Using a Neural Network Codec Approximation Loss to Improve Source Separation Performance in Limited Capacity Networks

Ishwarya Ananthabhotla
Responsive Environments
MIT Media Lab
Cambridge, USA
ishwarya@media.mit.edu

Sebastian Ewert
Spotify
Berlin, Germany
sewert@spotify.com

Joseph A. Paradiso
Responsive Environments
MIT Media Lab
Cambridge, USA
joep@media.mit.edu

***Abstract*—A growing need for on-device machine learning has led to an increased interest in light-weight neural networks that lower model complexity while retaining performance. While a variety of general-purpose techniques exist in this context, very few approaches exploit domain-specific properties to further improve upon the capacity-performance trade-off. In this paper, extending our prior work [1], we train a network to emulate the behaviour of an audio codec and use this network to construct a loss. By approximating the psychoacoustic model underlying the codec, our approach enables light-weight neural networks to focus on perceptually relevant properties without wasting their limited capacity on imperceptible signal components. We adapt our method to two audio source separation tasks, demonstrate an improvement in performance for small-scale networks via listening tests, characterize the behaviour of the loss network in detail, and quantify the relationship between performance gain and model capacity. Our work illustrates the potential for incorporating perceptual principles into objective functions for neural networks.**

***Index Terms*—Deep Neural Networks, Audio, Psyachoacoustics, Perception, Audio Coding, Source Separation**

I. Introduction

In the last few years, deep neural networks have led to a substantial increase in performance in speech and audio processing tasks. The capacity provided by these methods (measured in the number of free parameters) enables modeling data and underlying distributions with high accuracy. This capacity, however, also prohibits the applicability of neural networks to many resource-constrained device classes, including phones and Internet-of-Things (IoT) devices. Thus, maximizing performance under computing and memory constraints has recently gained considerable interest. In this context, model weight quantization [2], memory-efficient architectures [3], [4], parameter pruning and sharing strategies [5], and student-teacher training [6] were successfully employed to reduce resources while trying to maintain modeling power. None of these techniques, however, is specific to the audio domain, which leaves considerable potential for improvement. For example, systems producing audio, including those for noise removal in speech, speech synthesis or musical source separation, are currently often trained in a supervised fashion against ℓ_1, ℓ_2, or cross-entropy losses, such as in [7], [8]. Such simple losses, however, do not take human perception into account and thus force networks to waste their limited capacity on modeling aspects of the signal that cannot be perceived.

In this paper, we present a strategy for optimizing the performance of capacity-limited networks, employing a loss trained to remove perceptually irrelevant elements of the signal. More precisely, we train a neural network to emulate the operation of a low-bitrate audio codec (e.g. MP3). We therefore obtain a differentiable function-approximation that can be used to eliminate signal components that are perceptually less important before the signals are compared in a supervised form. This way, we can employ any psychoacoustic model available in audio codecs as a perceptual model for neural network training without expert knowledge. In contrast to adversarial losses (which have yet to be shown to be effective as perceptual models for capacity-constrained generators), our loss preserves the ability to train in a supervised fashion and thus training remains stable and straightforward. We explored the basic principles behind this idea in [1] and while we were able to demonstrate that there is merit to the idea, the results were limited to synthetic examples and so it became clear that the concept had to be developed further to be applicable in real world scenarios.

In the following, we extend this first idea to a stable strategy to optimize resource-constrained audio separation models. Our main contributions in this paper are as follows: (1) we extend the technique proposed in [1] and demonstrate that it is effective for two real-world audio separation tasks – speech denoising and vocal separation – and adapt the procedure as necessary; (2) we show an improvement in performance for both applications over a baseline ℓ_1 loss for resource-constrained models; (3) we conduct a series of listening tests to understand the contributions of different configurations of the trained codec-loss, resulting in a more efficient method for computing our loss (which accelerates training as compared to [1]); and (4) we further use listening tests to characterize the behaviour of our loss method in detail and show that the benefit obtained by employing our perceptual objective

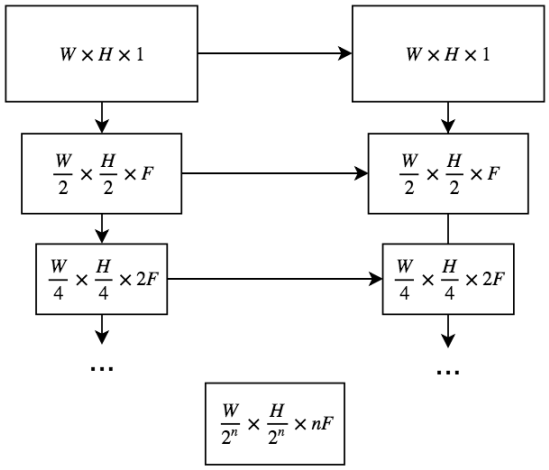

Fig. 1. An illustration of the u-network architecture used for our separation and loss networks.

function increases with smaller capacity models and converges to the performance of the baseline objective function for larger capacity models.

II. Related Work

Developing objective functions that incorporate principles of perception is not a well-explored area. Some attempts have been made to approximate metrics used in existing perceptual evaluation toolkits (e.g., STOI [9] and PESQ [10]), such as in [11]–[13]. These metrics, however, are either not differentiable functions, thus requiring numerical approximations for back propagation which is highly inefficient, or can be represented as differentiable functions with the consequence of being limited to rather simple models. Most recently, the authors in [14] suggested a perceptual weighting derived from psychoacoustic models applied to a mean-squared-error objective function and highlighted improvements in the performance of small scale neural networks. While this work provides a foundational step in exploring the intersection of psychoacoustic objective functions and limited capacity networks, we note that it does not incorporate subjective listening tests as a part of the evaluation, and employs a per-spectrum calculation of the global masking threshold from the PAM-1 model, which is a non-differentiable approximation.

III. Model Description

To describe our approach, let f_Θ denote a function representing a neural network with parameters Θ. Our aim is to train f_Θ to maximize performance for a specific audio separation task, while taking resource constraints for Θ into account. In this paper, we consider noise removal in speech and vocal separation as applications; since they are source separation tasks, we refer to f as the *separation network* in the following. In this context, f_Θ will operate on short snippets of magnitude spectrograms, with $X_M \in \mathbb{R}^{F \times N}$ denoting the input mixture and $X_S \in \mathbb{R}^{F \times N}$ the desired output for the target source. Given this notation, a baseline speech noise removal or vocal separation method can be trained in a supervised fashion using a standard ℓ_1 loss:

$$\Theta^* = \underset{\Theta}{\operatorname{argmin}} \underset{(X_M, X_S)}{\mathbb{E}} ||f_\Theta(X_M) \odot X_M - X_S||_1, \quad (1)$$

where $\odot$ denotes the Hadamard product and $f_\Theta(X_M) \in [0,1]^{F \times N}$ represents a mask to be applied to X_M (resembling Wiener filtering).

For our method, we follow [1] and replace the ℓ_1 term with a new expression that takes human perception into account. To this end, we define a second function g_Φ, which we train to approximate the operation of an audio codec. More precisely, let X denote a snippet of a magnitude spectrogram for an audio signal and let X_C be the corresponding representation for the signal after applying a codec, we train g_Φ to approximate the codec via:

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} \underset{(X, X_C)}{\mathbb{E}} ||g_\Phi(X) - X_C||_1.$$

This way, we can construct a new supervised loss $\tilde{L}$

$$\tilde{L}(X, Y) := ||g_{\Phi^*}(X) - g_{\Phi^*}(Y)||_1$$

and by replacing the ℓ_1 term in Eq. 1 with $\tilde{L}(f_\Theta(X_M) \odot X_M, X_S)$ we obtain a first version of a loss that removes signal components that are perceptually less relevant before computing the actual comparison. We refer to g_Φ as the *loss network* in the following.

While $\tilde{L}$ can work, we observed in practice slow convergence and sometimes even instabilities during training. Therefore, we incorporate ideas found useful in the image domain [15], [16], where trained classifiers were used as losses, which is conceptually related to our approach. More precisely, let $g^m_\Phi(X)$ denote the output of the m-th layer of the multilayer network g_Φ. In this context, $g^m_\Phi(X)$ corresponds to representations or features the network extracts intermittently to fulfill its task, i.e. the input signal is represented at various semantic levels. Thus, we can compare the two inputs not only at the final output layer but also at additional semantic levels. In [15], [16], this was shown to considerably stabilize the use of such a loss and we observed similar behaviour in our setting as well. Our proposed perceptual loss is thus defined as:

$$L_{\mathcal{M}}(X, Y) := \sum_{m \in \mathcal{M}} \lambda_m ||g^m_{\Phi^*}(X) - g^m_{\Phi^*}(Y)||_1, \quad (2)$$

where λ_m are weights to adjust the importance of individual layers and $\mathcal{M} \subset \{1, \ldots, M\}$, where M is the number of layers. In practice, we first train for 10 epochs with $\lambda_m = 1$, and then set $\lambda_m = \frac{1}{||g^m_{\Phi^*}(X) - g^m_{\Phi^*}(Y)||_1}$ for the remainder of the training to equally weight the contribution of the selected layers, following the suggestion in [15]. Since it is not clear which semantic levels are useful for our task, we conduct a series of listening tests in our experiments (see Section V), to investigate the importance of individual layers.

The architectures for our loss and separation networks closely follow the U-Net architecture described in [17], [18], as shown in Figure 1. Similar to Wavelets, the architecture is

TABLE I
A LIST OF THE MODEL ARCHITECTURE PARAMETERS AND HYPERPARAMETERS USED IN TRAINING THE LOSS AND SEPARATION NETWORKS FOR ALL EXPERIMENTS.

Parameter	Loss Network	Speech Denoising Network Loss Configuration Experiment	Speech Denoising Network Model Capacity Experiment	Vocal Separation Network Model Capacity Experiment
Number of Layers	6	2	{1,1,1,2,2}	{2,2,3,4,5}
W	128	128	128	128
H	512	512	512	512
F	28	1	{1,2,4,2,4}	{1,4,2,2,4}
Batch Normalization	All layers	All layers	All layers	All layers
Dropout	50% (first 3 upsampling layers)	50% (first 3 upsampling layers)	50% (first 3 upsampling layers)	50% (first 3 upsampling layers)
Kernel Size (Downsampling)	(5,5), Stride=2	(5,5), Stride=2	(5,5), Stride=2	(5,5), Stride=2
Kernel Size (Upsampling)	(5,5), Stride=2	(5,5), Stride=2	(5,5), Stride=2	(5,5), Stride=2
Activation	*ReLu*, *sigmoid* in final layer	*ReLu*, *sigmoid* in final layer	*ReLu*, *sigmoid* in final layer	*ReLu*, *sigmoid* in final layer
Learning Rate	0.0001	0.001	0.001	0.001
Decay	5e-6	5e-6	5e-6	5e-6
Batch Size	32	16	16	16
Optimizer	Adam	Adam	Adam	Adam

TABLE II
NUMBER OF TRAINABLE PARAMETERS ASSOCIATED WITH EACH LIMITED-CAPACITY CONFIGURATION.

Model Type	Num of Parameters (Speech)	Num of Parameters (Vocals)
P1	54	188
P2	107	1,949
P3	213	2,411
P4	575	9,683
P5	1,949	153,653

designed to represent the signal at multiple scales, via a series of down- and up-sampling blocks, which are implemented as convolutional or transposed convolutional layers with stride. As demonstrated in [18], the addition of skip connections between the layers enables the network to focus on higher-level semantics at higher layers, while still being able to access low-level information to reconstruct the signal as needed. This architecture was found useful for various tasks, including source separation [18] and lyrics alignment [19]. One may observe that using a U-Network architecture also for the loss network does not directly emulate the typical encoder-decoder structure that is characteristic of an audio codec, as the presence of skip connections circumvents the introduction of a true information bottleneck. In other words, we do not choose a network that would imitate an audio codec also on the architecture side. In particular, as the MP3 compressed audio data is already limited in information compared to the original audio, there is no need to introduce a separate information bottleneck in the network itself, which would limit the network's capacity to reproduce the audio codec faithfully. Instead, we use specific regularizers to provide a balance between approximation accuracy for the audio codec and smoothness of the function described by the loss network – we found this to be essential to be able to back-propagate through the loss network in a meaningful way. We use different configurations of this architecture in our experiments, which are given in Table I.

IV. MODEL CAPACITY EXPERIMENTS

We conducted a series of experiments to investigate the benefit of our proposed loss strategy for limited capacity networks in the context of the speech denoising and vocal separation tasks. We choose these tasks due to their relevance to on-device applications – examples include speech enhancement for phone calls and song identification based on lyric transcription. For the former task, we used the dataset first presented in [20], selecting the 56 speaker corpus. To increase the difficulty of the task, we select only those examples where the speech and noise are mixed at 0dB SNR, and sub-divide these examples into training, validation, and test sets of approximately 4000, 1200, and 600 samples respectively. For vocal separation, we employ the MUSDB18 dataset [21], which consists of pairs of mixes and corresponding stems for entire songs. We choose the mixture stem as the noisy input X_M, and attempt to predict the vocal stem as the clean output X_S. This dataset is sub-divided into train, validation, and test sets consisting of 100, 25, and 25 tracks respectively, with each track being several minutes in length. All of the speech segments/ music tracks are downsampled to 22050Hz, magnitude spectrograms are computed with a window size of 1024 and a hop size of 512 samples, and are broken into non-overlapping snippets of size 128. Some speech segments are simply tiled if they do not meet this minimum input width of the separation network.

We select and fix the loss network parameters as in Table I, and then choose five different sets of parameters determining model capacity for the separation networks performing each of the two tasks, denoted P1, P2, P3, P4, P5. In this context it should be noted that the size of the loss network does not contribute to the model capacity for the separation network associated with each experiment – the loss network is only used during training to improve the performance of the separation network at inference time. A set of parameters P is determined by the values for W and F in the U-Network (see Fig. 1), and are given in Table I; the corresponding total number of trainable parameters is shown in Table II. Note that the values for W and F for a given model type P may not be identical between the two tasks; state-of-the-art results in music separation tasks have been achieved with significantly larger networks than those needed for speech denoising tasks. We choose a range of model capacities whose extremes still demonstrate meaningful outputs, and discuss results from a

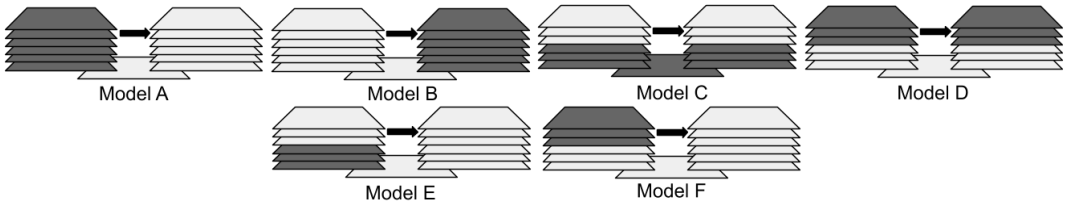

Fig. 2. An illustration of the six configurations of layers from the loss network tested in our characterization experiments; the dark shaded layers represent the regions of the loss network used to compute the custom loss in each configuration.

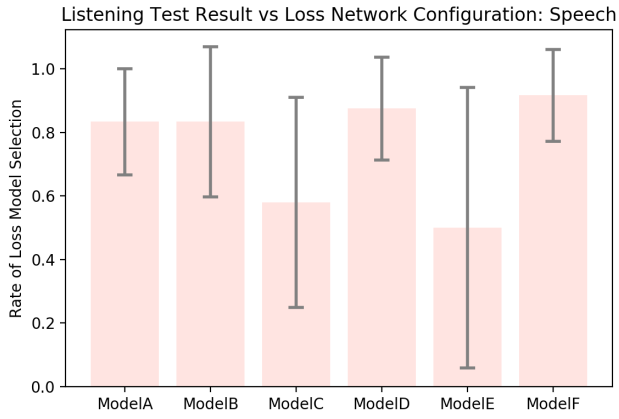

Fig. 3. Results from listening experiments plotting the rate that a sample associated with the proposed loss was preferred over the baseline, as a function of loss network configuration; Model F is selected as the best performing configuration.

few points sub-sampled in this range.

We begin by training the loss network in a fashion similar to our previous work, and utilize the dataset detailed in [1] consisting of lossless music tracks paired with their 16kbps MP3 coded counterparts; we pre-process the training examples to a sample rate of 22050Hz (as we intuit that perception will be influenced by higher frequency spectral detail in speech and music), a window size of 1024 and a hop size of 512, and use an ℓ_1 loss with early stopping to terminate training. Once this is complete and coding behavior is verified on the test set as in [1], this network is fixed without any further training. We then proceed to train the combined system of the separation network in each configuration P with the loss network, using Eq. 2 applied to only a subset of layers from the loss network (employing the best configuration from experiments in Section V). We additionally train the separation network in each configuration using an ℓ_1 loss to illustrate the respective performance improvement over the loss used in state-of-the-art systems for source separation, such as [18] and [22]. Our training is performed on a single GPU machine, using early stopping to terminate training; each experiment takes approximately 8-10 hours and 4-6 hours for the speech denoising and vocal separation tasks respectively.

We finally generate several examples from the test set for each configuration by applying the phase of the input mixture to the predicted output and inverting the resulting spectrogram. We evaluate the outcomes by conducting an online listening test, recruiting 20 participants in a crowd-sourced experiment for a small fee. We found that performing an actual listening test yielded more reliable results compared to approximative metrics such as PESQ or STOI. Each task in a study consisted of an A/B/X evaluation of a sample track or speech sample comparing our proposed loss metric to the baseline, corresponding to a particular model capacity type P. Each participant evaluated the same five speech samples/ tracks for each of the five configurations in a random order, for a total of 25 comparisons. For the speech task, participants were asked to select the sample that was more intelligible; for the vocal separation task, participants were asked to select the sample where the vocals were more distinct and stronger compared to the background track; in both cases, participants could select "I Don't Know" if they were unable to decide. We present a discussion of the results in Section VI.

V. Loss Configuration Experiments

In [1], we simply follow the literature on feature losses from the image domain and use all of the downsampling layers in the loss network to compute the objective function. However, since we might expect different layers of the network to extract features from the input at different levels of abstraction, we hypothesize that selecting certain subsets of layers $\mathcal{M}$ will lead to better performing objective functions. Using the speech denoising task, we evaluate this systematically by testing six different layer configurations defining $\mathcal{M}$ for the loss network illustrated in Figure 2, setting the parameters λ_m as discussed in Section III. We assign the separation network a fixed capacity with the parameters given in Table I, and train the combined separation network and loss network system in each configuration and with a baseline ℓ_1 loss for approximately 8-10 hours on a single GPU machine, using early stopping to terminate training.

Following the same reconstruction strategy as in Section IV, we run a listening experiment to compare the output of each loss configuration with the output of the baseline model. Participants were asked to choose the sample that contained the more intelligible speech, or choose "I Don't Know" if they could not decide. The study consisted of 15 crowd-sourced participants recruited for a small fee; each participant was

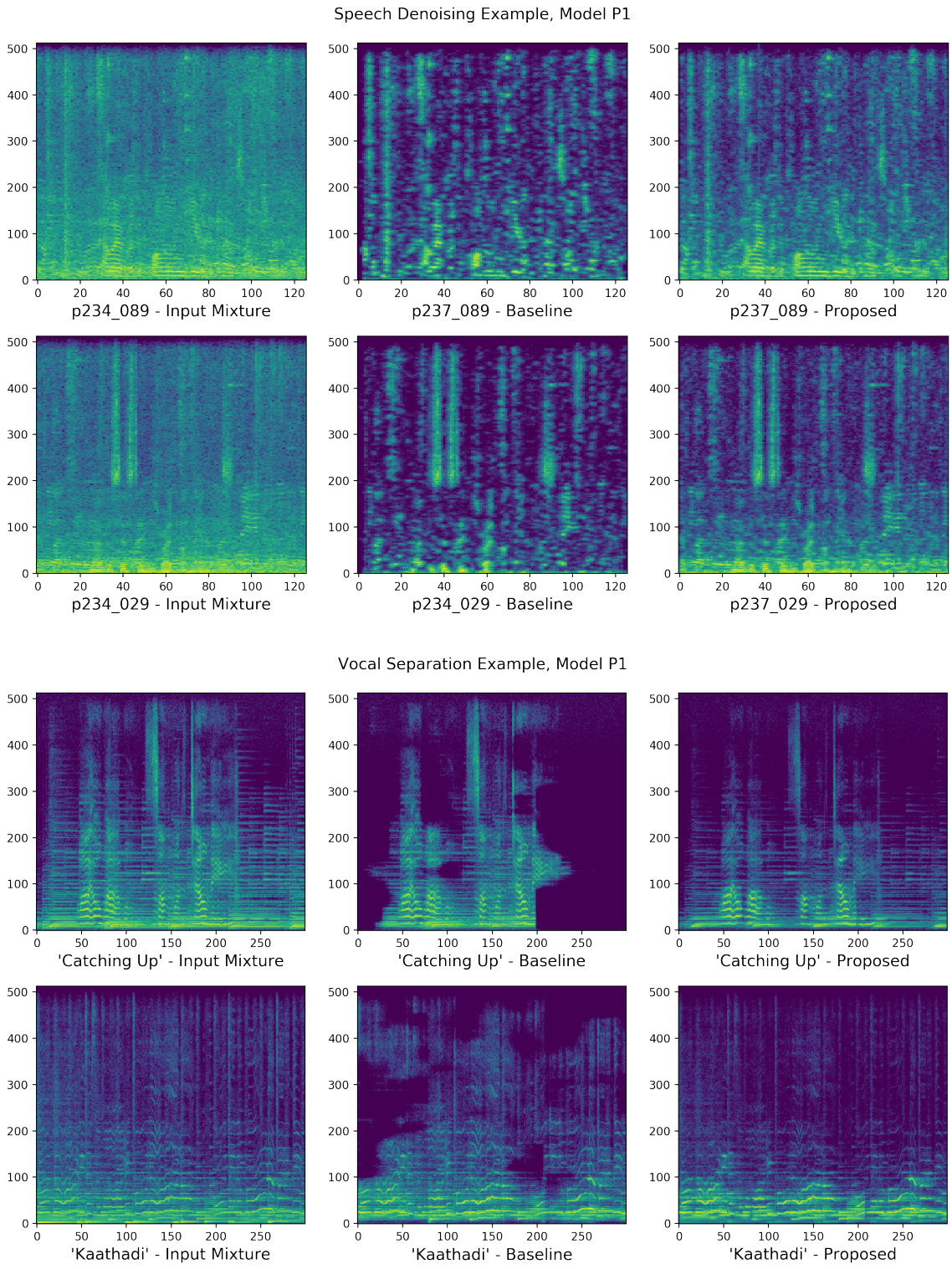

Fig. 4. Magnitude spectrogram examples comparing the output of P1 models trained with the baseline (center) and custom loss (right), referenced against the input (left), for the speech denoising (top) and vocal separation (bottom) tasks.

presented with the same five tracks from each of the six loss configurations in a random order, performing a total of 30 comparisons. A summary of the quantitative results can be seen in Figure 3 (we reserve the discussion of the qualitative findings for Section VI). The study participants were most likely to chose configurations “A”, “B”, “D”, “F”, suggesting

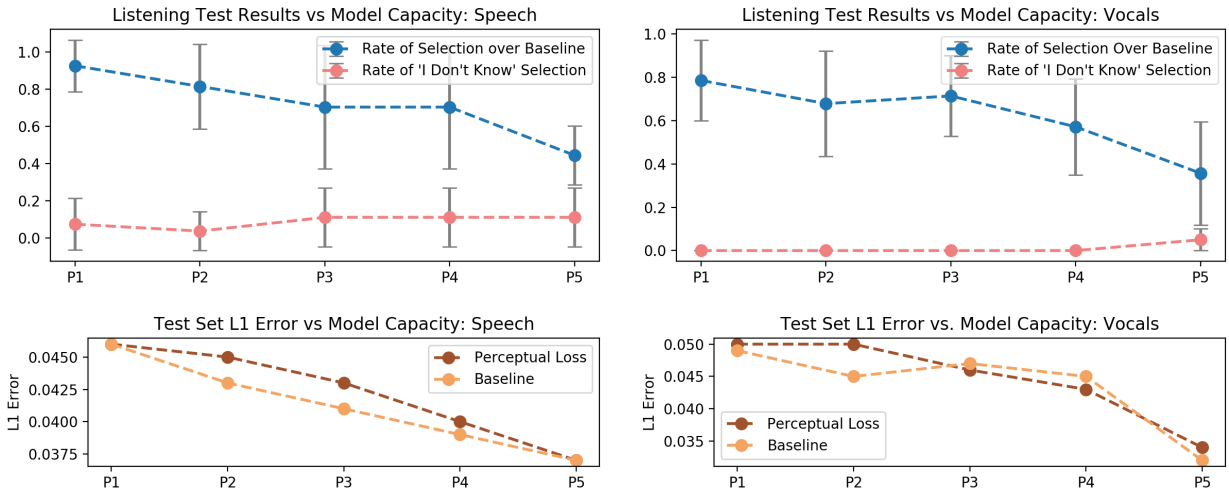

Fig. 5. Results from the listening tests comparing the proposed loss with the baseline for different model capacity configurations; (Top) For both the speech and vocal separation task, use of the proposed loss leads to better performance for lower capacity models; (Bottom) The ℓ_1 metric resulting from the loss case closely follows or is greater than that which results from the baseline case, suggesting that the loss network optimizes for a different set of constraints.

that lower level processing of the input spectrogram is a valuable contribution to the optimization problem. Furthermore, choosing model F from this set enables us to only use the activation outputs from the first three downsampling layers of the loss network, which reduces both the computational runtime for the loss and the on-GPU memory needed to train the separation network (the weights of these first few layers are saved independently). For the experiments detailed in Section IV, we employ loss configuration "F" to compute the objective function.

VI. Results and Discussion

A. Performance with Decreasing Network Capacity

Audio samples from both the model capacity and loss configuration experiments can be found at http://ishwaryaanant.github.io/small-network-perceptual-loss. In Figure 5, we summarize the quantitative results from our listening experiments by plotting the rate of selection of a sample associated with our proposed loss over the baseline loss strategy, as a function of model capacity for both the speech denoising and vocal separation tasks. We observe that the likelihood that a sample generated using a network trained with our proposed loss is preferred over a sample from the baseline procedure decreases as model capacity increases, for both tasks; the likelihood of a participant choosing option "X" ("I Don't Know") also increases with model capacity for both tasks. For example, we see that for Model P1, 80% or more of the participants were likely to choose the sample associated with our proposed loss for both tasks. Conversely, this number falls to 50% or less for Model P5. We also note the inter-rater variance by the error bars in both cases. While this variance is roughly constant for the vocal separation tasks, we see a significant drop in the variance for configurations P1 and P5 in the speech denoising tasks, indicating high confidence in rater agreement on preferring the proposed loss sample (P1) or the baseline sample (P5). Taken together, this behavior suggests that perceptual gains are afforded by our proposed objective function particularly in the case of the smallest source separation networks, while performance converges to the baseline with an increase in model capacity.

Additionally in Figure 5, we plot the final test set ℓ_1 error for both training procedures as a function of model capacity. We show that the ℓ_1 error for the baseline case tightly follows the perceptual loss case; this suggests that our loss strategy is not simply a form of a regularizer that leads to better ℓ_1 optimization, but an error metric that optimizes for a different set of aims.

B. Redistributing Noise

In Figure 4, we show spectrograms for sound samples from the test set corresponding to both the speech denoising and vocal separation tasks. Visually inspecting the samples provides an interesting observation – that the spectrogram resulting from the perceptual loss strategy appear to be "noisier" than their baseline counterparts, or that the noise appears in different time-frequency regions than in the baseline. This suggests that our proposed loss enables the network to optimize for regions of the spectrogram that more strongly influence our audition and ignore other regions, rather than optimize uniformly across the spectrogram – particularly in the case of limited capacity networks.

VII. Conclusion

In this work, we presented a method to improve source separation performance in networks with limited capacity. The underlying idea is to employ a network as a loss that is trained to remove perceptually irrelevant signal components before we compare the source separation results to ground truth.

More specifically, we adapted our perceptually relevant loss metric introduced in [1] to two audio source separation tasks, and demonstrated improved performance over an ℓ_1 baseline in listening tests. Additionally, we investigated the behaviour of our proposed loss by characterizing the contributions of different layers of our loss network, allowing us to accelerate training times, and explored the relationship between model capacity and performance. As a result, we found that designing neural network objective functions that optimize for principles of perception is not only feasible but also leads to considerable improvements in performance.

REFERENCES

[1] I. Ananthabhotla, S. Ewert, and J. A. Paradiso, "Towards a perceptual loss: Using a neural network codec approximation as a loss for generative audio models," in *Processings of the ACM International Conference on Multimedia (ACM MM)*, 2019.

[2] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[3] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1251–1258.

[4] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6848–6856.

[5] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126–136, 2018.

[6] J. Ba and R. Caruana, "Do deep nets really need to be deep?" in *Advances in neural information processing systems*, 2014, pp. 2654–2662.

[7] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2014, pp. 577–581.

[8] S. I. Mimilakis, K. Drossos, T. Virtanen, and G. Schuller, "A recurrent encoder-decoder approach with skip-filtering connections for monaural singing voice separation," in *2017 IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2017, pp. 1–6.

[9] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[10] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ) - a new method for speech quality assessment of telephone networks and codecs," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, 2001, pp. 749–752.

[11] Y. Zhao, B. Xu, R. Giri, and T. Zhang, "Perceptually guided speech enhancement using deep neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5074–5078.

[12] H. Zhang, X. Zhang, and G. Gao, "Training supervised speech separation system to improve stoi and pesq directly," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5374–5378.

[13] Y. Zhao, B. Xu, R. Giri, and T. Zhang, "Perceptually guided speech enhancement using deep neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018, pp. 5074–5078.

[14] K. Zhen, A. Sivaraman, J. Sung, and M. Kim, "On psychoacoustically weighted cost functions towards resource-efficient deep neural networks for speech denoising," *arXiv preprint arXiv:1801.09774*, 2018.

[15] F. G. Germain, Q. Chen, and V. Koltun, "Speech denoising with deep feature losses," *arXiv preprint arXiv:1806.10522*, 2018.

[16] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 694–711.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proceedings of the International Conference on Medical Mmage Computing and Computer-assisted Intervention*. Springer, 2015, pp. 234–241.

[18] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep U-Net convolutional networks," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2017, pp. 234–241.

[19] D. Stoller, S. Durand, and S. Ewert, "End-to-end lyrics alignment for polyphonic music using an audio-to-character recognition model," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Brighton, UK, 2019, pp. 181–185.

[20] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, "Speech enhancement for a noise-robust text-to-speech synthesis system using deep recurrent neural networks." in *Proceedings Interspeech*, 2016, pp. 352–356.

[21] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "The MUSDB18 corpus for music separation," 2017. [Online]. Available: https://doi.org/10.5281/zenodo.1117372

[22] D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," *arXiv preprint arXiv:1806.03185*, 2018.