UNetGAN: A Robust Speech Enhancement Approach in Time Domain forExtremely Low Signal-to-noise Ratio Condition

백혜림

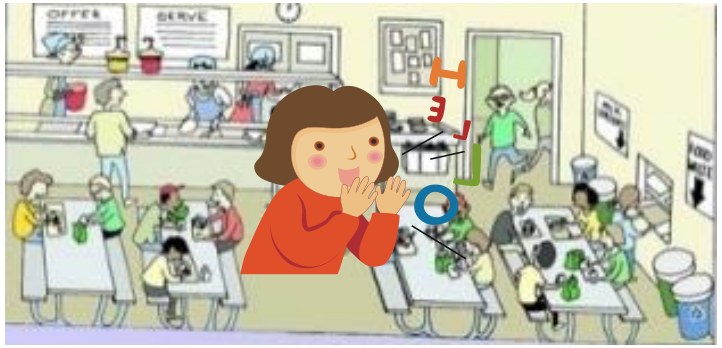
Abstract

- U-net 및 generative adversarial learning에 기반한 speech enhancement 접근법 UNetGAN을 제안
- 시간 영역에서 직접 작동하는 generative network와 discriminator 네트워크로 구성
- Generator network는 U-Net과 유사한 구조 쓰고 dilated convolution을 사용
- STOI와 PESQ에서 우수한 성능을 보임
  - SEGAN
  - cGAN for SE
  - Bidirectional LSTM using phase-sensitive spectrum approximation cost function
  - Wave-U-Net

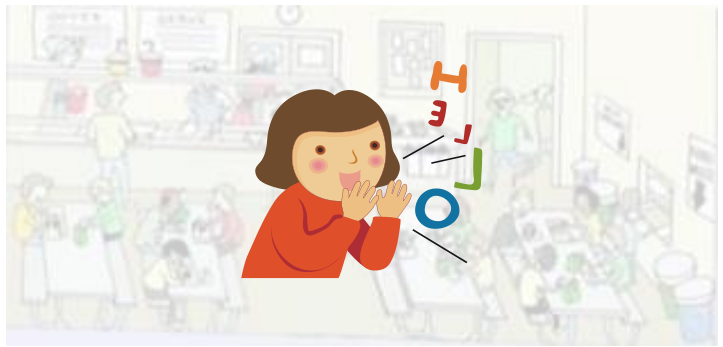
Introduction

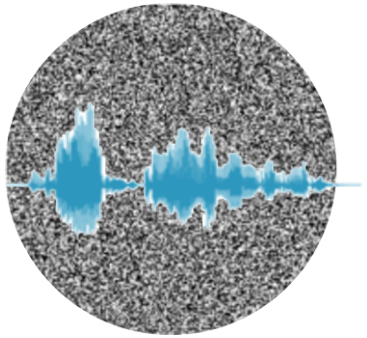
Speech enhancement

Noise + speech

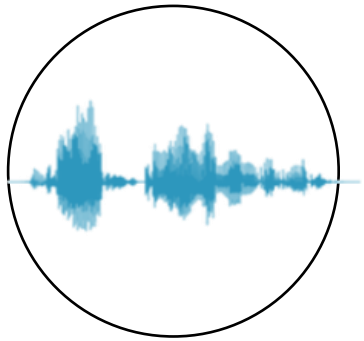


speech





Clean speech



[0.9, 0.8, 0.235, 0.876, 0.124, …]

- 낮은 snr에서의 speech enhancemen가 높은 snr보다 더 중요하다!

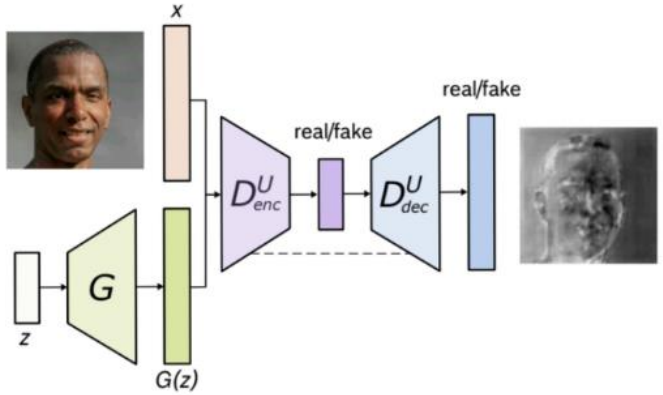
Introduction

- 잡음과 clean speech인 noisy한 환경에서 clean speech를 분리하는 speech separation으로 본다.
- GAN은 generator network와 discriminator network사이의 min-max game을 통해 generator network의 성능을 향상시킬 수 있다.
- Dilated convolution은 receptive filter 크기를 확장하고 큰 시간적 context를 고려할 수 있다.
- Generator network는 down sampling과 upsampling 블록 사이에 dilated convolution이 사용되는 U-Net 유사 구조를 사용한다.
- Discriminator network는 batch normalization과 leaky Relu를 포함하는 일반적인 convolution 신경망이다.

Related work

- 기존 모델인 SEGAN과 다르다!
- 시간 영역에 작동한다!
- SEGAN은 입력데이터에 high-frequency preemphasis filter를 적용한 것!

Architecture



- Discriminator D는 음성들을 훈련데이터 (실제, 1에 가깝게) or Generator G(가짜, 0에 가까움)에 나온 것으로 분류하도록 훈련

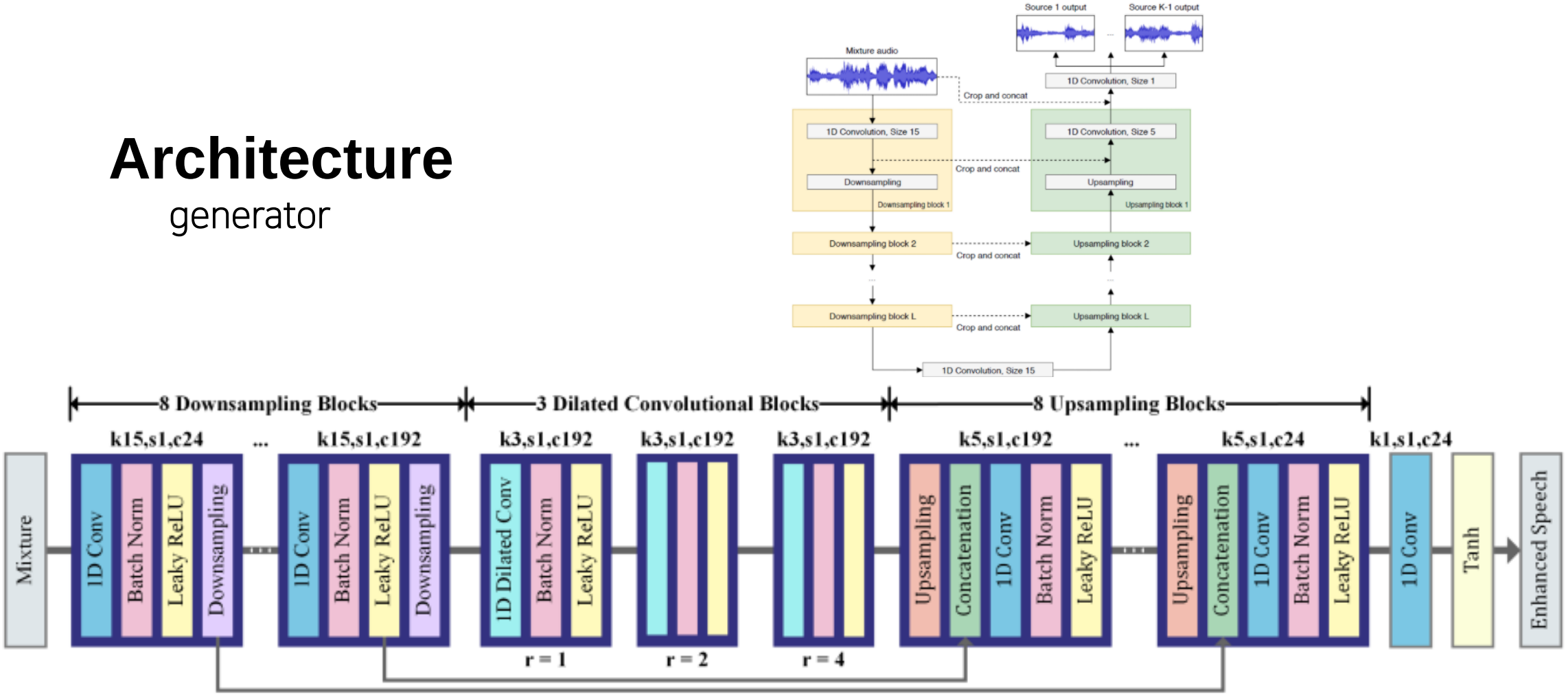
$$D(x, y)\ or\ D(x, \hat{y}) -\succ (0,1)$$

- Noisy speech x에 맞춰 조절되고, clean speech or enhanced speech $\hat{y}$를 실제 데이터 분포에 매핑한다.
- Generator G는 noisy speech x에서 enhanced speech $\hat{y}$: $G(x) \rightarrow \hat{y}$로 매핑되며, discriminator D를 혼동한다. Mini-max game을 하고 objective funtion은 다음과 같이 표현된다.

$$\arg\min_{G}\max_{D}(\mathbb{E}_{x,y}[logD(x,y)] + \mathbb{E}_{x,\hat{y}}[log(1 - D(x,\hat{y})])$$

Architecture

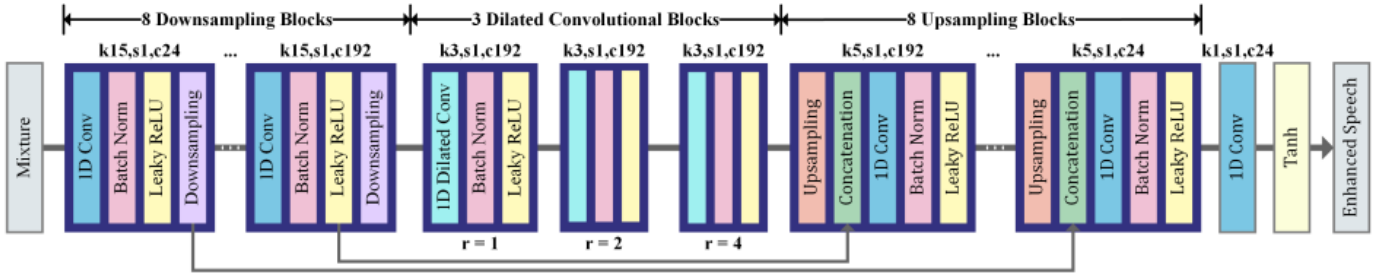
generator



- Noisy speech x는 downsampling블록을 사용하여 더 좁은 시간척도에서 점점 더 많은 상위 레벨 기능으로 변환한다.
- 세 개의 연속적인 dilated convolution블록에 의해 처리되어 더 큰 context를 통합!
- Skip connection을 통해 upsampling(UP)블록을 사용하여 초기 local 고해상도 기능과 결합되어 예측을 위해 사용되는 다중 스케일 기능을 산출한다.

Architecture

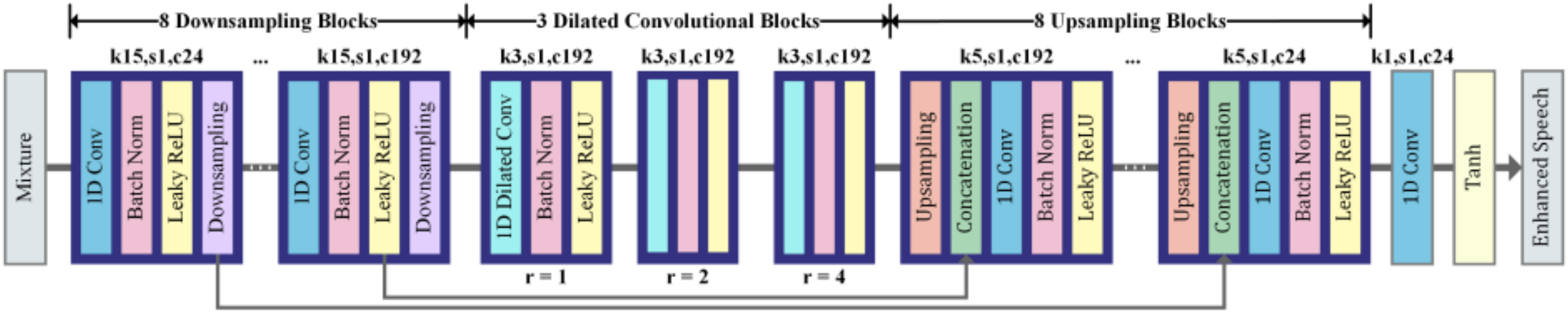
generator



- Generator의 downsampling블록에는 총 8개의 level이 있다.
- 각 연속 level은 이전 레벨과 같은 절반의 time resolution으로 작동하는 반면 채널 수는 24개의 간격으로 점진적으로 증가한다.
- 각 downsampling블록은 batch normalization, leaky relu와 down sampling에 따라 1Dconvolution을 수행
- 1D convolution의 매개변수는 각 downsampling블록 위에서 찾을 수 있으며, k, s 및 c는 각각 1D convolution의 커널크기, stride 및 channel를 나타낸다.
- Input과 동일한 time resolution의 출력을 생성하기 위해 동일한 패딩 수행
- Batch normalization은 네트워크 성능과 안정성을 보장하기 위해 사용
- 최종layer만 tanh를 사용하고, leaky relu를 각 layer의 활성화 함수로 사용
- Time resolution을 절반으로 낮추기 위해 다른 모든 시간 단계의 feature을 decimate.

Architecture

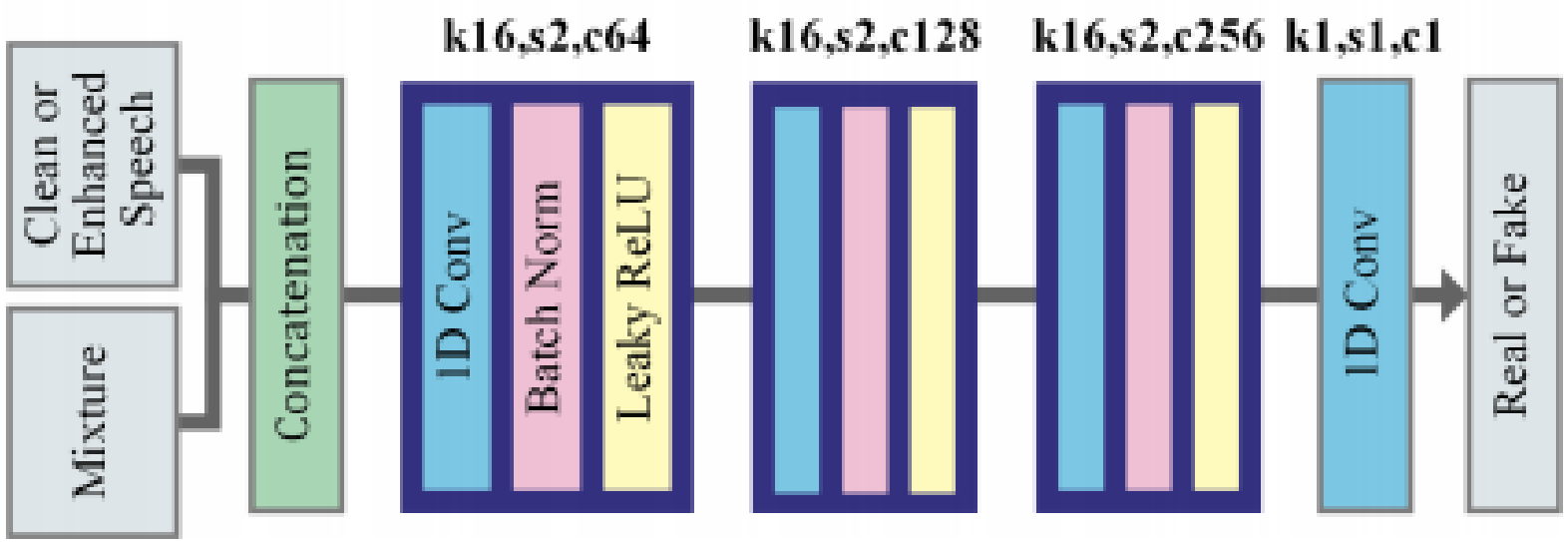
generator



- Dilated convolution블록에서 서로 다른 확장 속도(r=1,2,4)를 가진 세 개의 연속적인 dilated convolution연산을 사용하여 바람직한 time resolution에서 점진적 추출

Architecture

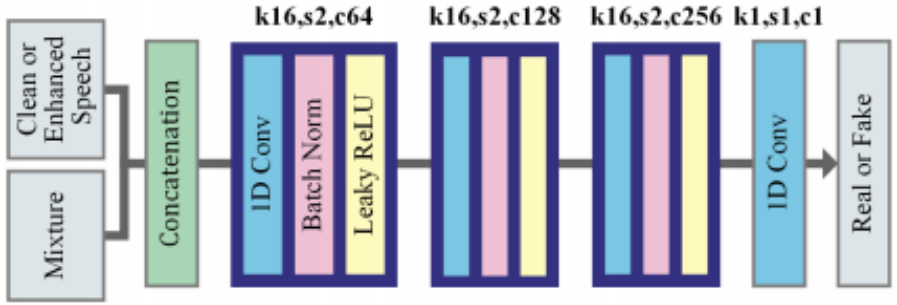
discriminator



k – kernel size
s – stride
c – channel의 수

Architecture

discriminator



- 결과 feature map은 Upsampling input으로 사용된다.
- Dilated convolution은 다음 절에 설명
- Upsampling은 time 방향에서 2배 인자로 linear interporation을 수행
- 각 레벨의 채널 번호가 24번 간격으로 감소
- Clean speech or enhanced speech는 noisy speech와 연결되고 1Dconvolution, batch normalization 및 leaky relu를 사용하여 점점 더 많은 수의 feature map으로 변환
- 3 개의 convolution블록 후에 feature map은 높은 수준으로 압축
- 두 네크워크는 번갈아 훈련
- Generator G의 경우, discriminator D는 clean speec와 enhanced speec와 구별하도록 훈련하고, discriminator가 최적일 때, 이를 동결하고, generator G를 계속 훈련시켜 discriminator의 정확도를 낮출 수 있다.

Architecture

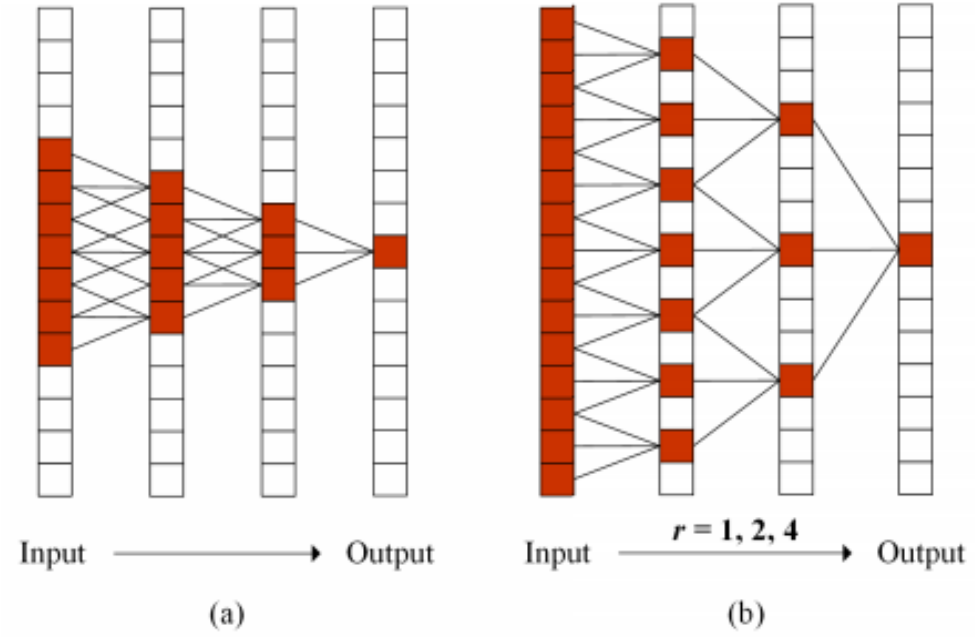
Dilated convolution

- Dilated convolution은 generator network에서 사용
- Kernel사이에 공백을 삽입하여 커널을 팽창시킨다.
- Receptive filed크기를 확대하여 더 큰 context를 통합할 수 있다.
- 1-D입력 신호 x(i)의 경우 길이 k의 필터 w(k)를 가진 확장 convolution의 출력 y(i)는 다음과 같이 정의된다.

$$y[i] = \sum_{k=1}^{K} x[i + r \cdot k] w[k] \qquad (1)$$

Architecture

Dilated convolution



- 1-D신호의 기존 convolution과 dilated convolution (r=1,2,4)를 보여주고, 여기서 stride = 1, 커널크기 =3
- 그림 (a)는 Receptiv filed 크기가 layer 수에 따라 선형이 세 개의 순차 convolution 연산 후 7이라는 것을 보여준다.
- 그림 (b)와 같이 기하급수적으로 증가하는 확장 속도(r=1,2,4)를 사용하면 receptive field크기가 15로 기하급수적으로 증가한다.
- 모델의 dilated convolution블록은 그림 3(b)과 같은 지수적으로 증가하는 확장 속도 (r=1,2,4)를 사용하여 receptive 필드의 크기가 기하급수적으로 증가

Architecture

Loss function

- Gan 기반 네트워크는 적대적 손실을 loss function으로 채택
- Generator loss function

$$\mathcal{L}_G = \underbrace{\mathbb{E}_x[log(1 - D(x, G(x)))]}_{Adversarial\ Loss} + \lambda \underbrace{\mathbb{E}_{x,y}[y - G(x)]^2}_{MSE} \quad (2)$$

- Discriminator
  - Noisy speech x에서 clean speech를 참으로 매핑하고, enhanced speech G(x)를 거짓 조건화 하여 거짓으로 매핑하는 역할

$$\mathcal{L}_D = -\mathbb{E}_x[log(1 - D(x, G(x)))] - \mathbb{E}_{x,y}[logD(x, y)] \quad (3)$$

Experiment

Dataset and metrics

- Training set
  - Noisx92 : Babble, factory floor1, destroyerengine 및 destroyerops, factoryfloor2
  - 각 noise의 초반 2분은 4SNR(0dB, -5dB, -10dB, -15dB) 중 하나에 training part로 clean speech와 섞인다.
  - 총 9600개의 훈련 샘플을 산출. 각각은 noisy speech와 그에 상응하는 Clean speech로 구성
  - Training set에 noise외에도, 일반적인 성능을 평가하기 위해 factoryfloor2를 선택
  - 각 noise의 마지막 2분은 9SNR(0dB, -3dB, -5dB, -7dB, -10dB, -12dB, -15dB, -17dB, -20dB) 중 하나에 test 발화로 섞는다.
- Validation set
  - 2250 sample를 포함하는 Test set과 같은 방법으로 만들어진다.
- 모든 sample의 sampling rate은 16000Hz
- Test Noise가 training set에서 반복되지 않도록 noise를 두 섹션으로 나눔.
- STOI랑 PESQ

Experiment

training

- Training
  - Learning rate = 0.0002
  - Adam, Decay rates $\beta 1 = 0.9$, $\beta 2 = 0.999$
  - Leaky relu의 batch size : 150
  - 음의 기울기 0.1
  - Generator loss는 초기 진동 동작으로 나타내며, 900epoch이후에 점진적 수렴
  - 0.771에 가까움.

Experiment

Result and discussion

N1 babble,
N2 factoryfloor1
N3 destroyerengine
N4 destroyerengine
N5 factoryfloor2

Noise	Target	STOI									PESQ								
		Seen				Unseen					Seen				Unseen				
		0dB	-5dB	-10dB	-15dB	-3dB	-7dB	-12dB	-17dB	-20dB	0dB	-5dB	-10dB	-15dB	-3dB	-7dB	-12dB	-17dB	-20dB
N1	Mixture	0.735	0.635	0.531	0.440	0.676	0.593	0.492	0.412	0.378	1.863	1.483	1.158	0.915	1.629	1.343	1.049	0.874	0.833
	Enhanced	0.903	0.879	0.840	0.780	0.890	0.866	0.818	0.749	0.695	2.659	2.467	2.226	1.975	2.550	2.380	2.129	1.870	1.722
N2	Mixture	0.755	0.650	0.551	0.475	0.692	0.609	0.517	0.452	0.426	1.783	1.507	1.309	1.188	1.609	1.405	1.249	1.163	1.132
	Enhanced	0.897	0.874	0.841	0.793	0.884	0.862	0.824	0.766	0.716	2.517	2.339	2.146	1.918	2.413	2.266	2.063	1.811	1.651
N3	Mixture	0.755	0.665	0.566	0.477	0.703	0.626	0.528	0.449	0.415	1.935	1.540	1.190	0.928	1.696	1.390	1.078	0.867	0.762
	Enhanced	0.910	0.888	0.853	0.802	0.898	0.876	0.835	0.775	0.723	2.751	2.572	2.354	2.091	2.651	2.485	2.252	1.983	1.810
N4	Mixture	0.721	0.610	0.507	0.432	0.654	0.566	0.473	0.411	0.390	1.775	1.392	1.100	0.859	1.547	1.262	0.963	0.820	0.792
	Enhanced	0.897	0.869	0.830	0.774	0.882	0.855	0.810	0.745	0.691	2.656	2.458	2.230	1.980	2.541	2.368	2.131	1.869	1.701
N5	Mixture	0.830	0.755	0.664	0.567	0.787	0.720	0.625	0.530	0.481	2.231	1.851	1.487	1.183	2.004	1.701	1.354	1.083	0.972
	Enhanced	0.876	0.813	0.708	0.575	0.843	0.775	0.657	0.540	0.500	2.502	2.179	1.806	1.425	2.313	2.030	1.653	1.285	1.137

- STOI와 PESQ측면에서 0dB ~ -20dB의 enhanced speech와 noisy speech
- Seen은 training set에 SNR 존재하는 것, Un-seen은 training set에서 존재하지 않은 SNR조건

Experiment

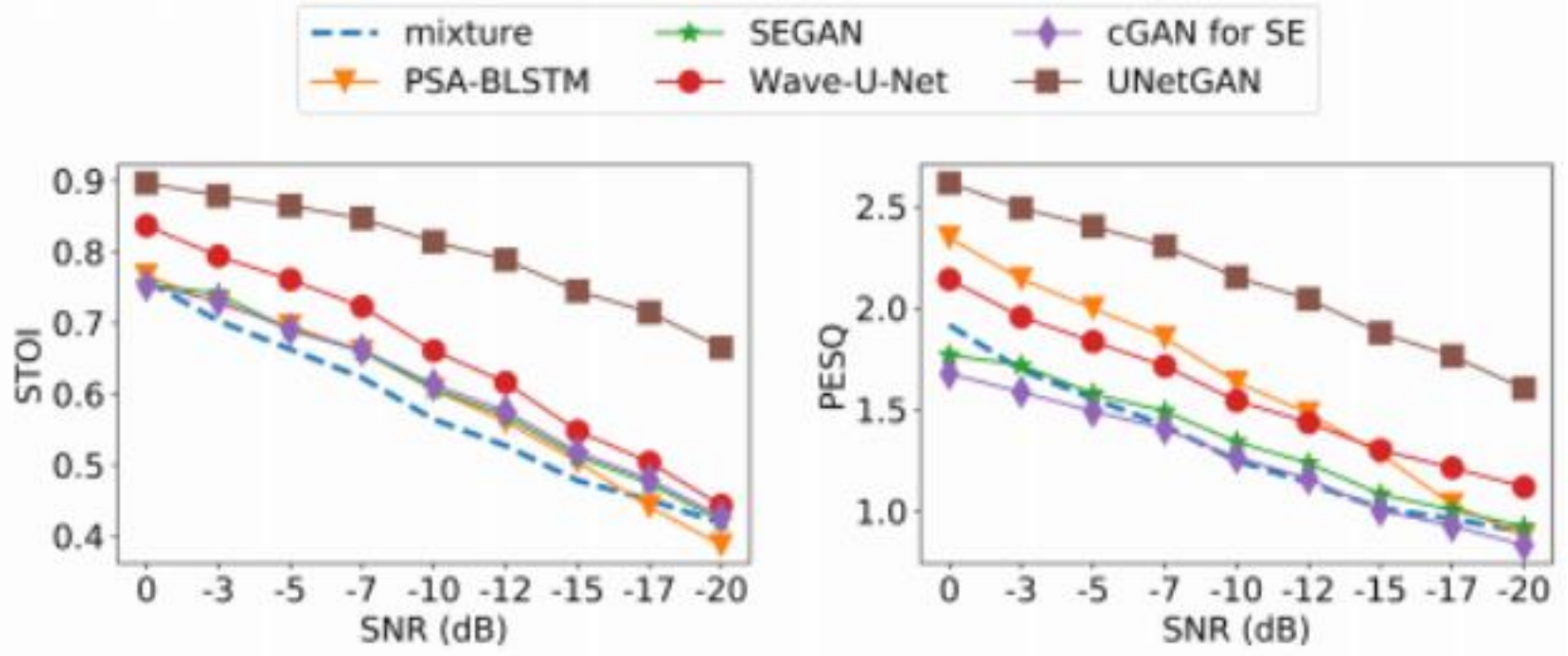
Result and discussion

Table 2: *The average performances of UNetGAN and the baseline approaches on the test set.*

Method	STOI	PESQ
Mixture	0.576	1.317
SEGAN	0.586	1.303
cGAN for SE	0.590	1.220
PSA-BLSTM	0.646	1.820
Wave-U-Net	0.654	1.584
UNetGAN	**0.802**	**2.140**

Experiment

Result and discussion



- UnetGAN은 STOI 및 PESQ와 서로 다른 snr에서의 기준 접근 방식
- UnetGAN이 모든 SNR에서 훨씬 우수한 성능보임
- SNR이 매우 낮을 때, 강한 robust를 보인다.

Thank you