Supplementary Materials for Query Adaptive Few-Shot Object Detection with Heterogeneous Graph Convolutional Networks

Guangxing Han, Yicheng He, Shiyuan Huang, Jiawei Ma, Shih-Fu Chang
Columbia University
{gh2561,yh3330,sh3813,jiawei.m,sc250}@columbia.edu

The supplementary materials are organized as follows. First, we describe the implementation details of our model architecture in Section 1. Then, we describe the implementation details of our training framework in Section 2. We show the visualization of the category-specific proposals in Section 3, and the visualization of the pairwise cosine similarity of all classes (80) in MSCOCO in Section 4.

1. Our Model Architecture with Implementation Details

We illustrate our model architecture in Fig. 1. Our model is built upon the baseline FSOD model in [1]. Next we describe each component in our model in details, including the backbone feature extractor, the Attention-RPN [1], the RoI feature extractor, the heterogeneous GCNs and the pairwise classifier [1].

1.1. The backbone feature extractor

We use ResNet-50/101 as our backbone feature extractor by default for fair comparison with other STOAs. Specifically, we use the output of the $res4$ block as the input image features of the detection head. The backbone feature extractor is shared for both query image I_q and support image I_s to extract the features $r(I_q)$ and $r(I_s)$ respectively.

1.2. The Attention-RPN

Following [1], we use the Attention-RPN module to generate category-specific proposals for each novel class. Formally, for each novel class $c \in \mathcal{C}_{novel}$, we first take the average feature of all support images belonging to that novel class as $r(c)$, and then conduct spatial average pooling to get the spatial-averaged feature $r(c)_{pool} = \frac{1}{H*W}\sum_{h,w} r(c)$, such that $r(c)_{pool}$ captures global representation of class c. Then we modulate the query image features using $r(c)_{pool}$, which can highlight important and relevant features in the query image for class c,

$$r(I_q)_c = r(I_q) \odot r(c)_{pool}, c \in \mathcal{C}_{novel} \quad (1)$$

where $\odot$ represents channel-wise Hadamard product.

After that, we use the original RPN to generate proposals using the modulated query image features, such that the proposals are category-specific. Typically we generate 100 proposals for each novel class by default.

1.3. The RoI feature extractor

After generating the proposals in the query image, we use RoI Align and the $res5$ block in ResNet to extract the feature $f(p_i^c)$ for proposal p_i^c from the query image feature $r(I_q)$. The same layers are applied to $r(I_s)$ for each support image, and we take the average feature of all support images belonging to the novel class c as the class prototype $f(c)$.

1.4. Our Heterogeneous GCNs Module

We propose the heterogeneous GCNs in this paper to enable efficient message passing among the proposal and class nodes, such that we could learn context-aware proposal features and query-adaptive, multiclass-enhanced prototype representations for each class.

For the Inter-Class Subgraph, we build a fully-connected graph (80 nodes) with all base classes (60 nodes) and novel classes (20 nodes). For each base class, we use 30 support images to extract the prototype representation. We emphasize that **proposals and prototypes should go through the same number of learnable transformation layers** before the final pairwise classification, to make sure the two features are in the same feature space. This is shown in Table 4 of the main paper that using GCN layers w/o W is better than that w/ W. We adhere to this principle throughout the model design, and do not use the learnable parameter W in our Inter-Class Subgraph. Meanwhile, the Inter-Class Subgraph is query-agnostic and we perform message passing on it before processing any query image.

The Intra-Class Subgraphs are built upon the query image. For each query image, we build an Intra-Class Subgraph for each novel class. For each Intra-Class Subgraph, we have in total 102 nodes, including 1 class node c, 1 global image node g and 100 category-specific proposal nodes P_c of the class c. We establish proposal-proposal edges between proposals if the IoU is more than the threshold $\theta = 0.7$.

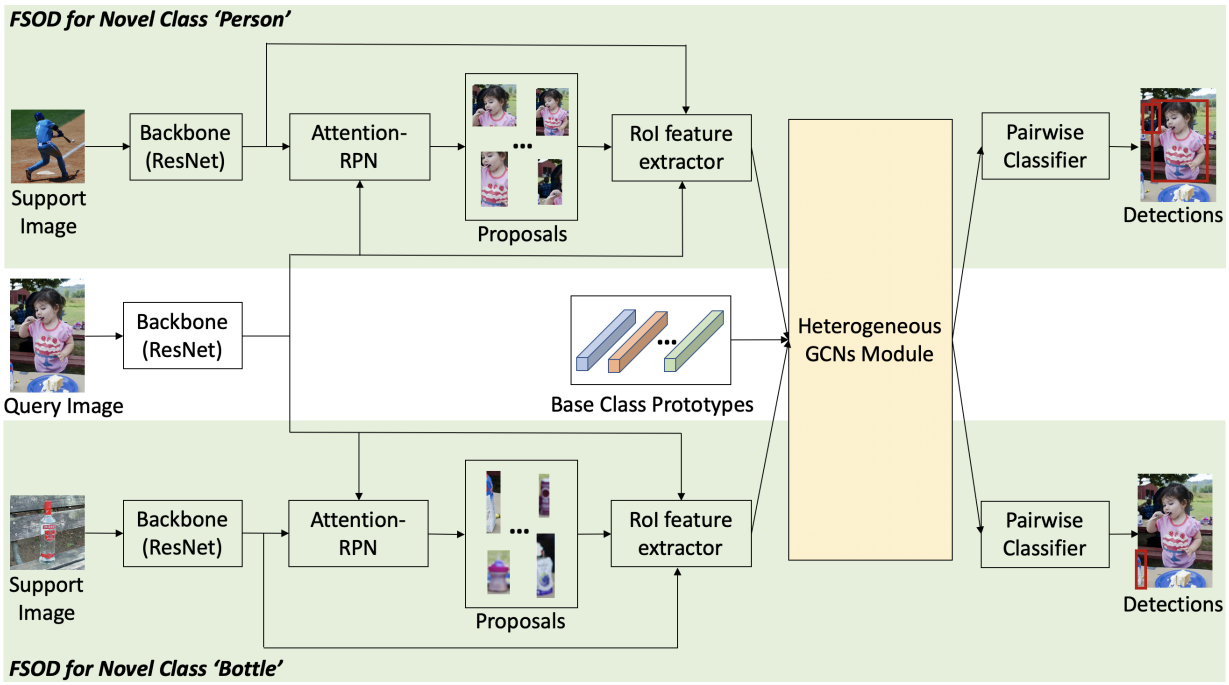

Figure 1. Our model architecture. We use two novel classes as an example. The backbone feature extractor, the Attention-RPN [1], the RoI feature extractor the and pairwise classifier [1] are shared for all the branches.

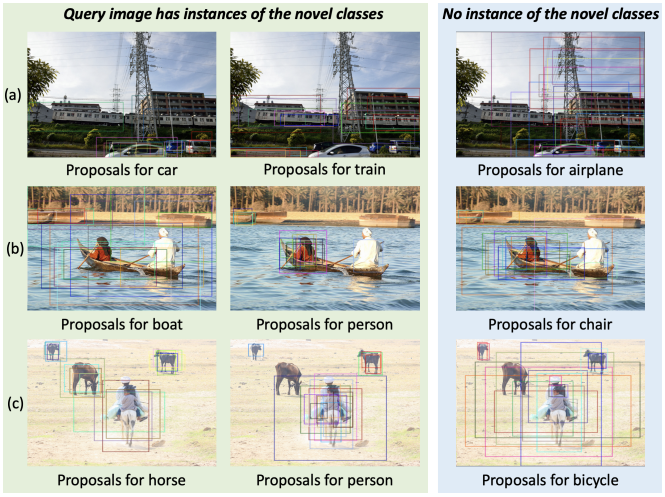

Figure 2. Visualization of category-specific proposals generated using Attention-RPN in [1]. We show both cases of whether the query images have instances of the target class or not.

The far-away proposals could hardly provide meaningful information. We enrich the proposals with global image features by adding edges to the global image node g. For the class-proposal edge, we connect bidirectional edges between the class node and all the 100 proposal nodes.

Note that we only use one GCN layer for both the Inter-Class Subgraph and Intra-Class Subgraphs as shown in Table 4 and 5 of the main paper because we already connect edges to all neighbors that a node needs in our model. Using more GCN layers are not helpful due to the over-smoothing problem [2] in GCNs.

1.5. The pairwise classifier

After getting the enhanced features for proposals and classes, we use the multi-relation network in [1] for pairwise classification. Specifically, for each novel class, we calculate the similarity score between the class prototype and category-specific proposal features, and produce the final detection results of this class using post-processing steps in [3].

2. Implementation Details of Model Training

Meta-learning with Base Classes. We train our FSOD model on base classes using episodic training.

For model training on the MSCOCO dataset, we use the SGD optimizer with an initial learning rate of 0.004, momentum of 0.9, weight decay of 0.0001, and a batch size of 8. The learning rate is divided by 10 after 60000 iterations. The total number of training iterations is 80000.

Similarly, we use a smaller number of training iterations for meta-training on the PASCAL VOC dataset. The initial learning rate is 0.004, divided by 10 after 10000 iterations. The total number of training iterations is 15000.

Fine-tuning with Novel Classes. We fine-tune the few-shot detection model on novel classes in this step. We use the

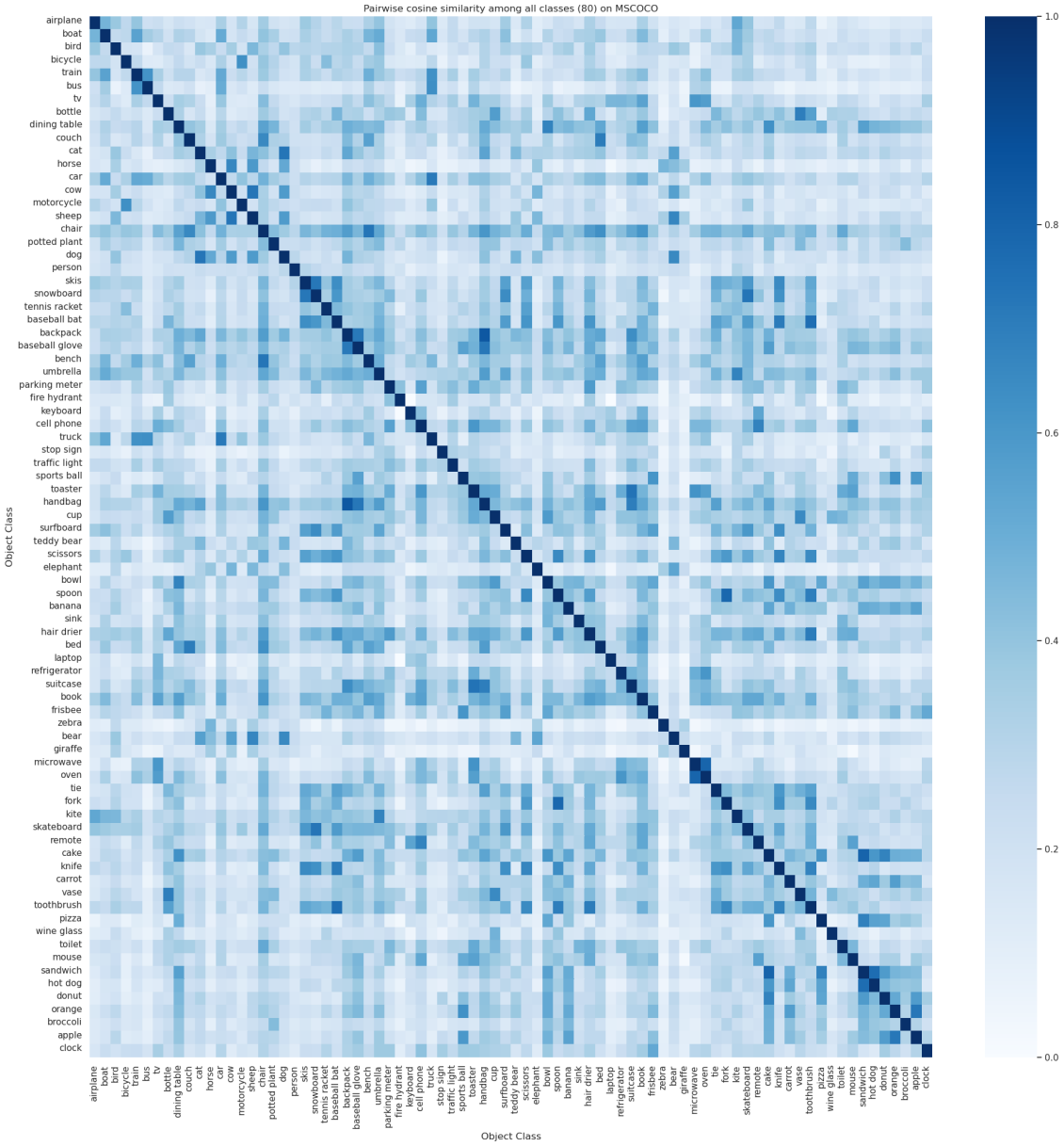

Figure 3. Paiwise cosine similarity of all classes (80) in MSCOCO. The first 20 classes are novel classes, and the rest are base classes.

original novel class images to generate positive and negative proposals of the novel classes for training.

Similar to meta-learning, we use the SGD optimizer with an initial learning rate of 0.001, momentum of 0.9, weight decay of 0.0001, and a batch size of 8. The difference is that we use a much smaller number of training iterations for fine-tuning. For 30-shot fine-tuning, the learning rate is divided by 10 after 2000 iterations, and the total number of training iterations is 3000. For 10-shot fine-tuning or fewer, the learning rate is divided by 10 after 1000 iterations, and the total number of training iterations is 1500.

As shown in Table 7 and 8 of the main paper, we can conclude that meta-learning is crucial for extreme few-shot (e.g., 1/2 shot) settings due to the strong generalization ability, and fine-tuning turns out to be more useful for larger shot (e.g., 10/30 shot) settings with more training images.

3. Visualization of the category-specific proposals by Attention-RPN [1]

As shown in Fig. 2, we can find that if the query image contains instances of the target class, the category-specific

proposals would be more likely the nearby regions. If not, the proposals could be regions of similar classes or just random regions. Motivated by this observation, we propose to build multiple class-specific Intra-Class Subgraphs for each query image, which is more efficient compared with building one single graph with all proposals and classes.

4. Pairwise cosine similarity of all classes in MSCOCO

We show in Fig. 3 the pairwise cosine similarity of all 80 classes (including 60 base classes and 20 novel classes) in MSCOCO. We can find that there are some similar classes between the novel classes and base classes, for example, car and truck, chair and bench. This motivates us to 'borrow' the robust features from these base classes to enhance novel class prototypes.

References

[1] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2020. 1, 2, 3

[2] Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2

[3] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2