Supplementary Material for: Learning to Have an Ear for Face Super-Resolution

1. Details of Network Architectures and Implementation

We provide details of the used network architectures in Tables 1 to 3. All the networks are convolutional using strided convolutions to reduce the spatial resolution. We apply instance normalization [4] to both the high-resolution encoder E_h and the low-resolution encoder E_l. Notice that we also process the audio spectrogram using a CNN architecture. We found however that applying instance normalization to the audio-encoder E_a leads to significantly worse performance. Consequently, no normalization was applied for E_a. We use the leaky ReLU activation function in all our networks with a leak of 0.2.

To train the high resolution encoder E_h we used a perceptual loss on features of an ImageNet pre-trained VGG16 network. We extracted features from the outputs of the layers `conv1_1`, `conv1_2`, `conv3_2` and `conv4_2`.

The fusion network F consists of three fully-connected layers each with a hidden dimension of 6'144. We again applied leaky ReLU activations in the hidden layers and did not use any normalization.

All networks were trained with multi-GPU training on 4 NVIDIA GTX 1080Ti GPUs.

Low-resolution encoder E_l

Layer	Kernel	Stride	Norm.	Activation	# Filters
conv	3×3	1	-	lReLU	128
conv	3×3	2	IN	lReLU	128
conv	3×3	1	IN	lReLU	256
conv	3×3	2	IN	lReLU	256
conv	3×3	1	IN	lReLU	512
conv	3×3	2	IN	lReLU	512
dense	-	-	-	lReLU	6144
dense	-	-	-	linear	6144

Table 1: The network architecture of the low-resolution encoder E_l. Images are assumed to be of size 8×8. The output size of 6144 matches the targets z_i.

High-resolution encoder E_h

Layer	Kernel	Stride	Norm.	Activation	# Filters
conv	4×4	1	-	lReLU	64
conv	4×4	2	IN	lReLU	64
conv	4×4	1	IN	lReLU	128
conv	4×4	2	IN	lReLU	128
conv	4×4	1	IN	lReLU	256
conv	4×4	2	IN	lReLU	256
conv	4×4	1	IN	lReLU	512
conv	4×4	2	IN	lReLU	512
conv	4×4	1	IN	lReLU	1024
conv	4×4	2	IN	lReLU	1024
conv	4×4	1	IN	lReLU	1024
conv	4×4	2	IN	lReLU	1024
dense	-	-	-	linear	6144

Table 2: The network architecture of the high-resolution encoder E_h. Input images are of size 128×128. The output size of 6144 matches the input input of the generator which is of size 12×512.

Audio encoder E_a

Layer	Kernel	Stride	Norm.	Activation	# Filters
conv	4×4	2	-	lReLU	64
conv	4×4	1	-	lReLU	64
conv	4×4	2	-	lReLU	64
conv	4×4	1	-	lReLU	128
conv	4×4	2	-	lReLU	128
conv	4×4	1	-	lReLU	256
conv	4×4	2	-	lReLU	256
conv	4×4	1	-	lReLU	512
conv	4×4	2	-	lReLU	512
conv	4×4	1	-	lReLU	1024
conv	4×4	2	-	lReLU	1024
conv	4×4	1	-	lReLU	2048
conv	4×4	2	-	lReLU	2048
dense	-	-	-	lReLU	8192
dense	-	-	-	linear	6144

Table 3: The network architecture of the audio encoder E_a. The input spectrograms are of size 257×257. The output size of 6144 matches the targets z_i.

2. Quantitative Results

Method	**PSNR**	**SSIM**	**Acc** C_i	**Acc** C_g	**Err** C_a
SRGAN ([2])	26.21	0.85	52.95%	97.01%	1.94
SRFeat ([3])	27.02	0.83	93.40%	99.27%	2.63
LapSRN ([1])	31.99	0.91	93.83%	99.38%	2.81

Table 4: Results of general-purpose super-resolution methods at a super-resolution factor of $4\times$. We report PSNR and SSIM along with identity, gender and age performance obtained on the closed test set. Note that the target resolution is fixed at 128×128 pixels and therefore the inputs to the $4\times$ methods is of size 32×32.

References

[1] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[2] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[3] Seong-Jin Park, Hyeongseok Son, Sunghyun Cho, Ki-Sang Hong, and Seungyong Lee. Srfeat: Single image super-resolution with feature discrimination. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[4] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.