Perceptual Losses for Real-Time Style Transfer and Super-Resolution: Supplementary Material

Justin Johnson, Alexandre Alahi, Li Fei-Fei
{jcjohns, alahi, feifeili}@cs.stanford.edu

Department of Computer Science, Stanford University

1 Network Architectures

Our style transfer networks use the architecture shown in Table 1 and our super-resolution networks use the architecture shown in Table 2. In these tables "$C \times H \times W$ conv" denotes a convolutional layer with C filters size $H \times W$ which is immediately followed by spatial batch normalization [1] and a ReLU nonlinearity.

Our residual blocks each contain two 3×3 convolutional layers with the same number of filters on both layer. We use the residual block design of Gross and Wilber [2] (shown in Figure 1), which differs from that of He *et al* [3] in that the ReLU nonlinearity following the addition is removed; this modified design was found in [2] to perform slightly better for image classification.

Layer	**Activation size**
Input	$3 \times 256 \times 256$
$32 \times 9 \times 9$ conv, stride 1	$32 \times 256 \times 256$
$64 \times 3 \times 3$ conv, stride 2	$64 \times 128 \times 128$
$128 \times 3 \times 3$ conv, stride 2	$128 \times 64 \times 64$
Residual block, 128 filters	$128 \times 64 \times 64$
Residual block, 128 filters	$128 \times 64 \times 64$
Residual block, 128 filters	$128 \times 64 \times 64$
Residual block, 128 filters	$128 \times 64 \times 64$
Residual block, 128 filters	$128 \times 64 \times 64$
$64 \times 3 \times 3$ conv, stride 1/2	$64 \times 128 \times 128$
$32 \times 3 \times 3$ conv, stride 1/2	$32 \times 256 \times 256$
$3 \times 9 \times 9$ conv, stride 1	$3 \times 256 \times 256$

Table 1. Network architecture used for style transfer networks.

2 Residual vs non-Residual Connections

We performed preliminary experiments comparing residual networks for style transfer with non-residual networks. We trained a style transfer network using *The Great Wave Off Kanagawa* as a style image, replacing each residual block

×4		×8	
Layer	**Activation size**	**Layer**	**Activation size**
Input	$3 \times 72 \times 72$	Input	$3 \times 36 \times 36$
$64 \times 9 \times 9$ conv, stride 1	$64 \times 72 \times 72$	$64 \times 9 \times 9$ conv, stride 1	$64 \times 36 \times 36$
Residual block, 64 filters	$64 \times 72 \times 72$	Residual block, 64 filters	$64 \times 36 \times 36$
Residual block, 64 filters	$64 \times 72 \times 72$	Residual block, 64 filters	$64 \times 36 \times 36$
Residual block, 64 filters	$64 \times 72 \times 72$	Residual block, 64 filters	$64 \times 36 \times 36$
Residual block, 64 filters	$64 \times 72 \times 72$	Residual block, 64 filters	$64 \times 36 \times 36$
$64 \times 3 \times 3$ conv, stride 1/2	$64 \times 144 \times 144$	$64 \times 3 \times 3$ conv, stride 1/2	$64 \times 72 \times 72$
$64 \times 3 \times 3$ conv, stride 1/2	$64 \times 288 \times 288$	$64 \times 3 \times 3$ conv, stride 1/2	$64 \times 144 \times 144$
$3 \times 9 \times 9$ conv, stride 1	$3 \times 288 \times 288$	$64 \times 3 \times 3$ conv, stride 1/2	$64 \times 288 \times 288$
-	-	$3 \times 9 \times 9$ conv, stride 1	$3 \times 288 \times 288$

Table 2. Network architectures used for ×4 and ×8 super-resolution.

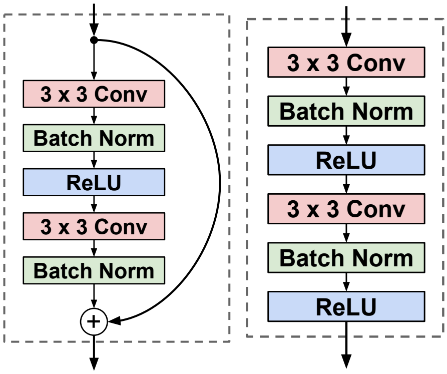
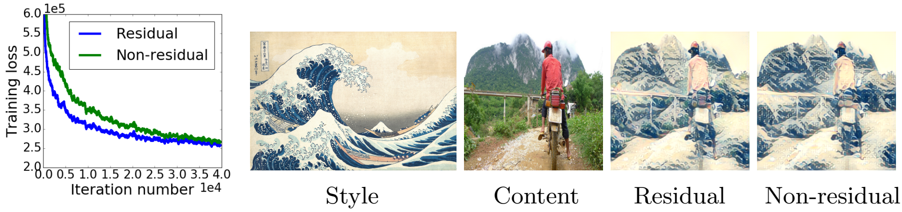


Fig. 1. Left: Residual block design used in our networks. Right: An equivalent convolutional block.



Style Content Residual Non-residual

Fig. 2. A comparison of residual vs non-residual networks for style transfer.

in Table 1 with an equivalent non-residual block consisting of a pair of 3×3 convolutional layers with the same number of filters as shown in Figure 1.

Figure 2 shows the training losses for a residual and non-residual network, both trained using Adam [4] for 40,000 iterations with a learning rate of 1×10^{-3}. We see that the residual network trains faster, but that both networks eventually achieve similar training losses. Figure 2 also shows a style transfer example from the trained residual and non-residual networks; both learn similar to apply similar transformations to input images.

Our style transfer networks are only 16 layers deep, which is relatively shallow compared to the networks in [3]. We hypothesize that residual connections may be more crucial for training deeper networks.

3 Super-Resolution Examples

We show additional examples of ×4 single-image super-resolution in Figure 4 and additional examples of ×8 single-image super-resolution in Figure 3.

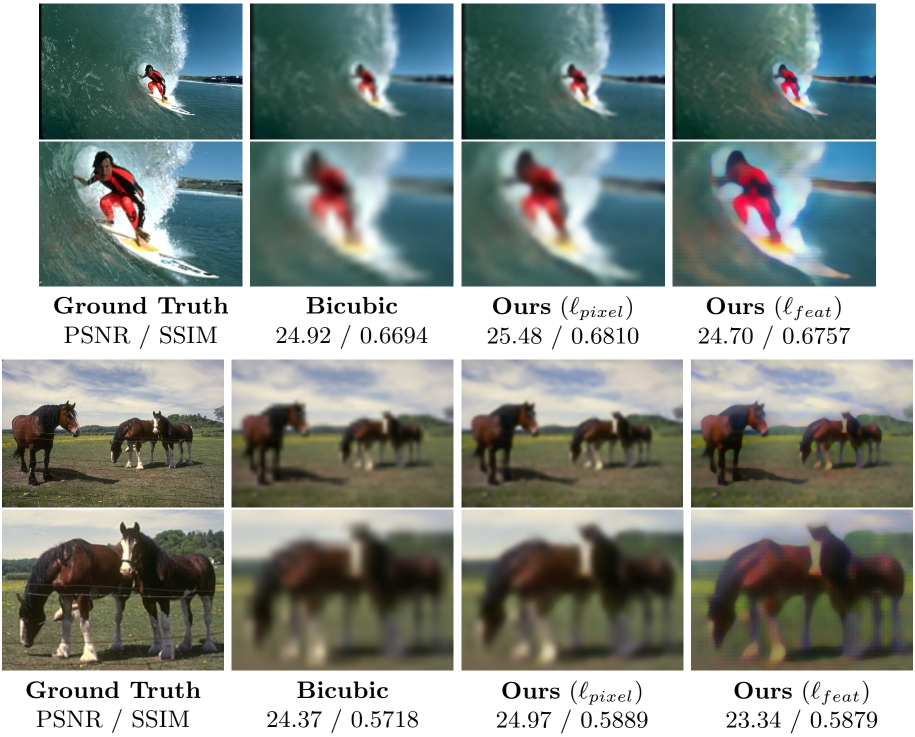


Fig. 3. Additional examples of ×8 single-image super-resolution on the BSD100 dataset.

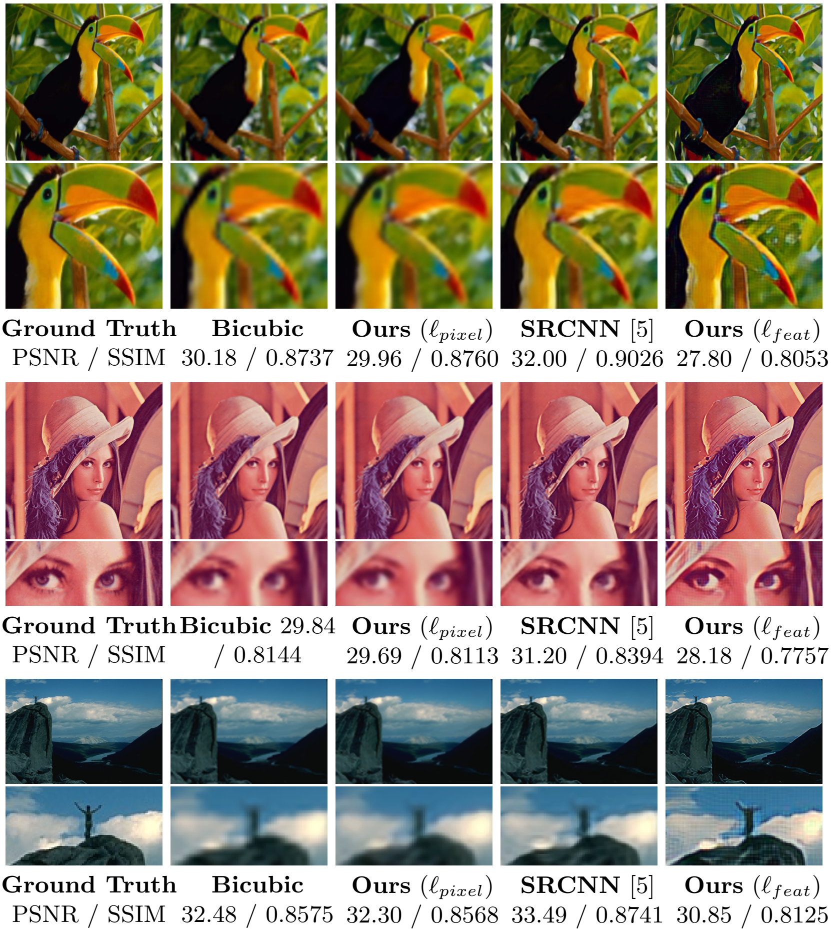


Fig. 4. Additional examples of ×4 single-image super-resolution on examples from the Set5 (top), Set14 (middle) and BSD100 (bottom) datasets.

References

1. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of The 32nd International Conference on Machine Learning. (2015) 448–456
2. Gross, S., Wilber, M.: Training and investigating residual nets. http://torch.ch/blog/2016/02/04/resnets.html (2016)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015)
4. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
5. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Computer Vision–ECCV 2014. Springer (2014) 184–199