Deep Learning-based Speech Enhancement

Paul Aimé,
Antoine Bertrand

3A SICOM EEH

January 22, 2020

Plan

Jeu de parole

- *TIMIT Speech Corpus* [Abdelaziz, 2017].
- 630 locuteurs des huit dialectes majeurs de l'anglais américain,
- Lecture de phrases phonétiquement riches de $\sim$ 3 secondes.
- Ratio d'entraînement (train - val) = (87,8% - 12,2%)

	Train Set	Validation Set	Test Set
Nombre	4056	564	1680
Pourcentage	64,4%	9,0%	26,6%

Table 1: Répartition des enregistrements de parole dans les différents jeux de données.

Jeu de bruit

- Fichier .wav de 3min30
- Bruit d'ambiance de voix humaines
- Ratio d'entraînement (train - val) = (87,8% - 12,2%)

	babble_train.wav	babble_val.wav	babble_test.wav
Tronçon	0:00 - 3:00	3:00 - 3:25	3:25 - 3:55
Durée	180s	25s	30s
Pourcentage	76,6%	10,6%	12,8%

Table 2: Découpage du fichier de bruit.

Mixage

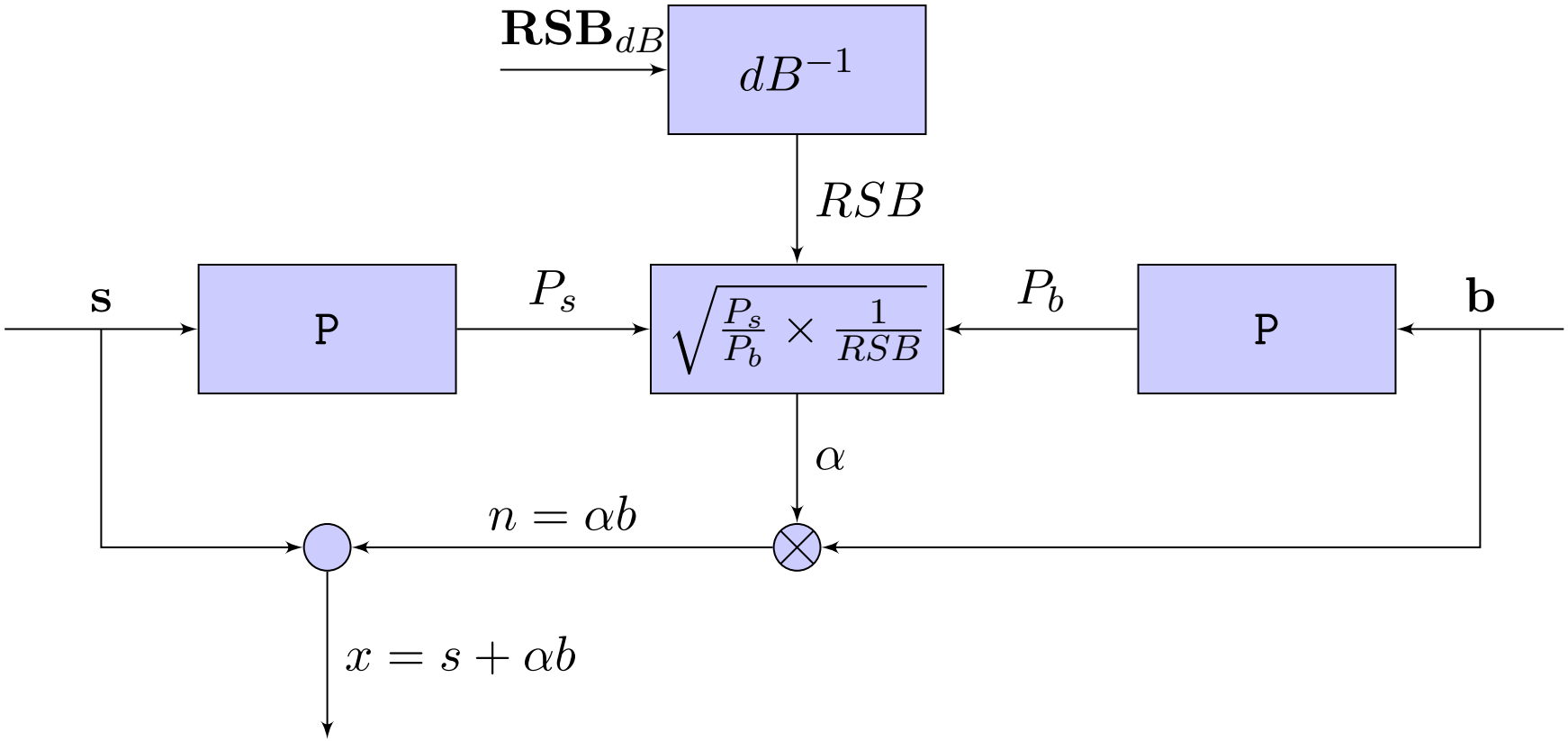
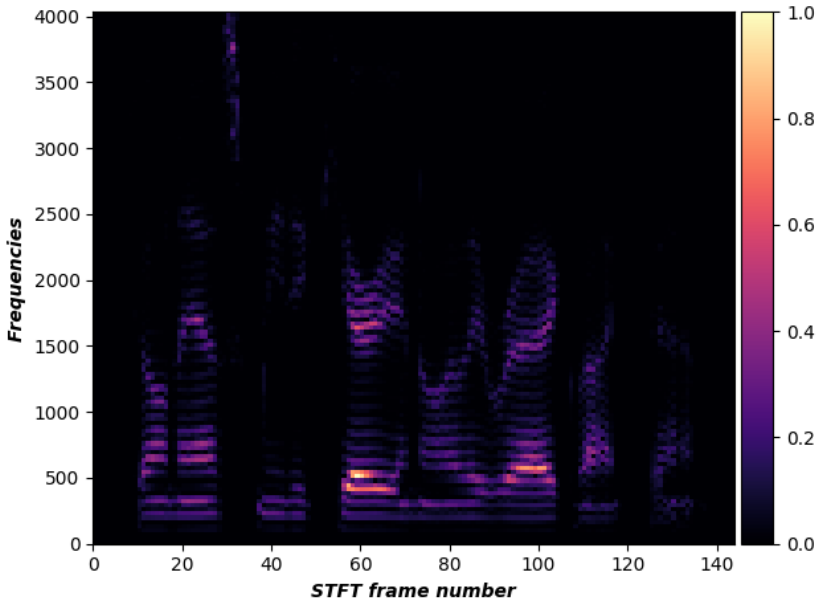


Figure 1: Schéma bloc de la fonction de mixage `add_noise_snr(signal, noise, snr_dB)`

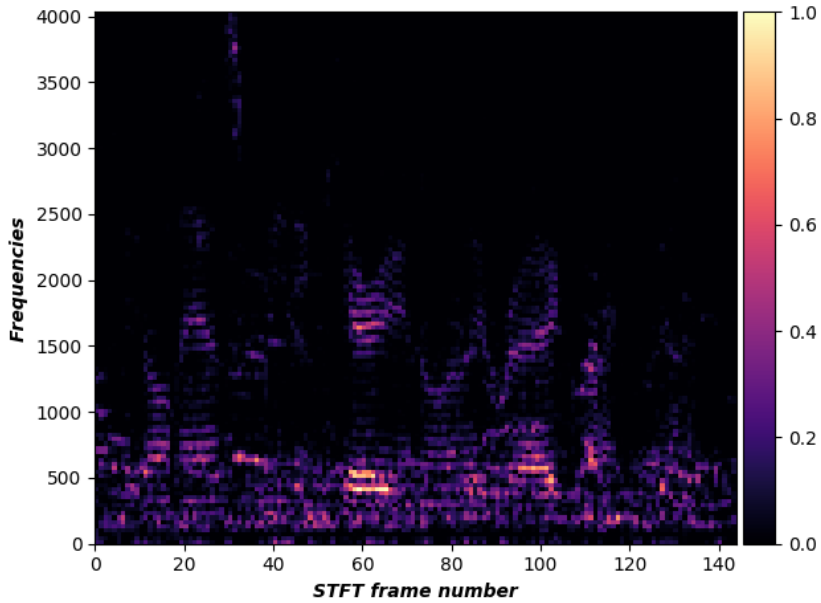
STFT

fréquence : 8 kHz
nfft : 256 (32 ms)
overlap : 50%

apodisation : hanning
centrage : oui
padding : réflection



(a) Clean



(b) Noisy

Figure 2: Exemples de STFTs de signal d'origine et de signal bruité.

Plan

Encodeur-Décodeur Convolutionnel

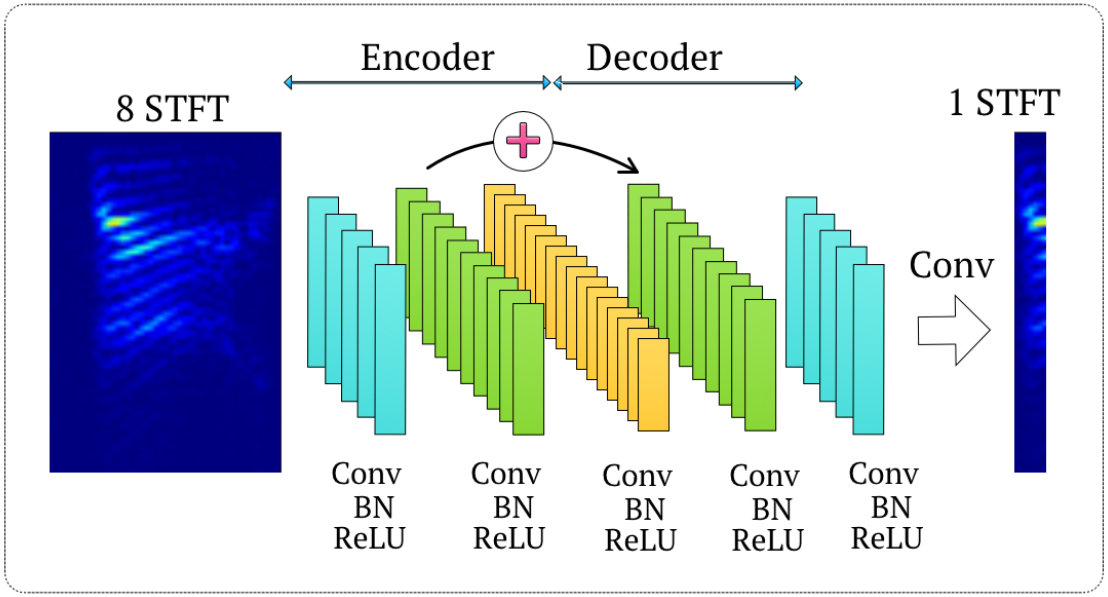


Figure 3: Architecture du réseau R-CED proposé dans [Park and Lee, 2017].
Figure issue de l'article correspondant.

Paramètres d'architecture

Entrée : $(H, W) = (129, 7)$

Durée : $32 + 6 \times 16 = 128$ ms[1]

Nb paramètres : 32611

	Encoder				Latent	Decoder				Out
Layer	1	2	3	4	5	6	7	8	9	10
(in, out)	(1, 12)	(12, 16)	(16, 20)	(20, 24)	(24, 32)	(32, 24)	(24, 20)	(20, 16)	(16, 12)	(12, 1)
kernel size	(13, W=7)	(11, 1)	(9, 1)	(7, 1)	(7, 1)	(7, 1)	(9, 1)	(11, 1)	(13, 1)	(H=129, 1)

Table 3: Paramètres de *feature maps* et de taille du noyau de convolution pour chaque couche du modèle.

[1] [Park and Lee, 2017] utilise 88ms.

Construction du batch

Entrée : $(H, W) = (129, 7)$

Taille du batch : un fichier son

Saut entre les entrées : 1

Entrées par batch : N, le nombre de frame de la STFT du son (3s $\rightarrow$ 187)

Sortie : $(N, H, 1) \longrightarrow (N, H)$

Plan

Procédure

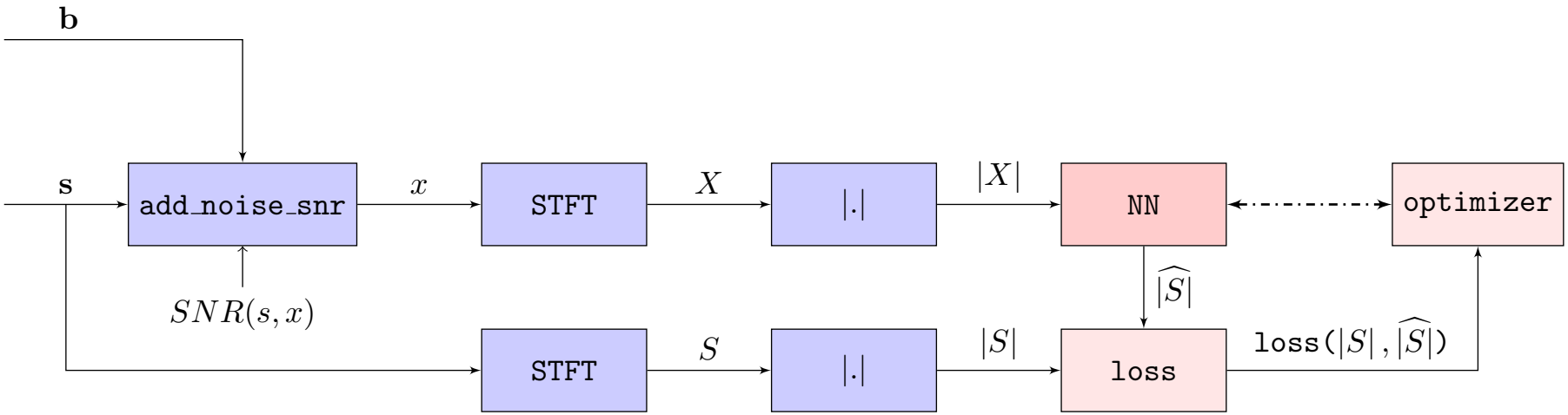
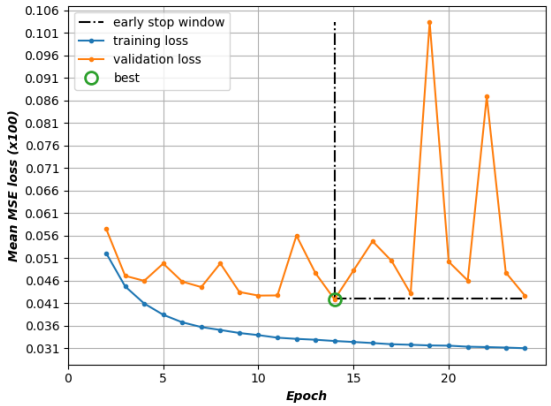
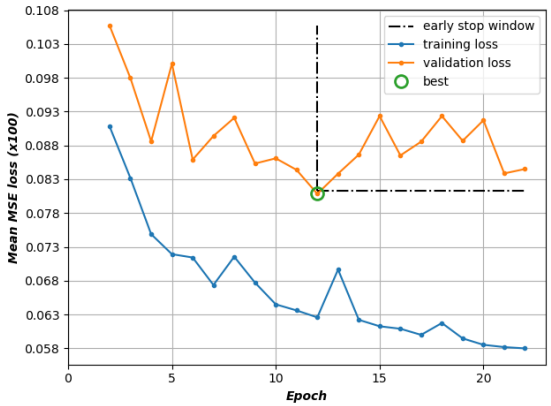


Figure 4: Schéma bloc de la procédure d'apprentissage.

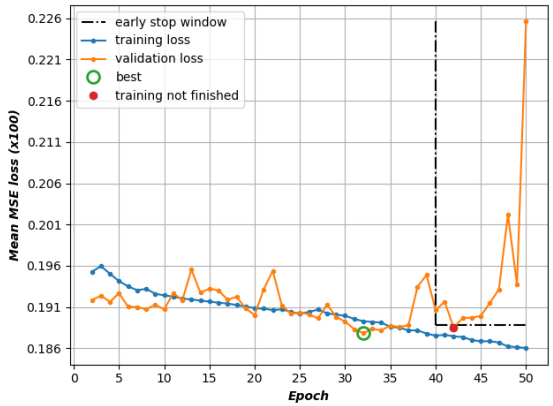
Évaluation de l'apprentissage



(a) $RSB = 0dB$



(b) $RSB = -5dB$



(c) $RSB = -20dB$

Figure 5: Évolution des erreurs quadratiques moyennes (EQM) d'entraînement, pour des modèles entraînés sur des jeux de données à différents niveau de RSB. La première époque n'est pas représentée.

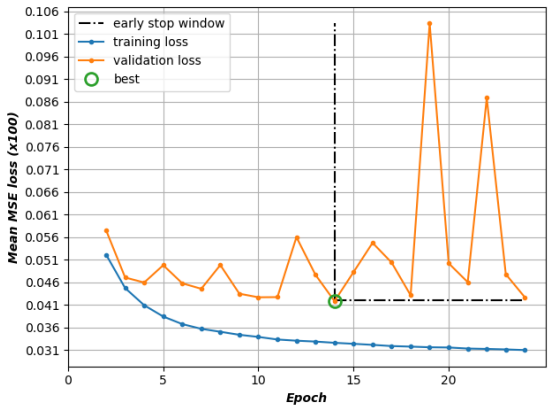
Évaluation de l'apprentissage

Librairie : PyTorch [Paszke et al., 2019].

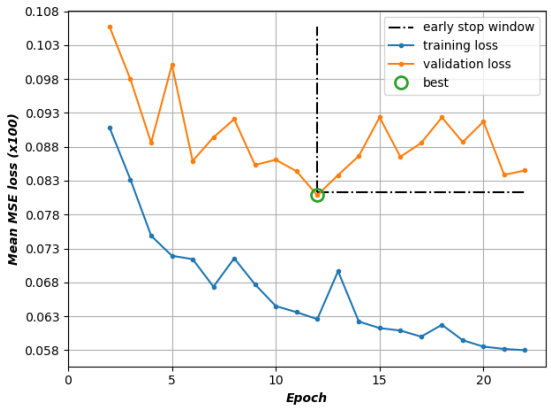
Machine : GCP Compute Engine `n1-highmem-8`

Puissance : 8 vCPU et 1 GPU NVIDIA Tesla P4.

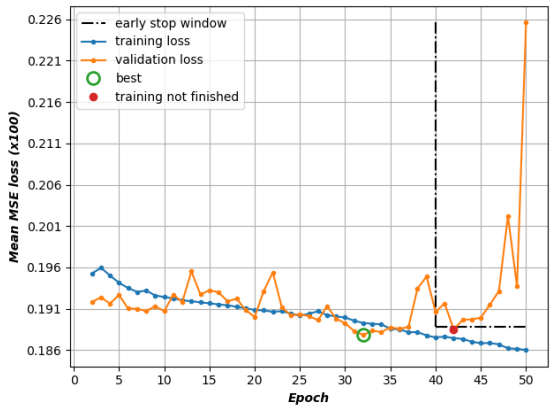
Durée d'une époque : 6 min 15 + 15 sec (50 époques = 6 heures)



(a) $RSB = 0dB$



(b) $RSB = -5dB$



(c) $RSB = -20dB$

Figure 5: Évolution des erreurs quadratiques moyennes (EQM) d'entraînement, pour des modèles entraînés sur des jeux de données à différents niveau de RSB. La première époque n'est pas représentée.

Plan

Procédure

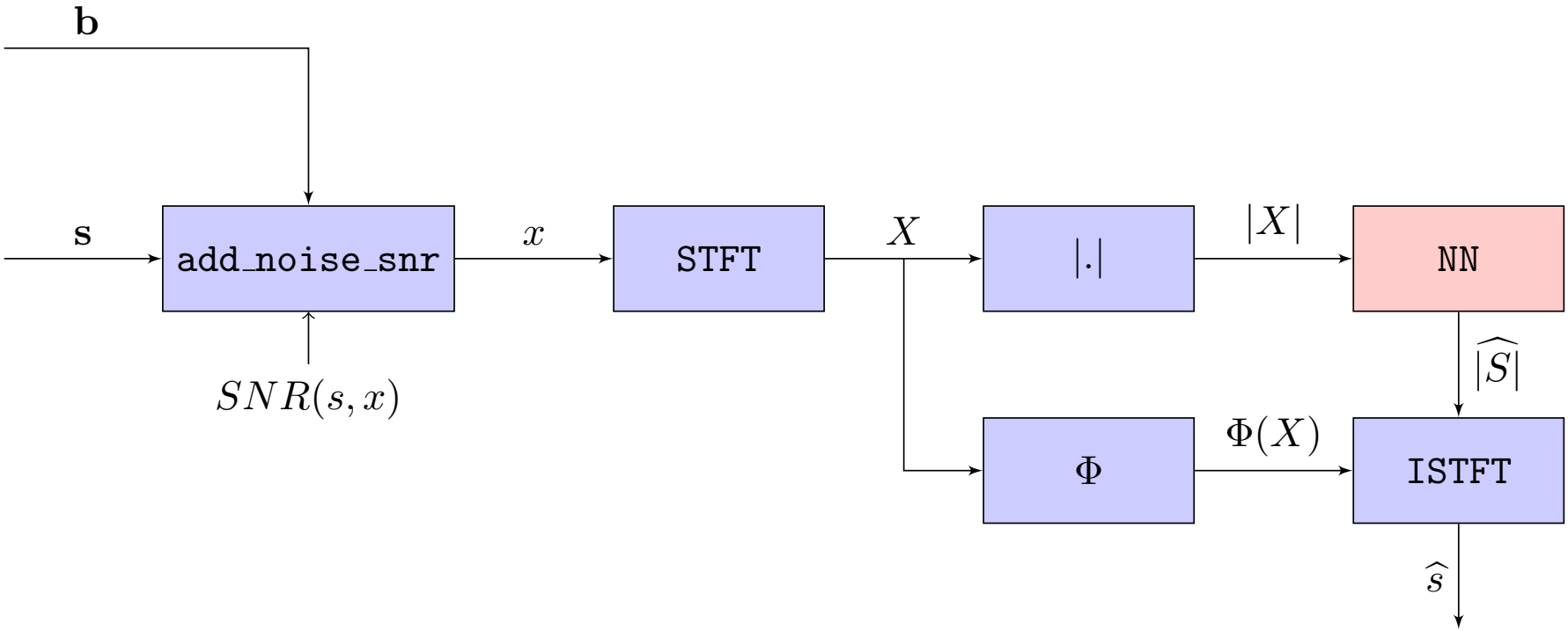


Figure 6: Schéma bloc de la procédure de débruitage.

Reconstruction du signal audio

RSB **sans** étape de normalisation: 140.33 dB

RSB **avec** étape de normalisation: 2.76 dB

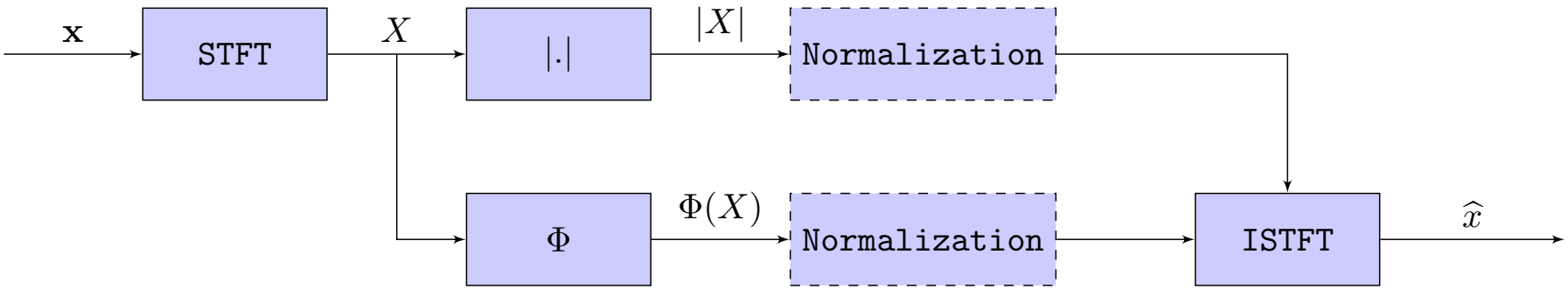


Figure 7: Schéma bloc d'une procédure de reconstruction simple.

Plan

Évaluation par comparaison des spectrogrammes

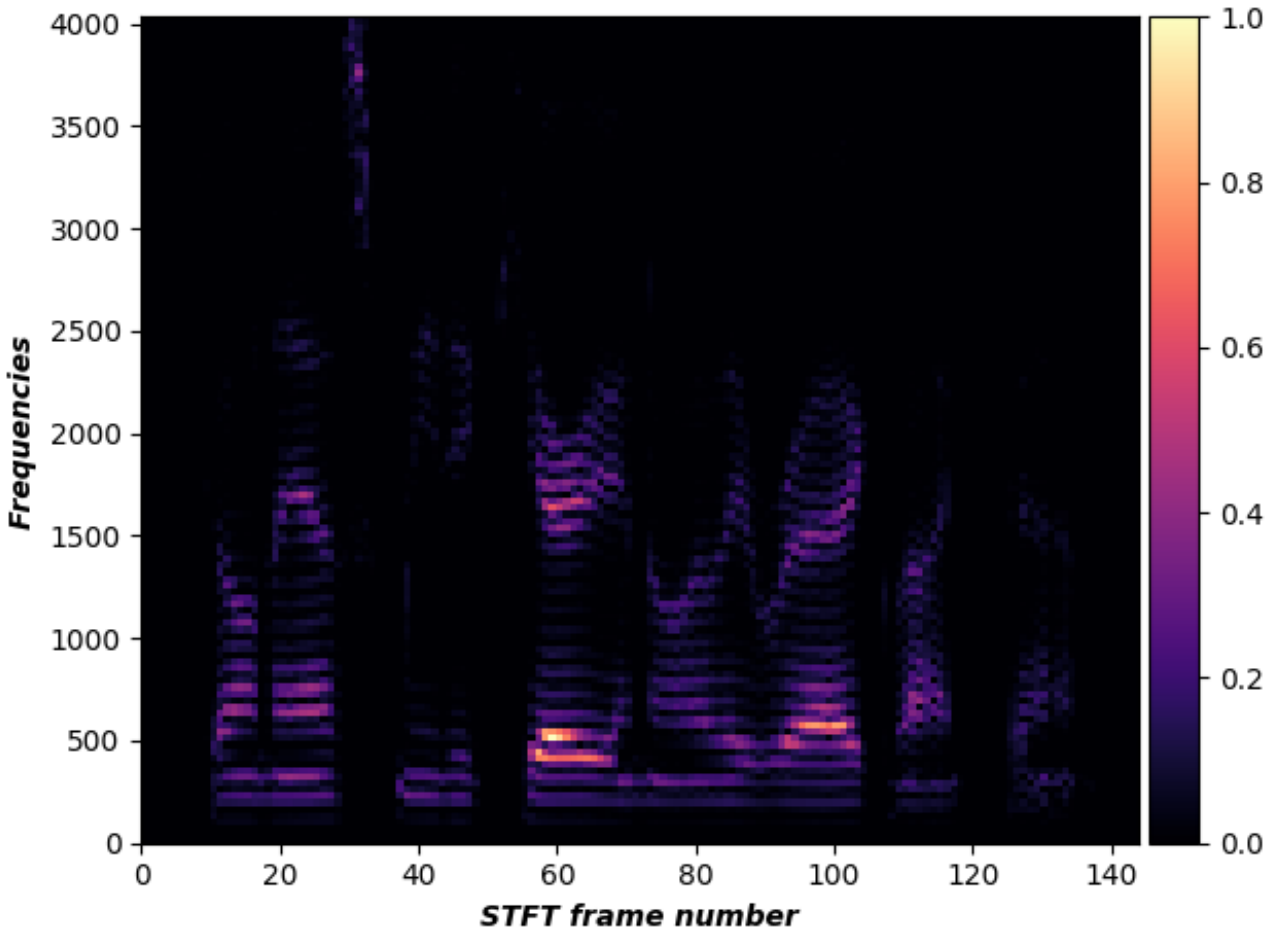


Figure 8: STFT du signal d'origine non bruité

Évaluation par comparaison des spectrogrammes

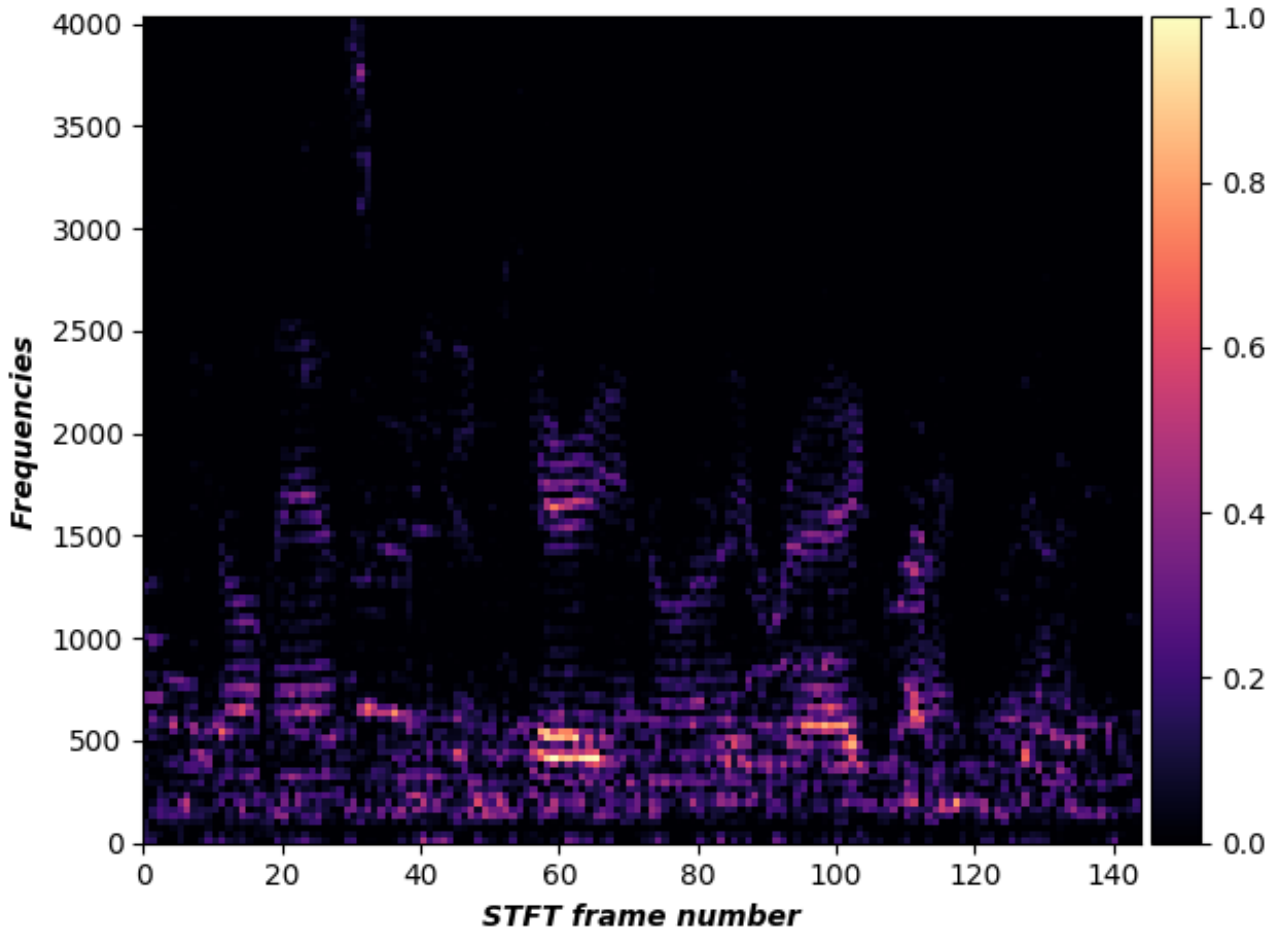


Figure 8: STFT du signal bruité

Évaluation par comparaison des spectrogrammes

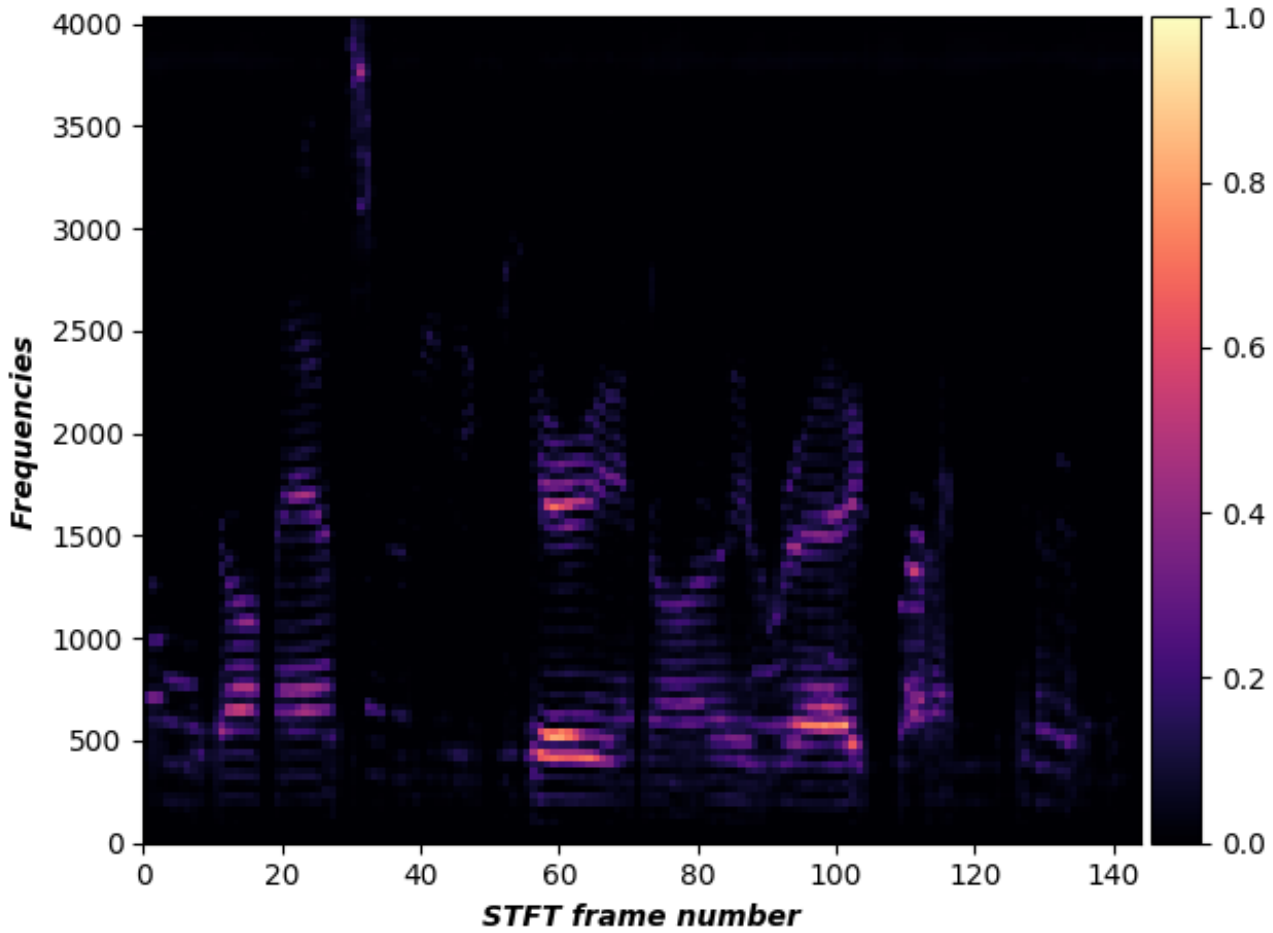
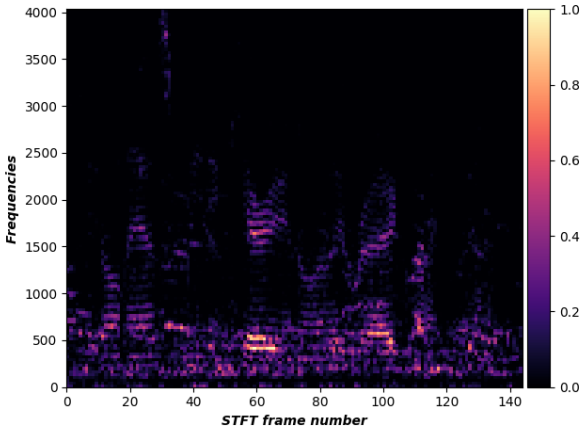
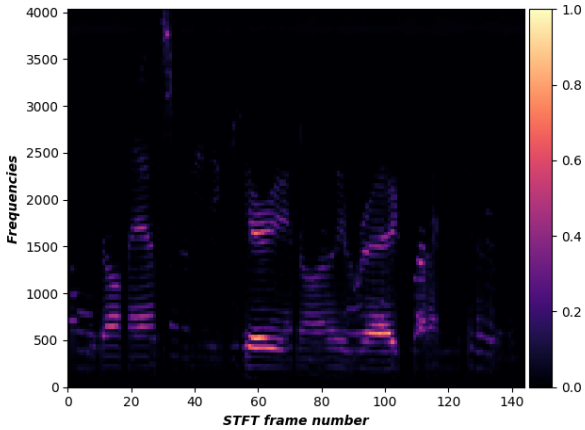


Figure 8: STFT débruitée prédite

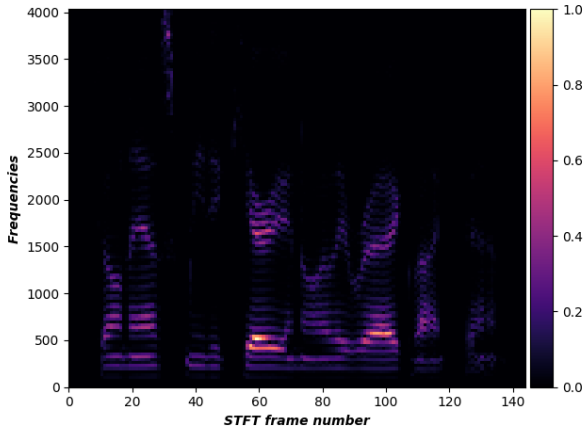
Évaluation par comparaison des spectrogrammes



(a) STFT du signal bruité



(b) STFT débruitée prédite



(c) STFT du signal d'origine non bruité

Figure 8: Visualisation des spectrogrammes d'un signal avant, pendant, et après débruitage.
(SNR d'entrée = 0dB)

Évaluation par comparaison des spectrogrammes

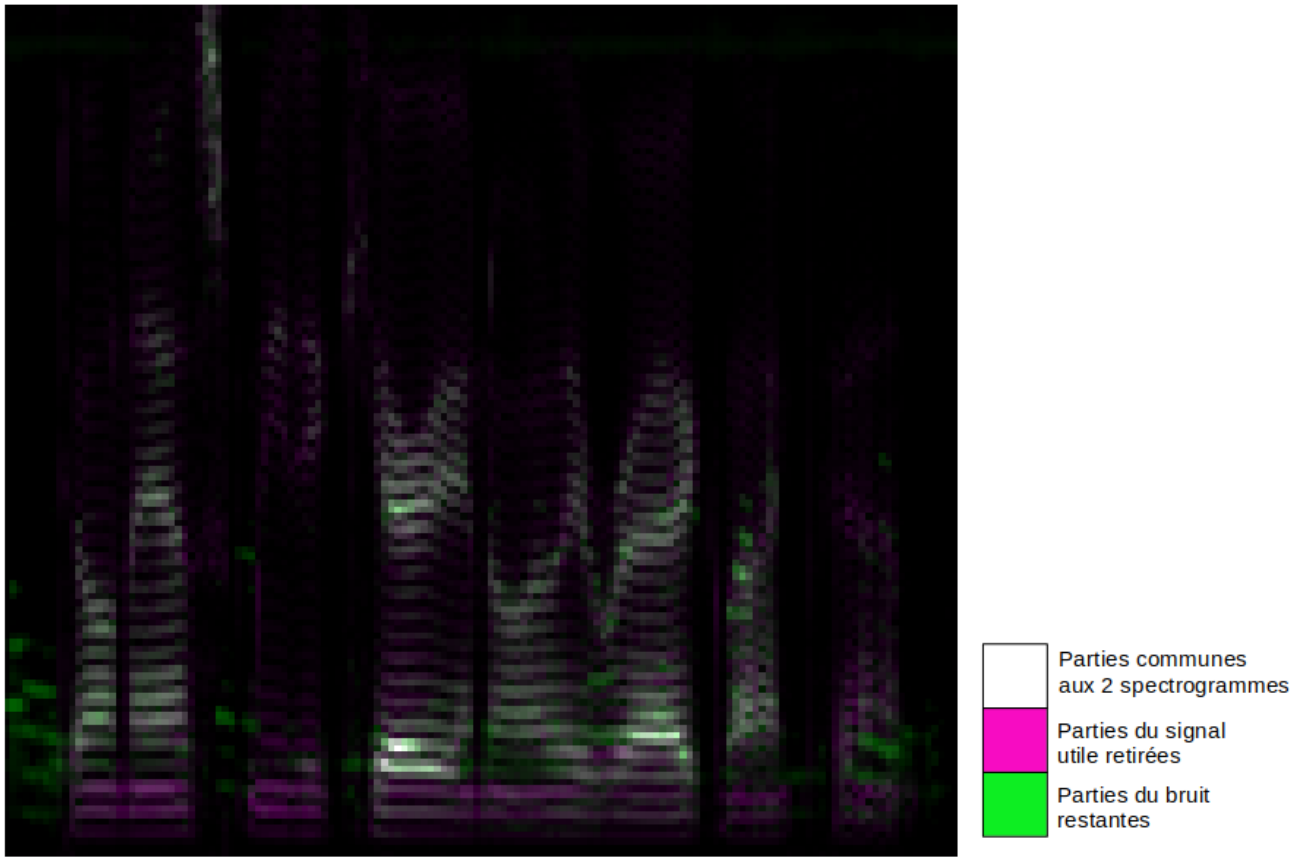


Figure 9: Visualisation des différences entre les spectrogrammes du signal non bruité et le spectrogramme débruité prédit.

Évaluation du gain en RSB

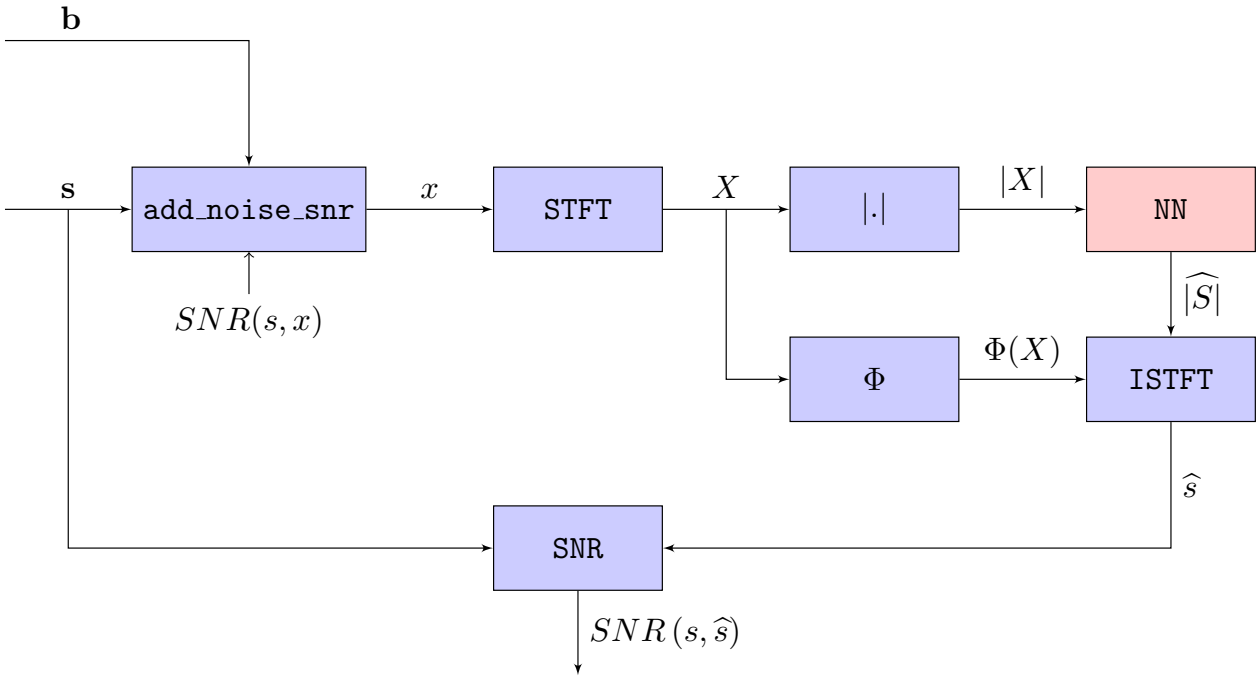


Figure 10: Schéma bloc de la procédure d'évaluation.

Input SNR (clean vs noisy)	**-20dB**	**-10dB**	**-5dB**	**0dB**
Mean SNR (clean vs pred)	1.859163	1.859163	1.859163	1.859163
STD SNR (clean vs pred)	0.734523	0.734523	0.734523	0.734523

Table 4: Statistiques calculées sur les valeurs de SNR des sons du jeu de test.

Évaluation subjective par écoute

Fin

References I

Abdelaziz, A. H. (2017).

Ntcd-timit: A new database and baseline for noise-robust audio-visual speech recognition.

In *Proc. Interspeech 2017*, pages 3752–3756.

Park, S. R. and Lee, J. W. (2017).

A fully convolutional neural network for speech enhancement.

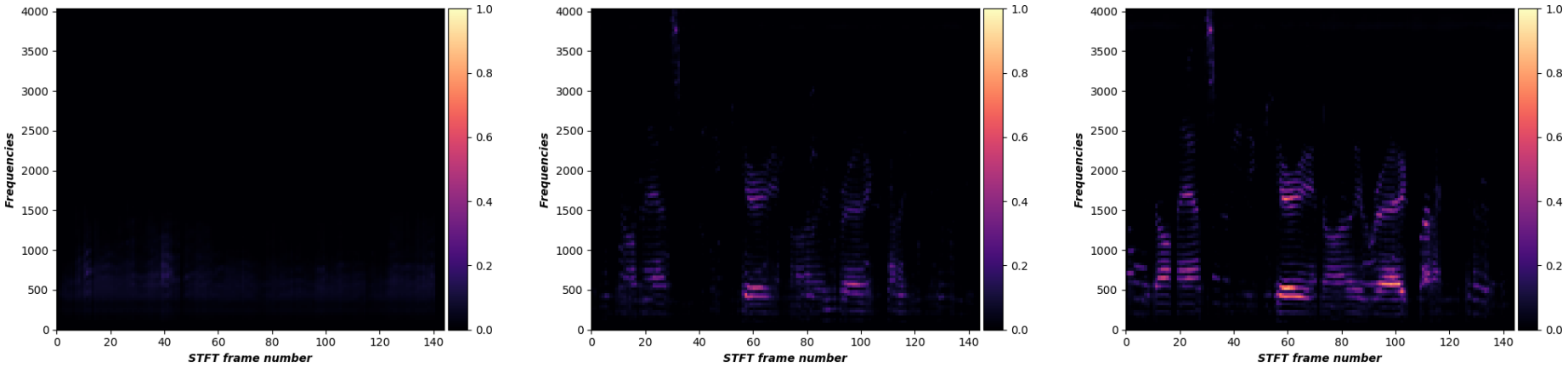
In *Proc. Interspeech 2017*, pages 1993–1997.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019).

Pytorch: An imperative style, high-performance deep learning library.

In Wallach, H., Larochelle, H., Beygelzimer, A., dAlché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

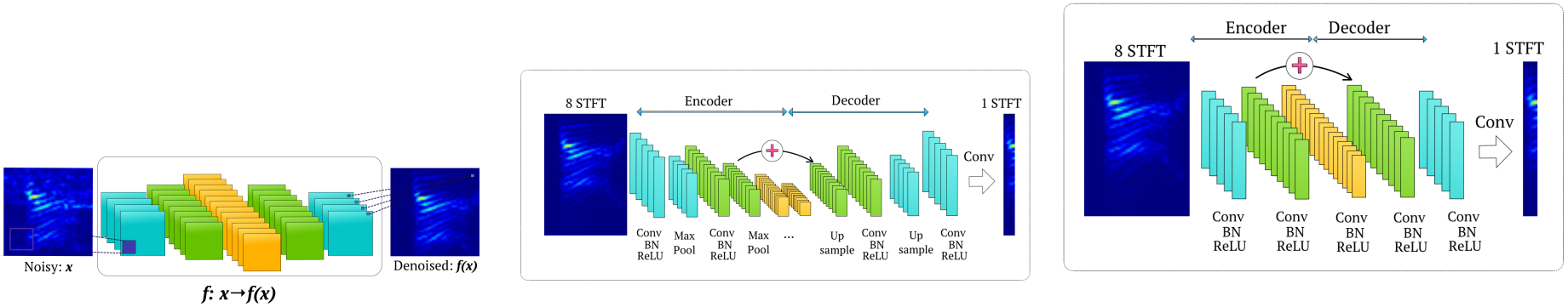
Annexe 1: Spectrogrammes prédits pour différents RSB d'entrée



(a) SNR d'entrée = -20dB (b) SNR d'entrée = -5dB (c) SNR d'entrée = 0dB

Figure 11: Spectrogrammes d'un même signal débruité, avec des niveaux de bruit différents

Annexe 2: Différentes architectures



(a) (b) (c)

Figure 12: Différentes architectures