SILCO: Show a Few Images, Localize the Common Object–supplementary file

Tao HU, Pascal Mettes, Jia-Hong Huang and Cees G. M. Snoek
University of Amsterdam

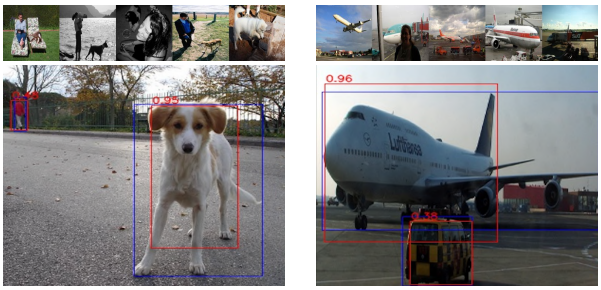

Figure 1. Localization from support images with multiple objects . Blue denotes ground truth, red denotes prediction. The two objects common in all support images are localized correctly in the query images.

Table 1. The #image of CL-VOC 07, CL-VOC 12, CL-COCO Few-Shot Common Localization Dataset.

dataset	train	val	test
CL-VOC07	2501	1010	1500
CL-VOC12	5717	2623	3200
CL-COCO	62783	20000	40504

Multiple common object. We also assess the ability to localize more than one common object. We have re-trained SILCO on images that also include multiple objects. Figure 1 shows two examples. In both cases, we detect the two objects that appear in all five support images. This further indicates the practical use of SILCO.

Dataset Details. we mainly elaborate the image numbers of CL-VOC07, CL-VOC12, CL-COCO in Table 1, the detail classes of all datasets is shown in Table 3, Table 4.

For experiment in VID, we only keeps the classes: bear, elephant, fox, giant panda, hamster, lion, lizard, monkey, rabbit, red panda, snake, squirrel, tiger, turtle, whale.

The data splitting principle of different subset is shown in Figure 2. For train,validation and test, the images are isolated with each other. When training, we random sample tuples(1+N for N-shot common-localization) from the pool.The detail algorithm is shown in Algorithm 1.

Another perspective to understand our method. From the aspect of space, our spatial similarity can be viewed as a kind of ***spatial*** attention mechanism between query feature and support feature. Furthermore, from the perspective of image, feature reweighting tries to rescale and balance the influence of different support ***images***. A natural question is whether we can introduce ***channel***-level attention into our framework to reach a ***spatial, channel, image***-level attention. Therefore, we try to adopt SENet[1] upon our framework and obtain the result in Table 2. It indicates that the performance can be further improved by channel attention. The result demonstrates that ***spatial, channel, image***-level attention mechanism can be combined in a mutually beneficial way.

Table 2. The comparison between "before channel attention" and "after channel attention". A typical channel attention method SENet[1] is deployed in our experiment.

dataset	method	Group 1	Group 2	mAP(%)
CL-VOC07	before	57.17	56.45	56.82
	after	**57.50**	**56.96**	**57.23**
CL-VOC12	before	55.11	58.62	56.86
	after	**56.55**	**59.22**	**57.89**

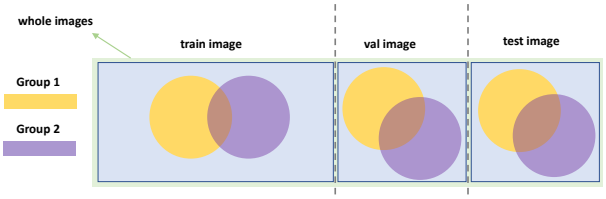

Figure 2. The dataset splitting demonstration. The total data are composed of train image, validation image, test image. Our labels are separate between different images.

Table 3. The class grouping for CL-VOC. One group is used to initialize the model, the other group is used to perform the few-shot common-localization.

Group	Semantic Classes
1	aeroplane, bicycle, bird, boat bottle, bus, car, cat, chair, cow
2	dining table, dog, horse, motorbike person, potted plant, sheep, sofa train, tv/monitor

Table 4. The class grouping for CL-COCO. One group is used to initialize the model, the other group is used to perform the few-shot common-localization.

Group	Semantic Classes
1	person, bicycle, car, motorcycle, airplane,bus, train, truck, boat, traffic light, fire hydrant, stop sign , parking meter, bench, bird, cat, dog, horse, sheep, cow , elephant, bear, zebra, giraffe, backpack, umbrella, handbag, tie , suitcase, frisbee, skis, snowboard, sports ball, kite, baseballbat, baseball glove, skateboard, surfboard, tennis racket bottle
2	wine glass, cup, fork, knife, spoon, bowl, banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake, chair, couch, potted plant, bed, dining table, toilet, tv, laptop, mouse, remote, keyboard, cell phone, microwave, oven, toaster, sink, refrigerator, book, clock, vase, scissors, teddy bear, hair drier, toothbrush

Algorithm 1 Dataset Preparation Algorithm

Notations:
$\mathcal{C}$: target class set
$\mathcal{D}_i$: the i-th image in the dataset $\mathcal{D}$
d: dictionary for (label, image_list) pairs

for $\mathcal{D}_i$ in $\mathcal{D}$ **do**
  for $\mathcal{C}_j$ in $\mathcal{C}$ **do**
    if $\mathcal{D}_i$ has the class of $\mathcal{C}_j$ and d has key $\mathcal{C}_j$ **then**
      d [$\mathcal{C}_j$] .append($\mathcal{D}_j$)
    else
      d [$\mathcal{C}_j$] = [$\mathcal{D}_j$]
    end if
  end for
end for
while Not Stop **do**
  random choose 1 item $\tilde{C}$ from $\mathcal{C}$
  random choose k+1 items $\mathcal{R}$ from $d[\tilde{C}]$
  yield $\mathcal{R}$
end while

References

[1] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 1