ADVERSARIAL AUDIO SUPER-RESOLUTION WITH UNSUPERVISED FEATURE LOSSES

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural network-based methods have recently demonstrated state-of-the-art results on image synthesis and super-resolution tasks, in particular by using variants of generative adversarial networks (GANs) with supervised feature losses. Nevertheless, previous feature loss formulations rely on the availability of large auxiliary classifier networks, and labeled datasets that enable such classifiers to be trained. Furthermore, there has been comparatively little work to explore the applicability of GAN-based methods to domains other than images and video. In this work we explore a GAN-based method for audio processing, and develop a convolutional neural network architecture to perform audio super-resolution. In addition to several new architectural building blocks for audio processing, a key component of our approach is the use of an autoencoder-based loss that enables training in the GAN framework, with feature losses derived from unlabeled data. We explore the impact of our architectural choices, and demonstrate significant improvements over previous works in terms of both objective and perceptual quality.

1 INTRODUCTION

Deep convolutional neural networks (CNNs) have become a cornerstone in modern solutions for image and audio analysis. Such networks have excelled at supervised descrimination tasks, for instance on ImageNet (Deng et al., 2009; Simonyan & Zisserman, 2014), where image classifier networks are trained on a large corpus of labeled data. More recently, CNNs have successfully been applied to data synthesis problems in the context of generative adversarial networks (GANs) (Goodfellow et al., 2014). In the GAN framework, a neural network is used to synthesize new instances from a modeled distribution, or resolve missing details given lossy observations. In the latter case, the GANs have been shown to greatly improve reconstruction of fine texture details for images, compared to standalone pixel-space losses that result in overly smoothed outputs (Dosovitskiy & Brox, 2016; Isola et al., 2017; Ledig et al., 2017). However, GANs are notoriously hard to train, and the use of conventional pixel-space objectives in conjunction with an adversarial loss either de-stabilizes training, or results in outputs with significant artifacts.

To address the smoothness problem described above, previous works typically augment or replace conventional pixel-space losses with a *feature loss* (also called a perceptual loss) (Dosovitskiy & Brox, 2016; Ledig et al., 2017; Johnson et al., 2016). Instead of distance in raw pixel-space, such feature losses reflect distance in terms of the feature maps of an auxiliary neural network. While classifier-based feature losses are effective, they require either a pre-trained neural network that is applicable to the problem domain (e.g., synthesizing images of cats), or a labeled dataset that is amenable to training a relevant classifier.

Training new classifiers for use in a feature loss can be non-trivial for several reasons. Besides the difficulty of training large classifiers that are commonly used for feature losses, such as VGG (Simonyan & Zisserman, 2014), creating a labeled dataset that is sufficiently large and diverse is often infeasible. Finally, it is unclear what level of label granularity and specificity is needed for a useful feature loss. For instance, for training a feature loss over the distribution of plants, we are not aware of any evidence to suggest that the plant species would necessarily be more useful than a less specific label, such as the genus.

In this work[1], we sidestep the difficulty of training auxiliary classifiers by developing a feature loss that is unsupervised. In particular, we focus on an audio modeling task called super-resolution, where the goal is to generate high-quality audio given down-sampled, low-resolution input. Inspired by previous work on audio and image super-resolution, we develop a neural network architecture for end-to-end super-resolution that operates on raw audio. We show that when trained in the GAN framework, the use of the unsupervised feature loss stabilizes training, and results in significant improvements in terms of both perceptual and objective metrics. In addition to providing new algorithms to model audio, our work suggests new techniques to improve GAN-based methods in other domains such as images and video.

2 Background & Related Work

Audio super-resolution Audio super-resolution is the task of constructing a high-resolution audio signal from a low-resolution signal that contains a fraction of the original samples. Concretely, given a low-resolution sequence of audio samples $x_l = (x_{1/R_l}, \ldots, x_{R_l T/R_l})$, we wish to synthesize a high-resolution audio signal $x_h = (x_{1/R_h}, \ldots, x_{R_h T/R_h})$, where R_l and R_h are the sampling rates of the low and high-resolution signals, respectively. We denote $R = R_h/R_l$ as the *upsampling ratio*, which ranges from 2 to 6 in this work. Note that since the bandwidth of a signal sampled at rate r is limited to the Nyquist frequency $r/2$, audio super-resolution is equivalent to reconstructing the missing frequency content between frequencies $R_l/2$ and $R_h/2$.

There is a vast body of prior work on audio super-resolution in the signal and audio processing communities under the term *artificial bandwidth extension* (Larsen & Aarts, 2004). Neural network-based methods in this domain generally apply a DNN on top of hand-crafted features as part of larger, complex bandwidth extension systems (Liu et al., 2015; Abel & Fingscheidt, 2018). Gaussian mixture and hidden Markov models have also been used (Bachhav et al., 2017; Tokuda et al., 2013), but these methods generally perform worse compared to neural network-based methods (Abel & Fingscheidt, 2018). In contrast with the works above, our method does not rely on hand-crafted features (e.g., transformations or Mel-frequency cepstrum coefficients), and is not specific to problems in speech modeling.

Audio modeling with neural networks Learning-based approaches for audio have also been explored in the largely in the context of representation learning, generative modeling, and text-to-speech (TTS) systems. Unsupervised methods such as convolutional deep belief networks (Lee et al., 2009) and bottleneck CNNs (Aytar et al., 2016) have been shown to learn useful representations from audio, such as phonemes and sound textures. Stacked autoencoders (Vincent et al., 2010) and variational autoencoders (Kingma & Welling, 2014; Sonderby et al., 2016) have been used for denoising, image generation, and music synthesis (Sarroff & Casey, 2014). Bottleneck-like CNNs have also demonstrated significant improvements for audio super-resolution in supervised settings compared to previous DNN and spline-based methods (Kuleshov et al., 2017). Donahue et al. (2018) is among the first to develop methods for raw audio synthesis with GANs. Notably, Donahue et al. (2018) highlight important differences between audio and image generation, and show that non-trivial modifications of GAN architectures are required to generative diverse and plausible audio outputs. We build on the works above by developing a GAN framework for audio super-resolution with an improved bottleneck-style generator, and show that leveraging representations learned from unsupervised training greatly aid the super-resolution task. Although computationally intensive, autoregressive probabilistic models have recently demonstrated state-of-the-art results for generation of music (Engel et al., 2017), general audio (van den Oord et al., 2016; Mehri et al., 2017), and for parametric TTS systems (Sotelo et al., 2017). We are not aware of any efforts to explore autoregressive modeling for super-resolution, but we believe it may be a promising future direction.

Generative adversarial networks for images Generative methods have been extensively explored for image generation and super-resolution. Building upon the original formulation of Goodfellow et al. (2014), GANs have been continuously improved to generate plausible, high-fidelity images (Radford et al., 2015; Denton et al., 2015; Berthelot et al., 2017; Karras et al., 2018). GAN variants conditioned on class labels or object sketches have also demonstrated promising results on tasks such as in-painting and style transfer (Mirza & Osindero, 2014; Isola et al., 2017).

[1] Audio samples are available at https://sites.google.com/view/unsupervised-audiosr/home

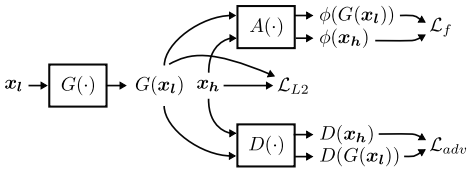

Figure 1: Overview of the model architecture and corresponding loss terms.

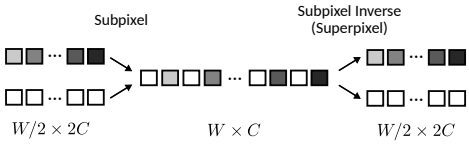

Figure 2: Subpixel and superpixel layers for increasing and decreasing spatial resolution, respectively.

3 METHOD

GANs for Super-Resolution GANs developed for super-resolution tasks have several important differences compared to the original formulation of Goodfellow et al. (2014). When used to generate new instances from a modeled data distribution p_{data}, the generator (G) parameterized by θ_G learns the mapping to data space as $G(\boldsymbol{z}; \theta_G)$, where $\boldsymbol{z}$ is a low-dimensional latent vector sampled from a noise prior. The discriminator (D) parameterized by θ_D then estimates the probability that $G(\boldsymbol{z}; \theta_G)$ was drawn from p_{data} rather than the generator distribution p_g. In contrast, for deterministic super-resolution, G is no longer conditioned on noise and learns the mapping to high-resolution data space p_h as $G(\boldsymbol{x_l}; \theta_G)$, where $\boldsymbol{x_l}$ is drawn from the low-resolution data distribution p_l. The task of D is to discriminate between samples drawn from the high-resolution and super-resolution (generator) distributions p_h and p_g, respectively. Since low-resolution data $\boldsymbol{x_l}$ corresponds directly to downsampled, high-resolution data $\boldsymbol{x_h}$ during training, we expect $G(\boldsymbol{x_l}; \theta_G) \approx \boldsymbol{x_h}$. This is in contrast to general GANs where $G(\boldsymbol{z}; \theta_G)$ is ideally dissimilar to instances from the training set. G and D are optimized according to the two-player minimax problem:

$$\min_{\theta_G} \max_{\theta_D} \mathbb{E}_{\boldsymbol{x_h} \sim p_h(\boldsymbol{x_h})} \left[\log D\left(\boldsymbol{x_h}; \theta_D\right)\right] + \mathbb{E}_{\boldsymbol{x_l} \sim p_l(\boldsymbol{x_l})} \left[\log(1 - D\left(G\left(\boldsymbol{x_l}; \theta_G\right)\right))\right] \tag{1}$$

This framework enables the joint optimization of two neural networks - G generates super-resolution data with the goal of fooling D, and D is trained to distinguish between real and super-resolved data. Thus, the GAN approach encourages G to learn solutions that are hard to distinguish from real, high-resolution datum.

Architecture overview MU-GAN (Multiscale U-net GAN) is composed of three models - a Generator (G), Discriminator (D), and stacked convolutional autoencoder (A) (Figure 1). The generator's task is to learn the mapping between the low and high-resolution data spaces, corresponding to signals $\boldsymbol{x_l}$ and $\boldsymbol{x_h}$, respectively. The discriminator's task is then to classify whether presented data instances are real, or produced by the generator. In addition to G and D, the stacked autoencoder extracts perceptually-relevant features from both real and super-resolved data for use in feature-space loss functions. The use of A is crucial in the GAN framework, as generators trained solely on L2 or other pixel-space losses suffer from training instability or output artifacts (Ledig et al., 2017).

Multiscale convolutional layers In comparison to images, audio signals are inherently periodic with time-scales on the order of 10's to 100's of samples. As a consequence, filters with very large receptive fields are required to create high quality, raw audio (Donahue et al., 2018; van den Oord et al., 2016). At the same time, previous work with classifier models suggests that varying the filter size within a network helps capture information at multiple scales (Szegedy et al., 2015). Leveraging these observations, we develop a multiscale convolutional building block composed of concatenated 3x1, 9x1, 27x1, and 81x1 filters. In practice, and with a fixed number of parameters for a given layer, we found that filters larger than 81x1 provided no additional benefit, while omitting large filter sizes resulted in significantly degraded audio quality. We interpret the poor performance of small filters as being a byproduct of their frequency selectivity; it is well known from signal processing theory that the resolution of an FIR filter's frequency response is proportional to the length of the filter.

Superpixel layers Methods for manipulating spatial resolution are a key component in image and audio synthesis models. Recently, it has been shown that pooling and strided convolutions tend to induce periodic "checkerboard" artifacts (Odena et al., 2016; Donahue et al., 2018). These artifacts

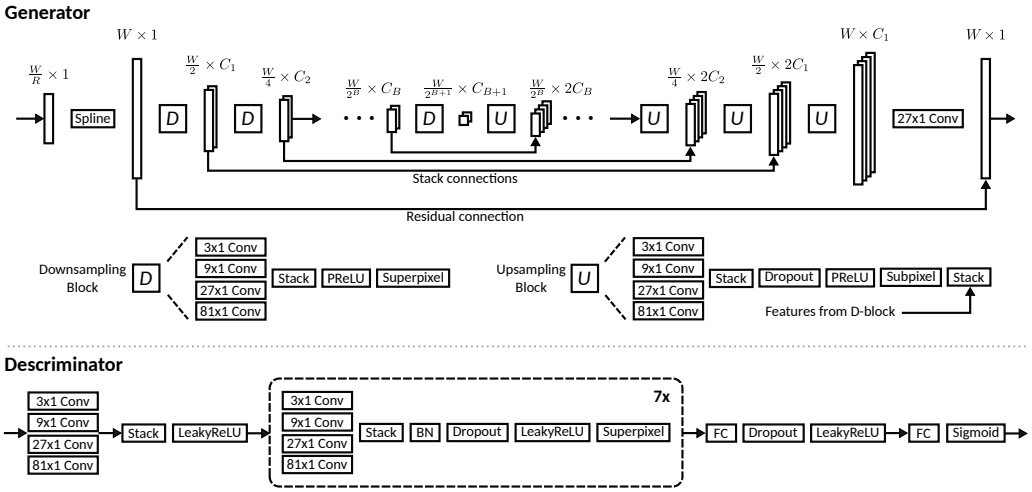

Figure 3: Generator and discriminator models.

can also result from analogous techniques that increase spatial resolution, such as transposed convolution. To increase spatial resolution, we use the subpixel layer developed in Shi et al. (2016) and Kuleshov et al. (2017), as it has been shown to be less prone to such checkerboard artifacts (Odena et al., 2016). As an alternative to conventional strided convolution or pooling, we simply use the inverse of the subpixel layer to reduce spatial resolution (Figure 2). We refer to this inverse operation as a *superpixel* layer.

Generator network The high-level architecture for the generator network (Figure 3, top) is inspired by autoencoder-like U-net models (Ronneberger et al., 2015; Isola et al., 2017; Kuleshov et al., 2017). In a U-net-style model, the first half of the network consists of B downsampling blocks (D-blocks) that perform feature extraction at multiple scales and resolutions. The second half the model consists of B upsampling blocks (U-blocks), which successively increase the spatial resolution of the signal. A crucial feature of the U-net is that each U-block receives not only features extracted from along the main trunk, but also raw features directly from the D-block of matching spatial resolution. This increases feature reuse, and prevents information loss that would otherwise occur if the signal was imperfectly compressible along the main trunk. Note that to have matching resolutions at the input and output of the generator, we first upsample the low-resolution signal to the target resolution with a cubic spline. While having matching U-net input/output resolutions is not strictly necessary, it enables an additive residual connection that accelerates training and improves performance with deeper models (He et al., 2015; Kuleshov et al., 2017).

Descriminator network The Descriminator (Figure 3, bottom) is used during training to differentiate between real, high-resolution audio and super-resolved signals produced by the generator. Our design is loosely based on the recommendations of Radford et al. (2015), and the image discriminator from Ledig et al. (2017). All discriminator activations are LeakyReLU (Maas et al., 2013) with $\alpha = 0.2$. As with the generator, we use the superpixel layer described above instead of strided convolutions, which reduces artifacts in the loss gradients (Odena et al., 2016). Hence, the feature map spatial resolution decreases by a factor of two with each convolutional layer. The output of the discriminator is composed of two fully-connected layers, with a standard sigmoid as the final activation.

Autoencoder network The autoencoder A is used to extract perceptually relevant features from the low and high-resolution signals. The features extracted by A are incorporated in the generator's feature loss $\mathcal{L}_f$, which is described in more detail in following sections. For the specific implementation of A, we use a modified version of the generator model that excludes all stacking and residual connections. Hence, the model for A is a stacked autoencoder, augmented with multiscale convolutional layers, and super/sub-pixel layers for down/up-sampling.

Loss functions MU-GAN incorporates several loss terms for training the generator and discriminator. The first term in the generator loss is the pixel-space L2 loss, given by[2]

$$\mathcal{L}_{pix} = \frac{1}{W} \sum_{i=1}^{W} \|\boldsymbol{x_h}_{,i} - G(\boldsymbol{x_l})_i\|_2^2 \,. \tag{2}$$

We found that using only the pixel-space and adversarial losses either resulted in little to no improvement over the baseline non-GAN model, or introduced audible artifacts (e.g., high-frequency tones). These findings are in line with those of Ledig et al. (2017), who experience similar issues with images. Ledig et al. (2017) posit that the poor performance of the pixel-space loss in the GAN setting is due to the competing nature of the adversarial and pixel-space losses.

As described in Section 2, the use of a feature loss with GAN training encourages the generator to learn solutions that incorporate perceptually relevant details, such as texture in images. Given the autoencoder A, we denote the output feature tensor at the bottleneck of the autoencoder as ϕ. The feature loss $\mathcal{L}_f$ is then given by

$$\mathcal{L}_f = \frac{1}{C_f W_f} \sum_{c=1}^{C_f} \sum_{i=1}^{W_f} \|\phi(\boldsymbol{x_h})_{i,c} - \phi(G(\boldsymbol{x_l}))_{i,c}\|_2^2 \,, \tag{3}$$

where W_f and C_f denote the width and channel dimensions for the feature maps of autoencoder bottleneck.

The adversarial loss $\mathcal{L}_{adv}$ is determined by discriminator's ability to discern whether data produced by the generator is real or fake. We use the gradient-friendly formulation originally posed in Goodfellow et al. (2014), given by

$$\mathcal{L}_{adv} = -\log D(G(\boldsymbol{x_l})). \tag{4}$$

The composite loss $\mathcal{L}_G$ for the generator is then given by the sum of the losses above, and the discriminator loss $\mathcal{L}_D$ derives directly from the GAN optimization objective in Equation 1, i.e.,

$$\mathcal{L}_G = \mathcal{L}_{pix} + \lambda_f \mathcal{L}_f + \lambda_{adv} \mathcal{L}_{adv}, \tag{5}$$

$$\mathcal{L}_D = -\left[\log D(\boldsymbol{x_h}) + \log(1 - D(G(\boldsymbol{x_l}))\right], \tag{6}$$

where λ_f and λ_{adv} are constant scaling factors.

4 Experiments

Datasets We evaluate our methods on three super-resolution tasks derived from the VCTK Corpus (Yamagishi), and the non-vocal music dataset from Mehri et al. (2017). VCTK consists of 44 hours of audio from 109 different speakers of varying ages and accents. The *Piano* dataset consists of 10 hours of non-vocal music from Beethoven's 32 piano sonatas. For speech from VCTK, we compose a dataset using recordings from a single speaker (the *Speaker1* task), and a dataset using recordings from multiple speakers (the *Speaker100* task). The dataset for *Speaker1* consists of the first 223 recordings from VCTK speaker ID#225 for training, and the final 8 recordings for testing. *Speaker100* uses all of the recordings from the first 100 VCTK speakers for training, and recordings from the last 9 speakers for testing. Finally, for *Piano*, we use the standard 88%-6%-6% train/validation/test split. For all tasks, the dataset is created by first applying an anti-aliasing lowpass filter, and then sampling random patches of fixed length from the resulting audio. We sample patches of length 16384, with a fixed stride of 4096 samples. Note that for the sake of direct comparison, the datasets above are the same as those used in Kuleshov et al. (2017).

[2]We write losses with respect to a single sample, with an implicit mean over the minibatch dimension.

Training methodology For *Speaker1*, we instantiate variants of MU-GAN and train for 400 epochs. For the larger datasets *Speaker99* and *Piano*, models are trained for 150 epochs. The epoch number is empirically selected based on observed convergence, and performance saturation on the validation set. For all models, we use the ADAM optimizer with learning rate 1e-4, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a batch size of 32. For the autoencoder feature losses, we instantiate a model with $L = 4$, and train for 400 epochs on the same dataset as its associated GAN model.

Performance metrics We use three metrics to assess the quality of super-resolved audio: (1) signal-to-noise ratio (SNR), (2) log-spectral distance (LSD), and (3) mean opinion score (MOS). The SNR is a standard metric in signal processing communities, defined as

$$\text{SNR}\left(x, x_{ref}\right) = 10 \log_{10} \frac{\|x_{ref}\|_2^2}{\|x - x_{ref}\|_2^2}, \tag{7}$$

where x is an approximation of reference signal x_{ref}. However, it has been shown that measures of sample-wise differences, such as SNR, have limited correlation to perceptual quality (Wang et al., 2004; Emiya et al., 2011). LSD (Gray & Markel, 1976) measures differences between signal frequencies, and has better correlation with perceptual quality compared to SNR (Jie et al., 2014; Kuleshov et al., 2017). Given short-time discrete Fourier transforms X and X_{ref}, the LSD is given by

$$\text{LSD}\left(X, X_{ref}\right) = \frac{1}{W} \sum_{w=1}^{W} \sqrt{\frac{1}{K} \sum_{k=1}^{N} \left(\log_{10} \frac{|X(w,k)|^2}{|X_{ref}(w,k)|^2} \right)^2}, \tag{8}$$

where w and k are the time window and frequency bin indices, respectively. We use non-overlapping Fourier transform windows of length 2048. Note that LSD is symmetric with respect to its arguments, while SNR is not. Perceptual evaluation of speech quality (PESQ) (ITU-T, 2001) is an industry-standard methodology for the assessment of audio quality in voice communication systems. Given reference and degraded audio signals, PESQ models the mean opinion score (MOS) of a group of listeners. Specifically, we use PESQ to produce MOS-LQO (listening quality objective) scores (ITU-T, 2003), which range from 1 (bad) to 5 (excellent).

Impact of superpixel layers We evaluate the impact of the superpixel layer proposed in Section 3 by comparing a baseline multiscale U-net with strided convolutions (*Strided*) against a multiscale U-net with superpixel layers (*Super*). We halve the number of convolutional kernels in each downsampling layer for *Super* such that the output feature map dimensions at each downsampling and upsampling layer are identical to those in *Strided*. We train both model types with the baseline L_2 loss for 400 epochs on the *Speaker1* task.

Table 1a shows the average wall time per minibatch with both $L = 4$ and $L = 8$ (L upsampling and L downsampling layers). Notably, the use of superpixel layers results in $\sim$14% improvement in training time, invariant to model size. Profiling layer execution times confirmed that the convolutional layers in *Super* are roughly twice as fast as those in *Strided* in both the forward and backward training passes. the reduction in convolutional kernels prior to the superpixel operation.

Table 1b shows that across varying upsampling ratios, the difference in audio quality between *Super* and *Strided* is marginal. Differences in audio produced by the two methods were also imperceptible in informal self-blinded listening tests.

Performance comparison Table 2 shows the quantitative performance of MU-GAN against other recent works. We denote *MU-GAN8* as an instance of MU-GAN with a depth parameter of $L = 8$, i.e., with 8 downsampling and 8 upsampling blocks. *U-net4* is the model with $L = 4$ from Kuleshov et al. (2017). To eliminate depth as a factor in the performance comparison, we implement a version of the model from Kuleshov et al. (2017) with $L = 8$, which we denote as *U-net8*.

Table 2 shows that *MU-GAN8* often performs worse in terms of SNR compared to other models. However, *MU-GAN8* has superior performance in terms of perceptual quality, as shown by the LSD

Table 1: Comparison of superpixel and strided convolutional layers.

(a) Training time per minibatch

	Depth Parameter	
	$L = 4$	$L = 8$
Strided	149.8 s	195.1 s
Super	128.1 s	168.0 s
Speedup	**14.5%**	**13.8%**

(b) Quality metrics

		Upsampling Ratio		
		$R = 2$	$R = 4$	$R = 6$
SNR	*Strided*	21.67	18.45	14.91
	Super	21.75	18.41	14.89
LSD	*Strided*	1.67	2.20	2.73
	Super	1.70	2.08	2.39
MOS-LQO	*Strided*	3.57	3.13	2.71
	Super	3.53	3.15	2.78

Table 2: Quantitative comparison with previous works

		Up. Ratio $R = 2$			Up. Ratio $R = 4$			Up. Ratio $R = 6$		
		U-net4	*U-net8*	*MU-GAN8*	*U-net4*	*U-net8*	*MU-GAN8*	*U-net4*	*U-net8*	*MU-GAN8*
Speaker1	SNR	21.1	**21.9**	21.4	17.1	**18.7**	17.7	14.4	**14.9**	14.0
	LSD	3.2	2.2	**1.6**	3.6	2.3	**1.9**	3.4	2.9	**2.0**
	MOS-LQO	-	3.5	**3.5**	-	2.7	**2.9**	-	2.7	**2.9**
Speaker99	SNR	**20.7**	20.1	20.0	**16.1**	14.3	14.0	10.0	**11.1**	10.9
	LSD	3.1	2.2	**2.1**	3.5	2.9	**2.7**	3.7	3.2	**3.0**
	MOS-LQO	-	3.1	**3.1**	-	2.3	**2.3**	-	1.8	**1.9**
Piano	SNR	30.1	45.0	**52.0**	23.5	31.7	**32.3**	16.1	22.5	**24.7**
	LSD	3.4	1.1	**0.9**	3.6	1.4	**1.3**	4.4	1.5	**1.4**

and MOS-LQO metrics. This supports previous evidence (Ledig et al., 2017) that feature losses in the GAN framework are well-suited for tasks that involve content synthesis or reconstruction. The exception is with the *Piano* task, where *MU-GAN8* achieves over 20 dB over the U-net baseline. We hypothesize that despite being a larger dataset, the *Piano* dataset is intrinsically sparse in terms of musical features, compared variation in speech.

5 Conclusion

In this paper we develop methods to enable the application of GANs to audio processing, in particular with classifier-free feature losses. In addition to several new model building blocks, we show that a stacked autoencoder can be used to implement a high-performance feature loss in the context of audio super-resolution. Demonstrated on several speech and music super-resolution tasks, we show significant performance improvements. While our model is specialized for audio processing, we believe that methods such as the unsupervised feature loss will be useful for other problem domains.

References

Johannes Abel and Tim Fingscheidt. Artificial speech bandwidth extension using deep neural networks for wideband spectral envelope estimation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(1):71–83, 2018.

Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, 2016.

Pramod Bachhav, Massimiliano Todisco, Moctar Mossi, Christophe Beaugeant, and Nicholas Evans. Artificial bandwidth extension using the constant q transform. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5550–5554, 2017.

David Berthelot, Tom Schumm, and Luke Metz. BEGAN: boundary equilibrium generative adversarial networks. *CoRR*, abs/1703.10717, 2017. URL http://arxiv.org/abs/1703.10717.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Emily L Denton, Soumith Chintala, Arthur Szlam, and Rob Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in Neural Information Processing Systems*, pp. 1486–1494. 2015.

Chris Donahue, Julian McAuley, and Miller Puckette. Synthesizing audio with generative adversarial networks. *CoRR*, abs/1802.04208, 2018. URL http://arxiv.org/abs/1802.04208.

Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. In *Advances in Neural Information Processing Systems*, pp. 658–666. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6158-generating-images-with-perceptual-similarity-metrics-based-on-deep-networks.pdf.

Valentin Emiya, Emmanuel Vincent, Niklas Harlander, and Volker Hohmann. Subjective and objective quality assessment of audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2046–2057, 2011.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi. Neural audio synthesis of musical notes with wavenet autoencoders. *CoRR*, abs/1704.01279, 2017. URL http://arxiv.org/abs/1704.01279.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pp. 2672–2680. 2014.

Augustine Gray and John Markel. Distance measures for speech processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(5):380–391, 1976.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5967–5976, 2017.

ITU-T. Perceptual evaluation of speech quality (pesq), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs. *ITU-T Rec. P.862*, 2001.

ITU-T. Mapping function for transforming p.862 raw result scores to MOS-LQO. *ITU-T Rec. P.862.1*, 2003.

Zhang Jie, Xiaoqun Zhao, Jingyun Xu, and Zhang Yang. Suitability of speech quality evaluation measures in speech enhancement. In *2014 International Conference on Audio, Language and Image Processing*, pp. 22–26, 2014.

Justin Johnson, Alexandre Alahi, and Fei-Fei Li. Perceptual losses for real-time style transfer and super-resolution. *CoRR*, abs/1603.08155, 2016. URL http://arxiv.org/abs/1603.08155.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *International Conference on Representation Learning*, abs/1710.10196, 2018. URL http://arxiv.org/abs/1710.10196.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Conference on Representation Learning*, 2014.

Volodymyr Kuleshov, S. Zayd Enam, and Stefano Ermon. Audio super resolution using neural networks. *CoRR*, abs/1708.00853, 2017. URL http://arxiv.org/abs/1708.00853.

Erik R. Larsen and Ronald M. Aarts. *Audio Bandwidth Extension: Application of Psychoacoustics, Signal Processing and Loudspeaker Design*. John Wiley & Sons, Inc., 2004. ISBN 0470858648.

Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 105–114, 2017.

Honglak Lee, Peter Pham, Yan Largman, and Andrew Y. Ng. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *Advances in Neural Information Processing Systems*, pp. 1096–1104. 2009.

Bin Liu, Jianhua Tao, Zhengqi Wen, Ya Li, and Danish Bukhari. A novel method of artificial bandwidth extension using deep architecture. In *INTERSPEECH*, 2015.

Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Internation Conference on Machine Learning*, volume 30, pp. 3, 2013.

Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron C. Courville, and Yoshua Bengio. Samplernn: An unconditional end-to-end neural audio generation model. *International Conference on Representation Learning*, 2017.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014. URL http://arxiv.org/abs/1411.1784.

Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016. URL http://distill.pub/2016/deconv-checkerboard.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *International Conference on Representation Learning*, 2015.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

Andy M Sarroff and Michael A Casey. Musical audio synthesis using autoencoding neural nets. In *International Computer Music Conference*, 2014.

Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1874–1883, 2016.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. URL http://arxiv.org/abs/1409.1556.

Casper Kaae Sonderby, Tapani Raiko, Lars Maaloe, Soren Kaae Sonderby, and Ole Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems*, pp. 3738–3746. 2016.

Jose Sotelo, Soroush Mehri, Kundan Kumar, Joo F. Santos, Kyle Kastner, Aaron Courville, and Yoshua Bengio. Char2Wav: End-to-end speech synthesis. In *International Conference on Learning Representations (Workshop Track)*, 2017.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.

Keiichi Tokuda, Yoshihiko Nankaku, Tomoki Toda, Heiga Zen, Junichi Yamagishi, and Keiichiro Oura. Speech synthesis based on hidden markov models. *Proceedings of the IEEE*, 101(5): 1234–1252, 2013.

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499, 2016. URL http://arxiv.org/abs/1609.03499.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11:3371–3408, 2010.

Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. ISSN 1057-7149.

Junichi Yamagishi. CSTR VCTK corpus. http://homepages.inf.ed.ac.uk/jyamagis/page3/page58/page58.html.