

6 Supplementary materials

This section contains the supplementary materials.

6.1 Source code

We share our implementation code to make it easy to reproduce our results. The source-code is attached to the supplementary materials in a ‘code’ directory. We also provide detailed instructions for training and evaluating our models in ‘README.md’ files.

6.2 Future works

In this paper we mostly focused on the texture bias [14]. Future works include studying the shape bias more carefully. We hope to answer the question of ‘Can we employ a technique of image generation, to diversify the shape attributes in a controlled way that can further help a down-stream task?’. Another direction for future work, is to investigate cases such as fine-grained classification on birds species, and evaluate the impact of texture and shape bias. These kinds of tasks might require special considerations/conditioning when generating new instances.

6.3 Additional visualizations

Figure 6 and 7 show the classification and object detection performance of various models for baselines and our method. The error bars of multiple runs demonstrate the statistical consistency of the improvements.

Next, we provide a large set of additional visualizations. Figure 8 shows patches of original images and several repainted versions. We observe from this figure the diversification of texture patterns of stripes on a zebra, parachute, and sea wave. Figure 9 and 10 illustrate additional pairs of original and repainted images. Figure 11, 12, 13, 14, 15, 16, and 17 demonstrate additional visualization for multiple instances generation from the COCO dataset. Similarly, Figure 18 and 19 show instance generation from the ImageNet dataset. In Figure 20, we provide several examples of failure in generating good-looking images, and explain that it’s not directly the main goal of the model to generate natural or good-looking images.

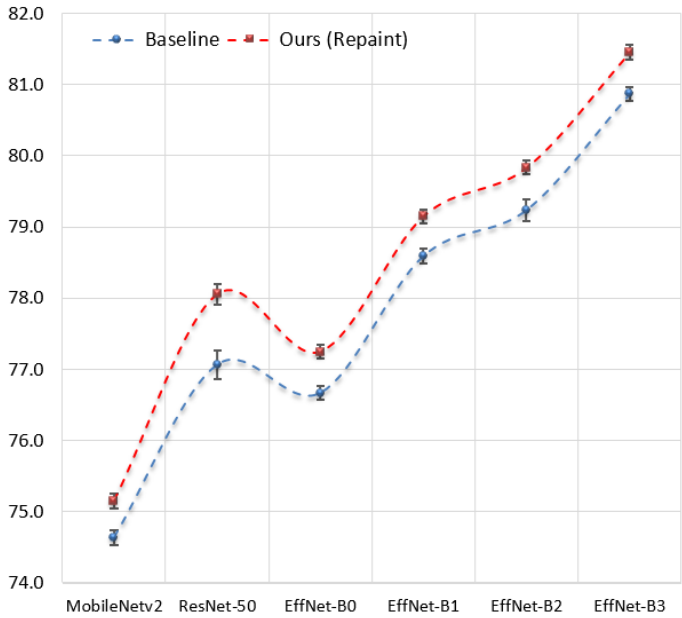


Figure 6: Performance evaluations on the ImageNet dataset. Top-1 (%) classification accuracy for baselines and repaint. Error-bars demonstrate consistent improvements.

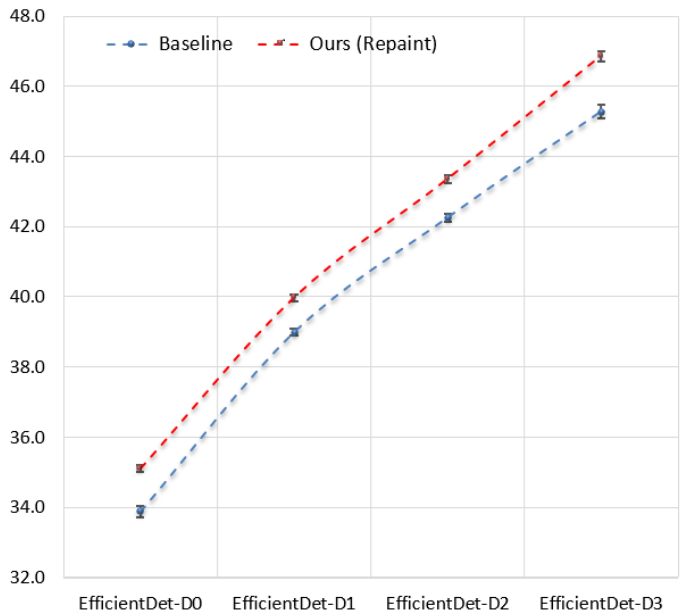


Figure 7: Performance evaluations on the COCO dataset. mAP @0.50:0.95 (%) for baselines and repaint. Error-bars demonstrate statistically consistent improvements.

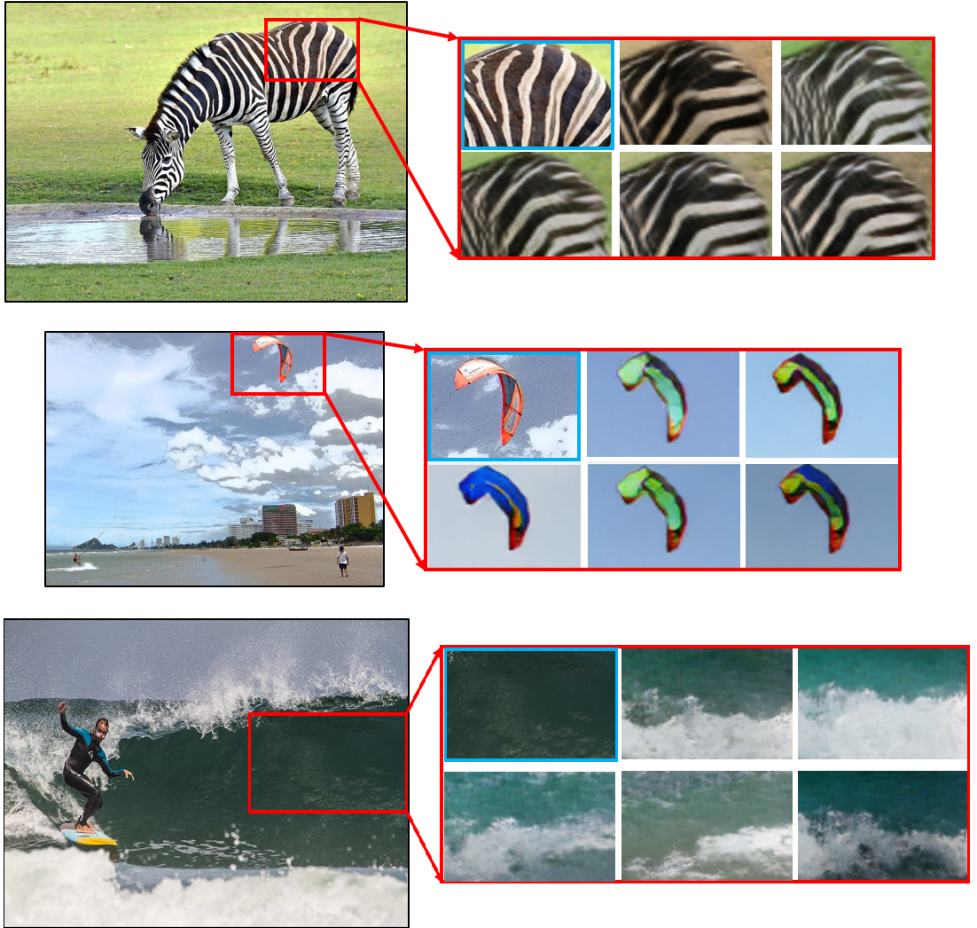


Figure 8: Both texture and color are repainted. Here we compare an original patch (top-left) with multiple repainted instances.

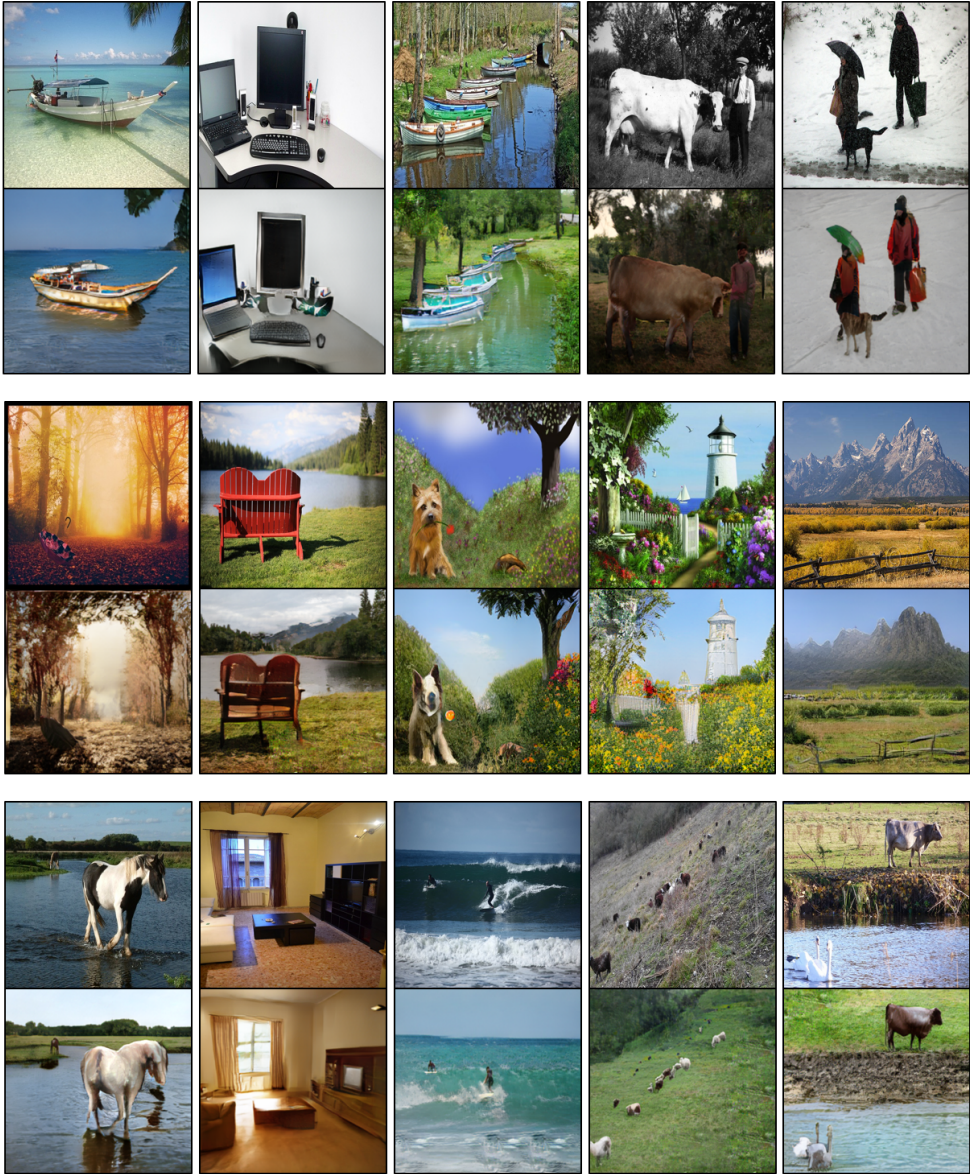


Figure 9: Example generated images in pairs: in each case, the upper row is an original image, and the lower row shows the repainted version.

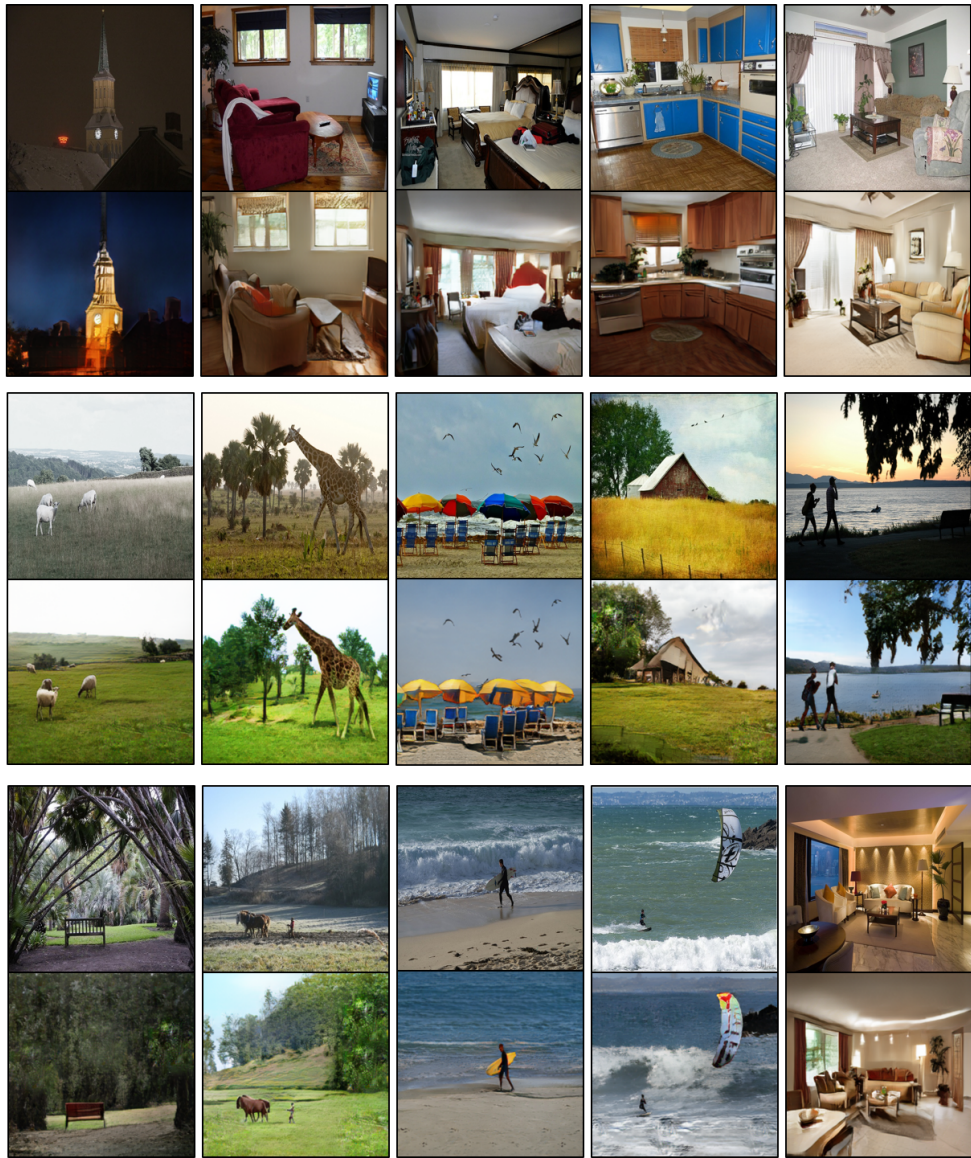


Figure 10: Example generated images in pairs: in each case, the upper row is an original image, and the lower row shows the repainted version.

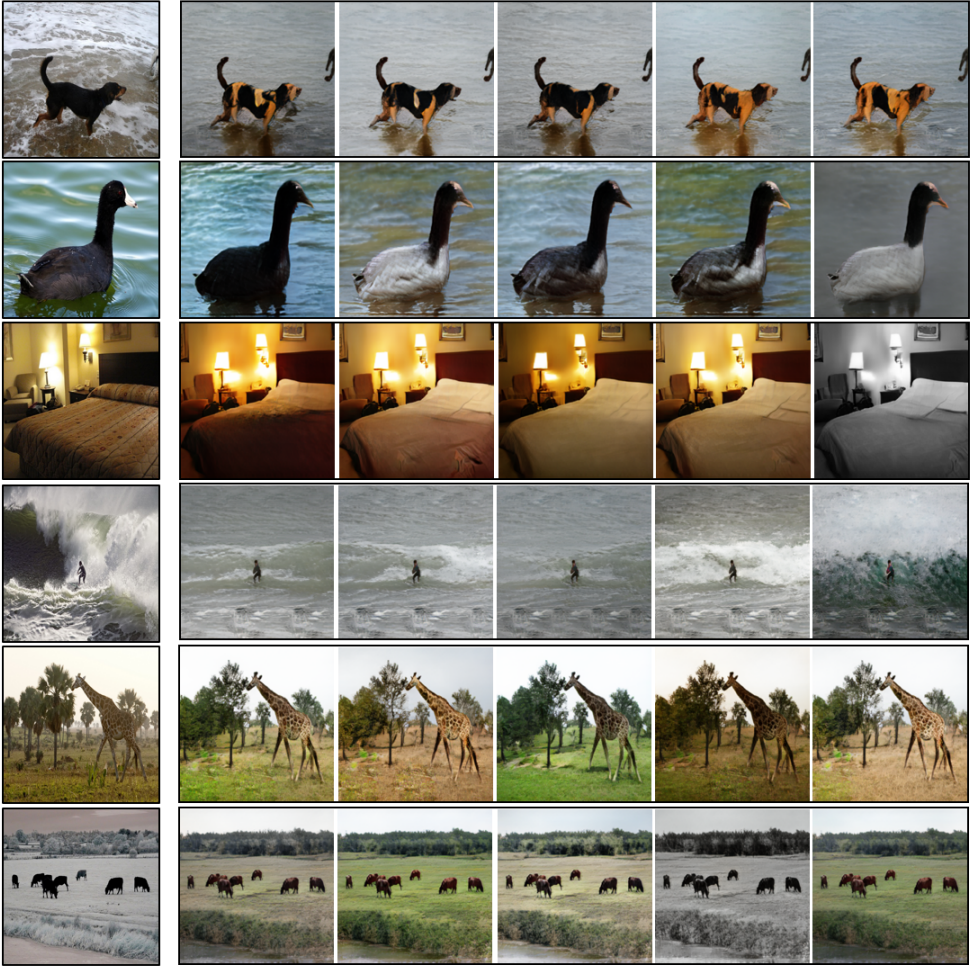


Figure 11: Example generated images from the COCO dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

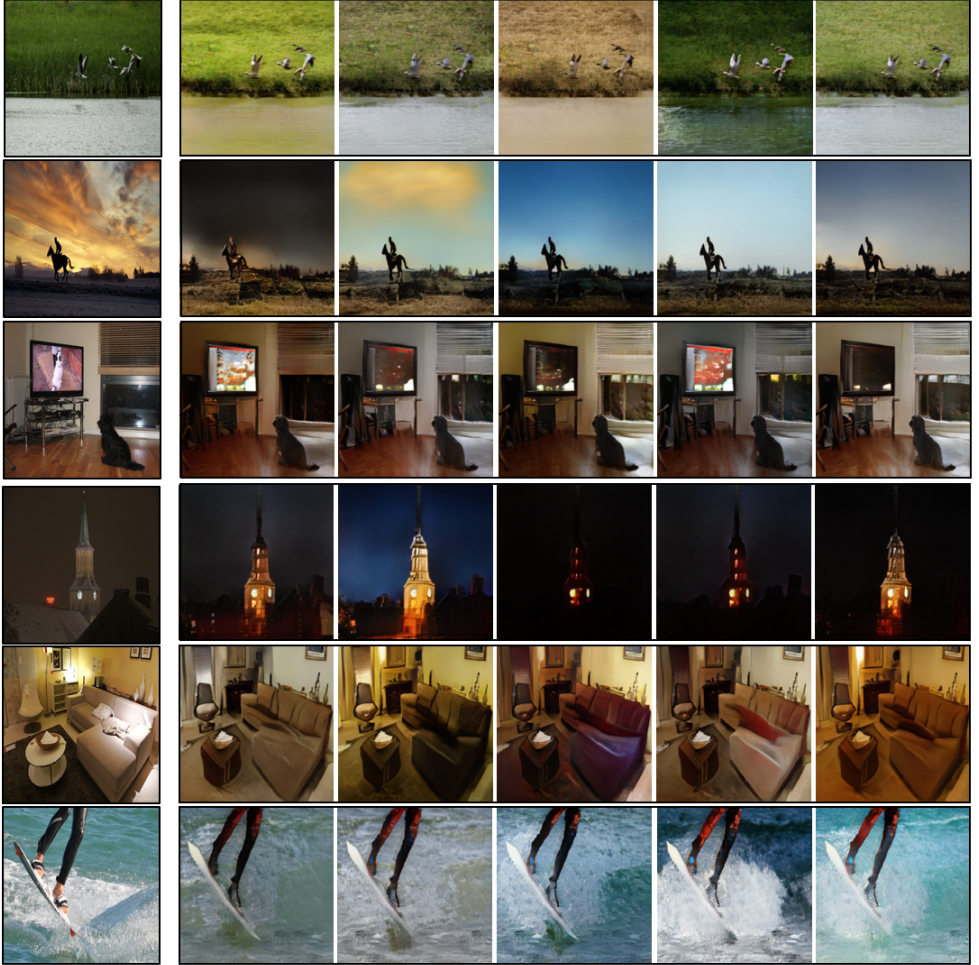


Figure 12: Example generated images from the COCO dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

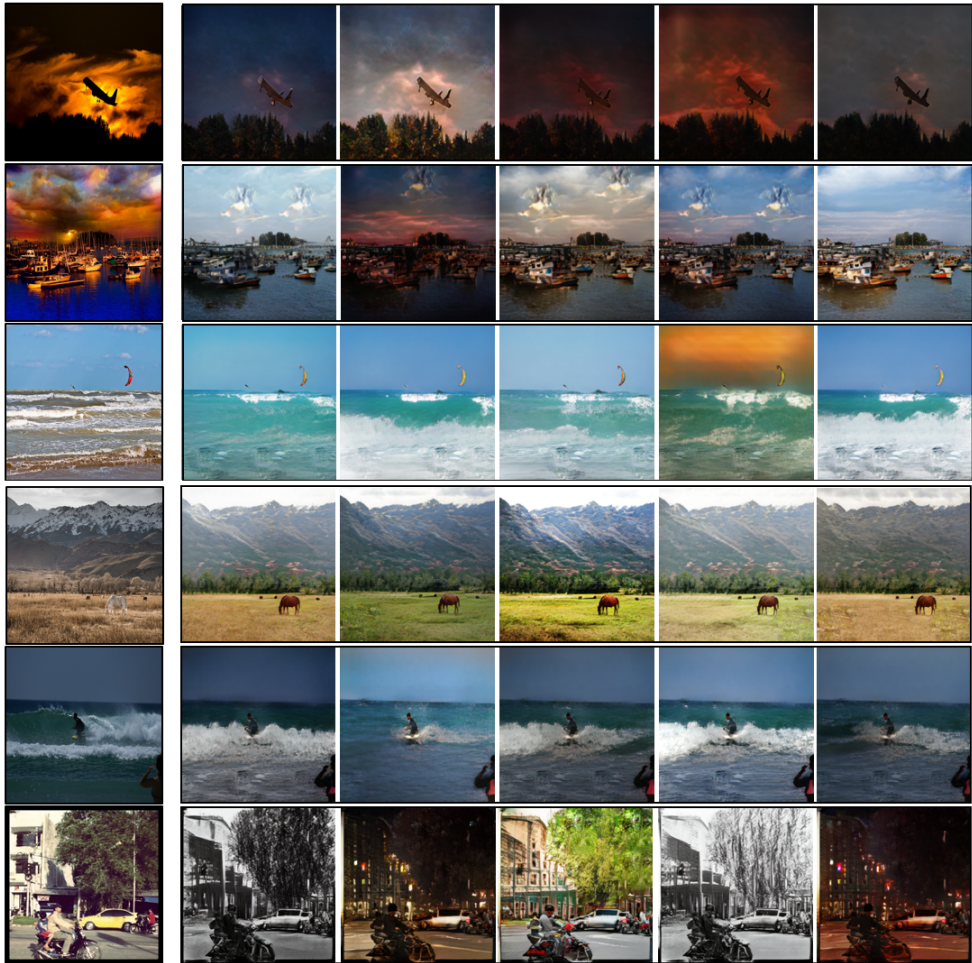


Figure 13: Example generated images from the COCO dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

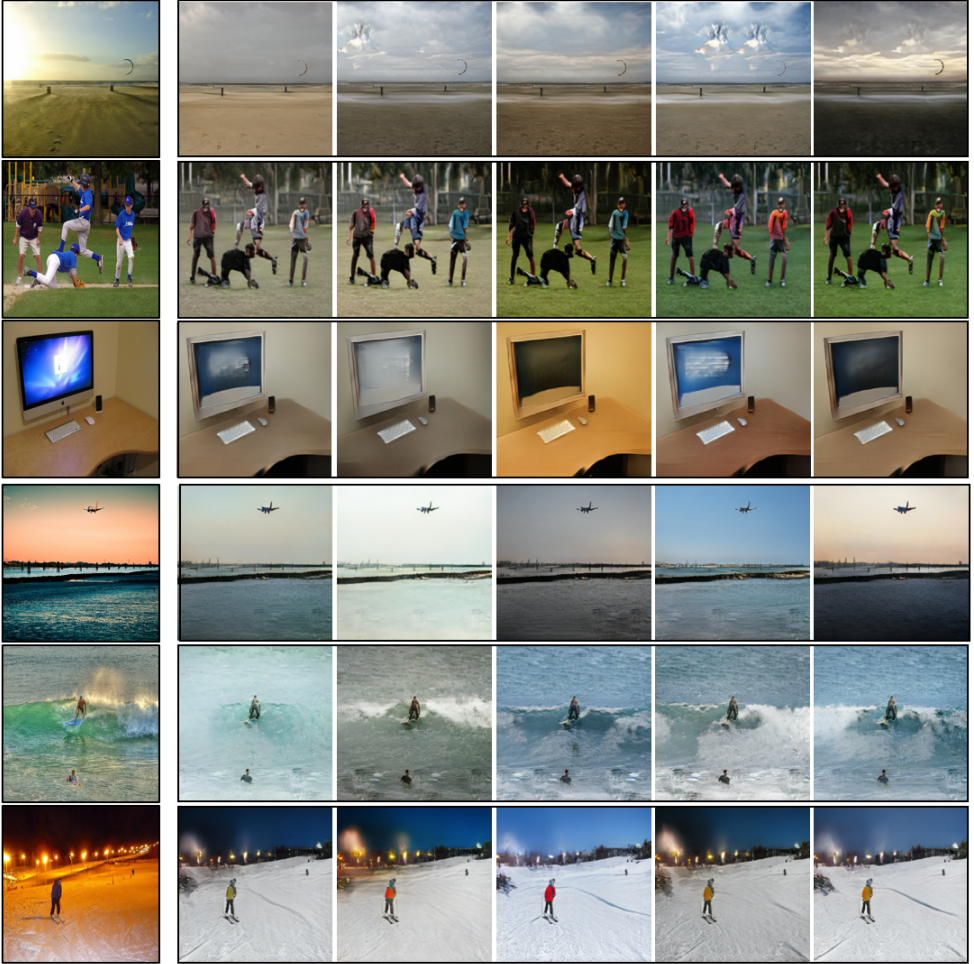


Figure 14: Example generated images from the COCO dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

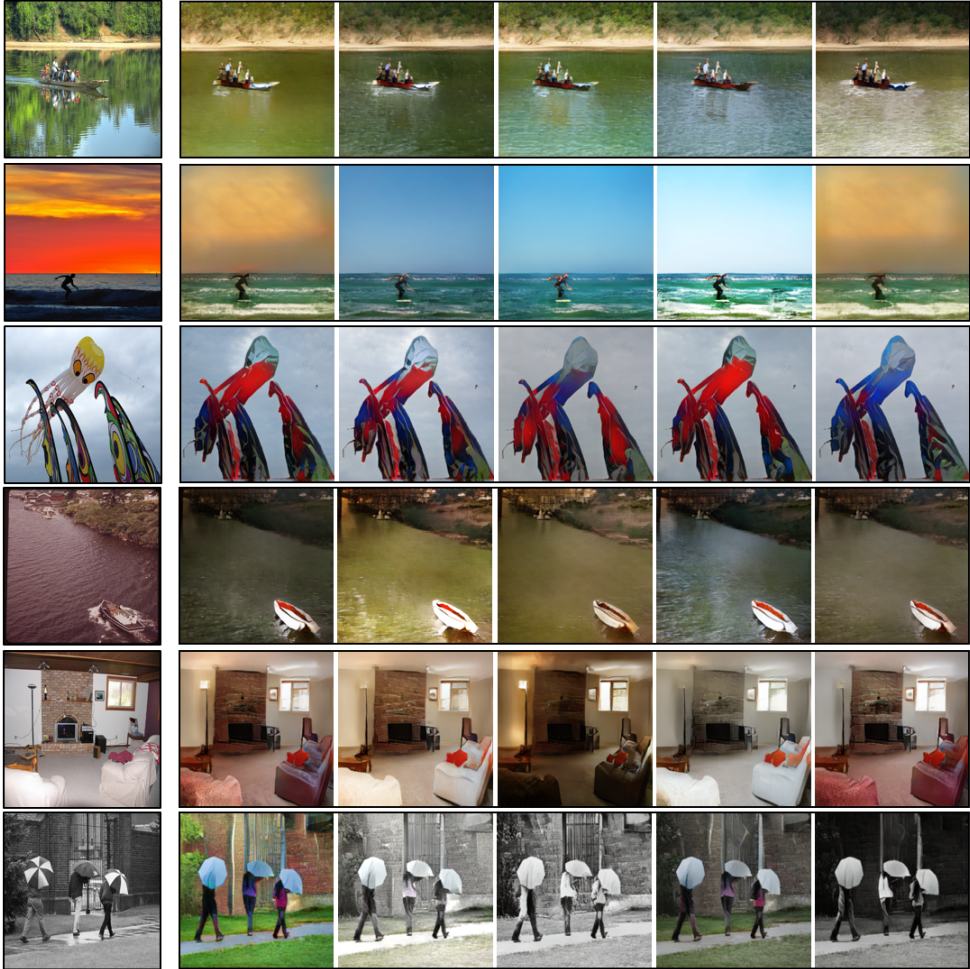


Figure 15: Example generated images from the COCO dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

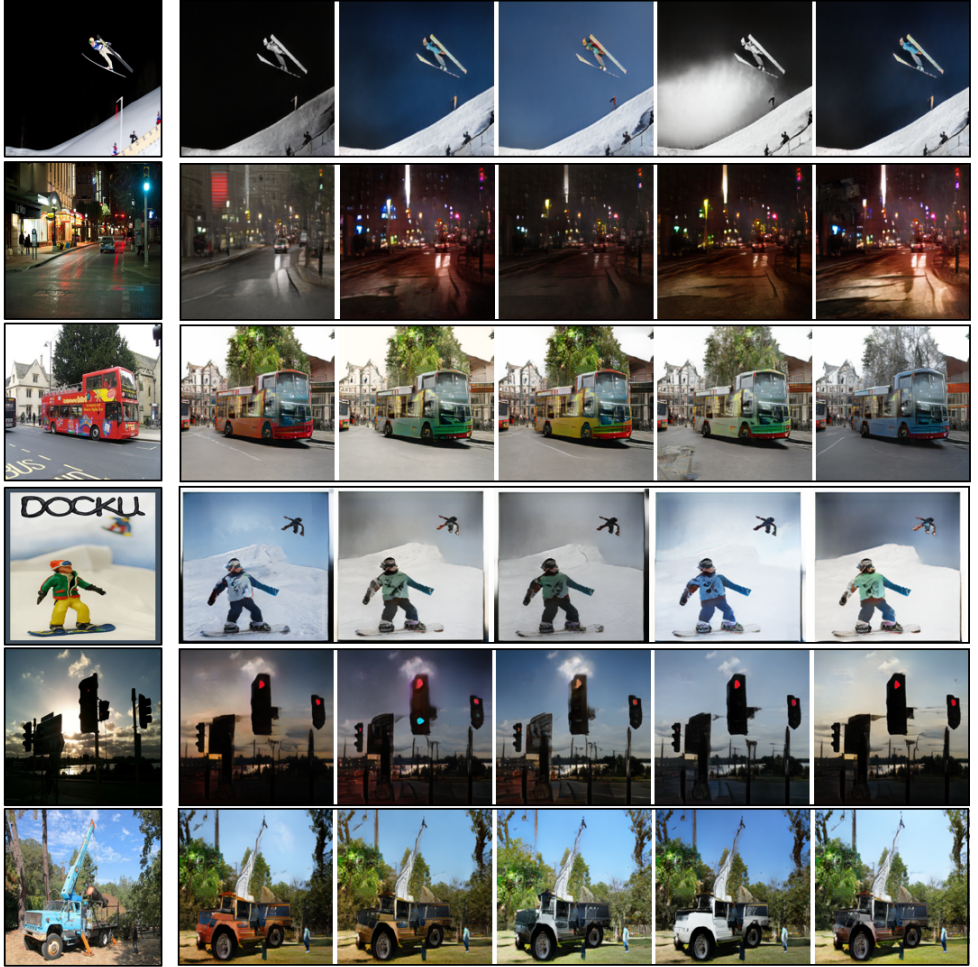


Figure 16: Example generated images from the COCO dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

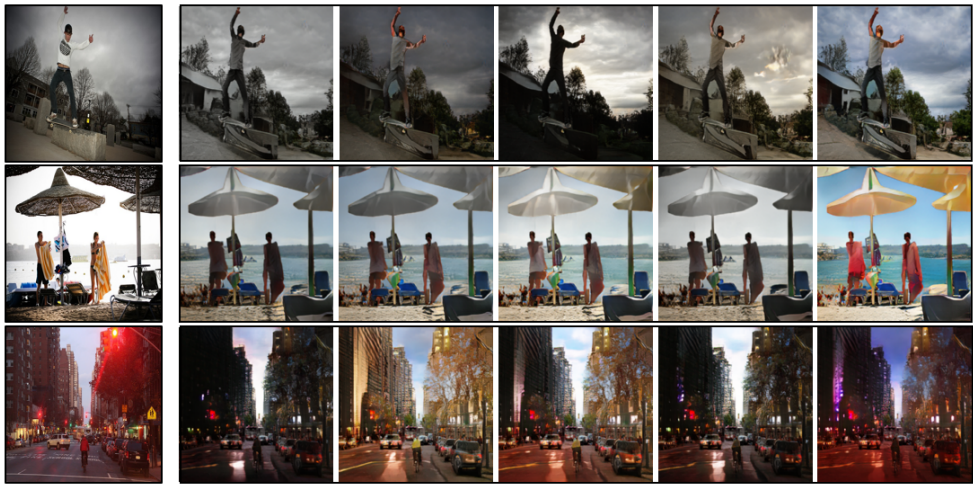


Figure 17: Example generated images from the COCO dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

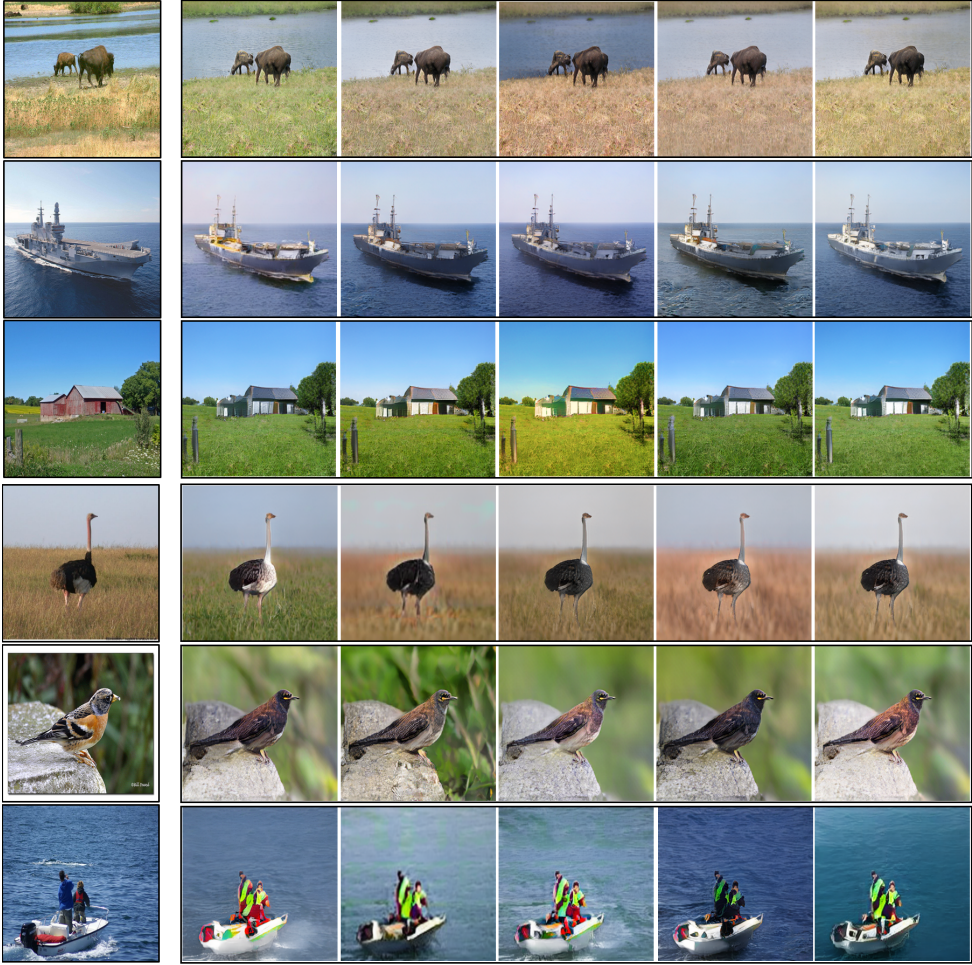


Figure 18: Example generated images from the ImageNet dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

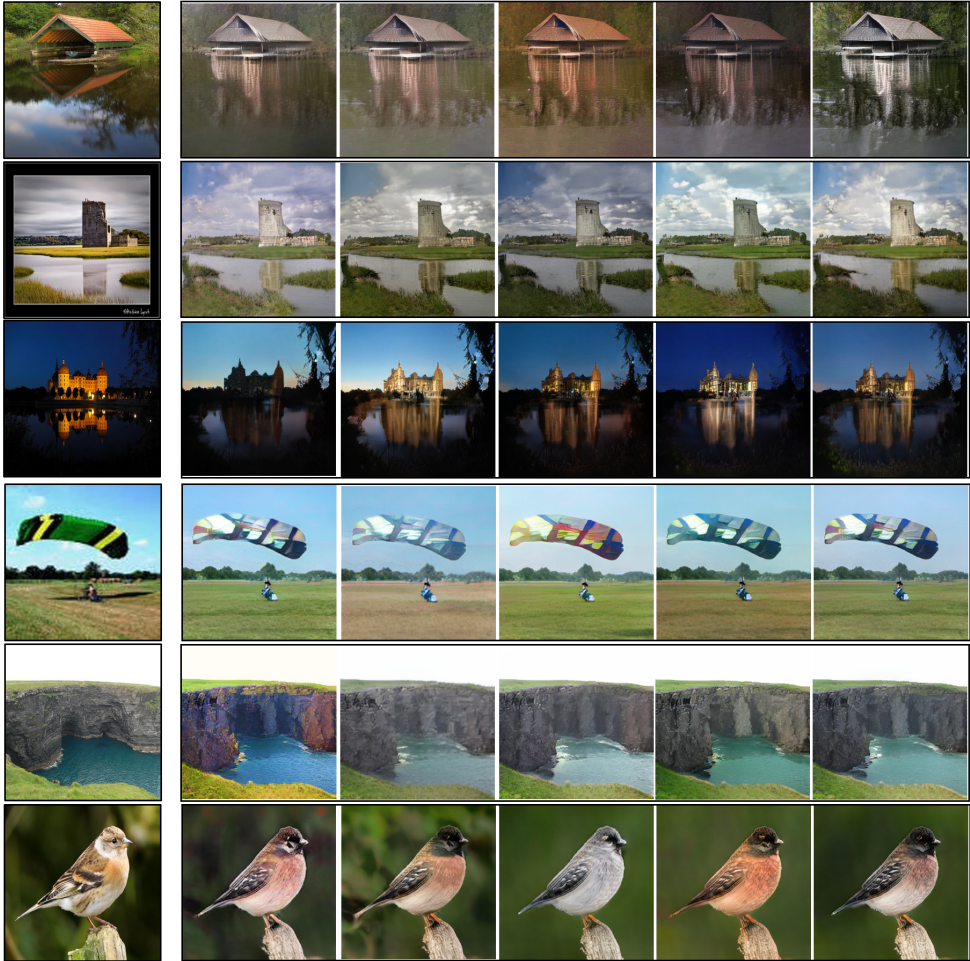


Figure 19: Example generated images from the ImageNet dataset: in each case, the left column is an original image, and the other columns show the repainted versions.

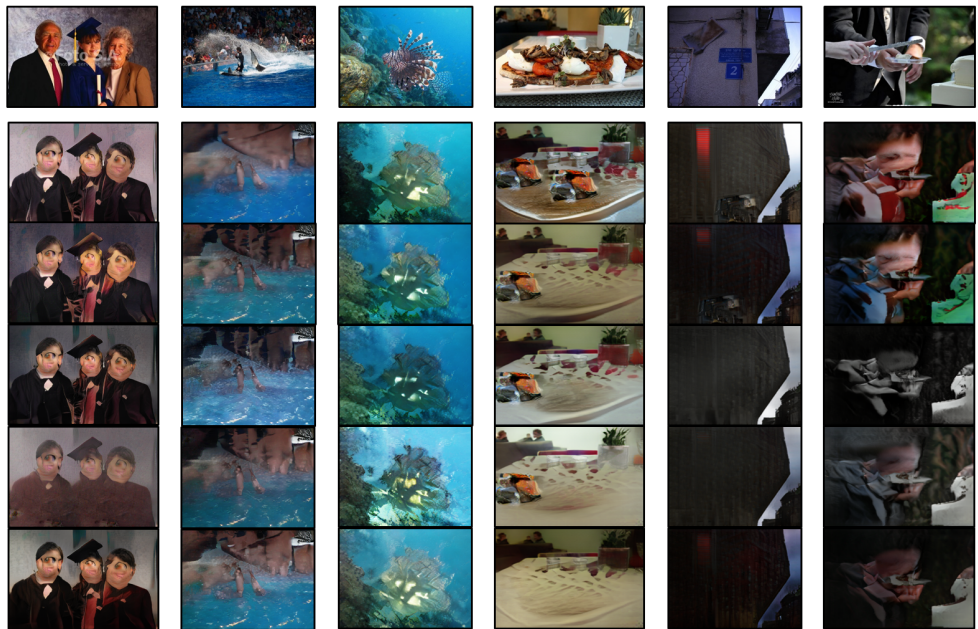


Figure 20: Examples of failure cases. In case of fine details such as facial features (in a natural diverse dataset like COCO or ImageNet) or in case of very rare objects, sometimes the algorithm fails to generate good-looking images. Nonetheless, the learning objective is to do well on the down-stream task, and not exactly only on image generation.