A Novel Unified Architecture for Low-Shot Counting by Detection and Segmentation

Jer Pelhan, Alan Lukežič, Vitjan Zavrtanik, Matej Kristan
Faculty of Computer and Information Science, University of Ljubljana
jer.pelhan@fri.uni-lj.si

Abstract

Low-shot object counters estimate the number of objects in an image using few or no annotated exemplars. Objects are localized by matching them to prototypes, which are constructed by unsupervised image-wide object appearance aggregation. Due to potentially diverse object appearances, the existing approaches often lead to overgeneralization and false positive detections. Furthermore, the best-performing methods train object localization by a surrogate loss, that predicts a unit Gaussian at each object center. This loss is sensitive to annotation error, hyperparameters and does not directly optimize the detection task, leading to suboptimal counts. We introduce GeCo, a novel low-shot counter that achieves accurate object detection, segmentation, and count estimation in a unified architecture. GeCo robustly generalizes the prototypes across objects appearances through a novel dense object query formulation. In addition, a novel counting loss is proposed, that directly optimizes the detection task and avoids the issues of the standard surrogate loss. GeCo surpasses the leading few-shot detection-based counters by $\sim$25% in the total count MAE, achieves superior detection accuracy and sets a new solid state-of-the-art result across all low-shot counting setups. The code is available on GitHub.

1 Introduction

Low-shot object counting considers estimating the number of objects of previously unobserved category in the image, given only a few annotated exemplars (few-shot) or without any supervision (zero-shot) [22]. The current state-of-the-art methods are predominantly based on density estimation [4; 14; 32; 26; 22; 31; 7; 31]. These methods predict a density map over the image and estimate the total count by summing the density.

While being remarkably robust for global count estimation, density outputs lack explainability such as object location and size, which is crucial for many practical applications [33; 30]. This recently gave rise to detection-based low-shot counters [20; 19; 35], which predict the object bounding boxes and estimate the total count as the number of detections. Nevertheless, detection-based counting falls behind the density-based methods in total count estimation, leaving a performance gap.

In detection-based counters, a dominant approach to identify locations of the objects in the image involves construction of object prototypes from few (e.g., three) annotated exemplar bounding boxes and correlating them with image features [20; 35; 19]. The exemplar construction process is trained to account for potentially large diversity of object appearances in the image, often leading to overgeneralization, which achieves a high recall, but is also prone to false positive detection. Post-hoc detection verification methods have been considered [20; 35] to address the issue, but their multi-stage formulation prevents exploiting the benefits of end-to-end training.

Currently, the best detection counters [20; 35] predict object locations based on the local maxima in the correlation map. During training, the map prediction is supervised by a unit Gaussian placed on

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

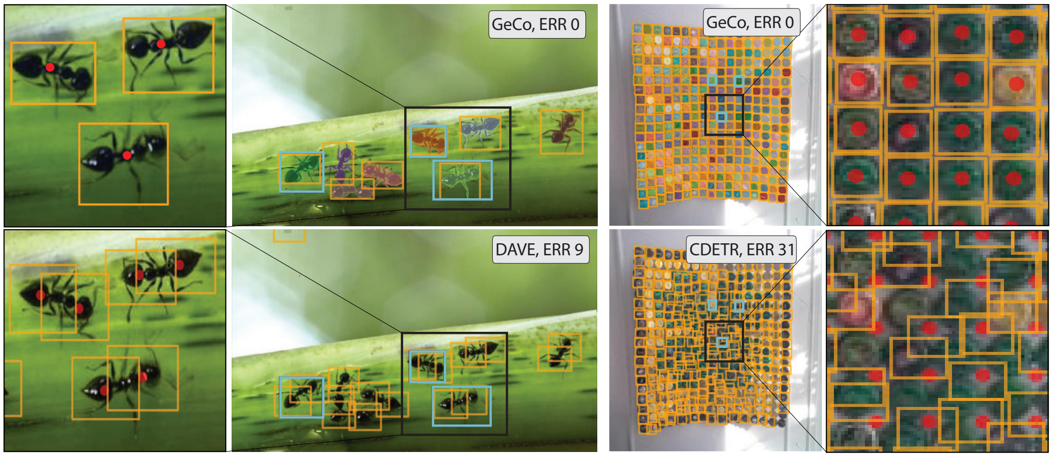

Figure 1: DAVE [20] predicts object centers (red dots) biased towards blob-like structures, leading to incorrect partial detections of ants (bottom left), while GeCo(ours) addresses this with the new loss (top left). CDETR [19] fails in densely populated regions (bottom right), while GeCo addresses this with the new dense query formulation by prototype generalization (top right). Exploiting the SAM backbone, GeCo delivers segmentations as well. Exemplars are denoted in blue.

each object center. However, the resulting surrogate loss is susceptible to the center annotation noise, requires nontrivial heuristic choice of the Gaussian kernel size and in practice leads to detection preference of compact blob-like structures (see Figure 1, column 1&2). Recently, DETR [1] inspired counter was proposed to avoid this issue [19], however, it fails in densely populated regions even though it applies a very large number of detection queries in a regular grid (see Figure 1, column 3&4).

We address the aforementioned challenges by proposing a new single-stage low-shot counter GeCo, which is implemented as an add-on network for SAM [12] backbone. A single architecture is thus trained for both few-shot and zero-shot setup, it enables counting by detection and provides segmentation masks for each of the detected objects. Our first contribution is a dense object query formulation, which applies a non-parametric model for image-wide prototype generalization (hence GeCo) in the encoder, and decodes the queries into highly dense predictions. The formulation simultaneously enables reliable detection in densely-populated regions (Figure 1, column 3&4) and prevents prototype over-generalization, leading to an improved detection precision at a high recall. Our second contribution is a new loss function for dense detection training that avoids the ad-hoc surrogate loss with unit Gaussians, it directly optimizes the detection task, and leads to improved detection not biased towards blob-like regions (Figure 1, column 1&2).

GeCo outperforms all detection-based counters on challenging benchmarks by 24% MAE and the density-based long-standing winner [4] by 27% MAE, while delivering superior detection accuracy. The method shows substantial robustness to the number of exemplars. In one-shot scenario, GeCo outperforms the best detection method in 5% AP50, 45% MAE and by 14% in a zero-shot scenario. GeCo is the first detection-based counter that outperforms density based counters in all measures by using the number of detections as the estimator, and thus sets a milestone in low-shot detection-based counting.

2 Related works

Traditional counting methods focus on predefined categories like vehicles[3], cells [5], people[15], and polyps, [33] requiring extensive annotated training data and lacking generalization to other categories, necessitating retraining or conceptual changes. Low-shot counting methods address this limitation by estimating counts for arbitrary categories with minimal or no annotations, enabling test-time adaptation.

With the proposal of the FSC147 dataset [24] low-shot counting methods emerged, which predict global counts by summing over a predicted density maps. The first method [24] proposed an adaptation of a tracking backbone for density map regression. BMNet+ [26] tackled learning representation and similarity metric, while SAFECount [32] introduced a new feature enhancement module, improving appearance generalization. CounTR [14] utilized a vision transformer for image feature extraction and a convolutional network for encoding the exemplar features. LOCA [4] argued that exemplar shape information should be considered along with the appearance, and proposed an iterative object prototype extraction module. This led to a simplified counter architecture that remains a top-performer among density-based counters.

To improve explainability of the estimated counts and estimate object locations as well, detection-based methods emerged. The first few-shot detection-based counter [19] was an extended transformer-based object detector [2] with the ability to detect objects specified by the exemplars. Current state-of-the-art DAVE [20] proposed a two-stage detect-and-verify paradigm for low-shot counting and detection, wherein the first stage it generates object proposals with a high recall, but low precision, which is improved by a subsequent verification step. PSECO [35] proposed a three-stage approach called point-segment-and-count, which employs more involved proposal generation with better detection accuracy and also applies a verification step to improve precision. Both DAVE and PSECO are multi-stage methods that train a network for the surrogate task of predicting density maps for object centers, from which the bounding boxes are predicted. Although detection-based counters offer additional applicability, they fall behind the best density-based counters in global count estimation.

3 Single-stage low-shot object counting by detection and segmentation

Given an input image $I \in \mathbb{R}^{H_0 \times W_0 \times 3}$ and a set of k exemplar bounding boxes $\boldsymbol{B}^{\mathrm{E}} = \{\mathbf{b}_i\}_{i=1:k}$ specifying the target category, the task is to predict bounding boxes $\boldsymbol{B}^{P} = \{\mathbf{b}_j\}_{j=1:N}$ for all target category objects in I, with the object count estimated as $N = |\boldsymbol{B}^{P}|$.

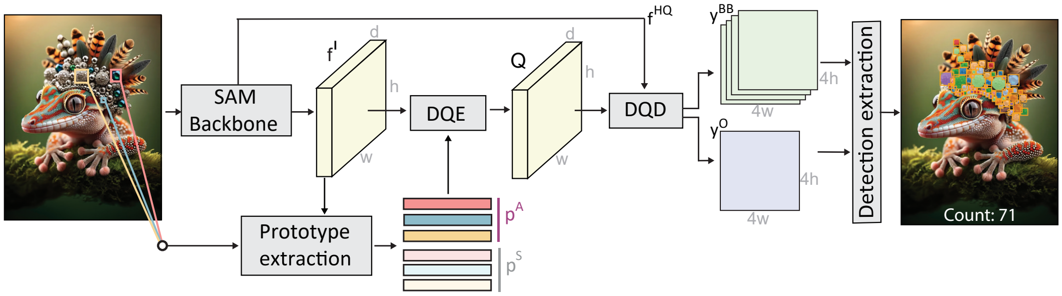

Figure 2: The architecture of the proposed single-stage low-shot counter GeCo.

The proposed detection-based counter GeCo pipeline proceeds as follows (see Figure 2). The image is encoded by a SAM [12] backbone into $\mathbf{f}^I \in \mathbb{R}^{h \times w \times d}$, where $h = H_0/r$, $w = W_0/r$ and d is number of feature channels. In the few-shot setup, two kinds of prototypes (appearance and shape) are extracted from each annotated object exemplar. The appearance prototypes $\mathbf{p}^A \in \mathbb{R}^{k \times d}$ are extracted by RoI-pooling [9] features $\mathbf{f}^I$ from the exemplar bounding boxes. Following [4], shape prototypes $\mathbf{p}^S \in \mathbb{R}^{k \times d}$ are extracted as well, by $\mathbf{p}_i^S = \Phi([W_{\mathbf{b}_i}, H_{\mathbf{b}_i}])$, where $W_{\mathbf{b}_i}$ and $H_{\mathbf{b}_i}$ are the width and height of the i-th exemplar bounding box, and $\Phi(\cdot)$ is a small MLP network. The concatenation of $\mathbf{p}^A$ and $\mathbf{p}^S$ yields $\mathbf{p} \in \mathbb{R}^{2k \times d}$ prototypes.

Note, however, that in a zero-shot setup, exemplars are not provided and the task is to count the majority-class objects in the image. In this setup, a single zero-shot prototype is constructed by attending a pretrained objectness prototype $\mathbf{p}^Z$ to the image features, i.e., $\mathbf{p} = \mathrm{CA}(\mathbf{p}^Z, \mathbf{f}^I, \mathbf{f}^I)$, where $\mathrm{CA}(a, b, c)$ is cross-attention [28] followed by a skip connection, with a, b and c as attention query, key and value, respectively.

The prototypes $\mathbf{p}$ (either from few-shot or zero-shot setup) are then generalized across the image, and dense object detection queries are constructed by the Dense query encoder (DQE, Section 3.1). These are decoded into dense detections by the Dense query decoder (DQD, Section 3.2). The final detections are extracted and refined by a post-processing step (Section 3.3). The aforementioned modules are detailed in the following sections.

3.1 Dense object query encoder (DQE)

To account for the variation of the object appearances in the image, the current state-of-the-art [4; 20; 35] aims at constructing a small number of prototypes (e.g., three) that compactly encode the object appearance variation in the image, often leading to overgeneralization and false detections. We deviate from this paradigm by considering image-wide prototype generalization with a non-parametric model that constructs $w \cdot h$ location-specific prototypes $\mathbf{P}_{N_P} \in \mathbb{R}^{w \cdot h \times d}$. Let $\mathbf{P}_0 = \mathbf{f}^I$ be the initial dense generalized prototypes (i.e., one for each location). The final dense generalized prototypes $\mathbf{P}_{N_P}$ are calculated by the following iterative adaptation via cross-attention

$$\mathbf{P}_i = \mathrm{CA}(\mathbf{P}_{i-1}, \mathbf{p}, \mathbf{p}), \tag{1}$$

where $i \in \{1, ..., N_P\}$. Note that spatial encoding is not applied, to enable spatially-unbiased information flow from the prototypes $\mathbf{p}$ to all locations.

Next, dense object queries are constructed from the generalized prototypes by the following iterations

$$\mathbf{Q}_j = \mathrm{CA}(\mathrm{SA}(\mathbf{f}^I), \mathbf{Q}_{j-1}, \mathbf{Q}_{j-1}), \tag{2}$$

where $j \in \{1, ..., N_Q\}$, $\mathbf{Q}_0 = \mathbf{P}_{N_P}$, and $\mathrm{SA}(\cdot)$ is a self-attention followed by a skip connection to adapt the input features to the current queries. In both cross- and self-attentions, positional encoding is applied to enable location-dependent query construction. In the remainder of the paper, the dense object queries $\mathbf{Q}_{N_Q}$ are denoted as $\mathbf{Q}$ for clarity

3.2 Dense object query decoder (DQD)

The dense queries $\mathbf{Q}$ from Section 3.1 are decoded into object detections by a dense object query decoder (DQD). Note that the spatial reduction of image by the SAM backbone may lead to encoding several small objects into the same query in $\mathbf{Q}$. To address this, the object queries are first *spatially unpacked* into high-resolution dense object queries i.e., $\mathbf{Q}^{HR} \in \mathbb{R}^{H \times W \times d}$, where $H = H_0/2$, $W = W_0/2$ and d is the number of feature channels. The unpacking process consists of three convolutional upsampling stages, with each stage composed of a 3×3 convolution, a Leaky ReLU and a $2\times$ bilinear upsampling. To facilitate unpacking of small objects, the features after the second stage are concatenated by the SAM-HQ features [11] $\mathbf{f}^{HQ}$ before feeding into the final stage.

Finally, the objectness score $\mathbf{y}^{\mathrm{O}} \in \mathbb{R}^{H \times W \times 1}$ is calculated by a simple transform, i.e., $\mathbf{y}^{\mathrm{O}} = \mathrm{LRelu}(\mathbf{W}_O \cdot \mathbf{Q}^{HR})$, where $\mathbf{W}_O$ is a learned projection matrix and $\mathrm{LReLU}(\cdot)$ is a Leaky ReLU. Each query is also decoded into the object pose by a three-layer MLP, i.e., $\mathbf{y}^{\mathrm{BB}} = \sigma(\mathrm{MLP}(\mathbf{Q}^{HR}))$, where $\sigma(\cdot)$ is a sigmoid function and $\mathbf{y}^{\mathrm{BB}} \in \mathbb{R}^{H \times W \times 4}$ are bounding box parameters in the *tlrb* format [27].

3.3 Detections extraction and refinement

The final detections are extracted from $\mathbf{y}^{\mathrm{O}}$ and $\mathbf{y}^{\mathrm{BB}}$ as follows. Bounding box parameters are read out from $\mathbf{y}^{\mathrm{BB}}$ at locations of local maxima on a thresholded $\mathbf{y}^{\mathrm{O}}$ (using a 3×3 nonmaxima suppression, NMS). The bounding boxes are refined by feeding them as prompts into a SAM decoder [12] on the already computed backbone features $\mathbf{f}^I$. The boxes are refitted to the masks by min-max operation and finally non-maxima suppression with $\mathrm{IoU} = 0.5$ is applied to remove duplicate detections. This process thus yields the predicted bounding boxes $\mathbf{B}^P$ and their corresponding masks $\mathbf{M}^P$.

3.4 A novel loss for dense detection training

GeCotraining requires supervision on the dense objectness scores $\mathbf{y}^O$ and the bounding box parameters $\mathbf{y}^{BB}$. Ideally, a network should learn to predict points on objects that can be reliably detected by a NMS, and also from which the bounding box parameters can be reliably predicted. We thus propose a new dense object detection loss that pursues this property.

Following the detection step (Section 3.2) in the forward pass, a set of local maxima $\{i\}_{i=1:N_{\mathrm{DET}}}$ is identified by applying a NMS on $\mathbf{y}^O$ and keeping all maxima higher than the median response, to ensure detection redundancy. The maxima are then labelled as *true positives* (TP) and *false positives* (FP) by applying Hungarian matching [13] between their bounding box parameters $\{\mathbf{y}_i^{BB}\}_{i=1:N_{\mathrm{DET}}}$ and the ground truth bounding boxes $\{\mathbf{B}_j^{GT}\}_{j=1:N_{\mathrm{GT}}}$. To account for missed detections, centers of the non-matched ground truth bounding boxes are added to the list of local maxima and labeled as *false negatives* (FN). The new training loss is thus defined as

$$\mathcal{L} = -\sum_{i \in \text{TP}} \text{gIoU}(\mathbf{y}_i^{BB}, \boldsymbol{B}_{\text{HUN}(i)}^{GT}) + \sum_{i \in \text{TP} \cup \text{FN}} (\mathbf{y}_i^{\text{O}} - 1)^2 + \sum_{i \in \text{FP}} (\mathbf{y}_i^{\text{O}} - 0)^2, \tag{3}$$

where $\text{gIoU}(\cdot,\cdot)$ is the generalized IoU [25], and $\text{HUN}(i)$ is the ground truth index matched with the i-th predicted bounding box. Note that the new loss simultaneously optimizes the bounding box prediction quality, promotes locations with better box prediction capacity that can be easily detected by a NMS, and enables automatic hard-negative mining in the objectness score via FP identification.

4 Experiments

Implementation details.

Using the SAM [12] backbone, GeCo reduces the input image by a factor $r = 16$, and projects the features into $d = 256$ channels (Section 3). In DQE (Section 3.1), $N_P = 3$ iterations are applied in prototype generalization (1) and $N_Q = 2$ iterations in dense object query construction (2). Following the established test-time practice [20; 19; 26], the input image is scaled to fit $W_0 = H_0 = 1536$ if the average of the exemplars widths and heights is below 25 pixels, otherwise it is downscaled to fit the average of the exemplar width and height to 80 pixels and zero-padded to $W_0 = H_0 = 1024$. As in [20], the zero-shot GeCo is run twice, first to estimate the objects size and then again on the resized image.

Training details. With the SAM backbone frozen, GeCo is pretrained with the classical loss [20] for initialization and is then trained for 200 epochs with the proposed dense detection loss (3) using a mini-batch size of 8, AdamW [16] optimizer, with initial learning rate set to 10^{-4}, and weight decay of 10^{-4}. The training is done on 2 A100s GPUs with standard scale augmentation [20; 4] and zero-padding images to 1024×1024 resolution. For the zero-shot setup, the few-shot GeCo is frozen and only the zero-shot prototype extension is trained for 10 epochs. Thus *the same trained network* is used in all low-shot setups.

Evaluation metrics and datasets. Standard datasets are used. The FSCD147 [19] is a detection-oriented extension of the FSC147 [24], which contains 6135 images of 147 object classes, split into 3659 training, 1286 validation, and 1190 test images. The splits are disjoint such that target object categories in test set are not observed in training. The objects are manually annotated by bounding boxes in the test set [19], while in the train set, the bounding boxes are obtained from point estimates by SAM [35]. For each image, three exemplars are provided. The second dataset is FSCD-LVIS [19], derived from LVIS [8] and contains 377 categories. Specifically, the unseen-split is used (3959 training and 2242 test images), which ensures that test-time object categories are not observed during training.

The standard evaluation protocol [24; 26; 32] with Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) is followed to evaluate the counting accuracy. Following [19], Average Precision (AP) and Average Precision at IoU=50 (AP50) is used on the same output to evaluate the detection accuracy.

4.1 Experimental Results

Few-shot counting and detection. GeCo is compared with state-of-the-art density-based counters (which only estimate the total count) LOCA [4], CounTR [14], SAFECount [32], BMNet+ [26], VCN [22], CFOCNet [31], MAML [7], FamNet [24] and CFOCNet [31], and with detection-based counters C-DETR [19], SAM-C [18], PSECO [35], and DAVE [20], which also provide object locations by bounding boxes. Results are summarized in Table 1.

GeCo outperforms both recent state-of-the-art detection-based counters DAVE [20] and PSECO [35] by a 24% and 39% MAE, and a remarkable 27% and 51% RMSE on the test split, setting a new state-of-the-art in detection-based counting. Notably, GeCo outperforms all single-stage density-based counters (top part of Table 1) by a large margin, which makes it the first detection-based counter that outperforms the longstanding total count estimation winner LOCA [4] by a remarkable 27% MAE and 4% RMSE on test split. In this respect, GeCo closes the performance gap that has been present for several years between state-of-the-art density-, and detection-based counters.

Table 1: Few-shot density-based methods (top part) and detection-based methods (bottom part) performances on the FSCD147 [19].

	Validation set				Test set			
Method	MAE (↓)	RMSE(↓)	AP(↑)	AP50(↑)	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
GMN [17] ACCV18	29.66	89.81	-	-	26.52	124.57	-	-
MAML [7] ICML17	25.54	79.44	-	-	24.90	112.68	-	-
FamNet [24] CVPR21	23.75	69.07	-	-	22.08	99.54	-	-
CFOCNet [31] WACV21	21.19	61.41	-	-	22.10	112.71	-	-
BMNet+ [26] CVPR22	15.74	58.53	-	-	14.62	91.83	-	-
VCN [22] CVPRW22	19.38	60.15	-	-	18.17	95.60	-	-
SAFEC [32] WACV23	15.28	47.20	-	-	14.32	85.54	-	-
CounTR [14] BMVC22	13.13	49.83	-	-	11.95	91.23	-	-
LOCA [4]ICCV23	10.24	32.56	-	-	10.79	56.97	-	-
C-DETR [19] ECCV22	20.38	82.45	17.27	41.90	16.79	123.56	22.66	50.57
SAM-C [18] arXiv23	31.20	100.83	20.08	39.02	27.97	131.24	27.99③	49.17
PSECO [35] CVPR24	15.31③	68.36③	32.12②	60.02③	13.05③	112.86③	42.98②	73.33②
DAVE [20] CVPR24	9.75②	**40.30**①	24.20③	61.08②	10.45②	74.51②	26.81	62.82③
GeCo (ours)	**9.52**①	43.00②	**33.51**①	**62.51**①	**7.91**①	**54.28**①	**43.42**①	**75.06**①

In terms of detection performance, GeCo surpasses all state-of-the-art methods, including PSECO [35] which uses both, SAM [12] and CLIP [21] backbones, by 1% AP, and 2% AP50. Note that GeCo also outperforms PSECO in count prediction by a large margin ($\sim$40%), which is crucial, as an ideal detection counter should deliver both accurate total count prediction as well as feature good object localization. In addition, GeCo also outperforms SAM-C, which is a low-shot counting and detection extension of SAM by 70%/55% MAE/AP. To demonstrate the impact of the refinement step in existing methods, we modified DAVE [20] by feeding predicted bounding boxes to SAM [12] as prompts, which results in a GeCo-like box refinement. Compared to modified DAVE, GeCo achieves 21% and 16% higher AP and AP50, respectively, indicating that the reason for the excellent performance of GeCo lies in its architecture, rather than in segmentation-based refinement.

Figure 3 visualizes detections for qualitative analysis[1]. GeCo predicts bounding boxes of superior quality for elongated objects (row 1), validating the selection of bounding box prediction locations. On detecting complex, non-blob-like objects (row 2), GeCo outperforms concurrent methods, by more accurately generalizing the prototypes. In densely populated scenes (row 3), GeCo achieves higher accuracy in both count and bounding box predictions. In comparison with state-of-the-art, GeCo features better object discrimination (row 4), which can be attributed to better prototype generalization in DQE (Section 3.1) and hard negative mining in the new loss from Section 3.4.

We further evaluate GeCo on FSCD-LVIS [19]. Results in Table 2 show that GeCo outperforms the best method by significant 178% and 73% in AP and AP50, respectively, and performs on-par in terms of MAE. The experiment supports the results on FSCD147.

Table 2: Few-shot counting and detection on the FSCD-LVIS [19] "unseen" split.

	Count		Detection	
Method	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
FSDetView-PB [29] TPAMI22	28.99	40.08	1.03	2.89
AttRPN-PB [6] CVPR22	39.16	46.09	3.15	7.87
C-DETR [19] ECCV22	23.50③	35.89③	3.85③	11.28③
DAVE [20] CVPR24	15.47②	**25.95**①	4.12②	14.16②
GeCo (ours)	**15.26**①	28.80②	**11.47**①	**24.49**①

One-shot counting and detection. In the one-shot counting setup, a single exemplar is considered. Table 3 shows comparison with the recent density- and detection-based methods. GeCo outperforms all state-of-the-art single-stage density-based counters, outperforming LOCA$_{1-\text{shot}}$ [4] version specifically trained for the one-shot setup, by a significant margin of 35% MAE and 20% RMSE on validation and test split, respectively. GeCo also outperforms state-of-the-art method PSECO [35] by

[1] See supplementary material for more visualizations

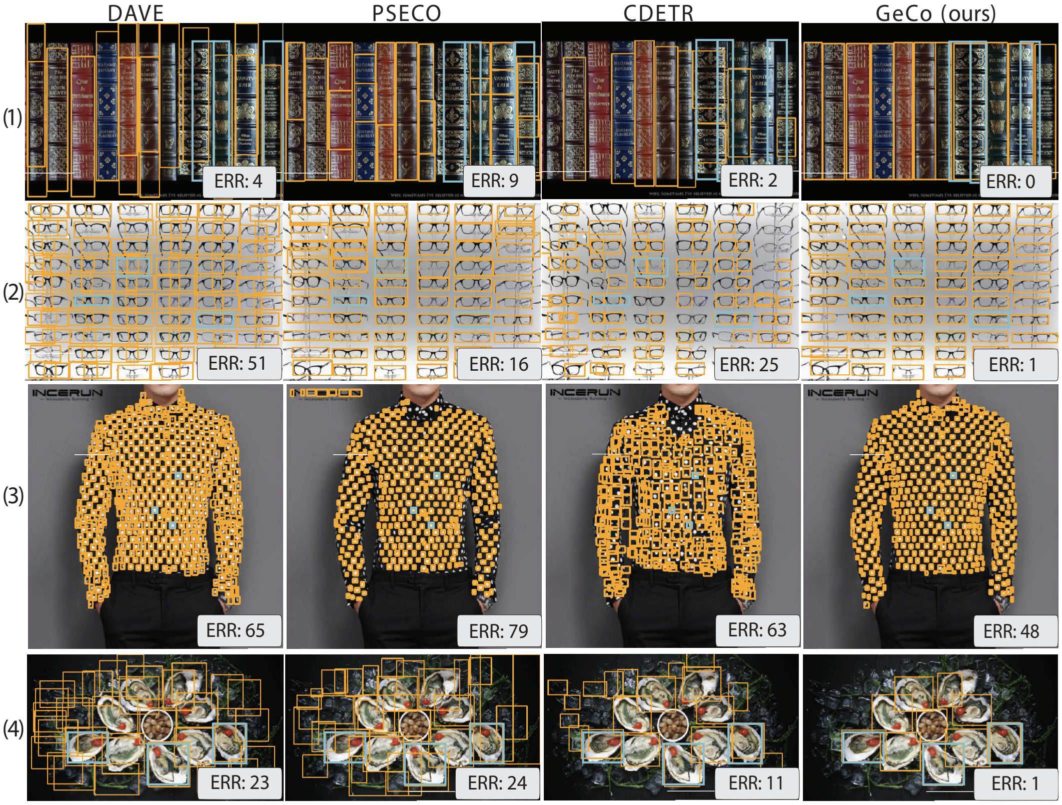

Figure 3: Compared with state-of-the-art few-shot detection-based counters DAVE [20], PSECO [35], and C-DETR [19], GeCo delivers more accurate detections with less false positives and better global counts. Exemplars are delineated with blue color, while segmentations are not shown for clarity.

4% AP and 5% AP50, and by significant 45% MAE and 49% RMSE on test split. These results show that GeCo features remarkable robustness to the number of exemplars since a single network (without re-training or fine-tuning) is used in both three- and one-shot setups. In particular, the performance drops by only 2%/11% of MAE/RMSE and 1%/1% AP/AP50 on the test split between both setups. In a *one-shot* setting, GeCo surpasses state-of-the-art *three-shot* models. Specifically, one-shot GeCo achieves 22% and 20% lower MAE and RMSE, respectively, compared to three-shot DAVE, and outperforms three-shot PSECO by 38% and 46% on the FSCD147 test set. These results highlight the robustness of GeCo to the number of exemplars, demonstrating its ability to handle inputs with lowered visual diversity.

Zero-shot counting and detection. Table 4 reports the results of the zero-shot GeCo compared with best zero-shot variants of the density-based counters, LOCA [4], CounTR [14], RepRPN-C [23], RCC [10] and with the zero-shot variant of the best detection-based counter DAVE [20]. GeCo outperforms DAVE [20] by a significant margin of 14% MAE and 6% RMSE on the test set. Furthermore, it outperforms all density-based methods and sets a new state-of-the-art result on FSC [24] benchmark, by outperforming the top-performer CounTR [14] by impressive 6% MAE on the test set. Since the zero-shot variant of the recent detection-based counter PSECO [35] does not exist, we include its prompt-based variant for complete evaluation (i.e., target object class is specified by a text prompt). Even in this setup, the zero-shot GeCo outperforms the prompt-based PSECO by 20% MAE 16% RMSE, and 2% AP50 demonstrating great robustness to different counting and detection scenarios.

Mutliclass images. To further verify the robustness of the proposed method, we validate it on a subset of FSCD147, that contain images with multiple object classes (FSCD147$_{\text{mul}}$) [20]. Results in Table 5 indicate that most state-of-the-art methods non-discriminatively count all objects in an image due to prototype over-generalization. GeCo outperforms all single-stage density-, and detection-based counters on multiclass images by at least 60%/67% in MAE/RMSE. This further verifies the

Table 3: One-shot density-based methods (top) and detection-based methods (bottom) on the FSCD147 [19].

	Validation set				Test set			
Method	MAE (↓)	RMSE(↓)	AP(↑)	AP50(↑)	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
GMN [17] ACCV18	29.66	89.81	-	-	26.52	124.57	-	-
CFOCNet [31] WACV21	27.82	71.99	-	-	28.60	123.96	-	-
FamNet [24] CVPR21	26.55	77.01	-	-	26.76	110.95	-	-
BMNet+ [26] CVPR22	17.89	61.12	-	-	16.89	96.65	-	-
CounTR [14] BMVC22	13.15	49.72	-	-	12.06	90.01	-	-
LOCA$_{\text{1-shot}}$ [4] ICCV23	11.36	38.04	-	-	12.53	75.32	-	-
PSECO [35] CVPR24	18.31③	80.73③	31.47②	58.53②	14.86③	118.64③	41.63②	70.87②
DAVE$_{\text{1-shot}}$ [20] CVPR24	10.98②	43.26②	18.00③	52.37③	11.54②	86.62②	19.46③	55.27③
GeCo (ours)	**9.97**①	**37.85**①	**32.82**①	**61.31**①	**8.10**①	**60.16**①	**43.11**①	**74.31**①

Table 4: Zero-shot density-based methods (top part), and detection-based methods (bottom part) on the FSCD147 [19]. The symbol $*$ denotes methods that also use text prompts as input.

	Validation set				Test set			
Method	MAE (↓)	RMSE(↓)	AP(↑)	AP50(↑)	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
RepRPN-C [23] ACCV22	29.24	98.11	-	-	26.66	129.11	-	-
RCC [10] arXiv22	17.49	58.81	-	-	17.12	104.5	-	-
CounTR [14] BMVC22	17.40	70.33	-	-	14.12	108.01	-	-
LOCA [4] ICCV23	17.43	54.96	-	-	16.22	103.96	-	-
PSECO [35]$*$ CVPR24	23.90③	100.33③	-	-	16.58③	129.77③	41.14②	69.03②
DAVE [20] CVPR24	15.71②	**60.34**①	16.31②	46.87②	15.51②	116.54②	18.55③	50.08③
GeCo (ours)	**14.81**①	64.95②	**31.04**①	**58.30**①	**13.30**①	**108.72**①	**41.27**①	**70.09**①

robustness of the proposed architecture, which benefits from the hard-negative mining in the proposed loss function, leads to more discriminative prototype construction and false positive reduction.

4.2 Ablation study

Dense object detection loss. To analyze the contribution of the new dense detection loss from Section 3.4, we trained GeCo using the standard loss [20; 35] that forms the ground truth objectness score by placing unit Gaussians on object centers – this variant is denoted by GeCo$_{\text{Gauss}}$. Table 6 shows that this leads to a substantial drop in total count estimation (38% RMSE, and 34% MAE) as well as in object detection (6% AP, and 3% AP50). Qualitative results are provided in Figure 4. As observed in columns 3 and 5, the classical unit-Gaussian-based loss [20; 35] forces the network to predict object locations from the object centers, which are not necessarily optimal for bounding box prediction. In contrast, the proposed dense detection loss enables the network to learn optimal point prediction, which more accurately aggregates information of the object pose. Columns 1 and 2 indicate that the new loss leads to superior detection of objects composed of blob-like structures avoiding false detections on individual object parts. Furthermore, the hard-negative mining integrated in the new loss design leads to better discriminative power of the detections and subsequent reduction of false positives (column 4).

Table 5: Performance on FSCD147 [19] test split, and its multiclass subset FSCD147$_{\text{mul}}$.

	FSCD147		FSCD147$_{\text{mul}}$	
Method	MAE(↓)	RMSE(↓)	MAE(↓)	RMSE(↓)
C-DETR [19] ECCV22	16.79	123.56	23.09	30.09
PSECO [35] CVPR24	13.05	112.86	25.73	44.95
LOCA [4] ICCV23	10.79③	56.97②	21.28	43.67
CounTR [14] BMVC22	11.95	91.23	14.56③	27.41③
DAVE [20] CVPR24	10.45②	74.51③	**3.09**①	**5.28**①
GeCo (ours)	**7.91**①	**54.28**①	5.88②	9.17②

Table 6: Ablation study on the FSCD147 [19] validation split.

	Counting		Detection	
Method	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
GeCo	**9.52**	**43.00**	**33.51**	**62.51**
$\text{GeCo}_{\text{Gauss}}$	12.79	59.33	31.43	60.73
$\text{GeCo}_{\overline{\text{HQ}}}$	10.04	47.11	33.08	62.50
$\text{GeCo}_{\overline{\text{p}^s}}$	9.97	46.93	32.56	61.19
$\text{GeCo}_{\overline{\text{Ref}}}$	10.26	43.33	24.63	61.57
$\text{GeCo}_{\overline{\mathbf{Q}}}$	10.32	45.14	33.01	61.68
$\text{GeCo}_{\text{DETR}}$	11.45	52.46	32.24	61.60

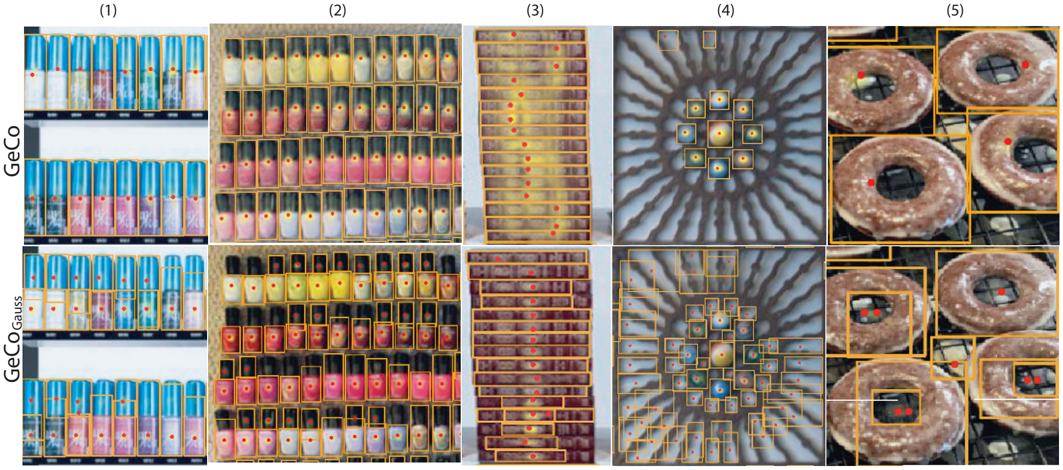

Figure 4: Response maps (in yellow), and locations for bounding box predictions (red dots) when using the proposed (first row) and the standard [20; 4; 35] (second row) training loss.

Architecture. To evaluate the impact of concatenating the SAM-HQ [11] features in the query *unpacking* process in the DQD module (Section 3.2), we remove these features in $\text{GeCo}_{\overline{\text{HQ}}}$. Table 6 shows a counting performance drop 5% MAE and 10% RMSE. To validate the importance of modeling exemplar shapes, i.e., width and height, with prototypes $\mathbf{p}^S$, we omit them in $\text{GeCo}_{\overline{\mathbf{p}^S}}$. We observe a substantial performance decrease of 5% MAE, and 9% RMSE. Finally, we remove bounding box refinement in the detection refinement module (Section 3.3), and denote the variant as $\text{GeCo}_{\overline{\text{Ref}}}$. While this does not affect the global count estimation accuracy, we observe a 26% decrease in AP and 2% decrease in AP50. It is worth noting, that bounding box refinement improves the accuracy of predicted bounding boxes, however it does not enhance object presence detection.

To verify the importance of the DQE module (Section 3.1), we replace the dense object queries $\mathbf{Q}$ construction step (2) with a standard self-attention, i.e., $\mathbf{Q} = \text{SA}(\mathbf{P})_{3\times}$. This leads to a 8% MAE and 5% RMSE performance drop, verifying the proposed approach. To evaluate the importance of using image features as queries in (2), we change the object query construction to $\mathbf{Q}_j = \text{CA}(\text{SA}(\mathbf{Q}_{j-1}), \mathbf{f}^I, \mathbf{f}^I)$ to follow a standard DETR [1]-like approach, and denote it as $\text{GeCo}_{\text{DETR}}$. We observe a 20% MAE and 22% RMSE decrease in counting performance.

5 Conclusion

We proposed GeCo, a novel single-stage low-shot counter that integrates accurate detection, segmentation, and count prediction within a unified architecture, and covers all low-shot scenarios with a single trained model. GeCo features remarkables dense object query formulation, and prototype generalization across the image, rather than just into a few prototypes. It employs a novel loss function specifically designed for detection tasks, avoiding the biases of traditional Gaussian-based losses. The loss optimizes detection accuracy directly, leading to more precise detection and counting.

The main limitation of the presented method is that it cannot process arbitrarily large images, due to memory constraints, since it, as all current methods, operates globally. In future work, we will explore local counting, incremental image-wide count aggregation, optimizing inference speed utilizing a faster backbone [34].

Extensive analysis showcases that GeCo surpasses the best detection-based counters by approximately 25% in total count MAE, achieving state-of-the-art performance in a few-shot counting setup and demonstrating superior detection capabilities. GeCo showcases remarkable robustness to the number of provided exemplars, and sets a new state-of-the-art in one-shot as well as zero-shot counting.

Acknowledgements. This work was supported by Slovenian research agency program P2-0214 and projects J2-2506, L2-3169, Z2-4459, J2-60054, and by supercomputing network SLING (ARNES, EuroHPC Vega - IZUM).

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[3] Zhe Dai, Huansheng Song, Xuan Wang, Yong Fang, Xu Yun, Zhaoyang Zhang, and Huaiyu Li. Video-based vehicle counting framework. *IEEE Access*, 7:64460–64470, 2019.

[4] Nikola Djukic, Alan Lukezic, Vitjan Zavrtanik, and Matej Kristan. A low-shot object counting network with iterative prototype adaptation. In *ICCV*, 2023.

[5] Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, et al. U-net: deep learning for cell counting, detection, and morphometry. *Nature methods*, 16(1):67–70, 2019.

[6] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *CVPR*, pages 4013–4022, 2020.

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[8] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[10] Michael Hobley and Victor Prisacariu. Learning to count anything: Reference-less class-agnostic counting with weak supervision. *arXiv preprint arXiv:2205.10203*, 2022.

[11] Lei Ke, Mingqiao Ye, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, Fisher Yu, et al. Segment anything in high quality. *Advances in Neural Information Processing Systems*, 36, 2024.

[12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[13] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[14] Chang Liu, Yujie Zhong, Andrew Zisserman, and Weidi Xie. Countr: Transformer-based generalised visual counting. In *BMVC*. BMVA Press, 2022.

[15] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. In *CVPR*, June 2019.

[16] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *ICLR*, 2019.

[17] Erika Lu, Weidi Xie, and Andrew Zisserman. Class-agnostic counting. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14*, pages 669–684. Springer, 2019.

[18] Zhiheng Ma, Xiaopeng Hong, and Qinnan Shangguan. Can sam count anything? an empirical study on sam counting, 2023.

[19] Thanh Nguyen, Chau Pham, Khoi Nguyen, and Minh Hoai. Few-shot object counting and detection. In *ECCV*, pages 348–365. Springer, 2022.

[20] Jer Pelhan, Alan Lukežič, Vitjan Zavrtanik, and Matej Kristan. Dave – a detect-and-verify paradigm for low-shot counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024.

[21] Alec Radford, Ilya Sutskever, Jong Wook Kim, Gretchen Krueger, and Sandhini Agarwal. Clip: connecting text and images. *OpenAI. https://openai. com/blog/clip/*, 2021.

[22] Viresh Ranjan and Minh Hoai. Vicinal counting networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4221–4230, June 2022.

[23] Viresh Ranjan and Minh Hoai Nguyen. Exemplar free class agnostic counting. In *Proceedings of the Asian Conference on Computer Vision*, pages 3121–3137, 2022.

[24] Viresh Ranjan, Udbhav Sharma, Thu Nguyen, and Minh Hoai. Learning to count everything. In *CVPR*, pages 3394–3403, 2021.

[25] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union. In *CVPR*, 2019.

[26] Min Shi, Hao Lu, Chen Feng, Chengxin Liu, and Zhiguo Cao. Represent, compare, and learn: A similarity-aware framework for class-agnostic counting. In *CVPR*, pages 9529–9538, June 2022.

[27] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[29] Yang Xiao, Vincent Lepetit, and Renaud Marlet. Few-shot object detection and viewpoint estimation for objects in the wild. *IEEE TPAMI*, 45(3):3090–3106, 2022.

[30] Weidi Xie, J Alison Noble, and Andrew Zisserman. Microscopy cell counting and detection with fully convolutional regression networks. *Computer methods in biomechanics and biomedical engineering: Imaging & Visualization*, 6(3):283–292, 2018.

[31] Shuo-Diao Yang, Hung-Ting Su, Winston H Hsu, and Wen-Chin Chen. Class-agnostic few-shot object counting. In *WACV*, pages 870–878, 2021.

[32] Zhiyuan You, Kai Yang, Wenhan Luo, Xin Lu, Lei Cui, and Xinyi Le. Few-shot object counting with similarity-aware feature enhancement. In *WACV*, pages 6315–6324, 2023.

[33] Vitjan Zavrtanik, Martin Vodopivec, and Matej Kristan. A segmentation-based approach for polyp counting in the wild. *Engineering Applications of Artificial Intelligence*, 88:103399, 2020.

[34] Xu Zhao, Wenchao Ding, Yongqi An, Yinglong Du, Tao Yu, Min Li, Ming Tang, and Jinqiao Wang. Fast segment anything. *arXiv preprint arXiv:2306.12156*, 2023.

[35] Huang Zhizhong, Dai Mingliang, Zhang Yi, Zhang Junping, and Shan Hongming. Point, segment and count: A generalized framework for object counting. In *CVPR*, 2024.

A Supplemental material

This supplementary material provides additional comparisons of GeCo with state-of-the-art under a non-standard experiment, and provides additional qualitative examples.

Performance analysis on a non-standard experiment. The analysis of the detection methods in Section 4 adheres to the standard evaluation protocol [19; 20], where a method predicts a set of bounding boxes for each image. The estimated count is the total number of predicted bounding boxes, and evaluated by the MAE/RMSE measures, while the detection accuracy is evaluated by AP/AP50 measures. Both measures are computed on *the same set* of output bounding boxes.

However, in the PSECO [35] paper, the reported evaluation deviated from the standard one in an important detail. Namely *different* outputs were evaluated under MAE/RMSE and AP/AP50 to fully evaluate the different properties of the method. AP/AP50 was computed in *all* output bounding boxes, while the MAE/RMSE were computed on a subset of the boxes, obtained by thresholding the response score. In Section 4, we evaluated all methods, including PSECO under the standard experiment. Nevertheless, we additionally report GeCo evaluated under the said non-standard PSECO experiment in Table 7.

Even in this setup, GeCo outperforms PSECO by 4%/4% AP/AP50, and 1%/2% AP/AP50, on validation and test set, respectively, again with a substantially lower global count errors (∼50% MSE/RMSE reduction). These results shed an important insight. A method producing false positives, which increase the count errors and reduce its usefulness for counting, might achieve good detection-oriented performance measures. Thus for counting performance evaluation, the MAE/RMSE should be considered primary measures, while AP/AP50 should be secondary, as they are less strict towards false positive detections.

Table 7: Few-shot detection-based counting evaluation on FSCD147 [19] under the non-standard evaluation protocol [35].

	Validation set				Test set			
Method	MAE (↓)	RMSE(↓)	AP(↑)	AP50(↑)	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
PSECO [35]	15.31②	68.34②	32.71②	62.03②	13.05②	112.86②	43.53②	74.64②
GeCo (ours)	**9.52**①	**43.00**①	**34.07**①	**64.23**①	**7.91**①	**54.28**①	**43.89**①	**76.18**①

Performance in crowded scenes. To evaluate counting performance in crowded scenes, we constructed a subset of the FSCD147 test set by including images with at least 200 objects and a maximal average exemplar size of 30 pixels. Notably, the new subset contains 42 images, averaging 500 objects per image, thus featuring dense scenes with small objects. Three top-performing methods from Table 1 were included in the comparison and are shown in Table 8. GeCo outperforms both PSECO and DAVE by a significant margin, e.g., outperforming DAVE by 23% in MAE and 36% in RMSE, which demonstrates superior counting performance on small, densely populated objects.

Table 8: Few-shot counting in crowded scenes, comparing the top-three detection-based counters from Table 1.

	MAE	RMSE
PSECO [35] CVPR24	173.64	594.91
DAVE [20] CVPR24	81.38	383.93
GeCo (ours)	62.60	242.82

Qualitative results. Figure 5 compares GeCo with PSECO [35], which achieves the best AP/AP50 measures among the related counters. GeCo shows robust performance, achieving high precision (see Figure 5 block 1), while achieving high recall (see Figure 5 block 2). This is challenging for related methods, particularly in densely populated scenes or with small objects. Furthermore, GeCo outperforms PSECO on elongated or more complex objects (see Figure 5 block 3), better exploiting the exemplars.

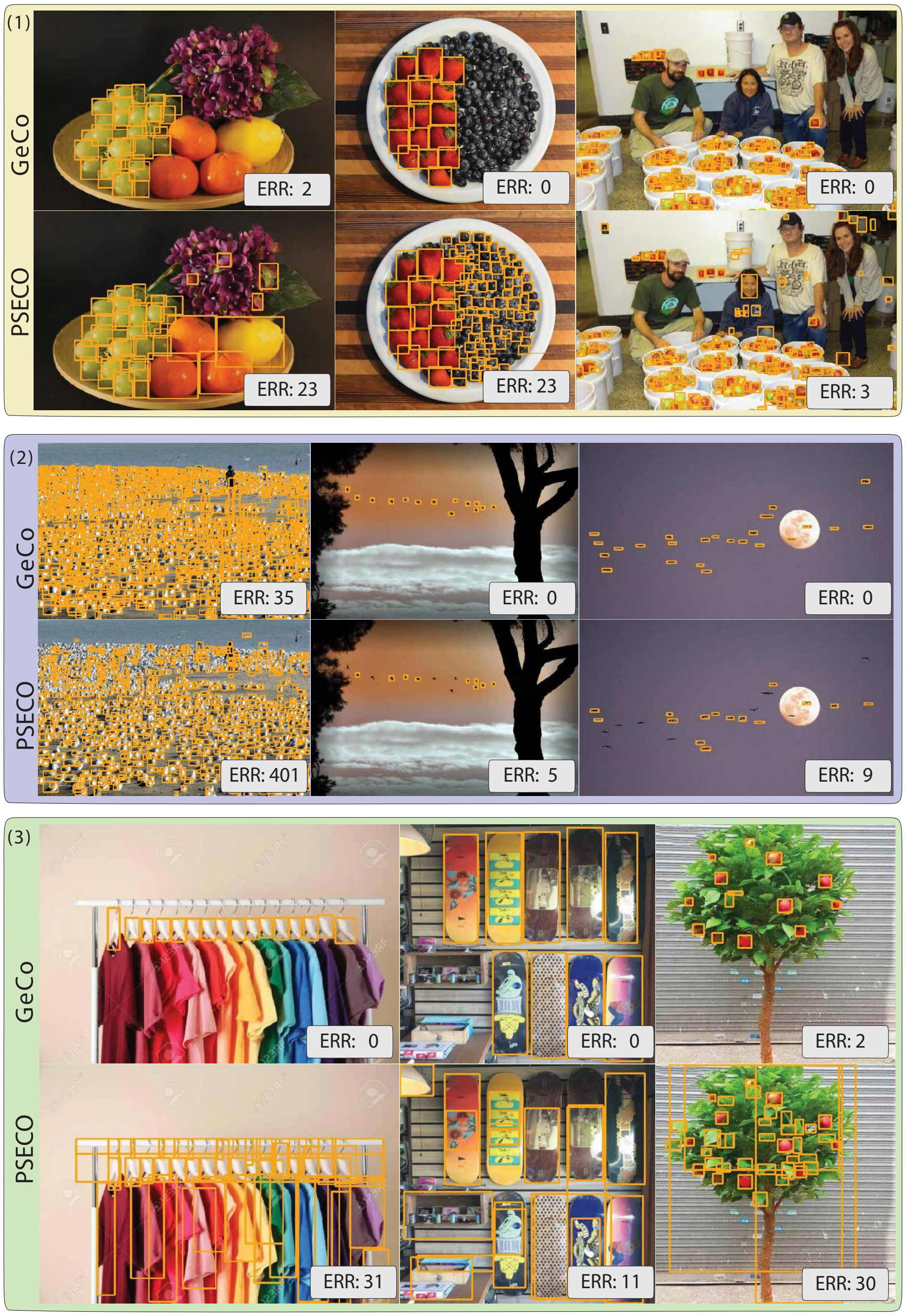

Figure 5: Comparison of few-shot counting on FSCD147. Exemplars are shown with red color and ERR indicates count error.

Figure 6 visualizes the segmentations produced by GeCo, in a few-shot setup, of various objects in diverse scenes. GeCo is robust to noise, achieves discriminative segmentations, and performs well on elongated, non-blob-like objects and in dense scenarios. Figure 7 compares GeCo with all state-of-the-art detection counters [20; 19; 35]. GeCo achieves superior counting performance, and predicts more accurate bounding boxes.

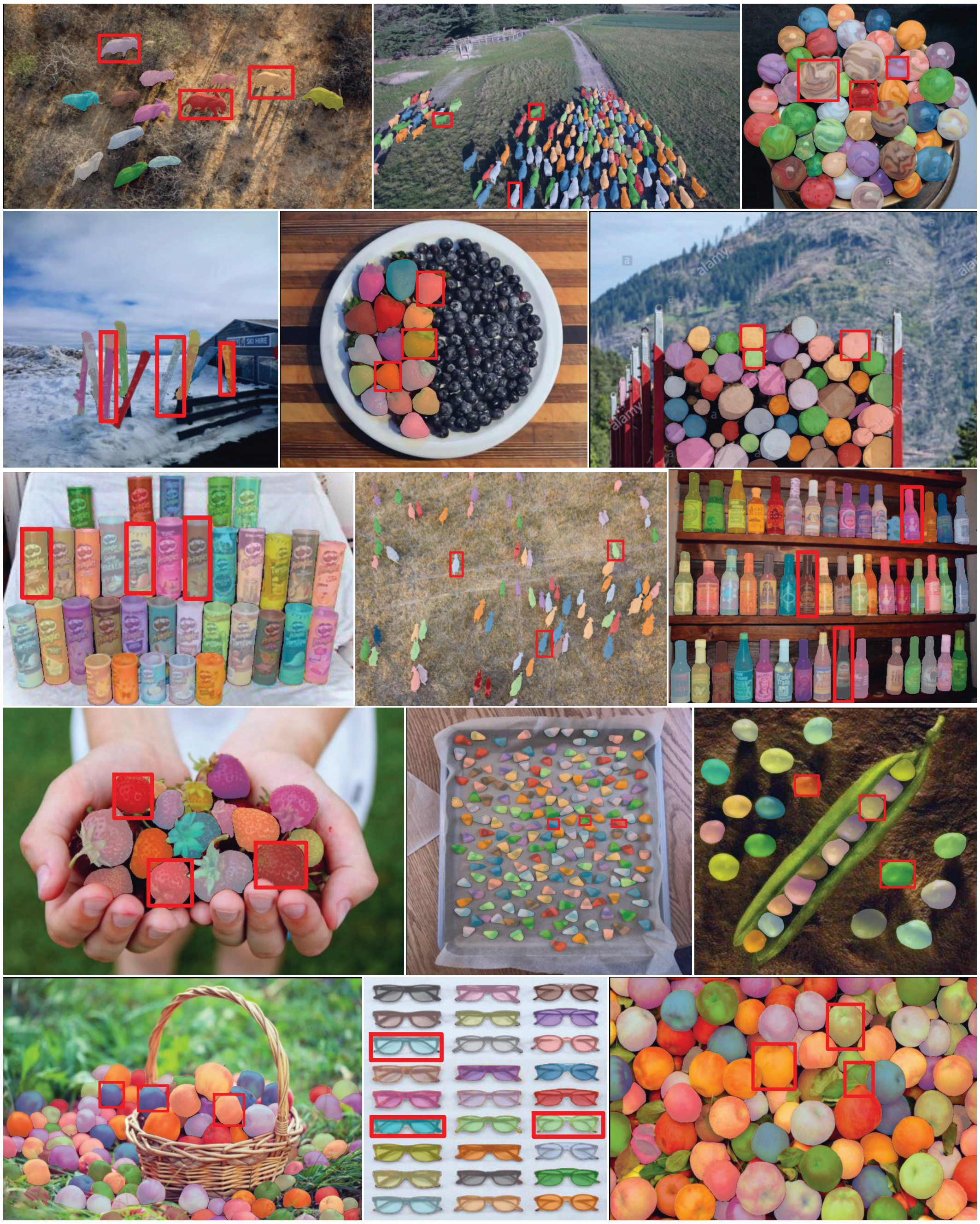

Figure 6: Segmentation quality of GeCo on diverse set of scenes and object types. Exemplars are denoted by red bounding boxes.

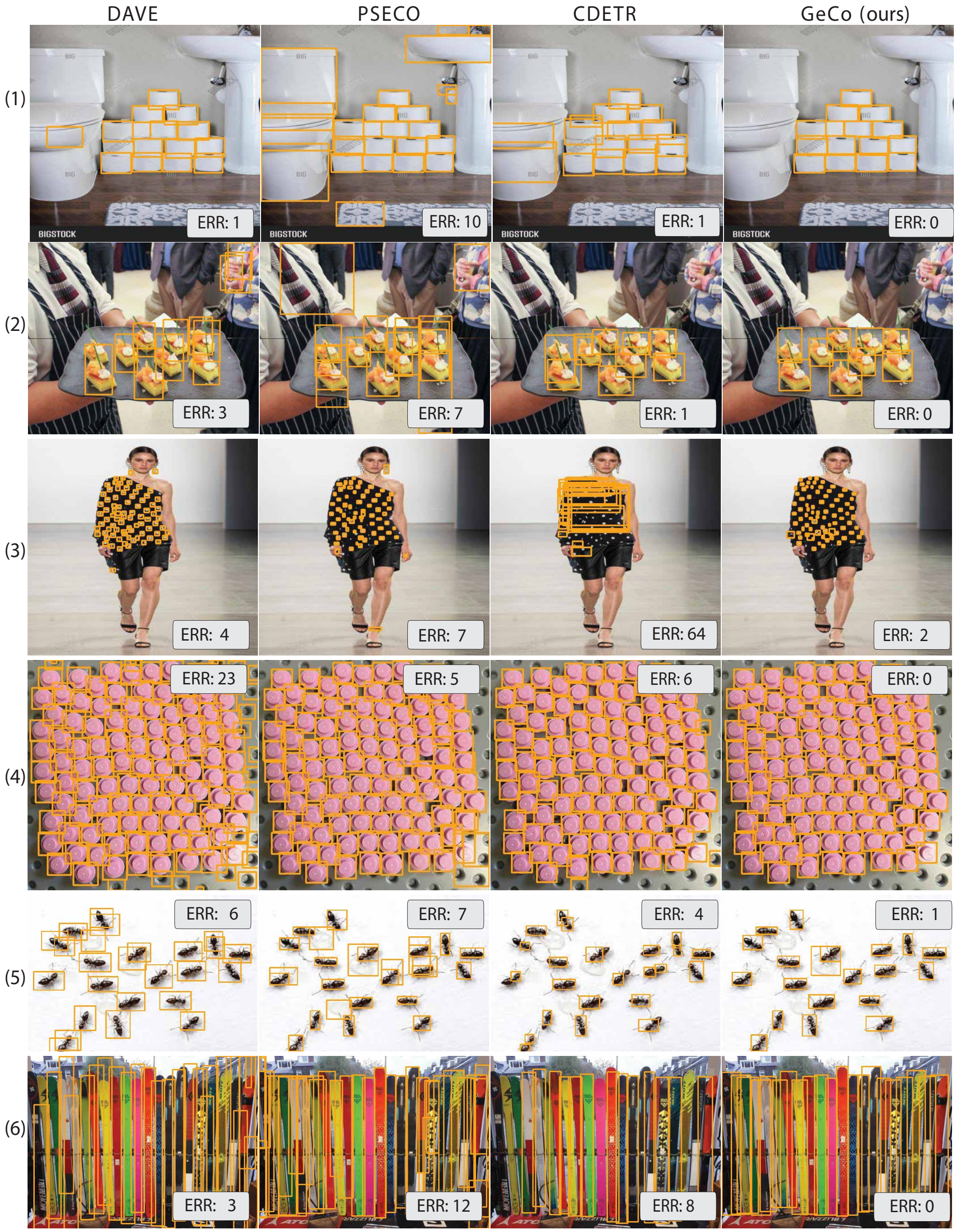

Figure 7: Comparison of few-shot counting and detection on FSCD147. ERR indicates count error.

In Figure 8 performance of GeCo is qualitatively demonstrated on examples with high intra-class variance. Image (a) displays marbles of various colors and textures (notable visual intra-class variance), all correctly detected and still distinguished from a visually similar coin. Example (b) shows donuts with different colors of decorations, all accurately counted and detected by GeCo. Image (c) contains bottles of various sizes, shapes, and colors, each with a distinct sticker. Image (d) features transparent food containers with differently colored and shaped fruits inside, successfully detected despite significant visual diversity. Examples (e) and (f) illustrate GeCo's robustness in detecting objects with high shape variance, including partially visible birds (notable object shape intra-class variance).

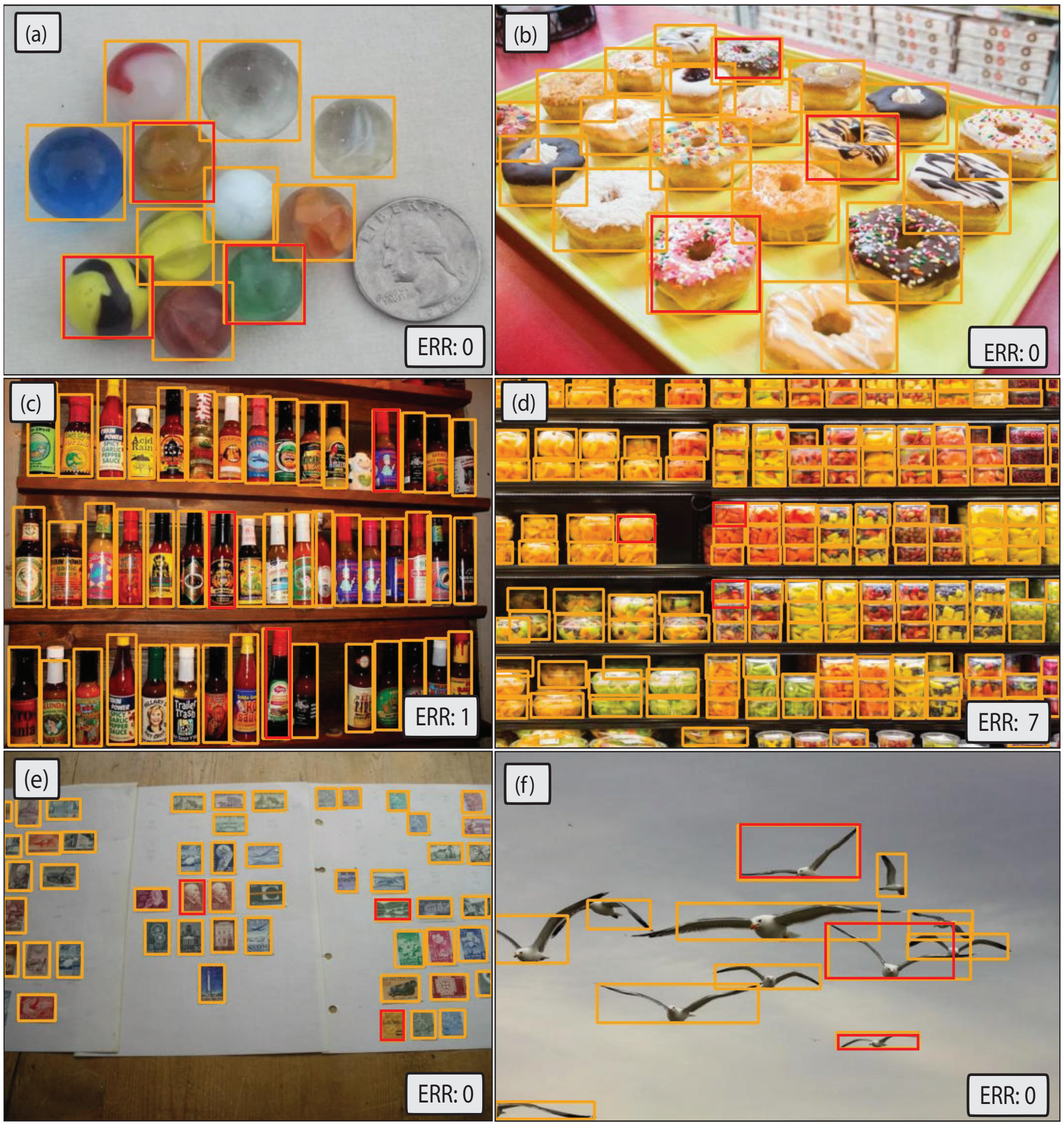

Figure 8: Few-shot detection and counting with GeCo on images with high intra-class object appearance variation. Orange and red bounding boxes denote detections and exemplars, respectively. Count error is denoted by ERR.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and precede the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS paper checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the abstract and introduction, we stress out the main contributions, and results of the presented method.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: We discuss the main limitation of the presented method in the conclusion.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: We do not derive theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The method is clearly described, making it possible to re-implement. We will make the code publically available upon acceptance.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Upon acceptance, we will make the code and models publically available.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper clearly specifies all implementation details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars and statistical significance are not reported by state-of-the-art methods in their respective papers. We omit the usage of error bars, as it would be too computationally expensive.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In the implementation details we clearly state the computer resources needed to train presented model.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We reviewed the Code of Ethics and found no violations in our work.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our work will not make any societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our data and models are not a risk of misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite all respective papers of publically available benchmarks and datasets used in our work.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not introduce any new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: We do not conduct research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our research does not include any research with human subjects.

    Guidelines:

    - We do not conduct research with human subjects.
    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.