Class-agnostic Few-shot Object Counting. In *Proceedings of the IEEE Workshop on Applications of Computer Vision*.

You, Z.; Yang, K.; Luo, W.; Lu, X.; Cui, L.; and Le, X. 2023. Few-Shot Object Counting With Similarity-Aware Feature Enhancement. In *Proceedings of the IEEE Workshop on Applications of Computer Vision*.

Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D.; Weng, F.; Yuan, Z.; Luo, P.; Liu, W.; and Wang, X. 2022. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. In *ECCV*.

Zhang, Y.; Zhou, D.; Chen, S.; Gao, S.; and Ma, Y. 2016. Single-Image Crowd Counting via Multi-Column Convolutional Neural Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Zhizhong, H.; Mingliang, D.; Yi, Z.; Junping, Z.; and Hongming, S. 2024. Point, Segment and Count: A Generalized Framework for Object Counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Zhou, X.; Girdhar, R.; Joulin, A.; Krähenbühl, P.; and Misra, I. 2022. Detecting Twenty-thousand Classes using Image-level Supervision. In *ECCV*.

Zhu, P.; Peng, T.; Du, D.; Yu, H.; Zhang, L.; and Hu, Q. 2021a. Graph Regularized Flow Attention Network for Video Animal Counting From Drones. In *IEEE Transactions on Image Processing*.

Zhu, P.; Wen, L.; Du, D.; Bian, X.; Fan, H.; Hu, Q.; and Ling, H. 2021b. Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.