BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2020
http://excel.fit.vutbr.cz

Speech Enhancement with Cycle-Consistent Neural Networks

Bc. Pavol Karlík*

Abstract
Speech enhancement aims to improve speech intelligibility and overall perceptual quality of speech by using various algorithms. Neural networks (NNs) have become a standard approach for solving such problems. NNs are usually trained by comparing the network output to the target sample. In our work, we incorporate cycle consistency constraint during the training period to improve the network robustness — we add another NN to the process. The second NN performs an opposite task — its goal is to introduce noise to clean speech recording. The networks are trained in a cycle, each taking the output of the other network as an input. Cycle-consistency, among other things, causes the network to see a much larger variety of noisy data, which improves the network's robustness. We perform experiments on both paired and unpaired data, which is enabled by adding adversarial training to the training. The DNN models are evaluated by using an automatic speech recognition system. The speech enhancement models trained and evaluated in this work are based on a recent publication. Our results have shown that adding cycle-consistency improves the models' performance significantly.

Keywords: Speech Enhancement — Deep Learning — Cycle-Consistency

Supplementary Material: *N/A*

*xkarli05@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Automatic speech recognition (ASR) is a widely used technology that allows transcription of a spoken speech utterance into a corresponding sequence of words. The field has been intensively researched in the past decades, with significant advances being made through the years. These improvements led to a surge in usage of intelligent human-machine speech communication systems, such as virtual speech assistants or interactive voice response systems.

Despite significant advances in this area, there are still certain factors that limit the performance of such systems. Most notably, reverberation and ambient noise drastically reduce speech quality of speech signal. There are many speech enhancement (SE) and ASR techniques to detect and combat the effects of noise and reverberation [1, 2, 3].

Current state-of-the-art speech enhancement methods primarily employ artificial neural networks (ANNs) [4]. These networks are trained using datasets that contain pairs of clean spoken utterances and the same spoken utterances with incorporated noise and reverberation — *paired* data. However, paired sets for training the network are not always available. *Generative adversarial network* (GAN) [5] is a framework that enables neural networks to train on *unpaired* data. In GANs, two neural networks are pitted against each other, each attempting to reach its objective, which is 'adversary' to the other network. Generative adversarial network essentially models the distribution of a given dataset.

One of the modifications of the aforementioned framework, *CycleGAN* [6], uses *cycle-consistency* for unpaired data training to further improve the architecture. CycleGAN as a whole is described in Section 3. In [6], it was demonstrated that enforcing cycle-consistency constraint significantly improves the

model robustness in image-to-image translation[1]. Cycle-consistency can be achieved by introducing an additional neural network to the framework. The network performs a dual task — in our case, it attempts to produce noisy speech signal given clean speech signal as input. Both networks are then used in conjunction to be consistent with each other. Therefore, we can run a corrupted sample through denoising network, and then use its output — an enhanced speech signal — as input to the other network, producing *reconstructed* corrupted sample. The constraint is therefore enforced by adding the reconstruction loss function to the main objective function.

The potential of CycleGAN has mostly been explored in the image processing field. The aim of this work is to evaluate the effect of cycle-consistency in the speech enhancement domain, for both paired and unpaired data. This work is based on a research paper recently published by Meng et al. (2018) [7] which proposes a framework inspired by [6] for a speech enhancement task. We implement and evaluate neural network models using the constraint during the training period for various architectures. First, using paired data, we train a standard NN without the constraint, which will serve as a baseline. Then, we train a network which inserts noise into a clean speech signal. We use that network and the baseline to further train the model with cycle-consistency. Lastly, we use the same dataset as if it contained no pairs to train a GAN with cycle-consistency constraint.

We perform experiments using automatic speech recognition (ASR) system on the CHiME-3 dataset [8]. We use evaluation set for evaluating the models with ASR system and training set for training the models. In addition, we re-train the acoustic model (AM) with data enhanced with our models and perform another set of experiments using the ASR system with re-trained AM.

Section 2 explains the problem of noise and reverberation in the speech recognition field. In Section 3, we briefly discuss the process of training standard neural networks and GANs. Additionally, we describe cycle-consistency constraint. In Section 4, we describe experiments performed in this work. Section 5, showcases our results. The article concludes with Section 6, in which we briefly summarize the results we have achieved and present possible improvements.

[1]In the case of mapping an input image to a specific output image, paired data is rarely available. For example, transforming a photo to a painting in the style of Van Gogh. There are no existing photo-Van Gogh painting pairs. However, an unpaired collection of the artist's paintings and photos can be used to train such model.

2. Speech Enhancement and Noise Reduction

Despite widespread use of ASR technologies in various systems, there is a large number of challenges that such systems need to handle to be applicable. When a speech signal is captured by a microphone, the picked up signal can get corrupted, causing loss of quality and intelligibility. Such an altered speech signal might then be erroneously processed by the ASR system. One of the fields that pursues this problem is *speech enhancement*.

This Section describes the impact of noise on speech recognition and briefly overviews current speech enhancement approaches.

2.1 Noise and Reverberation

Noise in a speech signal generally represents an unwanted modification that a signal may be subjected to when being captured or processed. According to the spectral distribution, the noises can be grouped into two categories - *stationary noise*, which keeps constant spectral distribution over time and *non-stationary noise*, which is more difficult to suppress because its statistics change over time.

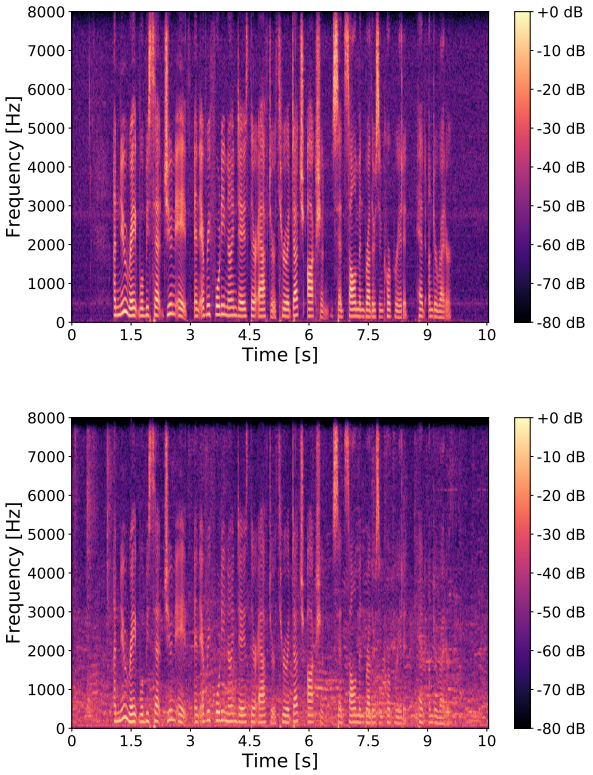

Figure 1. Comparison of clean and corrupted speech signal spectrogram.

Besides non-stationary noise, *reverberation* in a corrupted speech signals has a substantial impact on speech quality, as well [3]. Reverberation is a superpo-

sition of several time-shifted and attenuated versions of the clean signal. The signals differ in delay and amplitude, which makes the transition between phonemes in the signal less distinct. The presence of noise and reverberation is much longer than the Short Time Fourier Transform (STFT) analysis window size. This causes these artifacts to smear across several frames, as shown in Figure 1.

2.2 Speech Enhancement Methods

Generally, speech enhancement techniques modify the signal in the frequency domain. These techniques only modify the magnitude of the STFT spectrum [9], which can then be used to reconstruct the signal along with the original phase.

Common speech enhancement methods include *spectral substraction* [10], which uses noise spectrum estimated during non-speech period for denoising and *linear filter-based methods* [3], which enhance the signal in the STFT or time domain.

Neural network-based methods

These approaches vastly outperform standard speech enhancement techniques and their usage is currently considered a standard [2, 4]. Both high-level features and raw speech can be used as an input and output of the network.

While the ASR performance in difficult noisy and reverberant conditions has significantly improved over the past years [2, 11], there still are certain areas of focus where such systems perform poorly. It has been observed that, when ASR systems are given a challenging environment with distant noisy and overlapping conversational speech, the system performance suffers significantly [12].

3. Neural Networks and Cycle Consistency

The main purpose of a neural network is to map input X to another output $\tilde{Y}$, formally written as

$$F : X \longrightarrow \tilde{Y} \approx Y, \tag{1}$$

where the network F attempts to produce an output that is similar to a reference sample Y with respect to the cost function. We can indirectly improve the robustness of F by enhancing the training process with a constraint, called *cycle-consistency*.

3.1 Cycle-Consistent Neural Network

Cycle-consistency is a technique initially used in machine translation and visual tracking, that can be applied to enforce additional constraints within the training framework [13, 14]. For example, when translating a sentence from language A to language B, the machine should be able to transform the translated sentence back to the original sentence in language A. This form of cycle-consistency is called *forward-backward consistency*.

Cycle-consistency can be achieved by introducing an additional neural network to the framework. The network serves as an inverse mapping function

$$G : Y \longrightarrow \tilde{X} \approx X, \tag{2}$$

where G is the neural network performing a *dual task* — attempting to produce X when given Y as an input. Both networks are then used in conjunction to be consistent with each other. The *forward cycle-consistency* objective aims to accurately reconstruct X, and can be defined as

$$X \longrightarrow F(X) \longrightarrow G(F(X)) \approx X. \tag{3}$$

Similarly, we can define *backward cycle-consistency*, where the goal is to reconstruct Y, as follows:

$$Y \longrightarrow G(Y) \longrightarrow F(G(Y)) \approx Y. \tag{4}$$

The constraints are enforced by adding the cost functions for (3) and (4) to the main objective function. A large advantage of this constraint is that it has no computational performance impact on use of the resulting model. Since network F produces the wanted output, network G is not used at beyond the training process. The full objective function is defined as follows:

$$\begin{aligned} L_{CSE} = \lambda_1 L(F) + \lambda_2 L(G) + \\ \lambda_3 L(F,G) + \lambda_4 L(G,F), \end{aligned} \tag{5}$$

where $L(F)$, $L(G)$, $L(F,G)$ and $L(G,F)$ is a cost function of F, G, a forward and a backward cycle, respectively, with λs being weight coefficients. The training framework is shown in Figure 3.

Enforcing forward-backward consistency can improve the robustness of speech enhancement models [7]. In this work, which is based on [7], we implement the cycle-consistency framework for speech enhancement. Besides a standard neural network framework, we use the constraint together with *generative adversarial network* (GAN).

3.2 Generative Adversarial Networks

The goal during the training of a standard neural network is to produce an output that resembles a certain target output — the label. It is a *supervised* approach

to learning. However, neural networks require a significant amount of training data and having a labeled sample for each training input can be a costly task. The main advantage of *generative adversarial networks* (GANs) [5] is that it can be trained with data that does not contain input-target pairs (paired data). In GANs, two neural networks are pitted against each other, each attempting to reach its objective, which is 'adversary' to the other network. These models learn to model the probability distribution of data that resembles a given training set.

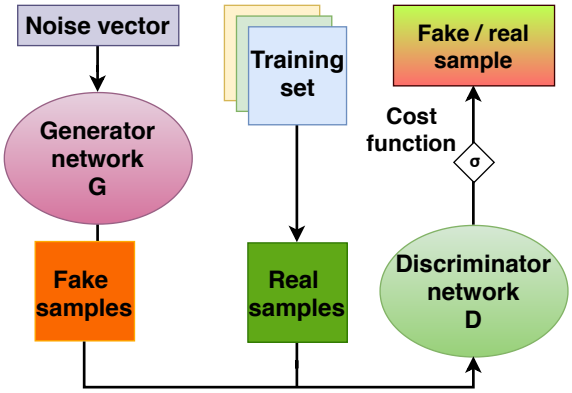

Figure 2. A structure of generative adversarial network.

Generative adversarial network consists of *two* neural networks, as seen in Figure 2. The generator, G, produces samples in the target domain, Y, given the generated noise (e.g., uniform noise) Z:

$$G : Z \longrightarrow Y. \tag{6}$$

Its adversary, D attempts to recognize whether its inputs have been drawn from the training set or not. The output of the discriminator is defined as

$$D : X \longrightarrow < 0, 1 > . \tag{7}$$

Generative adversarial networks are primarily used for image vision problems. In this work, we attempt to couple GANs with cycle-consistency constraint to solve speech enhancement.

3.2.1 CycleGAN

CycleGAN [6] is a GAN framework that uses a cycle-consistency loss to enable training without the need for paired data. It was originally proposed for image-to-image translation problems.

The goal of CycleGAN is to learn a mapping from the source domain to the target domain and vice versa. The framework consists of *four* neural networks in total — two generator-discriminator pairs. Forward and backward cycle-consistency losses are added to the cost function.

Additionally, the full cost function is extended with *identity mapping loss*. The generator networks are kept close to the identity mappings by the following constraints:

$$X \longrightarrow G(X) \approx X, \tag{8}$$

$$Y \longrightarrow F(Y) \approx Y. \tag{9}$$

The full objective function of CycleGAN is defined as

$$\begin{aligned} L_{CycleGAN} = \lambda_1 L(F,G) + \lambda_2 L(G,F) - \lambda_3 L_D(D_F) \\ -\lambda_4 L_D(D_G) + \lambda_5 L_I(F) + \lambda_6 L_I(G), \end{aligned} \tag{10}$$

where $L(F,G) + L(G,F)$ is a forward-backward cycle-consistency loss, $L_D(D_F)$, $L_D(D_G)$ are discriminator losses, and $L_I(F)$, $L_I(G)$ are F and G identity losses, respectively, with λs being weight coefficients. The whole CycleGAN architecture used in this work is shown in Figure 4.

Unlike standard GANs, the generator networks in CycleGAN do not take a sample from random noise as input. Instead, the input is a specific piece of information, such as noisy speech utterance [15].

Recently, experiments using CycleGAN for single-channel speech enhancement problem have been conducted [7]. In this work, we apply the same architecture to evaluate performance for speech enhancement task.

4. Experiments

For training and evaluation, we use the CHiME-3 dataset [8]. The dataset incorporates Wall Street Journal (WSJ) corpus sentences spoken in challenging noisy environments, specifically in café (CAF), street junction (STR), public transport (BUS) and pedestrian areas (PED). The *real* data consists of 6-channel recordings of sentences spoken live in the environments. The *simulated* data was constructed by mixing clean utterances into environment background recordings. The training set consists of 9137 pairs of clean and simulated noisy training utterances. For testing, we use real noisy speech utterances from the development test set. We use recordings from the 5th channel, as was done in [7].

We evaluate three models in total — noisy-to-clean mapping network, which will serve as a baseline, a network with cycle-consistency, and a generative adversarial network with cycle-consistency. These models were proposed in [7]. For testing, only the noisy-to-clean mapping network portion of the models is used to produce enhanced speech utterances.

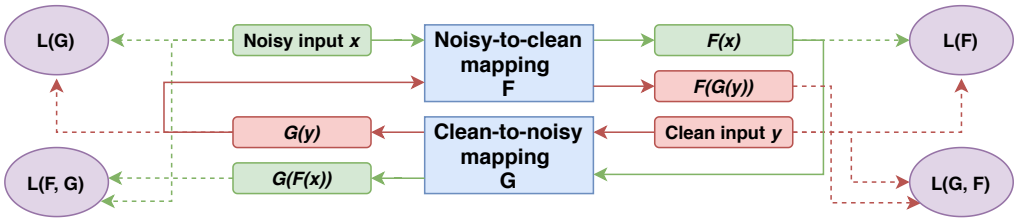

Figure 3. The architecture of cycle-consistency training framework for speech enhancement (CSE). Based on [7].

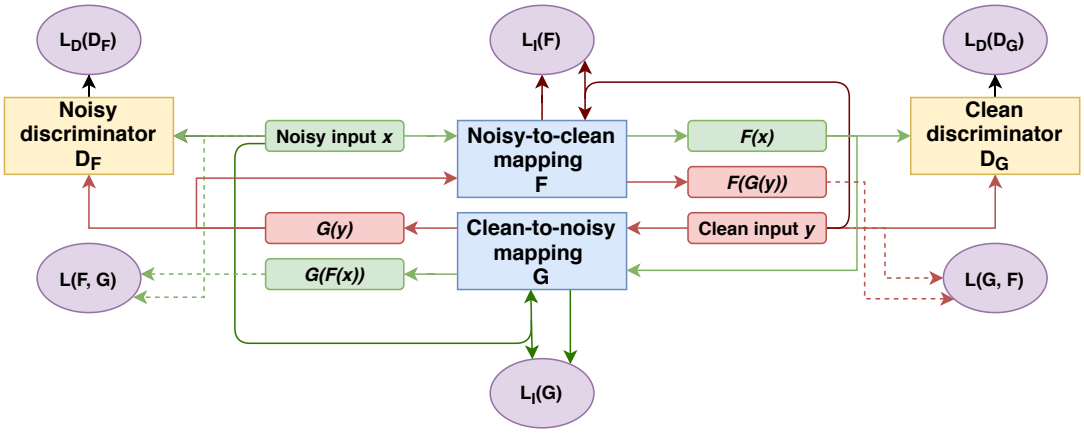

Figure 4. The architecture of adversarial training framework with cycle-consistency for speech enhancement (ACSE). Based on [7].

Training and evaluation was done using Sun Grid Engine[2] job scheduling system. The evaluation is performed by using a provided ASR system. ASR scripts and acoustic model re-training scripts were provided for this work. Training framework and the implemented neural network architectures are the author's own work.

4.1 Models Trained With Paired Data

This subsection describes models trained with *paired* data — each noisy utterance in the set has a corresponding clean utterance. We describe the training process of the baseline model and models with cycle-consistency constraints.

4.1.1 Baseline

Using standard supervised training, we first train a neural network for suppressing noise, F. The network input consists of log Mel-filterbank (MFB) features appended with first and second-order delta features, forming an 87-dimensional vector. The output is a 29-dimensional MFB without delta features. The network consists of two Long-Short Term Memory [16] layers followed by a linear layer. Each LSTM layer has 512 units. The input features were globally normal and mean variance normalized before being fed into the network.

We heavily tuned network parameters in order to achieve satisfying results. In LSTM layers, forget gate biases are initialized to 1 (otherwise 0) [17]. The weights were initialized using Xavier normal distribution [18]. For optimization, we use AdamW algorithm [19]. The learning rate is set at $9 \cdot 10^{-4}$ and batch size is set at 48. The weight decay of AdamW is set at $1 \cdot 10^{-4}$. We find that using recurrent dropout in the first LSTM layer slightly lowers the model performance. We use Mean Squared Error (MSE) as a cost function.

This baseline setup slightly deviates from [7], in which the network was optimized by using stochastic gradient descent (SGD) optimizer. No weight initialization techniques nor any other training parameters were mentioned in the reference paper.

4.1.2 Forward and Backward Cycle-Consistency

We train a neural network, G, that inserts noise into clean speech utterance. The input and the output feature dimensions are 29 and 87, respectively. The learning rate is set at $8 \cdot 10^{-4}$. Other parameters and a cost function are the same as specified in 4.1.1.

Then, we use the pre-trained networks, F and G, and jointly train them using cycle-consistency loss. We train the model with forward cycle-consistency and a model with both forward and backward cycle-consistency. When computing cycle-consistency loss, the input of one network is normalized before being fed to the other network. We set the learning rate at

[2] Sun Grid Engine - http://www.fit.vutbr.cz/CVT/cluster/SGE-UsersGuide.pdf

$4 \cdot 10^{-4}$. The batch size is set at 24. The λ loss function coefficients are the same as in [7].

While using a forward cycle alone has shown to degrade the model performance, having both forward and backward cycle constraints has shown to further improve the model's robustness. The training process of the best model converges in 7 epochs. We assume that, by pre-training F and G with carefully tuned hyperparameters, the networks adjust the weights to a relatively proper state rather quickly.

4.2 Models Trained With Unpaired Data

We use the same training set that contains noisy-clean sample pairs. However, the dataset is used as if it contained no related pairs. In practice, we take a batch of random noisy samples, a different batch of random clean samples, and work with these during the training iteration.

For generator networks, we use the same architecture as F and G. The discriminator networks consist of two fully-connected hidden layers. Each hidden layer has 512 units. The output layer has 1 unit. The discriminators, D_F and D_G, take 87-dimensional inputs (appended with delta features) and 29-dimensional inputs, respectively. D_F and D_G evaluate the probability of the input belonging to the noisy and clean set, respectively. We use AdamW optimizer for both generator and discriminator training.

Generally, GANs are difficult to train as a whole, as they can be very sensitive to changing hyperparameters. A large amount of minor training process adjustments was proposed [20, 21] that can significantly improve convergence and prevent common pitfalls, such as mode collapse [20].

Before beginning adversarial training, the generator networks need to be initialized in order to learn an underlying structure. Otherwise, the model would have trouble converging. From our experiments, these techniques were important to make the adversarial training converge:

- **initialization of generators** - The initialization is done by pre-training the generators as identity mapping functions — the target sample is the same as the input sample, but without normalization. The training hyperparameters for noisy-to-clean and clean-to-noisy generator networks are the same as of F and G, respectively. The initialization procedure in [7] may differ, as the initialization details were not mentioned.
- **buffer of generated samples** - As suggested by Shrivastava et al. [21], we update the discriminators by using a history of generated utterances rather than the ones produced by the latest generators. We store two sample buffers of size 72 that keep previously generated noisy and clean samples. The original Cycle-GAN uses a history of 50 samples [6]. It is not specified whether the framework in [7] uses such technique.
- **one-sided label smoothing** - We modify the cross-entropy cost functions of the discriminators by employing one-sided label smoothing. Label smoothing is a regularization technique that prevents the discriminators from predicting the labels too confidently during training, which can result in poor generalization.

Using pre-trained generators, we perform adversarial training. The learning rate and weight decay are set at $1 \cdot 10^{-6}$. During each iteration, the discriminator networks are trained before the generator networks. The λ loss function coefficients are the same as in [7].

5. Results

In this Section, we show the performance of our best-performing models for each category. We discuss the re-training of an acoustic model portion of the ASR system and discuss its impact on the system performance. We compare our approach to the reference publication. The results show that cycle-consistency plays a vital role in improving the model's robustness.

5.1 Models Trained With Paired Data

Due to carefully optimizing training hyperparameters and using proper weight and bias initialization methods, the baseline model alone reduces the ASR word error rate (WER) by 17.40% as opposed to no enhancement.

Model	Noise Environment					
	BUS	PED	CAF	STR	Avg.	RWERR
None	41.27	17.48	27.09	24.97	27.70	-
Baseline	28.91	16.36	27.58	18.68	22.88	17.40
CSE-FW	29.41	15.68	26.68	18.82	22.65	18.23
CSE	**28.35**	**15.40**	**25.24**	**18.57**	**21.89**	**20.97**

Table 1. The ASR WER (%) performance of real noisy test data in CHiME-3 enhanced by different models. Relative WER reductions (%) are shown in the last column. BUS, PED, CAF, STR refer to 4 different recording environments.

Further training the model with using only a forward cycle-consistency (CSE-FW) slightly boosts the model performance, up to 18.23% relative WER reduction (RWERR). The model with both cycle-consistency

constraints has shown to perform better, increasing the RWERR to 20.97%.

5.2 Models Trained With Unpaired Data

As shown in Table 2, we have reached 12.71% relative WER improvement over noisy data with our variation of CycleGAN (named ACSE), which is only slightly worse than the baseline from Table 1, which was trained with paired data.

Model	Noise Environment					
	BUS	PED	CAF	STR	Avg.	RWERR
None	41.27	17.48	27.09	24.97	27.70	-
ACSE	**32.91**	**16.27**	**26.39**	**21.16**	**24.18**	**12.71**

Table 2. The ASR WER (%) performance of real noisy test data in CHiME-3 enhanced by different models. Relative WER reductions (%) are shown in the last column.

5.3 Re-training the Acoustic Model

The authors of [7] performed acoustic model re-training in order to improve ASR performance. The acoustic model portion of the ASR is trained by taking speech recordings and their text transcriptions, from which a statistical representation of the sounds that make up each word is created. The ASR system provided for this work, trained using Kaldi[3] has DNN-HMM acoustic model, which we re-trained on speech utterances enhanced by our models. The dataset used for re-training the acoustic model is the same set that was used for training the neural networks.

In the publication, the re-training was performed on data enhanced from the adversarial model (ACSE), but not others. We have re-trained the acoustic model not only on ACSE, but also on baseline and CSE, as well.

Acoustic Model	Architecture			
	Baseline	CSE	ACSE	ACSE ([7])
Clean	22.88	21.89	24.18	29.44
Re-trained	19.16	18.42	**14.72**	18.20

Table 3. Comparison of ASR WER (%) performances of speech enhancement models evaluated with clean and re-trained ASR acoustic model.

As seen in Table 3, re-training an acoustic model significantly boosts the performance, causing total relative WER reduction up to 46.86% for ACSE. Surprisingly, acoustic model re-trained using ACSE-enhanced speech samples shows biggest performance improvement. Similar improvements can be seen in ACSE ([7]). CSE and baseline only slightly improve the ASR performance. The possible reason for this is that while ACSE performs worse on a test set, it can generalize to unseen data better, and thus be a more appropriate candidate for re-training the acoustic model.

[3]Kaldi ASR - https://kaldi-asr.org/

5.4 Summary

To summarize our work, we present a table with relative WER reductions (RWERR) to overview our achieved results. The table depicts relative WER improvements over noisy data. The WERs of noisy data in our work and the publication are 27.70% and 29.44%, respectively.

Model origin	Architecture			
	Baseline	CSE	ACSE	ACSE (re-trained AM)
Our work	**17.40**	**20.97**	**12.71**	**46.86**
Publication	12.33	19.60	6.9	38.17

Table 4. Comparison of relative WER improvement (%) of models over noisy data.

Table 4 shows that by carefully picking hyperparameters and using various NN training enhancements, our models have performed significantly better compared to [7]. The table shows relative word error rate reduction over results obtained from data without any form of enhancement. While the models in [7] were evaluated using a different ASR system, the relative WER improvement is shown over WER of noisy data from the publication, which was 29.44%, whereas the noisy data WER in our work was 27.70%.

6. Conclusions

The goal of this work was to apply cycle-consistency constraint during the training process to improve the performance of speech enhancement models. The constraint only alters the process of training the target neural network, and the second neural network is not used in the evaluation/application phase.

We evaluated multiple models, whose goal was to enhance speech utterances using only a single channel. We trained models with paired data to extend standard NN with the constraint and unpaired data using a slight modification of the CycleGAN architecture.

Our results have shown that the cycle-consistency constraint significantly improved the performance of the models. Training with paired data, CSE has reached a relative WER reduction of 20.97% when compared to noisy data, while [7] achieves 19.60% RWERR. Our

Cycle-GAN variant, ACSE, achieved 12.71% RWERR on unpaired data, which is a significant improvement compared when to [7]'s 6.69 % RWERR. The models were used to re-train the acoustic model, which was then used to re-evaluate the ASR WER of those models. The baseline, CSE, and ACSE has reached 30.83%, 33.50% and 46.86% relative WER improvement over noisy data, respectively.

For future work, training features in the time domain, as opposed to the frequency domain, can be considered [22, 23], as certain information can be lost when transforming speech into higher-level features. Temporal convolutional neural networks [24] have recently been used with great success for speaker separation [25], but have not yet been much explored for speech enhancement.

Acknowledgements

I would like to express my sincere gratitude and thanks to my supervisor, Ing. Kateřina Žmolíková for her valuable advice, support, and guidance.

References

[1] Keisuke Kinoshita, Marc Delcroix, Sharon Gannot, Emanuël AP Habets, Reinhold Haeb-Umbach, Walter Kellermann, Volker Leutnant, Roland Maas, Tomohiro Nakatani, Bhiksha Raj, et al. A summary of the reverb challenge: state-of-the-art and remaining challenges in reverberant speech processing research. *EURASIP Journal on Advances in Signal Processing*, 2016(1):7, 2016.

[2] Shinji Watanabe, Marc Delcroix, Florian Metze, and John R Hershey. *New era for robust speech recognition: exploiting deep learning*. Springer, 2017.

[3] Takuya Yoshioka, Armin Sehr, Marc Delcroix, Keisuke Kinoshita, Roland Maas, Tomohiro Nakatani, and Walter Kellermann. Making machines understand us in reverberant rooms: Robustness against reverberation for automatic speech recognition. *IEEE Signal Processing Magazine*, 29(6):114–126, 2012.

[4] Zixing Zhang, Jürgen Geiger, Jouni Pohjalainen, Amr El-Desoky Mousa, Wenyu Jin, and Björn Schuller. Deep learning for environmentally robust speech recognition: An overview of recent developments. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 9(5):49, 2018.

[5] 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*, pages 2672–2680, 2014.

[6] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[7] Zhong Meng, Jinyu Li, Yifan Gong, et al. Cycle-consistent speech enhancement. *arXiv preprint arXiv:1809.02253*, 2018.

[8] Jon Barker, Ricard Marxer, Emmanuel Vincent, and Shinji Watanabe. The third 'chime'speech separation and recognition challenge: Dataset, task and baselines. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 504–511. IEEE, 2015.

[9] Jont B Allen and Lawrence R Rabiner. A unified approach to short-time fourier analysis and synthesis. *Proceedings of the IEEE*, 65(11):1558–1564, 1977.

[10] Steven Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on acoustics, speech, and signal processing*, 27(2):113–120, 1979.

[11] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig. Achieving human parity in conversational speech recognition. *arXiv preprint arXiv:1610.05256*, 2016.

[12] Jon Barker, Shinji Watanabe, Emmanuel Vincent, and Jan Trmal. The fifth'chime'speech separation and recognition challenge: Dataset, task and baselines. *arXiv preprint arXiv:1803.10609*, 2018.

[13] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pages 820–828, 2016.

[14] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Forward-backward error: Automatic detection of tracking failures. In *2010 20th International Conference on Pattern Recognition*, pages 2756–2759. IEEE, 2010.

[15] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[16] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[17] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *International Conference on Machine Learning*, pages 2342–2350, 2015.

[18] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[20] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[21] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2107–2116, 2017.

[22] Ashutosh Pandey and Deliang Wang. A new framework for supervised speech enhancement in the time domain. In *Interspeech*, pages 1136–1140, 2018.

[23] Ashutosh Pandey and DeLiang Wang. Tcnn: Temporal convolutional neural network for real-time speech enhancement in the time domain. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6875–6879. IEEE, 2019.

[24] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[25] Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27(8):1256–1266, 2019.