

A Low-Shot Object Counting Network With Iterative Prototype Adaptation

Nikola Đukić, Alan Lukežič, Vitjan Zavrtanik, Matej Kristan

Faculty of Computer and Information Science, University of Ljubljana, Slovenia

nikola.m.djukic@gmail.com, {alan.lukezic, vitjan.zavrtanik, matej.kristan}@fri.uni-lj.si

Abstract

We consider low-shot counting of arbitrary semantic categories in the image using only few annotated exemplars (few-shot) or no exemplars (no-shot). The standard few-shot pipeline follows extraction of appearance queries from exemplars and matching them with image features to infer the object counts. Existing methods extract queries by feature pooling which neglects the shape information (e.g., size and aspect) and leads to a reduced object localization accuracy and count estimates.

We propose a Low-shot Object Counting network with iterative prototype Adaptation (LOCA). Our main contribution is the new object prototype extraction module, which iteratively fuses the exemplar shape and appearance information with image features. The module is easily adapted to zero-shot scenarios, enabling LOCA to cover the entire spectrum of low-shot counting problems. LOCA outperforms all recent state-of-the-art methods on FSC147 benchmark by 20-30% in RMSE on one-shot and few-shot and achieves state-of-the-art on zero-shot scenarios, while demonstrating better generalization capabilities. The code and models are available here: <https://github.com/djukicn/loca>.

1. Introduction

Object counting considers estimation of the number of specific objects in the image. Solutions based on object detectors have been extensively explored for categories such as people [1, 33], cars [20, 12] or animal species [2, 32]. However, these methods require huge annotated training datasets and are not applicable to counting new, previously unobserved, classes with potentially only few annotations. The latter problem is explored by low-shot counting, which encompasses few-shot and zero-shot counting. Few-shot counters count all present objects of some class with only few of them annotated by bounding boxes (exemplars), while zero-shot counters consider counting the most frequent class without annotations.

Few-shot counters have recently gained momentum with the emergence of a challenging dataset [24] and follow a

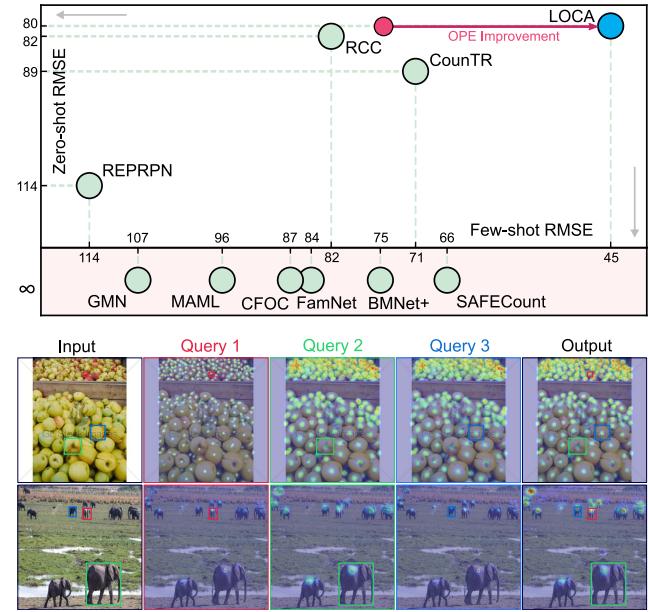


Figure 1. LOCA injects shape and appearance information into object queries to precisely count objects of various sizes in densely and sparsely populated scenarios. It also extends to a zero-shot scenario and achieves excellent localization and count errors across the entire low-shot spectrum.

common pipeline [18, 24, 13, 26, 31]. Image and exemplar features are extracted into object prototypes, which are matched to the image by correlation. Finally, the obtained intermediate image representation is regressed into a 2D object density map, whose values sum to the object count estimate. The methods primarily differ in the intermediate image representation construction method, which is based either on Siamese similarity [18, 24], cross-attention [16, 13] or feature and similarity fusion [26, 31]. While receiving much less attention, zero-shot counters follow a similar principle, but either identify possible exemplars by majority vote from region proposals [22] or implicitly by attention modules [11].

All few-shot counters construct object prototypes by pooling image features extracted from the exemplars into fixed-sized correlation filters. The prototypes thus fail to encode the object shape information (i.e., width, height

and aspect), resulting in a reduced accuracy of the density map. Recent works have shown that this information loss can be partially addressed by complex architectures for learning a nonlinear similarity function [26]. Nevertheless, we argue that a much simpler counting architecture can be used instead, by explicitly addressing the exemplar shape and by applying an appropriate object prototype adaptation method.

We propose a Low-shot Object Counting network with iterative prototype Adaptation (LOCA). Our main contribution is the new object prototype extraction module, which separately extracts the exemplar shape and appearance queries. The shape queries are gradually adapted into object prototypes by considering the exemplar appearance as well as the appearance of non-annotated objects, obtaining excellent localization properties and leading to highly accurate counts (Figure 1). To the best of our knowledge, LOCA is the first low-shot counting method that explicitly uses exemplars shape information for counting. In contrast to most works [26, 24, 30, 31], LOCA does not attempt to transfer exemplar appearance onto image features, but rather constructs strong prototypes that generalize across the image-level intra-class appearance.

LOCA outperforms all state-of-the-art (in many cases more complicated methods) on the recent FSC147 benchmark [24]. On the standard few-shot setup it achieves $\sim 30\%$ relative performance gains, on one-shot setup even outperforms methods specifically designed for this setup, achieves state-of-the-art on zero-shot counting. In addition, LOCA demonstrates excellent cross-dataset generalization on the car counting dataset CARPK [12].

2. Related work

Historically, object counting has been addressed by class-specific detectors for people [1, 33], cars [20, 12] and animals [2], but these methods do not cope well with extremely crowded scenes. In a jellyfish polyp counting scenario, [32] thus proposed to segment the image and interpret the segmentation as a collection of circular objects. Alternatively, [1, 6] framed counting as a regression of object density map, whose summation predicts the number of objects. A major drawback of these methods is that they require large annotated training datasets for each object class, which is often an unrealistic requirement.

In response, class-agnostic counters have been explored, that specialize to the object category at test-time using only a few user-provided object exemplars. An early representative [18] proposed a two-stream Generic Matching Network, that extracts the image and exemplar object features, concatenates them and regresses the representation into the final density map. CFOCNet [30] noted that a mere concatenation leads to unreliable localization and proposed a Siamese correlation network inspired by the tracking liter-

ature [3] to improve the localization and counts. Ranjan et al. [24] proposed a further improvement of correlation robustness by test-time Siamese backbone adaptation. Shi et al. [26] proposed an alternative approach for jointly learning the representation as well as a nonlinear similarity metric for improved localization and applied self-attention to reduce the within-class appearance variability in the test image. You et al. [31] combined the similarity map with the image features before applying location regression to improve count accuracy and proposed a learnable similarity metric to guide the fusion of exemplar and image features. Liu et al. [16] adopted a vision transformer [7] for image feature extraction and a convolutional encoder to extract the exemplars. Cross-attention is used to fuse image and exemplar features and a convolutional decoder regresses the density map. Recently, few-shot counting has been extended to few-shot detection [21] by adopting the transformer-based object detector [29] to predict also the object bounding box in addition to location.

While most works addressed situations with several (typically three) exemplars available, only few recent works considered reducing this number. Lin et al. [13] proposed a counting method that requires only a single exemplar. Their method is based on a transformer architecture and formulates correlation between image and exemplar features by several self- and cross-attention blocks. An extreme case of zero-shot counting [22, 11] has been explored as well. Ranjan and Hoai [22] proposed RepRPN-Counter, which combines a region proposal network [25] that also predicts a repetition score of each proposal. Proposals with the highest repetition scores are used as exemplars and sent through FamNet [24] to predict multiple density maps. On the other hand, Hobley and Prisacariu [11] developed a weakly-supervised method that implicitly identifies object category most likely to be counted and predicts a density map for that category. Vision transformer with a unsupervised training stage [16] has also shown success in zero-shot counting.

3. A low-shot prototype adaptation counter

Without loss of generality, we present our low-shot counting method LOCA in the context of few-shot counting. Given an input image $\mathbf{I} \in \mathbb{R}^{H_0 \times W_0 \times 3}$ and a set of n bounding boxes denoting a few selected objects, LOCA predicts a density map $\mathbf{R} \in \mathbb{R}^{H_0 \times W_0}$ whose values sum into the number of all objects of the selected class present in \mathbf{I} .

The LOCA architecture (Figure 2) follows four steps: (i) image feature extraction (encoder), (ii) object prototype extraction, (iii) prototype matching and (iv) density regression (decoder). The input image is resized to $H_{IN} \times W_{IN}$ pixels and encoded by a ResNet-50 [10] backbone. Multi-scale features are extracted from the second, third and fourth block, resized to a common size of $h \times w$ and reduced

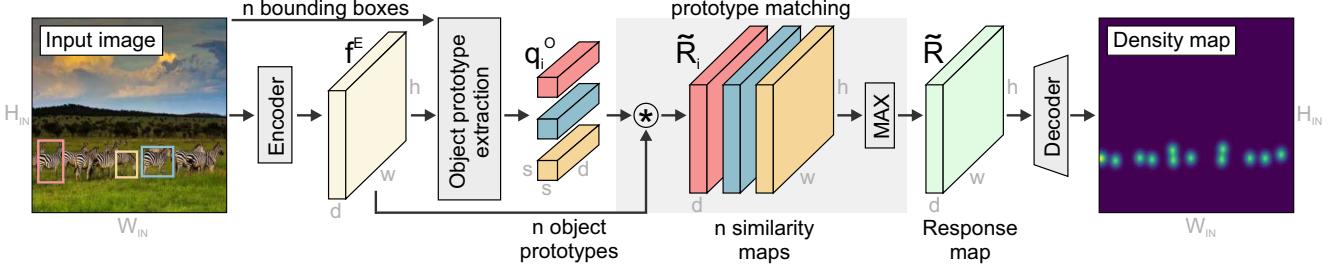


Figure 2. The LOCA architecture. Input image is encoded into features \mathbf{f}^E , which are depth-wise correlated (*) by n object queries predicted by the object prototype extraction module. The response map $\tilde{\mathbf{R}}$ is obtained by computing per-element maximum of n similarity maps $\tilde{\mathbf{R}}_i$ and then upsampled by decoder to the final density map.

by 1×1 convolutional layer into d channels. To further consolidate the encoded features and increase the similarity between same-category objects, a global (image-wide) self-attention block [28, 4] is applied, thus producing the encoded image features $\mathbf{f}^E \in \mathbb{R}^{h \times w \times d}$.

Next, n object prototypes $\{\mathbf{q}_i^O \in \mathbb{R}^{s \times s \times d}\}_{i=1:n}$ with spatial size $s \times s$, corresponding to the annotated bounding boxes are computed by the *object prototype extraction module*, which considers the annotated objects shape and appearance properties (detailed in Section 3.1). The image features \mathbf{f}^E are depth-wise correlated with the prototypes. Each prototype thus generates a multi-channel similarity tensor $\tilde{\mathbf{R}}_i$, i.e.,

$$\tilde{\mathbf{R}}_i = \mathbf{f}^E * \mathbf{q}_i^O, \quad (1)$$

where (*) is a depth-wise correlation. The individual n prototype similarity tensors are fused by a per-channel, per-pixel max operation, yielding a joint response tensor $\tilde{\mathbf{R}} \in \mathbb{R}^{h \times w \times d}$.

Finally, a regression head predicts the final 2D density map $\mathbf{R} \in \mathbb{R}^{H_{IN} \times W_{IN}}$. The regression head consists of three 3×3 convolutional layers with 128, 64 and 32 feature channels, each followed by a Leaky ReLU, a $2 \times$ bilinear upsampling layer, and a linear 1×1 convolution layer followed by a Leaky ReLU. The number of objects in the image N is estimated by summing the density map values, i.e., $N = \text{sum}(\mathbf{R})$.

3.1. Object prototype extraction module

The object prototype extraction module (OPE) (Figure 3) constructs n object prototypes $\{\mathbf{q}_i^O\}_{i=1:n}$, with $\mathbf{q}_i^O \in \mathbb{R}^{s \times s \times d}$, using the image feature map $\mathbf{f}^E \in \mathbb{R}^{h \times w \times d}$ and the set of n bounding boxes $\{b_i\}_{i=1:n}$. Ideally, the prototypes should generalize over the appearance of the selected object category in the image and retain good localization properties. Shape information is injected by initializing the prototypes with exemplar width and height features. The appearance of the remaining objects is then iteratively transferred into the final prototypes, with the exemplar appearance supervising the process. We details this process next.

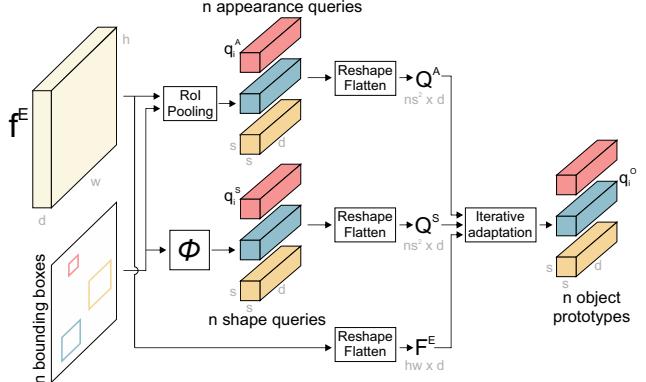


Figure 3. Object prototype extraction module (OPE). Shape and appearance queries are extracted separately and iteratively adapted considering the image-wide information into n object prototypes.

First, n appearance queries $\mathbf{q}_i^A \in \mathbb{R}^{s \times s \times d}$ are extracted from the annotated objects by ROI pooling [9] the image features \mathbf{f}^E from individual bounding boxes b_i into $s \times s$ tensors. The pooling operation makes the appearance queries shape-agnostic, since it maps features from different spatial shapes into rectangular queries of the same size. We introduce shape queries \mathbf{q}_i^S to recover the lost information as follows.

The shape query corresponding to the i -th bounding box is computed by a nonlinear mapping $\mathbb{R}^2 \rightarrow \mathbb{R}^{s \times s \times d}$ of its width and height $[b_i^w, b_i^h]$ into a high-dimensional tensor $\mathbf{q}_i^S = \phi([b_i^w, b_i^h])$. The mapping $\phi(\cdot)$ is implemented as a three-layer feed-forward network ($2 \rightarrow 64 \rightarrow d \rightarrow s^2 d$) with ReLU activations following each linear layer.

The shape and appearance queries are converted into object prototypes by an iterative adaptation module (Figure 4) using a recursive sequence of cross-attention blocks. Specifically, the shape queries \mathbf{q}_i^S are reshaped into a matrix $\mathbf{Q}^S \in \mathbb{R}^{ns^2 \times d}$ and in the same way the appearance queries \mathbf{q}_i^A and image features \mathbf{f}^E are reshaped into $\mathbf{Q}^A \in \mathbb{R}^{ns^2 \times d}$ and $\mathbf{F}^E \in \mathbb{R}^{hw \times d}$, respectively. The adaptation iteration

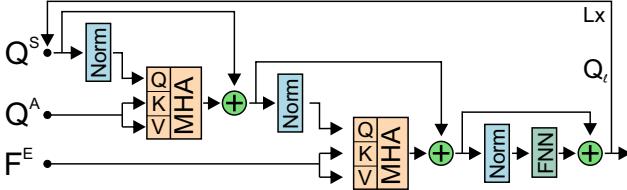


Figure 4. The iterative adaptation module applies attention to gradually generalize prototypes to the object instances indicated by few input exemplars.

then follows the sequence

$$\mathbf{Q}'_\ell = \text{MHA}(\text{LN}(\mathbf{Q}_{\ell-1}), \mathbf{Q}^A, \mathbf{Q}^A) + \mathbf{Q}_{\ell-1} \quad (2)$$

$$\mathbf{Q}''_\ell = \text{MHA}(\text{LN}(\mathbf{Q}'_\ell), \mathbf{F}^E, \mathbf{F}^E) + \mathbf{Q}'_\ell \quad (3)$$

$$\mathbf{Q}_\ell = \text{FFN}(\text{LN}(\mathbf{Q}''_\ell)) + \mathbf{Q}''_\ell, \quad (4)$$

where the inputs at $\ell = 0$ are initialized by the shape queries (i.e., $\mathbf{Q}_0 = \mathbf{Q}^S$), MHA is the standard multi-head attention [28], LN is layer normalization and FFN is a small feed-forward network. The process is performed for L iterations, i.e., $\ell \in \{1, \dots, L\}$. The output $\mathbf{Q}_L \in \mathbb{R}^{ns^2 \times d}$ is finally reshaped into a set of n object prototypes $\mathbf{q}_i^O \in \mathbb{R}^{s \times s \times d}$.

3.1.1 Adaptation for zero-shot setup

In the zero-shot scenario, the annotation-specific shape and appearance queries cannot be extracted due to absence of object annotations. Thus a minor modification of the OPE module is required to compute the object prototypes \mathbf{q}_i^O . In particular, the step (2) is skipped, and \mathbf{Q}'_ℓ is initialized by trainable objectness queries $\mathbf{q}_i^{S'} \in \mathbb{R}^{s \times s \times d}$. The iterative adaptation module computational sequence then becomes (3) and (4).

3.2. Training loss

LOCA is trained using the ℓ_2 loss between the predicted density map \mathbf{R} and the ground-truth map $\hat{\mathbf{G}}$ normalized by the number of objects,

$$\mathcal{L}_{OSE} = \frac{1}{M} \|\hat{\mathbf{G}} - \mathbf{R}\|_2^2, \quad (5)$$

where M is the number of objects in the mini-batch. The normalized loss emphasizes the errors in images with many objects, which usually contain the most challenging situations with high local object densities.

Auxiliary losses are added to better supervise the training of the iterative adaptation module (Figure 4). In particular, every intermediate output \mathbf{Q}_ℓ is reshaped into n queries $\{\mathbf{q}_i^\ell \in \mathbb{R}^{s \times s \times d}\}_{i=1:n}$ and applied to image features \mathbf{f}^E as in (1), generating an intermediate multi-channel response

tensor $\tilde{\mathbf{R}}_i^\ell$. This is followed by the max operation and regression head to obtain an intermediate density map \mathbf{R}^ℓ . The auxiliary loss is then computed as

$$\mathcal{L}_{AUX} = \frac{1}{M} \sum_{\ell=1}^{L-1} \|\hat{\mathbf{G}} - \mathbf{R}^\ell\|_2^2. \quad (6)$$

The final loss is thus $\mathcal{L} = \mathcal{L}_{OSE} + \lambda_{AUX} \mathcal{L}_{AUX}$, where λ_{AUX} is the auxiliary loss weight.

4. Experiments

4.1. Implementation details

Architecture details. LOCA resizes the input image to $H_{IN} = W_{IN} = 512$ pixels and applies the SwAV [5] pre-trained ResNet50 backbone with the features from the final three blocks upsampled to $h = w = 64$ pixels. This results in an activation map with 3584 channels, which is further projected into $d = 256$ channels by a 1×1 convolutional layer. The global self-attention block is a transformer encoder [4, 28] with 3 layers. MHA modules consist of 8 attention heads with the hidden dimension $d = 256$, while the FFN has the hidden dimension of 1024. Dropout [27] is applied after every MHA and FFN module with probability 0.1. The iterative adaption module contains $L = 3$ layers with the same MHA and FFN dimensions. The object prototype spatial size is $s \times s$ with $s = 3$, while the dropout is not used. The ground truth density maps are generated by placing unit densities on object locations and smoothing with the Gaussian kernel, whose size is determined for each image separately. In particular, the kernel size is determined as 1/8 of the average exemplar bounding box size.

Training details. Standard training image augmentation is applied, such as tiling, horizontal flipping and color jitter [16]. The backbone network parameters are frozen, while all other LOCA parameters are trained for 200 epochs using the AdamW [17] optimizer with the fixed learning rate 10^{-4} and weight decay 10^{-4} . The auxiliary loss weight in (6) is set to $\lambda_{AUX} = 0.3$ and gradient clipping with maximum norm of 0.1 is used. LOCA is trained on two Tesla V100 GPUs with batch size 8 (4 images per GPU) for approximately 10 hours.

4.2. Comparison with the state of the art

LOCA is evaluated on the recent few-shot counting dataset FSC147 [24]. The dataset contains 6135 images of 147 object categories split into training, validation and test sets consisting of 3659, 1286 and 1190 images, respectively. The sets of object categories present in each split are disjoint. Each image annotation consists of three bounding boxes of exemplar objects and point annotations for all objects of the same category as the exemplars.

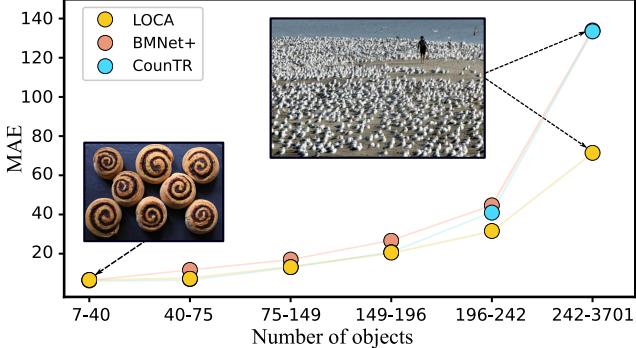


Figure 5. LOCA excels most in the highly challenging dense scenarios and outperforms state-of-the-art across the density levels.

In the few-shot counting scenario, we compare LOCA with GMN [18], MAML [8], FamNet [24] and the most recent state-of-the-art methods CFOCNet [30], BMNet+ [26], SAFECCount [31] and CounTR [16]. We follow the standard evaluation protocol [24, 26, 31] and compute Mean Absolute Error (MAE) and Root of Mean Squared Error (RMSE) given the predicted and ground truth object counts.

Results are summarized in Table 1. LOCA substantially outperforms all methods with a relative improvement of 22.0 %, 9.7 % in terms of MAE on validation and test sets, respectively, and 31.0 % and 33.4 % in terms of RMSE and sets a solid new state-of-the-art. Note that LOCA significantly outperforms even the most recent CounTR [16], which applies post-hoc error compensation routines (i.e., it estimates a correction factor for adjusting the estimated count).

Method	Validation set		Test set	
	MAE	RMSE	MAE	RMSE
GMN [19]	29.66	89.81	26.52	124.57
MAML [8]	25.54	79.44	24.90	112.68
FamNet [24]	23.75	69.07	22.08	99.54
CFOCNet [30]	21.19	61.41	22.10	112.71
BMNet+ [26]	15.74	58.53	14.62	91.83
SAFECCount [31]	15.28 ^③	47.20 ^②	14.32 ^③	85.54 ^②
CounTR [16]	13.13 ^②	49.83 ^③	11.95 ^②	91.23 ^③
LOCA (ours)	10.24 ^①	32.56 ^①	10.79 ^①	56.97 ^①

Table 1. Evaluation on a few-shot counting scenario.

For further insights we inspect the count errors with respect to the number of objects in the image (Figure 5). LOCA outperforms the state-of-the-art across the different object numbers and most significantly outperforms the state-of-the-art on images with very high object counts. These typically contain extremely high object densities, presenting substantial challenge to all previous methods. But LOCA copes very well even with these cases, reducing the count errors by nearly 50% compared to state-of-the-art.

4.2.1 Evaluation on one-shot counting

We inspect performance under minimal user supervision with a single annotation – a one-shot scenario. LOCA is compared with LaoNet [13], which is designed specifically for one-shot scenarios, as well as with the recent methods GMN [18], CFOCNet [30], FamNet [24], BMNet+ [26] and CounTR [16] which were specialized for the one-shot setting. The results are shown in Table 2. LOCA outperforms the current state-of-the-art with a relative improvement of 13.6 % MAE on the validation set, and 23.5% and 16.3% RMSE on validation and test set, respectively. This empirically confirms that LOCA generalizes well also to the minimal supervision counting case.

Method	Validation set		Test set	
	MAE	RMSE	MAE	RMSE
GMN [19]	29.66	89.81	26.52	124.57
CFOCNet [30]	27.82	71.99	28.60	123.96
FamNet [24]	26.55	77.01	26.76	110.95
BMNet+ [26]	17.89	61.12	16.89	96.65 ^③
LaoNet [13]	17.11 ^③	56.81 ^③	15.78 ^③	97.15
CounTR [16]	13.15 ^②	49.72 ^②	12.06 ^①	90.01 ^②
LOCA (ours)	11.36 ^①	38.04 ^①	12.53 ^②	75.32 ^①

Table 2. Evaluation on a one-shot counting scenario.

4.2.2 Evaluation on zero-shot counting

As noted in Section 3.1.1, LOCA can be easily applied to the unsupervised counting scenario with no user annotations, i.e., the zero-shot setup. We thus compare LOCA with zero-shot CounTR [16] and state-of-the-art methods RepRPN-C [23] and RCC [11] which are specialized for zero-shot counting. The results in Table 3 show that LOCA achieves relative improvements of 6.5 %, and 0.5 % in terms of RMSE on validation and test sets, respectively, compared to the state-of-the-art, and outperforms all zero-shot specialized architectures. This confirms that the proposed OPE module successfully adapts the trainable objectness queries into strong object prototypes capable of accurate count estimation even in the extreme case without manually annotated exemplars.

Method	Validation set		Test set	
	MAE	RMSE	MAE	RMSE
RepRPN-C [23]	29.24	98.11	26.66	129.11
RCC [11]	17.49 ^③	58.81 ^②	17.12 ^③	104.53 ^②
CounTR [16]	17.40 ^①	70.33 ^③	14.12 ^①	108.01 ^③
LOCA (ours)	17.43 ^②	54.96 ^①	16.22 ^②	103.96 ^①

Table 3. Evaluation on a zero-shot counting scenario.

4.2.3 Qualitative few-shot counting results

Figure 6 visualizes the predicted object density maps from LOCA and BMNet+ [26]. Note that LOCA produces density maps with high fidelity object localization. The prototypes generated by the OPE module discriminate well between the objects and the background (columns 1–3). In the second column, LOCA generates clear density peaks on the object centers. In columns 4, 5, 6 and 7 we see that LOCA outperforms BMNet+ on small objects, with objects localized far better in the density map. This is likely due to explicitly accounting for object shape and size by shape-specific objectness queries, in contrast to other methods that consider only scale-agnostic object appearance extraction. The shape-specific information enables LOCA to more robustly address the object size variability within the image. In column 8, BMNet+ misses several larger apples while LOCA accurately localizes apples of all sizes. Similarly, BMNet+ underestimates the density of larger marbles in column 9, while LOCA produces a much more crisp and accurate density map. Columns 10, 11 and 12 show examples with few objects. LOCA also performs well in such scenarios.

4.3. Comparison with object detectors

In limited cases where large training sets are available, objects can be counted using pretrained object detectors. It is thus instructive to evaluate the general few-shot object counters in these specialized cases in comparison with the classical detectors. The FSC147 dataset [24] in fact provides image subsets Val-COCO and Test-COCO containing only categories for which abundant annotated training images are available in COCO [15].

This allows comparing counting capabilities of LOCA with those of classical detectors FasterRCNN [25], MaskRCNN [9], and RetinaNet [14] as well as the recent few-shot counting state-of-the-art FamNet [24], BMNet+ [26] and CounTR [16]. The results are reported in Table 4. LOCA achieves state-of-the-art performance, most significantly on Val-COCO with a relative 31% MAE and 36% RMSE improvement over the best method.

Method	Val-COCO		Test-COCO	
	MAE	RMSE	MAE	RMSE
Faster-RCNN [25]	52.79	172.46	36.20	79.59
RetinaNet [14]	63.57	174.36	52.67	85.86
Mask-RCNN [9]	52.51	172.21	35.56	80.00
Famnet [24]	39.82	108.13	22.76	45.92
BMNet+ [26]	26.55③	93.63③	12.38③	24.76①
CounTR [16]	24.66②	83.84②	10.89②	31.11②
LOCA (ours)	16.86①	53.22①	10.73①	31.31③

Table 4. Evaluation on the COCO object-detection counting dataset.

4.4. Cross-dataset generalization

We evaluate the cross-dataset generalization capabilities of LOCA using the established evaluation protocol from [24]. In that protocol, a method is trained on the FSC147 dataset [24] and evaluated on the CARPK dataset [12], which is a car-counting dataset containing aerial images of parking lots, which are considerably different from the FSC147 images. To ensure there is no object class overlap between the training and test dataset, the car images are omitted from the FSC-147 training set. For counting purposes, twelve exemplars are sampled from the training CARPK images and used in all test CARPK images.

The results are reported in Table 5. LOCA achieves better cross-dataset generalization with a relative 4.5 % MAE and 9.2 % RMSE improvement compared to the most recently published state-of-the-art method BMNet+, thus setting a new dataset generalization state-of-the-art among the few-shot counting methods.

Method	MAE	RMSE
FamNet [24]	28.84③	44.47③
BMNet+ [26]	10.44②	13.77②
LOCA (ours)	9.97①	12.51①

Table 5. Cross-dataset generalization experiment on CARPK [12].

4.5. Ablation study

We finally analyze the architectural design choices and examine the influence of the object-normalized loss and the auxiliary losses. The experiments are performed on the FSC147 dataset in the few-shot setting. We report the performance by averaging a certain measure on validation and test sets.

Architecture design. Table 6 reports the performance of re-trained LOCA variants with individual computational blocks removed. To evaluate the importance of global attention in the image features encoder block, we removed this block ($LOCA_{no_att}$) and observe a 12% MAE performance drop. This indicates the importance of image feature consolidation by attention, which likely brings objects of the same category closer at feature level. The impact of the OPE module is evaluated by removing it and extracting the object prototypes directly from the encoder image features by pooling the features of the exemplar regions ($LOCA_{no_ope}$). This results in significant performance drop in order of 34% MAE. Removing both, global attention and OPE ($LOCA_{no_att_ope}$) leads to further performance drops of $\sim 39\%$ MAE compared to the original LOCA.

Next, we explored the importance of the exemplar shape information in addition to their appearance. A variant

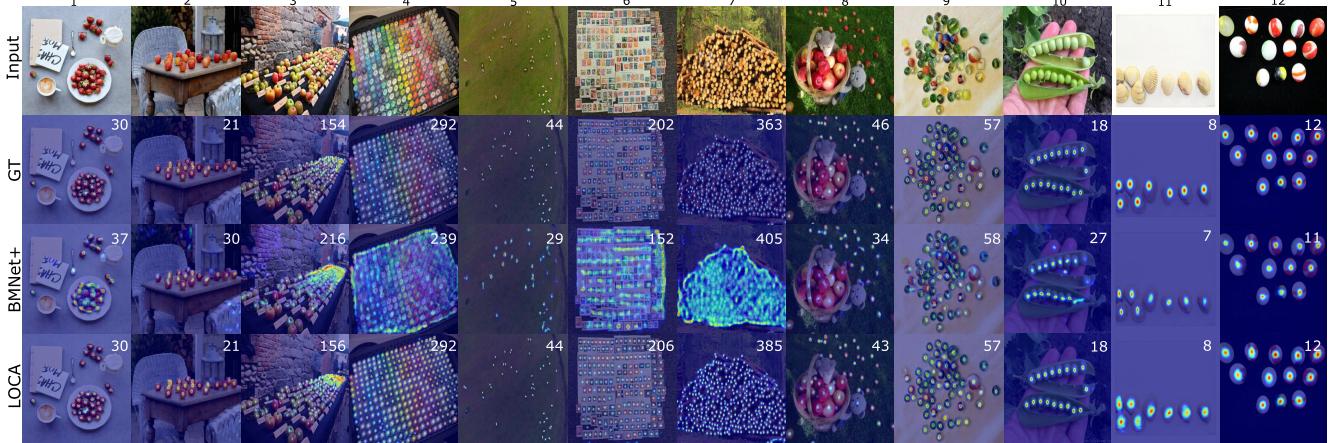


Figure 6. Qualitative results on the FSC147 dataset. Compared to related works, LOCA better discriminates between objects and background, predicts density maps with clear object locations and works well on smaller objects, while better capturing the intra-class variability and within-image size variability.

$LOCA_{no_shape}$ was constructed, which ignores the shape queries \mathbf{q}_i^S by omitting the first attention block in the OPE module (Figure 3) and replacing Q^S with Q^A in the second attention block. We observe a 25% MAE reduction compared to original LOCA. This confirms the importance of accounting for the shape information in addition to the appearance in OPE.

Importance of the mapping function that transforms the exemplar width and height into shape-specific objectness queries (Section 3.1) is analyzed in the following. Instead of predicting the shape queries from exemplars, we replace them with trainable queries ($LOCA_{pre_shape}$). Results show a significant drop in performance compared to LOCA with a 27% change in MAE, indicating that useful shape-specific objectness information is indeed extracted from the exemplars size parameters and that it significantly contributes to object localization and accurate counts.

We also analyzed the role of the first cross-attention in the first OPE iteration (Equation 2) by replacing it with a simple summation: $\mathbf{Q}'_1 = \mathbf{Q}^A + \mathbf{Q}^S$. This results in a 5% increase of MAE and a 22% increase of RMSE, which indicates that the first MHA in OPE should not be considered as a simple matching operation, but rather as a modulation of the prototype construction process by the exemplar shape information. This information is unique for every exemplar, thus optimally adjusting the resulting prototype to localize the objects of interest.

Finally, we analyzed the impact of the number of adaptation iterations L in the iterative adaptation module (Section 3.1) on the joined FSC-147 evaluation sets. Results are shown in Table 7. The choice of $L = 3$ provides the best performance while maintaining a low model complexity.

Complexity. As shown in Table 8, the proposed architecture has almost $3\times$ less parameters and almost $10\times$ less trainable parameters than CounTR while being compara-

ble to other state-of-the-art methods in both the number of parameters and computational complexity. These results demonstrate that excellent low-shot object counting performance of LOCA comes from the methodological improvements instead of increased complexity.

Method	Validation set		Test set	
	MAE	RMSE	MAE	RMSE
$LOCA_{no_att_ope}$	17.62	55.78	16.92	101.96
$LOCA_{no_ope}$	16.24	57.41	15.53	96.23
$LOCA_{no_shape}$	13.77	49.60	14.29	112.48
$LOCA_{pre_shape}$	13.00	44.61	15.80	122.87
$LOCA_{no_att}$	11.99	36.67	11.96	78.72
LOCA	10.24	32.56	10.79	56.97

Table 6. Ablation study of individual architectural components. $LOCA_{no_att_ope}$ removes the entire global self-attention block from the encoder and the OPE module, $LOCA_{no_ope}$ removes the OPE module, $LOCA_{no_shape}$ removes the use of the shape queries and $LOCA_{no_att}$ removes the global self-attention block from the encoder.

L	1	2	3	4	5	6
MAE	11.07	10.81	10.50	10.89	11.60	11.04

Table 7. Ablation of the number of iterations L in the iterative adaptation module.

Backbone and resolution. Importance of the backbone pre-training regime, input image resolution and the prototype spatial size are presented in Table 9. Replacing the SwAV-pretrained backbone with the ImageNet-pretrained one ($LOCA_{ImNet}$) results in only a slight performance drop (8% MAE and 4% RMSE). Reducing the input image resolution from 512×512 to 384×384 pixels ($LOCA_{384}$)

Method	GFLOPS	Number of parameters	
		Total	Trainable
FamNet	55	26M	760k
BMNet+	27	13M	12M
SafeCount	366	32M	20M
CounTR	91	100M	99M
LOCA (ours)	80	37M	11M

Table 8. Computational complexity and the number of parameters.

leads to the 9% performance drop in both MAE and RMSE. Without any hyperparameter modifications, $LOCA_{384}$ remains the top-performing method in three out of four metrics. Changing the prototype spatial size s from 3 to 1 ($LOCA_{s=1}$) or 5 ($LOCA_{s=5}$) does not lead to significant performance drops. MAE is reduced by 3% and 10% while RMSE is reduced by 6% and 11% for $s = 1$ and $s = 5$, respectively, which confirms that LOCA is not sensitive to the prototype spatial size. All these results further verify that the design of the OPE module is the main driver of LOCA’s superior performance.

Method	Validation set		Test set	
	MAE	RMSE	MAE	RMSE
$LOCA_{ImNet}$	11.40	37.10	11.56	55.89
$LOCA_{384}$	10.26	32.62	12.75	65.34
$LOCA_{s=1}$	10.90	38.66	10.79	56.97
$LOCA_{s=5}$	11.11	35.47	12.27	65.08
LOCA	10.24	32.56	10.79	56.97

Table 9. Impact of the backbone pre-training regime, input image resolution and the prototype spatial size on LOCA’s performance.

Model supervision. We explored the impact of object count normalization in \mathcal{L}_{OSE} (Equation 5) and the importance of using the auxiliary losses on OPE blocks (Section 3.2). Results are shown in Table 10. Avoiding the object count normalization leads to a 11% performance drop in terms of MAE. This shows the benefits of the object count normalization which places a larger penalty on images with larger object counts providing an emphasis on difficult cases with high local object densities. Additionally removing the auxiliary losses leads to a 17% performance drop in terms of RMSE. This drop in performance indicates that supervision on individual iterations in OPE is beneficial as it encourages the OPE module to provide informative features throughout the iterative process.

4.6. Qualitative analysis

Figure 7 qualitatively compares LOCA with the recent state-of-the-art method CounTR [16]. LOCA demonstrates superior performance in counting small objects (first and second row), large objects (third row) and objects of mixed sizes (fourth and fifth row), which supports the proposed design. Figure 8 qualitatively compares LOCA with a version that does not use shape information and a version without the OPE module. The shape information injection and the adaptation in OPE module both contribute to accurate localization and counts.

\mathcal{L}_{OSE}	Auxiliary loss	Validation set		Test set	
		MAE	RMSE	MAE	RMSE
✗	✗	10.87	35.68	11.93	72.83
✗	✓	10.86	31.89	12.83	62.73
✓	✓	10.24	32.56	10.79	56.97

Table 10. Ablation study on the object-normalized ℓ_2 loss (\mathcal{L}_{OSE}) and the auxiliary losses after every block in OPE. The mark \times with \mathcal{L}_{OSE} indicates that the standard ℓ_2 [24] loss is used.

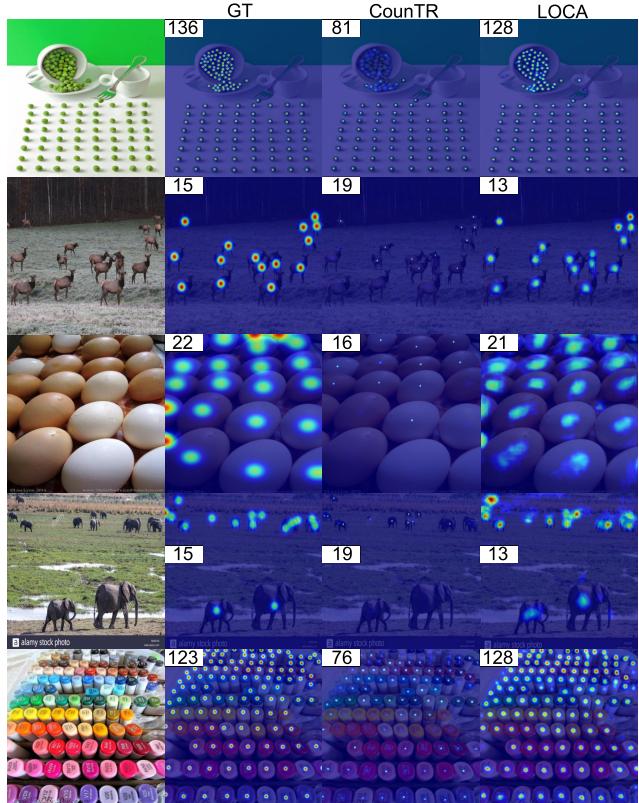


Figure 7. LOCA demonstrates superior performance on images with only small objects (first and second row), images with only large objects (third row), as well as images with objects of mixed sizes (fourth and fifth row).

sizes (fourth and fifth row), which supports the proposed design. Figure 8 qualitatively compares LOCA with a version that does not use shape information and a version without the OPE module. The shape information injection and the adaptation in OPE module both contribute to accurate localization and counts.

5. Conclusion

We presented a new low-shot counting method LOCA, that addresses the limitations of the current state-of-the-art methods. LOCA considers the exemplar shape and appearance properties separately and iteratively adapts these into object prototypes by a new object prototype extrac-

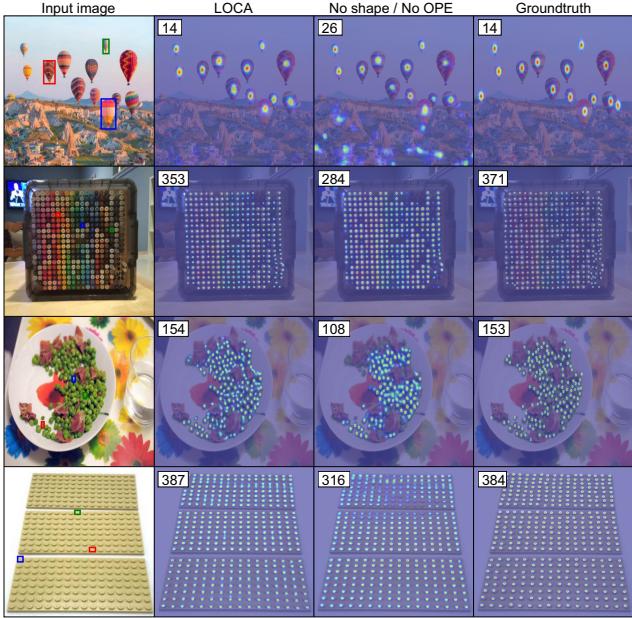


Figure 8. LOCA compared to a variant that does not use shape information (first two rows) and a variant without the OPE module (last two rows). Objects across scales are most accurately localized and counted when using the shape information and OPE.

tion (OPE) module considering the image-wide features. The prototypes thus generalize to the non-annotated objects in the image, leading to better localization properties and count estimates.

Experiments show that LOCA outperforms state-of-the-art on the FSC147 public benchmark in few-shot, one-shot and zero-shot settings. We observed a relative RMSE improvement of 33.4% in few-shot and 16.3% in zero-shot scenarios. On the COCO subsets of FSC147, LOCA outperforms recent state-of-the-art counting methods, as well as object detection methods, achieving a 36% RMSE improvement. On the CARPK dataset, LOCA achieves a relative improvement of 9.2% RMSE, which demonstrates excellent cross-dataset generalization. The quantitative results convincingly support the benefits of the new OPE module, which is our main contribution.

We envision several possible future research directions. Additional supervision levels such as introducing negative exemplar annotations could be introduced in LOCA for better specification of the selected object class. This could lead to interactive tools for accurate object counting. Furthermore, a gap between low-shot counters and object detectors could be further narrowed by enabling bounding box or segmentation mask prediction in LOCA to output additional statistics about the counted objects such as average size, etc., which is useful for many practical applications such as biomedical analysis.

Acknowledgements: This work was supported by Slovenian research agency program P2-0214 and projects J2-2506, Z2-4459, 23-20MR.R588 and J2-3169.

References

- [1] Shahira Abousamra, Minh Hoai, Dimitris Samaras, and Chao Chen. Localization in the crowd with topological constraints. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 872–881, 2021. [1](#), [2](#)
- [2] Carlos Arteta, Victor S. Lempitsky, and Andrew Zisserman. Counting in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 483–498, 2016. [1](#), [2](#)
- [3] Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-convolutional siamese networks for object tracking. In *ECCV Workshops*, 2016. [2](#)
- [4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. [3](#), [4](#)
- [5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9912–9924. Curran Associates, Inc., 2020. [4](#)
- [6] Antoni B. Chan and Nuno Vasconcelos. Bayesian poisson regression for crowd counting. *2009 IEEE 12th International Conference on Computer Vision*, pages 545–551, 2009. [2](#)
- [7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. [2](#)
- [8] 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. [5](#)
- [9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:386–397, 2020. [3](#), [6](#)
- [10] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. [2](#)
- [11] Michael A. Hobley and Victor Adrian Prisacariu. Learning to count anything: Reference-less class-agnostic counting with weak supervision. *ArXiv*, abs/2205.10203, 2022. [1](#), [2](#), [5](#)
- [12] Meng-Ru Hsieh, Yen-Liang Lin, and Winston H. Hsu. Drone-based object counting by spatially regularized regional proposal network. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4165–4173, 2017. [1](#), [2](#), [6](#)

- [13] Hui Lin, Xiaopeng Hong, and Yabin Wang. Object counting: You only need to look at one. *ArXiv*, abs/2112.05993, 2021. [1](#), [2](#), [5](#)
- [14] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. [6](#)
- [15] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. [6](#)
- [16] Chang Liu, Yujie Zhong, Andrew Zisserman, and Weidi Xie. Countr: Transformer-based generalised visual counting. In *BMVC*, 2022. [1](#), [2](#), [4](#), [5](#), [6](#), [8](#)
- [17] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. [4](#)
- [18] E. Lu, W. Xie, and A. Zisserman. Class-agnostic counting. In *Asian Conference on Computer Vision*, 2018. [1](#), [2](#), [5](#)
- [19] Erika Lu, Weidi Xie, and Andrew Zisserman. Class-agnostic counting. In *Asian conference on computer vision*, pages 669–684. Springer, 2018. [5](#)
- [20] Terrell N. Mundhenk, Goran Konjevod, Wesam A. Sakla, and Kofi Boakye. A large contextual dataset for classification, detection and counting of cars with deep learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 785–800, 2016. [1](#), [2](#)
- [21] Thanh Thoi Nguyen, Chau Khoa Pham, Khoi Duc Minh Nguyen, and Minh Hoai. Few-shot object counting and detection. In *ECCV*, page 348–365, 2022. [2](#)
- [22] Viresh Ranjan and Minh Hoai. Exemplar free class agnostic counting. *ArXiv*, abs/2205.14212, 2022. [1](#), [2](#)
- [23] Viresh Ranjan and Minh Hoai. Exemplar free class agnostic counting. *arXiv preprint arXiv:2205.14212*, 2022. [5](#)
- [24] Viresh Ranjan, Udbhav Sharma, Thua Nguyen, and Minh Hoai. Learning to count everything. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3393–3402, 2021. [1](#), [2](#), [4](#), [5](#), [6](#), [8](#)
- [25] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. [2](#), [6](#)
- [26] Minghan Shi, Hao Lu, Chen Feng, Chengxin Liu, and ZHIGUO CAO. Represent, compare, and learn: A similarity-aware framework for class-agnostic counting. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9529–9538, 2022. [1](#), [2](#), [5](#), [6](#)
- [27] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15:1929–1958, 2014. [4](#)
- [28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc., 2017. [3](#), [4](#)
- [29] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based detector. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2567–2575, 2022. [2](#)
- [30] Shuo Yang, Hung-Ting Su, Winston H. Hsu, and Wen-Chin Chen. Class-agnostic few-shot object counting. *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 869–877, 2021. [2](#), [5](#)
- [31] Zhiyuan You, Yujun Shen, Kai Yang, Wenhan Luo, Xin Lu, Lei Cui, and Xinyi Le. Few-shot object counting with similarity-aware feature enhancement. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023. [1](#), [2](#), [5](#)
- [32] Vitjan Zavrtanik, Martin Vodopivec, and Matej Kristan. A segmentation-based approach for polyp counting in the wild. *Eng. Appl. Artif. Intell.*, 88, 2020. [1](#), [2](#)
- [33] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. Cross-scene crowd counting via deep convolutional neural networks. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 833–841, 2015. [1](#), [2](#)