Self-Supervised Scene De-occlusion
Supplementary Materials

Xiaohang Zhan[1], Xingang Pan[1], Bo Dai[1], Ziwei Liu[1], Dahua Lin[1], and Chen Change Loy[2]

[1]CUHK - SenseTime Joint Lab, The Chinese University of Hong Kong
[2]Nanyang Technological University

[1]{zx017, px117, bdai, zwliu, dhlin}@ie.cuhk.edu.hk
[2]ccloy@ntu.edu.sg

A. Implementation Details

In our experiments, the backbone for PCNet-M is UNet [1] with a widening factor 2, and that for PCNet-C is a UNet equipped with partial convolution layers [2]; while note that PCNets do not have restrictions on backbone architectures. For both PCNets, the image or mask patches centering on an object are cropped by an adaptive square and resized to 256x256 as inputs.

For COCOA, the PCNet-M is trained using SGD for 56K iterations with an initial learning rate 0.001 decayed at iterations 32K and 48K by 0.1. For KINS, we stop the training process earlier at 32K. The batch size is 256 distributed on 8 GPUs (GTX 1080 TI). The hyper-parameter γ that balances the two cases in training PCNet-M is set to 0.8. In current experiments, we do not use RGB as an input to PCNet-M, since we empirically find that introducing RGB through concatenation makes little differences. It is probably because for these two datasets, modal masks are informative enough for training; while we believe in more complicated scenes, RGB will exert more influence if introduced in a better way.

For PCNet-C, we modify the UNet to take in the concatenation of image and modal mask as the input. Apart from the losses in [2], we add an extra adversarial loss for optimization. The discriminator is a stack of 5 convolution layers with spectral normalization and leaky ReLU (slope=0.2). The PCNet-C is fine-tuned for 450K iterations with a constant learning rate 10^{-4} from a pre-trained inpainting network [2]. We adapt the pre-trained weights to be compatible for taking in the additional modal mask.

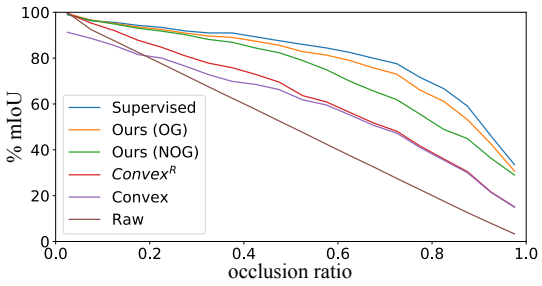

Figure 1. Performances of different approaches under a growing occlusion ratio, evaluated on KINS testing set.

B. Discussions

B.1. Analysis on varying occlusion ratio.

Fig. 1 show the amodal completion performances of different approaches under varying ratios of occluded area. Naturally, larger occlusion ratios result in lower performances. Under high occlusion ratios, our full method (*Ours (OG)*) surpasses the baseline methods by a large margin.

B.2. Does it support mutual occlusion?

As a drawback, our approach does not support cases where two objects are mutually occluded as shown in 2, because our approach focuses on object-level de-occlusion.

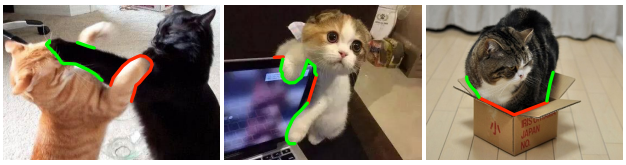

Figure 2. Mutual occlusion cases. Green boundaries show one object occlude the other and red boundaries vice versa.

(a-1) (a-2) (b) (c) (d)

Figure 3. (a-1) and (a-2) represent case 1 and case 2 in training, respectively; (b) - (d) represent possible cases in testing. Among the test cases, only the A in (b) will be completed.

For mutual occlusions, the ordering graph cannot be defined, therefore fine-grained boundary-level de-occlusion is required. It leaves an open question to scene de-occlusion problem. Nonetheless, our approach works well if more than two objects are cyclically occluded as shown in Fig. 7 in the main paper.

B.3. Will case 2 mislead PCNet-M?

As shown in Fig.3, one may have concerns that in case (a-2) when not-to-complete strategy is applied, the boundary between A and $B\backslash A$ might include a contour shown in green where A is occluded by a real object, namely C. Therefore, it might teach PCNet-M a wrong lesson if the yellow shaded region is taught not to be filled.

Here we explain why it will not teach PCNet-M the wrong lesson. First of all, PCNet-M learns to complete or not to complete the target object *conditioned on* a surrogate occluder. As shown in Fig. 3, as PCNet-M is taught to complete $A\backslash B$ in (a-1) while not to complete A in (a-2), it has to discover cues indicating that A is below B in (a-1) and A is above B in (a-2). The cues might include the shape of two objects, the shape of common boundary, junctions, *etc*. In testing time, *e.g*. in (b) when regarding the real C as the condition, it is easy for PCNet-M to tell that C is above A from those cues. Therefore PCNet-M actually inclines to case 1, when A will be completed conditioned on C.

Then which case does this not-to-complete strategy affect? The case in (c) shares very similar occlusion patterns with (a-2), especially in the upper right part of the common boundary, showing strong cues that A is above C, in which case PCNet-M will not complete A as expected. However, case (c) is abnormal and unlikely to exist in the real world. The situation where the not-to-complete strategy really takes effect lies in case (d). In this case when strong cues indicate that A is above D, the PCNet-M is taught not to extend A across $A\&D$ boundary to invade D.

C. Visualization

As shown in Fig. 4, our approach enables us to freely adjust scene spatial configurations to re-compose new scenes. The quality could be further improved with the advance of image inpainting, since the PCNet-C shares a similar network architecture and training strategy to image inpainting.

References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[2] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *ECCV*, 2018.

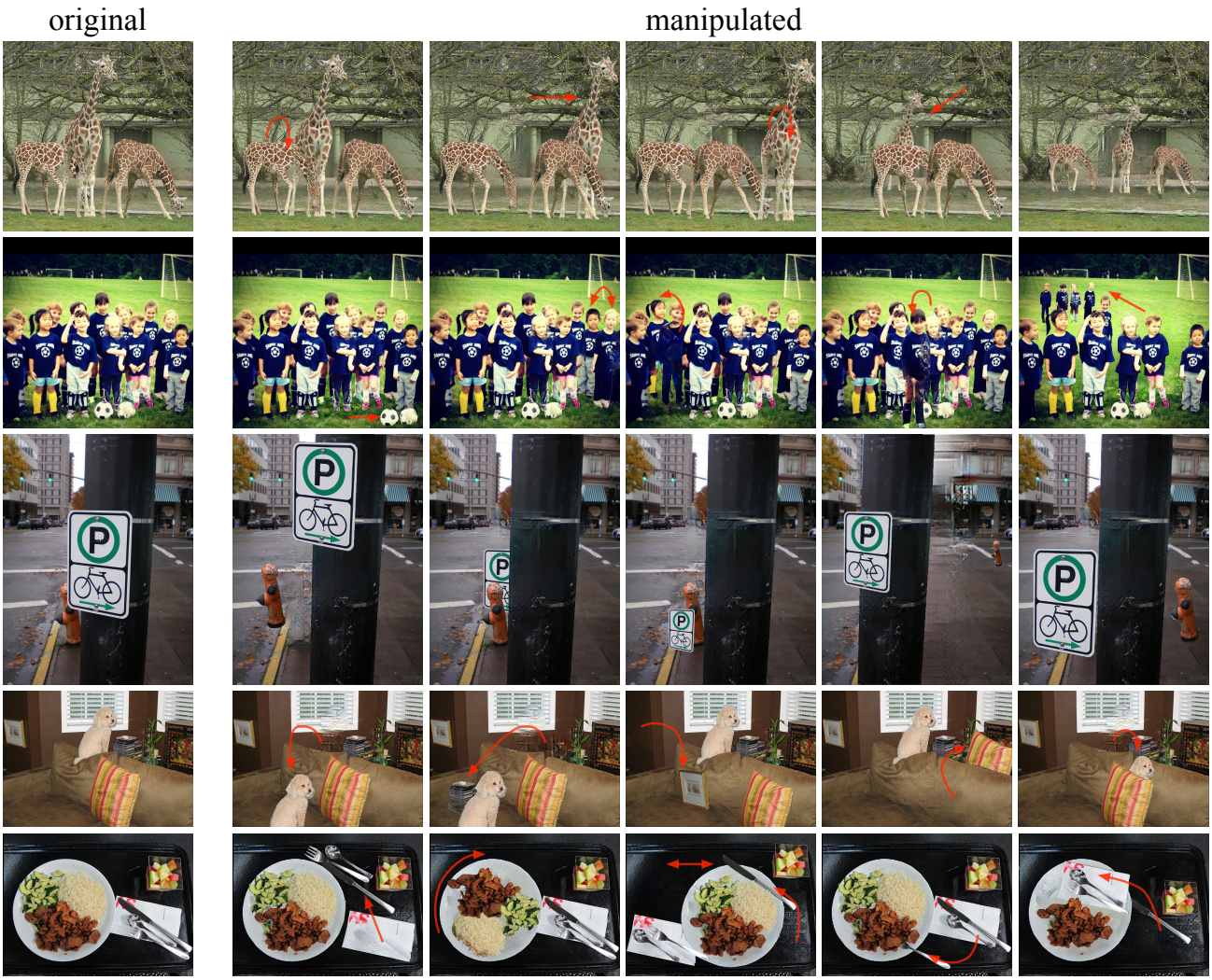

Figure 4. Scene manipulation results based on our de-occlusion framework. Inconspicuous changes are marked with red arrows. A video demo can be found in the project page: `https://xiaohangzhan.github.io/projects/deocclusion/`.