Supplementary Material for MetaStyle: Three-Way Trade-Off Among Speed, Flexibility, and Quality in Neural Style Transfer

Chi Zhang and **Yixin Zhu** and **Song-Chun Zhu**
{chizhang,yzhu,sczhu}@cara.ai
International Center for AI and Robot Autonomy (CARA)

1 Details of the Network Architecture

We provide further details on the network architecture used in MetaStyle in Table 1. Note that all the convolution layers use the "same" padding before the operation, and all the upsamplings are of nearest sampling with a scale factor of 2.

2 Additional Details of the Training

During training, we use the time-based learning rate decay for both the outer and the inner objective optimization, *i.e.*,

$$\kappa = \frac{1}{1 + k \times t} \kappa_0, \tag{1}$$

where κ denotes the learning rate for either the outer or the inner objective, t the number of iterations, and $k = 2.5 \times 10^{-5}$. To reduce the computation, we do not iteratively sample a new content batch from the training set $\mathcal{D}_{tr}$ in the inner objective optimization, but share the same content batch $\mathcal{B}_{tr}$ in each iteration. Similarly, we use the same content batch $\mathcal{B}_{val}$ from the validation set $\mathcal{D}_{val}$ during each outer objective update. Note that this procedure accelerates the convergence. In contrast to Finn, Abbeel, and Levine (2017) and Nichol, Achiam, and Schulman (2018), we find that the first-order gradient approximations lead to serious fluctuations during training and no convergence is observed. In addition, increasing T to the values as large as 5 does not notably improve performance. Therefore, we set $T = 1$ in the reported experiment results. Such a setting significantly reduce GPU memory consumption. To further stabilize training, we only update parameters in instance normalization layers in inner objective optimization. This design implicitly encourages the instance normalization layers to find a set of parameters that specializes in a style-neutral representation, corresponding to the finding in Dumoulin, Shlens, and Kudlur (2017).

3 Additional Neural Style Transfer Results

We include more examples in Page 3-8.

References

Dumoulin, V.; Shlens, J.; and Kudlur, M. 2017. A learned representation for artistic style. *International Conference on Learning Representations (ICLR)*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of International Conference on Machine Learning (ICML)*.

Nichol, A.; Achiam, J.; and Schulman, J. 2018. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*.

Operator	Channel	Stride	Kernel	Padding	Activation
Network — Input	3				
Convolution	32	1	9	Reflection	
Instance Norm	32				ReLU
Convolution	64	2	3	Reflection	
Instance Norm	64				ReLU
Convolution	128	2	3	Reflection	
Instance Norm	128				ReLU
Residual Block	128				
Residual Block	128				
Residual Block	128				
Residual Block	128				
Residual Block	128				
Upsampling					
Convolution	64	1	3	Reflection	
Instance Norm	64				ReLU
Upsampling					
Convolution	32	1	3	Reflection	
Instance Norm	32				ReLU
Convolution	3	1	9	Reflection	Sigmoid
Residual Block — Input	128				
Convolution	128	1	3	Reflection	
Instance Norm	128				ReLU
Convolution	128	1	3	Reflection	
Instance Norm	128				
Addition	128				

Table 1: Network architecture used in MetaStyle.