Supplementary for Foreground Object Search by Distilling Composite Feature

Bo Zhang[1], Jiacheng Sui[2], and Li Niu*[1]

[1]Center for Machine Cognitive Computing of Artificial Intelligence Institute
Artificial Intelligence Institute, Shanghai Jiao Tong University
{bo-zhang, ustcnewly}@sjtu.edu.cn
[2]Xian Jiao Tong University
rookiecharles99@gmail.com

In this document, we provide additional materials to supplement our main text. We will first provide more details of our Foreground Object Search (FOS) datasets and the implementation of our method in Section 1 and 2, respectively. In Section 3, we will present the quantitative comparison between our method and baseline approaches on different categories. Meanwhile, more qualitative results of different methods will be provided in Section 4. In Section 5, we will demonstrate that our method can be applied to FOS for a mixture of different categories. In Section 6, we will apply the proposed method to new categories that have not been seen during training, which further verifies the generalization ability of our model. Then, we will study the effect of different hyper-parameters adopted in our method in Section 7, including three trade-off parameters used in our loss function and the ratio of positive and negative foregrounds per background during training stage. In Section 8, we will show some failure cases generated by our method and discuss the limitation of our method.

1. Our FOS Datasets

Previous works [9, 10, 12] on FOS did not release their datasets, which inspired us to build our own FOS datasets: S-FOSD with synthetic composite images and R-FOSD with real composite images. In this section, we will present more details about our dataset and compare our datasets with previous datasets.

1.1. Rules for Foreground Selection

To accommodate our task, we delete some categories and objects according to the following rules: 1) The categories where most of the foregrounds look similar, so that most foregrounds can be considered compatible (*e.g.*, lighthouse, apple); 2) The categories where most of the foreground objects usually appear non-independently, as parts of larger objects (*e.g.*, clothing, wheel, flower); 3) The objects that are too large or too small in the background image (*e.g.*, smaller than 5% or larger than 50% of the whole image). 4) The objects that are occluded by other objects. 5) The categories with too few remaining objects after removing occluded objects and objects with inappropriate sizes. Summarily, the above categories and objects are either unsuitable for FOS task or beyond our focus (geometry and semantic compatibility).

1.2. Remaining Foreground Categories

Following the above rules, we select 32 foreground categories to construct our FOS dataset, which are airplane, bird, book, bottle, box, bread, bus, cake, camera, car, cat, coffee cup, keyboard, couch, dog, duck, fish, goose, guitar, horse, laptop, cellphone, monkey, motorcycle, pen, person, frame, taxi, toilet, train, wastebin, watch.

1.3. S-FOSD Test Set Building

Recall in Section 3.1 of the main text, we built the test set of S-FOSD dataset mainly concerning its diversity and quality. Here we will provide more details about the construction process. For each category, we first extract the features of all foreground images using ResNet [2] pretrained on ImageNet [1], and then cluster them into 100 clusters based on feature distance. Then we select the foreground objects closest to the cluster centers along with their background images as candidates and remove low-quality samples, including blurred objects, the background images whose light conditions are particularly dim, and so on. After that, we randomly select 20 background images from the remaining samples for each category. Based on the clusters where the 20 background images are located, we select 25 foreground images closest to each cluster as candidate foreground im-

*Corresponding author

Dataset	coarse/fine	compatible factors	category	fg/category	bg	synthetic/real	human	public
CAIS-Training [9]	coarse	semantics	8	2,962∼38,418	86,800	synthetic	N	N
CAIS-Evaluation [9]	coarse	semantics	8	114∼364	80	real	Y	N
IFO [5]	fine	geometry, semantics	-	-	-	synthetic	Y	N
FFR-Training [8]	fine	geometry, style	15	-	16,700	synthetic	N	N
FFR-Evaluation [8]	fine	geometry, style	3	150	15	synthetic	Y	N
GALA-Pixabay [12]	coarse	geometry, lighting	914	912	833,964	synthetic	N	N
GALA-OpenImages [12]	coarse	geometry, lighting	350	3,926	1,374,344	synthetic	N	N
Our S-FOSD	coarse	geometry, semantics	32	500∼5,000	57,859	synthetic	N	Y
Our R-FOSD	coarse	geometry, semantics	32	200	640	real	Y	Y

Table 1. Comparison with previous FOS datasets. “coarse/fine”: coarse-grained or fine-grained retrieval. “fg/category”: foreground images per category. “bg”: background images. “synthetic/real”: synthetic/real composite images. “human”: human annotation on compatibility.

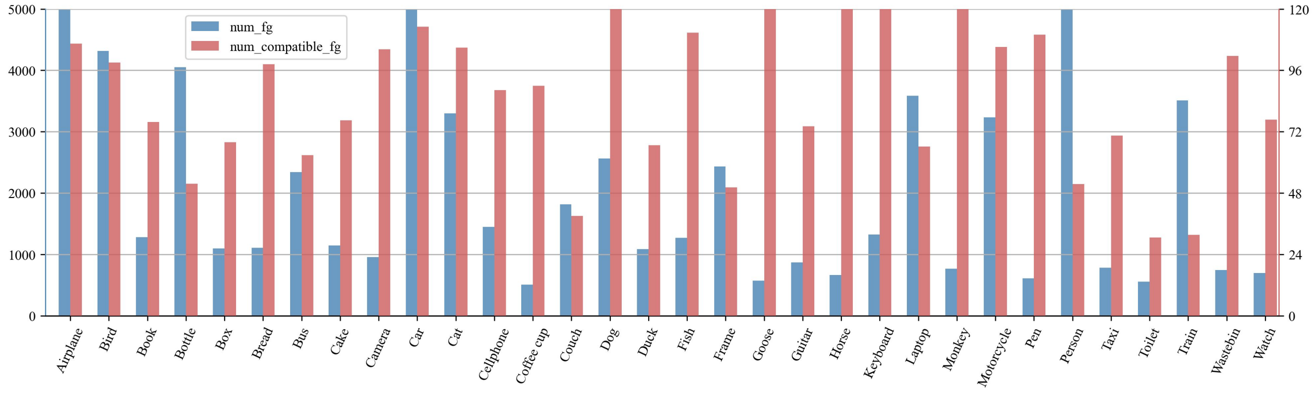


Figure 1. “num_fg”: the number of foreground images per category in our S-FOSD dataset. “num_compatible_fg”: the average number of compatible foreground images per background in one category of our R-FOSD dataset, in which we provide 200 candidate foregrounds for each background and the compatibility label is assigned by three human annotators.

ages. In this way, we obtain 20 background images and 20×25 = 500 candidate foreground images for each category. After filtering low-quality images, we randomly select 200 foregrounds and 20 backgrounds per category. By selecting high-quality samples from cluster centers, we ensure the quality and diversity of test samples in S-FOSD dataset, which helps provide more effective evaluation for FOS.

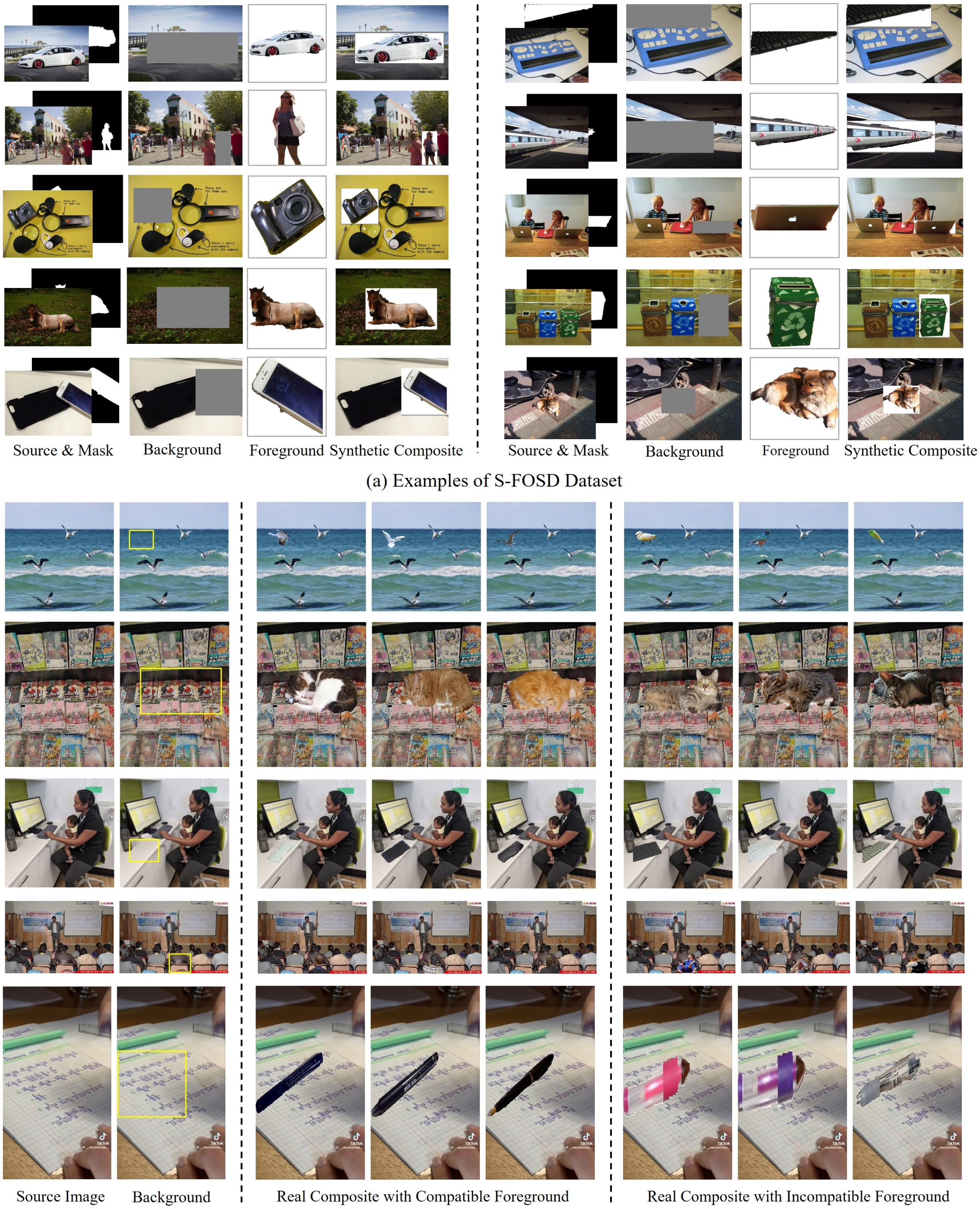
1.4. Comparison with Previous Datasets

Following [9, 12], we build our FOS datasets based on an existing large-scale real-world dataset, *i.e.*, Open Images [4]. Table 1 provides a summary comparison between our datasets and the datasets that are used in previous works [9, 10, 12, 5, 8]. Among these datasets, GALA-Pixabay and GALA-OpenImages [12] contain far more background and foreground images covering more categories, probably because that they did not filter some categories or objects like us. However, this may harm the quality of their dataset. In contrast, we remove some unsuitable categories and low-quality images to build dataset (see Section 1.1), which contributes to more effective training and evaluation on FOS task. Moreover, only the evaluation set of CAIS (CAIS-Evaluation) [9] and our R-FOSD dataset provide real composite images, which enables more practical evaluation for real-world applications. Moreover, different from the above works, we have released our datasets to facilitate research on FOS task.

1.5. Dataset Statistics

S-FOSD Dataset. The S-FOSD dataset contains totally 63,619 foreground images covering 32 categories. In Figure 1, we show the number of foregrounds per category in S-FOSD dataset. During experiments, S-FOSD dataset is divided into training set and test set. The training set has 57,219 pairs of foregrounds and backgrounds, with a maximum of 4800 pairs and a minimum of 300 pairs in one category. The test set provides 20 backgrounds and 200 foregrounds (including 20 foregrounds from the same images as the backgrounds) for each category.

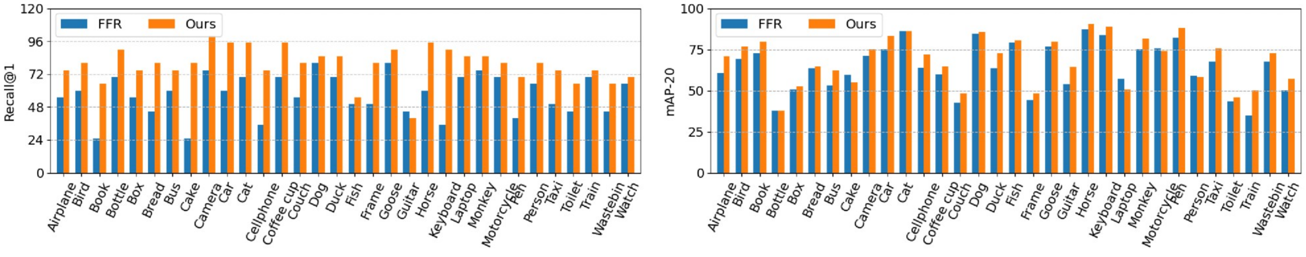
R-FOSD Dataset. The R-FOSD dataset shares the same foregrounds with the test set of S-FOSD, and has 20 back-



(a) Examples of S-FOSD Dataset

(b) Examples of R-FOSD Dataset

Figure 2. Some examples of our S-FOSD dataset and R-FOSD dataset. For R-FOSD dataset, we show real composite images generated by placing compatible/incompatible foregrounds in the query bounding box (yellow) on background, in which compatibility label is provided by human annotators.



(a) Comparison on S-FOSD Dataset

(b) Comparison on R-FOSD Dataset

Figure 3. Comparing our method with the most competitive baseline FFR [8] for each category in our S-FOSD and R-FOSD datasets.

grounds as well as 200 foregrounds per category. Each pair of background and foreground is evaluated by three human annotators in terms of their compatibility. Then, only the foreground objects that all annotators label as compatible are considered to be compatible. The resulting dataset contains 4∼190 compatible foregrounds per background. We present the average number of compatible foregrounds per background in a category in Figure 1.

1.6. Visualization Examples

In Section 3 of the main text, we have introduced the pipeline of constructing our S-FOSD and R-FOSD datasets. Here we present more examples of these two datasets in Figure 2. Similar to Figure 2 of the main text, we show source image with instance segmentation mask, background, foreground, and synthetic composite images for one example of S-FOSD dataset. As demonstrated in Figure 2 (a), the foreground and background in S-FOSD dataset are diverse enough to cope with various real-world scenarios. For each example of R-FOSD dataset, we show source image, background, and real composite images produced by inserting foreground into the given background image. Recall there are 200 candidate foregrounds for each background. We randomly select three compatible foregrounds and three incompatible foregrounds to composite with the background, in which compatibility labels are acquired from three human annotators. By observing the examples in Figure 2 (b), we can roughly verify the validness of the compatibility annotations in our R-FOSD dataset.

2. Implementation Details

Our method is implemented using PyTorch [6] and distributed on NVIDIA RTX 3090 GPU. We use the Adam optimizer [3] with a fixed learning rate of $1e^{-5}$ to train our model for 50 epochs. Following [10, 12], we adopt VGG-19 [7] pretrained on ImageNet [1] as backbone network for our discriminator D and encoders $\{E^f, E^b\}$. Before being fed into networks, both composite image and background image are directly resized to 224 × 224, while foreground image is first padded with white pixels to be a square image and then resized to 224 × 224. In this way, the composite feature map $\mathbf{F}^c$, background and foreground feature maps $\mathbf{F}^b, \mathbf{F}^f$ have the same shape $7 \times 7 \times 512$, *i.e.*, $h = w = 7, c = 512$. We implement the knowledge distillation module E^d with two convolution+relu operations. For discriminator D and distillation module E^d, we append a fully-connected layer with a $sigmoid$ function as binary classifier.

Recall we generate positive and negative foregrounds from S-FOSD dataset by using a pretrained classifier (see Section 4.1 of the main text). When training on S-FOSD dataset, we adopt the same 1:10 ratio of positive and negative foregrounds per background for different models. Additionally, we set the margin m in Eqn. (2) of the main text as 0.1, and $\lambda_{kd}, \lambda_{cls}$ in Eqn. (5) of the main text as 1 via cross-validation.

3. Comparison on Different Categories

In Table 1 of the main text, we have compared with different baseline methods [9, 11, 10, 8] on our S-FOSD and R-FOSD datasets, in which one metric is obtained by averaging over all categories. This comparison demonstrates that our method performs more favorably against previous baselines. To take a close look at the superiority of our method, we evaluate our model and the most competitive baseline (*i.e.*, FFR [8]) on each single category of our S-FOSD and R-FOSD datasets, in which we report their performance in term of Recall@1 and mAP-20, respectively. As demonstrated in Figure 3 (a), our method can generally achieve better results than FFR on different categories of S-FOSD dataset. In Figure 3 (b), it can be seen that our model also beats FFR in most categories. These results further prove the improvement of our method over previous baseline methods on FOS task.

4. More Qualitative Results

To better demonstrate the effectiveness of our method, we provide additional qualitative results of our method and baseline methods [9, 11, 10, 8] on our S-FOSD and R-FOSD datasets in Figure 4 and Figure 5, respectively. Given

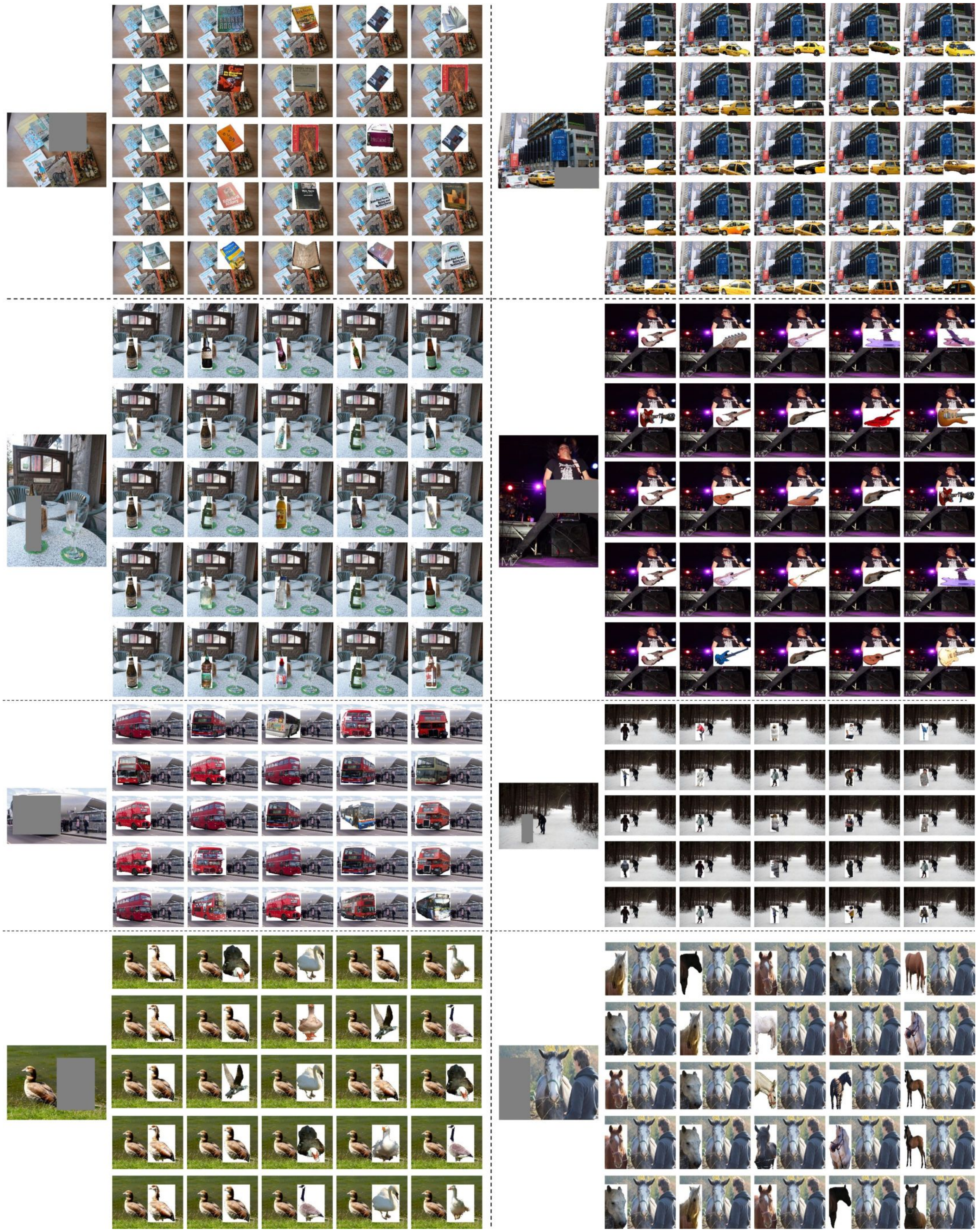


Figure 4. Qualitative comparison of different methods on our S-FOSD dataset. For a background image with query bounding box, which is filled by mean image pixel, we show several rows of the retrieval results from different methods, from top to bottom: CFO [9], UFO [10], GALA [12], FFR [8], and ours.

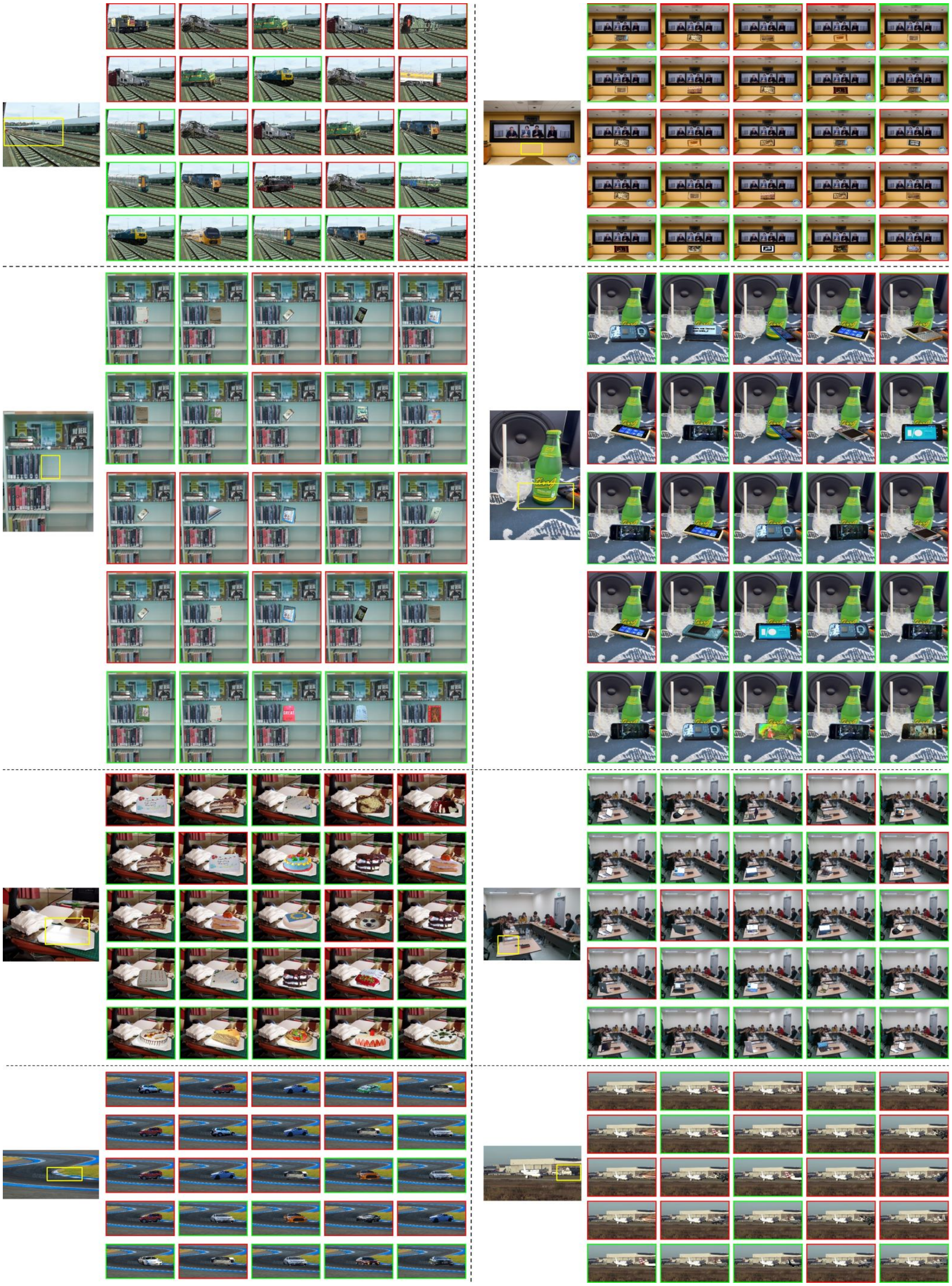


Figure 5. Qualitative comparison of different methods on our R-FOSD dataset. Each example contains a background image with query bounding box (yellow) and several rows of the retrieval results from different methods, from top to bottom: CFO [9], UFO [10], GALA [12], FFR [8], and ours. Additionally, we mark the foreground that is assigned with compatible (*resp.*, incompatible) label by human annotators using green (*resp.*, red) box.

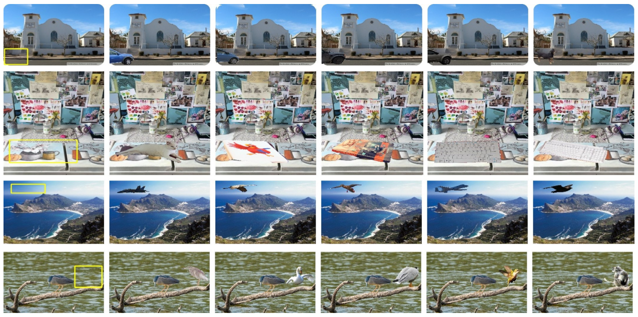


Figure 6. Applying our method to FOS from 32 categories of foregrounds. In each row, our method retrieves foregrounds for the background on the left and places the retrieved foreground in the query bounding box (yellow) on background to get composite image.

a background image with query bounding box, we show the composite images that are generated by placing the top-5 foregrounds of different methods in the query bounding box on the given background. The visualized results demonstrate that our method can generally find compatible foregrounds by considering both semantic and geometric factors. For example, the background on the left of the third row in Figure 4 has a sloping road, indicating that inserted “bus” should have a matching viewpoint, so as to generate a realistic composite image. Among the compared methods, only our method works well by returning compatible foregrounds with similar viewpoint. In the top-right example of Figure 5, our method retrieves more composite foregrounds (*i.e.*, bird) than other baselines for the given background scene, in which a flying “bird” appears more suitable to be placed on the background river. In summary, these qualitative comparisons further verify the effectiveness of the proposed method for FOS.

5. Retrieval from Different Categories

In this work, we focus on searching compatible foregrounds from specified category for a given background, which is referred to as constrained foreground object search. In real-world application scenario, user may retrieve foreground from different categories, which is referred to as unconstrained foreground object search. To investigate the performance of our model in this scenario, we employ our model to search compatible foregrounds from 32 categories in our R-FOSD dataset. It is worth noting that the retrieval process of our method is unchanged, which means that the model ranks different foregrounds by predicted compatibility scores. We provide several examples in Figure 6, which demonstrate that our method is capable of generating reasonable results in this scenario as well.

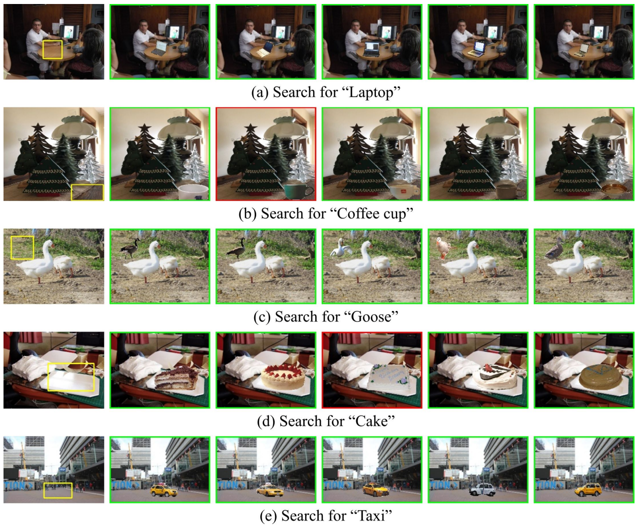

(a) Search for “Laptop”
(b) Search for “Coffee cup”
(c) Search for “Goose”
(d) Search for “Cake”
(e) Search for “Taxi”

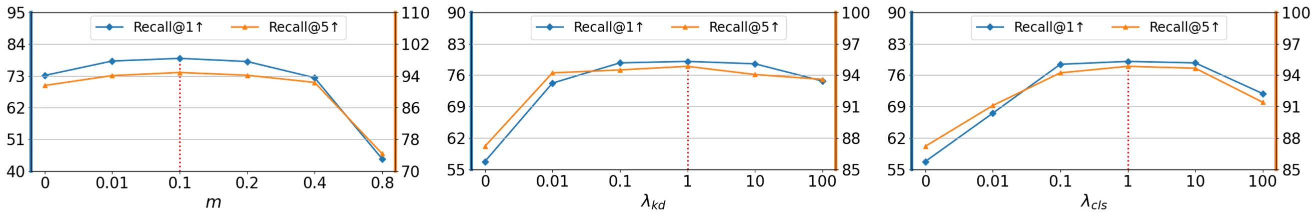
Figure 7. Evaluating our method on new categories that have not been seen during training. In each row, the left is one background image with query bounding box (yellow) in R-FOSD dataset and the other are composite images with retrieved foregrounds by our method, in which compatible (*resp.*, incompatible) foreground is marked with green (*resp.*, red) box.

6. Generalization to New Categories

To investigate the generalization ability of our learnt compatibility knowledge, we test our model on new categories that have not been seen during training. Specifically, we divide existing 32 categories of our S-FOSD dataset into five supercategories (*e.g.*, animal, carrier). We then randomly choose one item from each supercategory to build the test set and the rest forms the training set. After training, we evaluate our model on our R-FOSD dataset that adopts the same foregrounds as the test set of S-FOSD dataset. As shown in Figure 7, given a query background of R-FOSD dataset, our method typically can find compatible foregrounds (green box) from new categories even without training on these categories.

7. Hyper-parameter Analyses

Recall that we have three hyper-parameters in the main text, *i.e.*, margin m for triplet loss in Eqn. (2), distillation loss weight λ_{kd}, and classification loss weight λ_{cls} in Eqn. (5), which are respectively set as 0.1, 1, 1 via cross-validation by splitting 20% training samples of our S-FOSD dataset as validation set. Besides, the ratio of positive and negative foregrounds per background is also considered as a hyper-parameter. In this section, we study the performance variance of our method when varying those hyper-parameters, in which we evaluate on the test set of S-FOSD dataset and report results of Recall@k (R@k) in Figure 8.



(a) (b) (c)

Figure 8. Performance variation of our method with different hyper-parameters m in Eqn. (2), λ_{kd}, λ_{cls} in Eqn. (5) of the main text on our S-FOSD dataset. The dashed vertical lines denote the default values used in our other experiments.

Margin for Triplet Loss. It is worth mentioning that our triplet loss is calculated on the cosine distance between foreground feature and background feature, which means that the margin is meaningful only if its value lies in (0, 1). To evaluate the impact of different margins m for triplet loss, we vary m in the range of [0, 0.8], generating results shown in Figure 8 (a), in which we report the results of Recall@1 and Recall@5 on S-FOSD dataset. By comparing the results without triplet loss ($m = 0$) and the results with $m = 1$, we can see a clear gap between their performance, demonstrating the necessity of using triplet loss to learn discriminative foreground/background feature. When m varies in the range of [0.01, 0.2], Recall@1 is in the range of [77.97, 79.06] and Recall@5 is in the range of [94.06, 94.84], which indicates that our model is robust when setting m in a reasonable range.

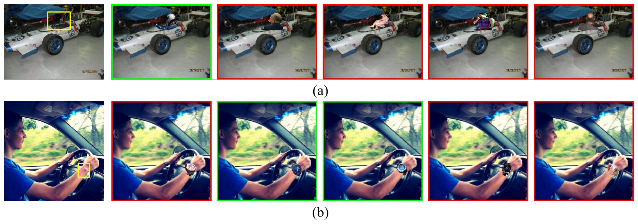
Distillation Loss Weight. With $m = 0.1$, we evaluate the results of our model adopting different distillation loss weights λ_{kd} in Figure 8 (b). When $\lambda_{kd} = 0$, the knowledge distillation module essentially degrades to a classifier and the resultant model is equivalent to the row 3 in Table 2 of the main text. In this case, the distilled feature cannot learn compatibility knowledge from composite image feature and using such feature may affect the prediction of foreground-background compatibility, leading to inferior performance. When $\lambda_{kd} \leq 1$, the performance increases as λ_{kd} increases, which implies that adding feature distillation could benefit the compatibility prediction via distilled feature. When λ_{kd} becomes larger than 1, the performance begins to drop. Moreover, we find that the model can achieve satisfactory results when λ_{kd} ranges from 0.1 to 10.

Classification Loss Weight. By setting $m = 0.1$ and $\lambda_{kd} = 1$, we further evaluate the performance of our model with different classification loss weights λ_{cls}, the results of which are shown in Figure 8 (c). For $\lambda_{cls} = 0$, the predicted compatibility scores cannot be guaranteed and thus we estimate the compatibility via foreground-background feature similarity. It can be seen that the model with $\lambda_{cls} = 1$ clearly outperforms the model with $\lambda_{cls} = 0$, which confirms the advantages of the proposed method over the two encoders. Moreover, Recall@1 is in the range of [78.42, 79.06] and Recall@5 is in the range of [94.22, 94.84] when λ_{cls} varies in the range of [0.1, 10], which implies that our model performs robust to λ_{cls} when setting λ_{cls} in a reasonable range.

	Pos	Neg	R@1↑	R@5↑	R@10↑	R@20↑
1	1	1	48.75	80.00	90.00	95.63
2	1	5	77.34	94.00	96.44	99.38
3	1	10	79.06	94.84	97.34	99.38
4	1	20	**80.72**	**95.23**	**98.50**	**99.56**
5	5	10	65.63	87.03	93.59	97.19
6	10	10	46.25	76.09	86.25	93.59

Table 2. The performance of our method trained using different ratios of positive and negative foregrounds per background on our S-FOSD dataset. “Pos” and “Neg” indicate the numbers of positive and negative foregrounds per background, respectively.

Ratio of Positive and Negative Foregrounds. Recall we generate positive and negative foregrounds from S-FOSD dataset by using a pretrained classifier in Section 4.1 of the main text. Then we set the ratio of positive to negative foregrounds per background as 1:10 for different models when training on S-FOSD dataset. To study the impact of training with different ratios, we vary the ratio and report the results of our method in Table 2. In row 1∼4, we use the only ground-truth positive foreground for one background and observe that the performance increases as the number of negative foreground increases, verifying the effectiveness of the selected negative foregrounds by the pretrained classifier. Based on row 3, we keep the number of negative foregrounds at 10 and increase the number of positive foregrounds in row 5 and 6, from which we can observe the significant performance drop. This may be attributed to the fact that the set of positive foregrounds identified by the classifier still contains some implausible samples. In summary, although using more training foregrounds per background may achieve better results, this would significantly slow down the training speed. To seek for the trade-off between training speed and model performance,


(a)

(b)

Figure 9. Failure cases of our method produced on R-FOSD dataset. Given a background image with query bounding box (yellow) on the left of a row, we composite each of the returned foregrounds by our method with the given background and present them on the right, in which the compatible (*resp.*, incompatible) foregrounds are indicated with green (*resp.*, red) boxes.

we finally adopt the 1:10 ratio of positive and negative foreground per background for all baselines and our method in experiments.

8. Discussion on Limitation

Although our method is able to find compatible foregrounds for most query backgrounds, it may fail on some challenging cases. For example, as shown in Figure 9 (a), all the retrieved foregrounds by our method are upper body images with non-frontal posture, yet most of them (red box) are different from the compatible one (green box) on boundary truncation and hand action, yielding implausible composite images. This can be attributed to the fact that our model mainly considers coarse-grained foreground retrieval, which makes it tougher to find foregrounds with particular attributes. In addition, as discussed in [12], the search space of FOS is bounded by the gallery of foregrounds and thus there may be a few or even no perfectly suitable foreground for a given background. In Figure 9 (b), we present a such case where our method fails to return satisfactory results, because there are only a few compatible foregrounds for the given background in database.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 4

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 4

[4] Alina Kuznetsova, Hassan Rom, Neil Gordon Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4. *International Journal of Computer Vision*, 128:1956–1981, 2020. 2

[5] Boren Li, Po-Yu Zhuang, Jian Gu, Mingyang Li, and Ping Tan. Interpretable foreground object search as knowledge distillation. In *ECCV*, 2020. 2

[6] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. In *NIPS*, 2019. 4

[7] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 4

[8] Zongze Wu, Dani Lischinski, and Eli Shechtman. Fine-grained foreground retrieval via teacher-student learning. In *WACV*, 2021. 2, 4, 5, 6

[9] Hengshuang Zhao, Xiaohui Shen, Zhe L. Lin, Kalyan Sunkavalli, Brian L. Price, and Jiaya Jia. Compositing-aware image search. In *ECCV*, 2018. 1, 2, 4, 5, 6

[10] Yinan Zhao, Brian L. Price, Scott D. Cohen, and Danna Gurari. Unconstrained foreground object search. In *ICCV*, 2019. 1, 2, 4, 5, 6

[11] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. Learning a discriminative model for the perception of realism in composite images. In *ICCV*, 2015. 4

[12] Sijie Zhu, Zhe Lin, Scott D. Cohen, Jason Kuen, Zhifei Zhang, and Chen Chen. GALA: Toward geometry-and-lighting-aware object search for compositing. In *ECCV*, 2022. 1, 2, 4, 5, 6, 9