A. Ablation on the Copy-Paste method

In this section we present ablations for our Copy-Paste method. We use Mask R-CNN EfficientNetB7-FPN architecture and image size of 640×640 for our experiments.

Subset of pasted objects. In our method, we paste a *random* subset of objects from one image onto another image. Table 10 shows that although we get improvements from pasting only *one* random object or *all* the objects of one image into another image, we get the best improvement by pasting a *random* subset of objects. This shows that the added randomness introduced from pasting a subset of objects is helpful.

Blending. In our experiments, we smooth out the edges of pasted objects using alpha blending (see Section 3). Table 10 shows that this is not an important step and we get the same results without any blending in contrast to [13] who find blending is crucial for strong performance.

Setup	Box AP	Mask AP
EfficientNetB7-FPN (640)	48.5	42.7
w/ Copy-Paste (one object)	(-0.9) 49.1	(-0.6) 43.1
w/ Copy-Paste (all objects)	(-0.3) 49.7	(-0.4) 43.3
w/ Copy-Paste (no blending)	**50.0**	**43.7**
w/ Copy-Paste	**50.0**	**43.7**

Table 10. Ablation studies for the Copy-Paste method on COCO. We study the value of applying blending to pasted objects along with how many objects to paste from one image to another.

Scale jittering. In this work, we show that by combining large scale jittering and Copy-Paste we obtain a significant improvement over the baseline with standard scale jittering (Figure 1). In the Copy-Paste method, we apply independent random scale jittering on both the pasted image (image that pasted objects are being copied from) and the main image. In Table 11 we study the importance of large scale jittering on both the main and the pasted images. Table 11 shows that most of the improvement from large scale jittering is coming from applying it on the main image and we only get slight improvement (0.3 box AP and 0.2 Mask AP) from increasing the scale jittering range for the pasted image.

Main Image	Pasted Image	Box AP	Mask AP
SSJ	SSJ	(-1.9) 48.1	(-1.6) 42.1
SSJ	LSJ	(-2.3) 47.7	(-1.9) 41.8
LSJ	SSJ	(-0.3) 49.7	(-0.2) 43.5
LSJ	LSJ	**50.0**	**43.7**

Table 11. Ablation study on scale jittering methods for the main image and the pasted image.

B. Copy-Paste provides more gain on harder categories of COCO

Figure 6 shows the relative AP gain per category obtained from applying Copy-Paste on the COCO dataset. Copy-Paste improves the AP of all the classes except hair drier. In Figure 6 classes are sorted based on the baseline AP per category. We observe most of the classes with the highest improvement are on the left (lower baseline AP) which shows Copy-Paste helps the hardest classes the most.

C. How likely objects are copied to an unmatched scene?

In our method, we copy objects from a random image to another random image without considering the context of the images. In this section we compute the probability of copying objects to an unmatched scene category (context) of indoor or outdoor.

COCO images do not have scene categories. But, we use COCO-panoptic labels to assign the COCO images to indoor or outdoor scene categories. We found there are 42538 indoor and 71017 outdoor images (we couldn't estimate the category of the rest 4732 images). Table 12 shows the probability of copying objects from one scene category to another. Therefore, we copy objects to an unmatched scene in about half (46.8%) of generated images.

from \ to	indoor	outdoor
indoor	14.1%	23.4%
outdoor	23.4%	39.1%

Table 12. Probability of copying objects from one scene category to another scene category for COCO dataset.

D. Benchmark results on different object sizes

In the table 1 we report Copy-paste performance of variety of model architectures. In table 13 we provide additional benchmarks on different object sizes.
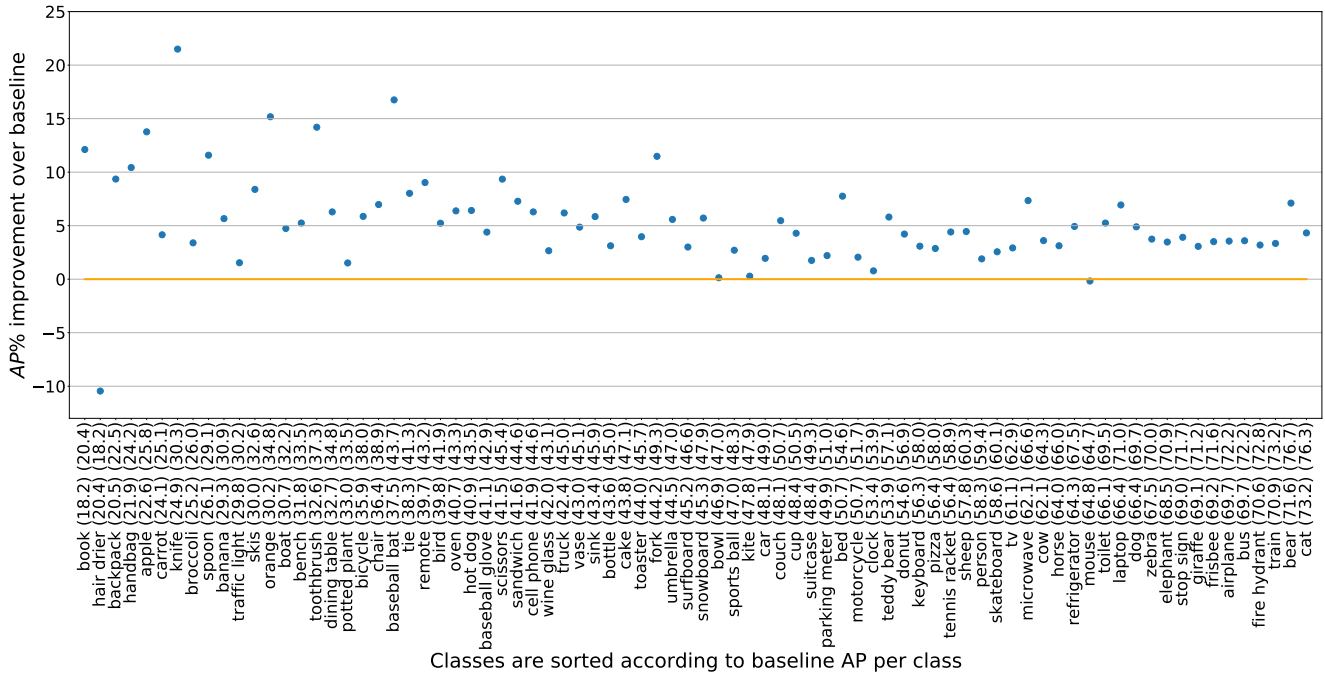
Figure 6. Per category relative AP improvement from Copy-Paste on 80 classes of COCO dataset. Numbers in the parentheses show the AP per category of the baseline model (first number) and the model trained with Copy-Paste (second number). Each number is the average over 5 runs. Classes are sorted based on the baseline AP per class.

Model	box AP	box AP$_s$	box AP$_m$	box AP$_l$	mask AP	mask AP$_s$	mask AP$_m$	mask AP$_l$
Res-50 FPN (1024)	47.2	28.5	49.6	64.6	41.8	23.0	44.3	60.1
w/ Copy-Paste	(+1.0) 48.2	(+0.4) 28.9	(+1.4) 51.0	(+1.8) 66.4	(+0.6) 42.4	(+0.2) 23.2	(+0.9) 45.2	(+1.1) 61.2
Res-101 FPN (1024)	48.4	29.2	51.1	65.8	42.8	23.5	45.5	60.4
w/ Copy-Paste	(+1.4) 49.8	(+1.3) 30.5	(+1.8) 52.9	(+1.1) 66.9	(+0.8) 43.6	(+1.0) 24.5	(+1.4) 46.9	(+1.1) 61.5
Res-101 FPN (1280)	49.1	30.4	51.9	66.6	43.1	24.5	46.0	61.6
w/ Copy-Paste	(+1.2) 50.3	(+1.3) 31.7	(+1.8) 53.7	(+0.6) 67.2	(+1.1) 44.2	(+1.2) 25.7	(+1.5) 47.5	(+0.2) 61.8
Eff-B7 FPN (1280)	51.1	33.3	53.9	67.9	44.8	26.6	47.9	62.7
w/ Copy-Paste	(+1.5) 52.6	(+1.0) 34.3	(+1.7) 55.6	(+2.3) 70.2	(+1.1) 45.9	(+0.9) 27.5	(+1.5) 49.4	(+1.8) 64.5

Table 13. Box AP and Mask AP benchmark results on different object sizes for models trained with different backbones and image sizes.