Apprentissage de bancs de filtres pour la séparation aveugle de sources sonores

Félix MATHIEU[1,2], Thomas COURTAT[2], Gaël RICHARD[1], Geoffroy PEETERS[1]

[1]LTCI, Télécom Paris, IP-Paris
19 Place Marguerite Perey, 91120 Palaiseau, France

[2]Thales SIX, advanced studies AI Lab
11 Av. Augustin Fresnel, 91120 Palaiseau, France

felix.mathieu@telecom-paris.fr, thomas.courtat@thalesgroup.com,
gael.richard@telecom-paris.fr, geoffroy.peeters@telecom-paris.fr

Résumé – L'utilisation d'encodeurs audio paramétrés s'est révélée être une piste encourageante pour améliorer l'interprétabilité et les performances des modèles de séparation de sources bout-à-bout. Nous présentons des propriétés d'intérêt nécessaires à l'apprentissage des filtres de ces encodeurs ; et proposons une paramétrisation pour contraindre ces filtres. Sur la base de la transformée de Hilbert et du théorème de Bedrosian, nous proposons de construire un ensemble de filtres déphasés en modulant des sinusoïdes à travers des filtres passe-bas appris librement. Ces filtres permettent d'obtenir des invariances pour des décalages temporels, des décalages de phases tout en évitant l'utilisation de réseaux de neurones complexes grâce à une astuce de sur-paramétrisation de la phase pour une forme d'onde donnée.

Abstract – The use of parameterized audio encoders has proven to be an encouraging avenue for improving the interpretability and performance of end-to-end source separation models. We present properties of interest needed to learn the filters of these encoders ; and propose a parameterization to constrain these filters. Based on the Hilbert transform and the Bedrosian theorem, we propose to construct a set of phase-shifted filters by modulating sinusoids through freely learned low-pass filters. These filters allow to obtain invariances for time shifts, phase shifts while avoiding the use of complex neural networks thanks to a trick of over-parameterization of the phase for a given waveform.

1 Introduction

Ces dernières années, l'apprentissage profond s'est révélé particulièrement efficace pour traiter diverses tâches liées aux signaux audio, aussi variées que la classification des signaux, la Reconnaissance Automatique de la Parole (RAP) [1], la synthèse des signaux [2] ou les problèmes de reconstruction tels que la Séparation Aveugle de Sources (SAS) [3]. Dans cet article, nous étudions l'apprentissage de bases de filtres pour la SAS par masquage. La SAS par masquage [10] consiste à déterminer, pour chaque point de la projection, la proportion correspondante à chacune des sources à séparer. Pour la SAS, il existe principalement deux approches se distinguant par le prétraitement initial du signal audio (partie effectuée avant l'utilisation du Réseau de Neurones profonds (RN)).

La première approche consiste à encoder le signal dans un espace à deux dimensions à partir de transformations comme la Transformée de Fourier Court Terme (TFCT) [3, 4, 5] ; elle est communément appelée approche spectrale.

La seconde approche consiste à donner directement le signal brut au RN ; approche communément appelée temporelle [6].

L'apprentissage de bases de filtres dépasse le cadre de la SAS. Il a d'abord été étudié pour des tâches de classification et notamment la reconnaissance de parole [9, 8]. L'architecture SincNet [8] a été un véritable moteur dans l'évolution de cette problématique. SincNet propose de paramétrer les filtres par les fréquences de coupures de filtres passe-bandes rectangulaires. Dans le cadre de la SAS par masquage, Pariente et al. [11] proposent l'utilisation de la Transformée de Hilbert qui permet d'obtenir un banc de filtres ayant de bonne propriété d'invariance-par-translation-temporelle.

Les approches spectrales et temporelles sont en fait identiques si l'on considère que l'on projette le signal en utilisant un banc de filtres - fixé pour le premier cas (des sinusoïdes complexes pour la TFCT) ou - appris dans le second.

Objectifs et structure. L'objectif de cette article est de présenter des propriétés d'intérêt à incorporer sous forme de contraintes lors de l'apprentissage de bancs de filtres pour des tâches de SAS tout en maximisant la liberté de ces filtres. Pour ce faire, on présentera dans un premier temps les méthodes classiques de masquage pour la séparation de sources et nous discuterons de la structure des filtres obtenus à l'aide de bases apprises librement. Dans une deuxième partie, on présentera les différentes paramétrisations proposées jusqu'à aujourd'hui et ce qu'elles impliquent dans la structure de ces filtres. Enfin on présentera des idées de filtres à utiliser afin de maximiser la liberté d'apprentissage des banc de filtres tout en les contraignant à avoir les propriétés d'intérêt dégagées dans les parties précédentes.

2 Méthodes de masques

Pour un mélange $\mathbf{x} \in \mathbf{R}^T$où T est la longueur du signal, composé de C sources $(\mathbf{p}_i)_{i=1,\dots,C}$ tel que $\mathbf{x} = \sum_{i=1}^{C} \mathbf{p}_i$, les différentes étapes d'un algorithme de masquage sont :
(a) l'encodage f_e : le signal $\mathbf{x}$ est projeté dans un espace à deux dimensions : $f_e(\mathbf{x}) = X \in \mathbb{R}^{K \times N}$ où N est le temps sous-échantillonné et K la dimension de la projection égale au nombre de filtres.
(b) le masquage f_s : C masques strictement positifs sont construits, pour chaque source on a : $f_s(X) = M \in \mathbb{R}^{C \times K \times N}$;
(c) le décodage f_d : les signaux masqués ($X_i = M_i \odot X$) sont projetés à nouveau dans le domaine temporel : $\hat{\mathbf{p}}_i = f_d(X_i) \in \mathbb{R}^T$ [1].

3 Front-ends basés sur la TFCT

Les premières tentatives de SAS par des méthodes de masquage reposent sur la TFCT. Pour un signal discret $\mathbf{x} \in \mathbf{R}^T$, on rappelle que la TFCT sur K fréquences discrètes est définie par :

$$X(k,n) = \sum_{m=0}^{M-1} x(m) w(m - nR) e^{-j2\pi \frac{k}{K} m}, \qquad (1)$$

où $k \in \{0, \dots K-1\}$ est l'indice des fréquences, $n \in \{0, \dots N-1\}$ ($N = T/R$) est l'indice des trames temporelles, w est une fonction fenêtre de taille M valant 0 en dehors de l'intervalle $[0, M-1]$ et R est le pas d'avancement de la TFCT. En considérant la matrice W formée des filtres complexes $\cos + j \sin$ fenêtrés à différentes fréquences, et pour un pas d'avancement R, la TFCT peut se ré-écrire :

$$X = \mathbf{x} \circledast W, \qquad (2)$$

où $\circledast$ est l'opérateur de convolution. $X(k,n)$ peut se représenter - soit par sa partie réelle et imaginaire (ce qui peut être vue comme la contribution de deux filtres orthogonaux de même fréquence) : $\mathrm{Re}(X) + j\mathrm{Im}(X)$ - soit par sa composante d'amplitude et de phase : $X(k,n) = A_{X_{k,n}} e^{j\phi_{X_{k,n}}}$.
Les premiers algorithmes RN pour la SAS cherchaient à estimer le masque M_i à appliquer au module de la TFCT du mélange A_X afin d'obtenir leur module séparé A_{S_i}. Dans ce cas, la phase $e^{j\phi_X}$ du mélange est généralement directement utilisée :

$$\hat{S}_i = (M_i \odot A_X) e^{j\phi_X} \qquad (3)$$

La reconstruction de la phase de la TFCT des signaux sources a été largement étudiée dans le cadre de la SAS mais implique soit l'estimation de masques complexes [13] soit l'utilisation de méthodes trigonométriques annexes permettant la reconstruction d'une nouvelle phase compatible avec l'amplitude estimée [12]. Cependant ces deux méthodes entraînent généralement l'utilisation d'architectures spécifiques et se généralisent difficilement dans le cadre de l'apprentissage de banc de filtres.

1. $\odot$ correspond à un produit terme à terme

4 Front-end libre

Pour s'affranchir de ces problèmes de phase, Luo et al. [6] proposent pour f_e l'utilisation d'une simple convolution 1D réelle appliquée au signal, suivie d'une fonction d'activation ReLU. Dans ce cas, la phase n'est pas à reconstruire puisqu'elle est encodée implicitement dans les filtres appris (à réponses fréquentielles proches mais décalés en phase) [2]. Un séparateur réel f_s permet ensuite d'obtenir un masque et un décodeur f_d permet de reconstruire le signal. Si le nombre de filtres est suffisant, cela permet de reconstruire parfaitement le signal.
Cette représentation reste l'état de l'art pour le SAS. Depuis, les plus grandes améliorations ont principalement portées sur le masqueur f_s.
Cependant, des travaux sur les front-ends f_e ont montré qu'il est possible d'obtenir de meilleurs performances sur un séparateur f_s donné [14, 15] et que ces encodeurs sont moins sensibles à l'hyper-paramétrisation des réseaux (comme la taille de fenêtre ou le nombre de filtres utilisés dans l'encodeur).

5 Front-ends invariants à la phase

En paramétrant les filtres, il est possible d'aider le réseau de neurones à converger vers des filtres ayant de bonnes propriétés d'invariance à la phase. Ceci est possible en entraînant des filtres de base s_0 et - en les combinant avec leur transformée de Hilbert (l'amplitude du signal analytique étant invariante à la phase) [11] (partie 5.1), - en sur-paramétrisant explicitement la phase) [15] (partie 5.2), ou - en sur-paramétrisant implicitement en étendant la transformée de Hilbert (partie 5.3).

5.1 Transformée de Hilbert

C'est dans cette optique d'ajouter une invariance à la phase et aux légers décalages temporels que Pariente et al. [11] ont proposé d'apprendre librement la première moitié du banc de filtres s_0 et de construire la seconde moitié en prenant la transformée de Hilbert de la première, telle que pour un filtre s_0 on ait :

$$\mathcal{H}(S_0(f)) = \begin{cases} S_0(f).e^{j\pi/2} & \text{si } f > 0, \\ S_0(f).e^{-j\pi/2} & \text{si } f < 0, \\ 0 & \text{sinon.} \end{cases} \qquad (4)$$

Chaque paire de filtres peut ainsi être regroupée en un filtre complexe analytique :

$$\tilde{s}_0(t) = s_0(t) + j\mathcal{H}(s_0)(t), \qquad (5)$$

$$\tilde{s}_0(t) = A_0(t) e^{i\phi_0(t)}, \qquad (6)$$

La paire de filtres s_0 et $\mathcal{H}(s_0)$ appris assure une réponse en amplitude invariante à la phase du signal d'entrée x. En effet,

2. Dans l'article Conv-TasNet [14] Luo et al. constatent déjà que les filtres appris dans l'encodeur convergent vers des filtres très bien localisés en fréquences. Et, que pour une réponse fréquentielle donnée, des filtres avec différents décalages de phases semblent émerger de l'apprentissage.

quelle que soit la phase ψ choisie, la norme $||\langle \Re(\tilde{x}e^{j\psi})|\tilde{s}_0\rangle||$ reste constante. Ici, $\tilde{x}$ correspond au signal analytique associé à x et $\Re(\tilde{x}e^{j\psi})$ correspond à une version de x décalée en phase de ψ.
Cette transformation ne permet pourtant pas d'obtenir une invariance à de légers décalages temporels. En effet, si la partie modulante $A_0(t)$ du filtre $\tilde{s_0}(t)$ contient des fréquences dominantes trop grandes alors un léger décalage temporel peut entraîner une réponse en amplitude très différente.
Plus précisément, on peut décomposer un filtre s_0 et sa transformée de Hilbert de la manière suivante :

$$s_0(t) = \sum_{k=0}^{K-1} a_k \cos(2\pi f_k t + \psi_k), \tag{7}$$

$$\mathcal{H}(s_0)(t) = \sum_{k=0}^{K-1} a_k \sin(2\pi f_k t + \psi_k), \tag{8}$$

on peut donc décrire sa partie modulante A_0 telle que :

$$A_0(t) = \sqrt{s(t)^2 + \mathcal{H}(s)(t)^2}, \tag{9}$$

$$A_0(t) = \sqrt{\Big(\sum_k a_k \cos\big(2\pi f_k t + \psi_k\big)\Big)^2 + \Big(\sum_k a_k \sin\big(2\pi f_k t + \psi_k\big)\Big)^2}, \tag{10}$$

$$A_0(t) = \sqrt{\sum_k a_k^2 + \sum_{k_1,k_2|k_1\neq k_2} a_{k_1}a_{k_2}\cos\big(2\pi(f_{k_1} - f_{k_2})t + (\psi_{k_1} - \psi_{k_2})\big)}. \tag{11}$$

On constate que la transformée de Fourier de $A_0(t)$ peut être principalement décomposée en fréquences $f_{k_1} - f_{k_2}$ (les autres composantes fréquentielles auront une amplitude beaucoup plus faible dans le domaine de Fourier). Ainsi, le fait d'avoir des fréquences prépondérantes trop éloignées (f_{k_1} et f_{k_2}) dans le filtre de base $s_0(t)$ conduit à un comportement oscillatoire important dans sa partie modulante. Cette présence de hautes fréquences dans l'enveloppe du filtre peut entraîner des réponses en amplitudes différentes pour de petits décalages temporels. Idéalement, il faudrait ajouter aux filtres une contrainte sur leurs supports fréquentiels pour obtenir immanquablement une invariance aux petits décalages temporels.

5.2 Sur-paramétrisation explicite de la phase

Une autre idée de paramétrisation de filtre est celle proposée par Ditter et al. [15]. Ils proposent d'utiliser pour f_e des filtres "non appris" de type Gammatone tels que chaque filtre γ_k s'exprime :

$$\gamma_k(t|a,b,n,f_0,\psi_i) = at^{n-1}e^{-2\pi bt}\cdot\cos(2\pi f_0 t + \psi_i) \tag{12}$$

À la différence des fonctions cosinus/sinus de Fourier, les fonctions sont ici modulées en amplitude. À l'inverse de la projection sur fonctions orthogonales de Fourier, les décalages de phases ψ_i sont ici modelisés explicitement. Pour un tuple (a,b,n,f_0), le nombre de décalages de phase ψ_i est choisi en fonction de la fréquence f_0 de telle sorte que l'on ait des filtres décrivant un maximum de décalages de phase à mesure que f_0 est bas. Ainsi, on peut représenter précisément la phase (et donc la position) de la forme d'onde sans avoir à passer par une représentation complexe.

5.3 Sur-paramétrisation implicite de la phase

On peut généraliser cette surparamétrisation à un filtre quelconque s_0 en calculant une transformée de Hilbert "étendue" $\mathcal{H}_\psi$, que nous définissons par :

$$\mathcal{H}_\psi(S_0(f)) = \begin{cases} S_0(f).e^{j\psi} & \text{si } f > 0, \\ S_0(f).e^{-j\psi} & \text{si } f < 0, \\ 0 & \text{sinon.} \end{cases} \tag{13}$$

De manière équivalente on peut obtenir ce même décalage de phase ψ pour un filtre analytique $\tilde{s}_0 = A_0(t)e^{j\phi_0(t)}$ en prenant :

$$s_\psi(t) = A_0(t)\Re(e^{j\phi_0(t)}e^{j\psi}), \tag{14}$$

Ainsi, on peut à la fois apprendre librement un banc de filtres quelconque et obtenir la série de filtres de même amplitude fréquentielle mais de phase différente pour une enveloppe A_0 donnée. Néanmoins, cette paramétrisation ne permet toujours pas d'obtenir l'invariance aux petits décalages temporels pour les mêmes raisons que 5.1.

6 Front-ends invariants au décalage temporel

Les filtres basés sur la transformée de Hilbert, la surparamétrisation de la phase explicite ou implicite (transformée de Hilbert étendue) permettent d'obtenir une invariance à la phase. Cependant, dès lors que le filtre modélise également une modulation d'amplitude (partie $A_0(t)$ des filtres libres ou $at^{n-1}e^{-2\pi bt}$ des filtres Gammatone), il est important de considérer également une invariance au décalage de cet enveloppe. Pour cela, il faut contraindre l'enveloppe du filtre à n'être composée que de basses fréquences. L'objectif de cette partie est de construire de tels filtres tout en essayant de garder un potentiel d'apprentissage maximal.

6.1 Théorème de Bedrosian

Pour un filtre quelconque s_0 on cherche donc à construire une enveloppe temporelle $A_0(t)$ basse fréquence (eq.[9]) telle que la partie modulante de

$$s_0(t) = A_0(t)\cos(\phi(t)), \tag{15}$$

n'ait pas de comportement oscillatoire. Cependant, la contrainte (basse fréquence) sur A_0 ne permet pas de garantir que le filtre $\tilde{s} = A_0(t)e^{j\phi(t)}$ soit analytique. Pour garantir l'analycité du filtre nous utilisons le théorème de Bedrosian. Celui-ci nous donne une condition suffisante sur A_0 et $\cos(\phi(t))$ pour garantir l'analycité de $\tilde{s}$: *"la transformée de Hilbert du produit d'un signal passe-bas et d'un signal passe-haut dont les spectres ne se superposent pas est donnée par le produit du signal passe-bas et de la transformée de Hilbert du signal passe-haut"*, soit :

$$\mathcal{H}\big(A_0(t)\cos(\phi(t))\big) = A_0(t)\mathcal{H}\big(\cos(\phi(t))\big). \tag{16}$$

En appliquant cette contrainte sur les supports fréquentiels des parties modulantes et modulées, on peut donc construire un filtre ayant à la fois une invariance à la phase (Hilbert sur $e^{j\phi}$) mais aussi une invariance au décalage temporel (enveloppe A_0 basse fréquence). Cette décomposition permet aussi de bénéficier de la sur-paramétrisation de phase (eq. (14)) en ajoutant un terme ψ dans la partie modulée du filtre.

6.2 Proposition d'implémentation

On propose ici une paramétrisation simple de filtres permettant de respecter toutes les propriétés présentées dans les parties précédentes. Nous proposons ici de prendre pour partie modulée $\cos(\phi_0(t))$ une simple cosinusoïde paramétrée par sa fréquence f_0, $\phi_0(t) = 2\pi f_0 t$. Le paramètre d'amplitude $A_0(t)$ est obtenu en convoluant un filtre libre $a_0(t)$ par un filtre gaussien passe-bas (l'amplitude de sa transformée de Fourier doit être approximativement nul en f_0). L'ensemble du filtre s'écrit :

$$s_\psi(t|A_0, f_0) = A_0(t)\cos\left(2\pi f_0 t + \psi\right), \quad (17)$$

$$\text{avec } A_0(t) = a_0(t) \circledast e^{-(\frac{t}{\sigma_{t_0}})^2}, \quad (18)$$

$$\text{ou en fréquence } \mathcal{F}(A_0)(f) = \mathcal{F}(a_0)(f) * e^{-(\frac{f}{\sigma_{f_0}})^2}, \quad (19)$$

avec σ_{f_0} déterminé par rapport à f_0 (afin de garantir le non recouvrement fréquentiel). Pour chacun des k filtres où $k \in \{0, \dots K-1\}$ filtre, nous entraînons, sa fréquence f_0 et son enveloppe $\mathcal{F}(a_0)(f)$.

7 Conclusion

Dans cet article, nous avons étudié l'apprentissage des filtres de l'encodeur des modèles de masquage pour la SAS. En particulier, nous avons souligné l'importance d'obtenir une invariance à la phase et à l'enveloppe temporelle dans l'encodeur. Nous avons proposé des outils mathématiques permettant la mise en œuvre de telles invariances : la transformée de Hilbert (pour l'invariance à la phase), celle de Hilbert étendue (permettant une sur-paramétrisation de phase) et le théorème de Bedrosian (permettant d'assurer l'obtention de filtres analytiques ayant une partie modulante basse fréquence). Ces différents outils pourront être utile à la construction de nouveaux bancs de filtres pour des tâches de SAS mais aussi pour des tâches où la localisation de la phase est importante comme la localisation d'évènements sonores [3].

Références

[1] S. Kriman and S. Beliaev and B. Ginsburg and J. Huang and O. Kuchaiev and V. Lavrukhin and R. Leary and J. Li and Y. Zhang. *Quartznet : Deep Automatic Speech Recognition with 1D Time-Channel Separable Convolutions.* 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[2] A. van den Oord and S. Dieleman and H. Zen and K. Simonyan and O. Vinyals and A. Graves and N. Kalchbrenner and A. W. Senior and K. Kavukcuoglu. *WaveNet : A Generative Model for Raw Audio.*

[3] J. R. Hershey and Z. Chen and J. Le Roux and S. Watanabe. *Deep clustering : Discriminative embeddings for segmentation and separation.*, 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[4] Z. Chen and Y. Luo and N. Mesgarani. *Deep attractor network for single-microphone speaker separation*. 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[5] H.-S. Choi and J.-H. Kim and J. Huh and A. Kim and J.-W. Ha and K. Lee. *Phase-aware Speech Enhancement with Deep Complex U-Net*. 2018.

[6] Y. Luo and N. Mesgarani. *TaSNet : Time-Domain Audio Separation Network for Real-Time, Single-Channel Speech Separation*. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[7] D. Stoller and S. Ewert and S. Dixon *Wave-U-Net : A Multi-Scale Neural Network for End-to-End Audio Source Separation*. 2018.

[8] M. Ravanelli and Y. Bengio. *Speaker Recognition from Raw Waveform with SincNet*. 2018 IEEE Spoken Language Technology Workshop (SLT).

[9] N. Zeghidour and N. Usunier and I. Kokkinos and T. Schatz and G. Synnaeve and E. Dupoux. *Learning Filterbanks from Raw Speech for Phone Recognition*. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[10] D. Wang. *On Ideal Binary Mask As the Computational Goal of Auditory Scene Analysis*. Speech Separation by Humans and Machines. 2005.

[11] M. Pariente and S. Cornell and A. Deleforge and E. Vincent. *Filterbank Design for End-to-end Speech Separation*. 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

[12] Z.-Q. Wang and K. Tan and D. Wang. *Deep Learning Based Phase Reconstruction for Speaker Separation : A Trigonometric Perspective*, 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[13] Erdogan and Hershey and Watanabe and Le Roux. *Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks*. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[14] Y. Luo and N. Mesgarani. *Conv-TasNet : Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation*, 2019 IEEE/ACM Transactions on Audio, Speech, and Language Processing.

[15] Ditter, David and Gerkmann, Timo. *A Multi-Phase Gammatone Filterbank for Speech Separation Via Tasnet*, 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[16] F. Mathieu, T. Courtat, G. Richard and G. Peeters. *Phase Shifted Bedrosian Filterbank : An Interpretable Audio Front-End for Time-Domain Audio Source Separation* 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

3. On pourra se référer à l'article [16] pour obtenir des résultats sur l'utilisation de ces bancs de filtres sur une tâche de séparation de sources.