

# Improving Zero-Shot Voice Style Transfer via Disentangled Representation Learning

Siyang Yuan, Pengyu Cheng, Ruiyi Zhang, Weituo Hao, Zhe Gan, Lawrence Carin

Le transfert de voix consiste à faire prononcer un contenu source par une voix cible différente. Il repose généralement sur un système d'encodage-décodage où des *embeddings* de contenu et de style sont appris à partir d'exemples et appariés différemment pour générer un nouvel audio. La nouveauté du modèle IDE-VC (*Information Theoretic Disentangled Embedding for Voice Conversion*) est qu'il veille à démêler le style du contenu : il extrait des embeddings sémantiquement purs, distincts, et libres de toute information inutile. Cela permet de lancer une conversion sur des orateurs jamais rencontrés en *train* et de surpasser les deux modèles de l'état-de-l'art en *zero-shot learning* – AUTOVC [2] et Adaln-VC [1] – en naturel et en précision du transfert.

## Théorie de l'information


Pour chaque audio  $x$ , IDE-VC apprend un embedding de contenu  $c$  et un embedding de style  $s$  qui est l'empreinte vocale de l'orateur  $u$ . Sa particularité est qu'il mesure l'information mutuelle  $I(s; c)$  entre eux ; pour assurer leur pureté, il minimise


$$\min \mathcal{L} = I(s; c) - I(x; s, c) = I(s; c) - I(x; c|s) - I(x; s)$$

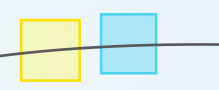
Si  $x$  est un exemple prononcé par  $u$ , alors  $I(u; s) \leq I(x; s)$ , d'où on dérive une borne supérieure plus simple à calculer :

$$\min \bar{\mathcal{L}} = I(s; c) - I(x; c|s) - I(u; s) \geq \mathcal{L}$$

IDE-VC repose sur des estimées des trois termes de la loss. En notant  $M$  le nombre d'orateurs et  $N$  le total des enregistrements, on a :

  $I(u; s) \geq \mathbb{E} \left[ \frac{1}{N} \sum_{u=1}^M \sum_{i=1}^{Nu} \left[ -\|s_{ui} - \mu_u^{(-ui)}\|^2 - \frac{e^{-1}}{N} \sum_{v=1}^M N_v \exp \left( -\|s_{ui} - \mu_v^{(-ui)}\|^2 \right) \right] \right]$   
LOSS DU STYLE VOCAL  $I_1$ . Maximiser l'information mutuelle entre  $s_{ui}$  tiré de  $x$  et l'identité complète de l'orateur  $u$  : rapprocher  $s_{ui}$  de la moyenne des autres embeddings  $\mu_u^{(-ui)}$  pour cette personne, et l'éloigner de la moyenne pour les autres.

$I(x; c|s) = \frac{1}{N} \sum_{u=1}^M \sum_{i=1}^{Nu} \left[ -\|x_{ui} - \text{decode}(c_{ui}, s_u)\|^2 - \log \left( \frac{1}{Nu} \sum_{j=1}^{Nu} \exp \left( -\|x_{uj} - \text{decode}(c_{ui}, s_u)\|^2 \right) \right) \right]$    
LOSS DU CONTENU  $I_2$ . Maximiser l'information mutuelle entre  $x$  et le contenu pour un orateur, de sorte que le décodage sur ce style vocal et ce contenu permette de reconstruire  $x$  et pas un autre enregistrement de cette personne.

  $I(s; c) = \frac{1}{N} \sum_{u=1}^M \sum_{i=1}^{Nu} \left[ \log q_{\theta}(s_{ui} | c_{ui}) - \frac{1}{N} \sum_{v=1}^M \sum_{j=1}^{Nv} \log q_{\theta}(s_{ui} | c_{vj}) \right]$   
LOSS DE DÉMÊLAGE  $I_3$ . Minimiser l'information mutuelle entre style et contenu. Intuitivement, un style vocal doit être moins représentatif du contenu de l'audio associé que de ceux des autres dans la base. Pour l'estimer, on a besoin de la distribution conditionnelle  $s_{ui} | c_{ui}$ , qui est inconnue : on l'approche par une distribution variationnelle  $q_{\theta}$  avec un réseau de neurones séparé.

## Éléments d'architecture

Encodeur 1: embedding orateur.

Encodeur 2 : embedding contenu.

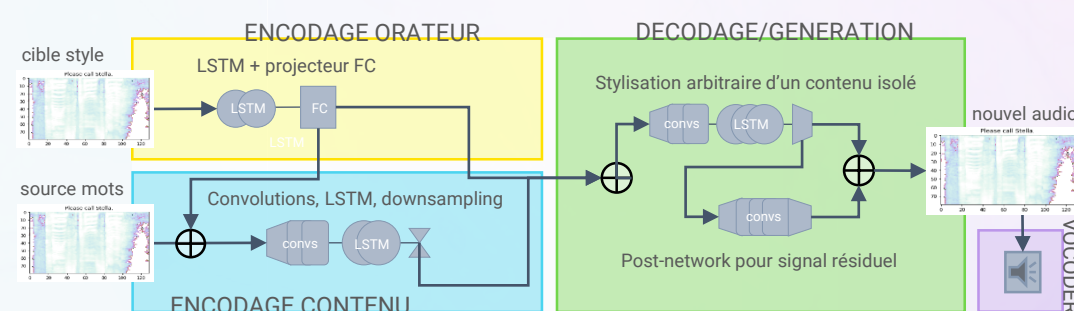
Décodeur : apprend à les assembler pour créer

un nouvel M-SP qui a les caractéristiques

requises. Un post-network affine la génération.

Un vocoder devra convertir la sortie en signal

audible pour l'appréciation humaine.



## Procédure d'entraînement et de génération

Cette structure évite l'entraînement *pairwise*. En train, le même audio est fourni aux deux encodeurs : la loss  $I_2$  optimise le décodage et l'encodage en comparant la sortie avec l'entrée et non avec une hypothétique version réelle de la phrase. En test, l'appariement entre des orateurs source et cible différents sera possible, car la manière correcte d'extraire le contenu et le style vocal aura été apprise. Des métriques (décrites dans la rubrique) permettront de tirer profit de l'existence de données *pairwise* si on a cette chance.

Un entraînement alterné est nécessaire entre l'apprentissage (maximisation de vraisemblance) de la distribution variationnelle  $q_{\theta}$ , dont dépend le démêlage, et celui des éléments du réseau à proprement parler avec la minimisation de la triple loss décrite.

L'ordre de grandeur des epochs n'est pas précisé, pas plus que les hyperparamètres. La puissance de calcul requise est très grande (Nvidia Titan Xp) et l'entraînement reste long. A titre de comparaison, 1500 epochs sur 10 orateurs prenaient deux jours pour AutoVC.

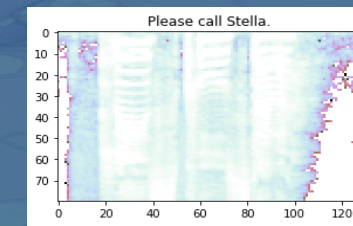
## Motivation

Les modèles de transfert de voix se limitent souvent à un entraînement *pairwise* ; l'orateur cible doit avoir aussi prononcé la phrase source pour calculer une erreur de reconstruction. Le système IDE-VC se concentre plutôt sur l'extraction de *features* des audios qui est applicable partout une fois apprise, de sorte qu'il saura convertir n'importe quel orateur vers n'importe quel autre - vu en entraînement ou non. Le principe est déjà appliqué dans les systèmes AutoVC et Adaln-VC, mais les résultats sont perfectibles.

## Données choisies

IDE-VC est entraîné sur VCTK, le dataset public le plus récent. Il contient 42000 enregistrements courts de 109 hommes et femmes de langue anglaise, et la possibilité du *pairwise* n'y est pas systématique.

Les audios sont présentés au modèle sous forme de mel-spectrogrammes (M-SP), représentation idéale pour la voix et tout ce qui est audible par l'humain.



## Métriques pertinentes

- (1) **Distance** Le M-SP de la phrase générée doit être proche de celui du même contenu prononcé par la cible – modulo un *Dynamic Time Warping*. La métrique Distance requiert donc d'avoir des données *pairwise*.
- (2) **Vérification**. Même sans données *pairwise*, un classifieur d'identité (*Resemblyzer*) peut, en complément, permettre de s'assurer qu'on identifie bien l'orateur cible dans la génération.
- (3) **Confusion** Si le démêlage fonctionne, la vérification ne doit pas être possible avec le code contenu seul. En pratique, on mesure la confusion via le score du *Resemblyzer* sur  $c$ .
- (4) **Naturel** Une métrique subjective notée sur 5.
- (5) **Similarité** Pourcentage subjectif pour juger de la proximité avec l'identité vocale cible.

## Résultats et analyses supplémentaires

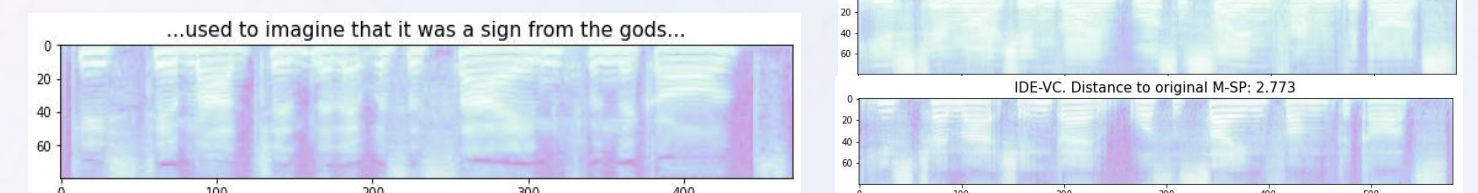
Les baselines Adaln-VC et AutoVC permettent déjà le zero-shot learning. Le partage du principe général et la proximité des architectures prouvent que seule la loss novatrice fondée sur le démêlage justifie la supériorité d'IDE-VC sur les cinq métriques.

Métrique	Distance	Vérification	Confusion	Naturel	Similarité
Objectif	$\ominus$	$\oplus$	$\ominus$	$\oplus$	$\oplus$
Adaln-VC	6.37	76.7	19.0	2.67	68.4
AUTOVC	6.68	60.0	9.5	2.19	58.6
IDE-VC	<b>6.31</b>	<b>81.1</b>	<b>8.1</b>	<b>3.33</b>	<b>76.4</b>

Cet avantage d'IDE-VC est prouvable : sur une t-SNE selon l'orateur, ses embeddings de style sont séparés en clusters, et ceux du contenu ne montrent aucun pattern particulier. C'est bien le but de la minimisation de l'information mutuelle, et cela corrige les défauts des deux autres mis en évidence ici. Pour Adaln-VC, l'isolation des orateurs est mauvaise ; et pour AutoVC, les embeddings de contenu sont répartis selon leur couleur, et contiennent donc des informations sur l'orateur alors qu'ils ne devraient pas.



La visualisation de l'impact requiert un test *pairwise*. Voici une comparaison des mel-spectrogrammes entre la cible théorique ci-dessous et les sorties des trois décodeurs :



Les M-SP en sortie sont plus longs que la cible du fait de la différence de rythme entre les locuteurs étudiés. Cet effet est quasiment systématique. L'intérêt de la métrique Distance et du DTW inclus est de permettre la comparaison de performances malgré cela, et c'est bien IDE-VC qui parvient au meilleur score.

**Critiques** : la conversion vers un speaker jamais vu était tout à fait concluante pour tous les modèles, ce qui est le but du zero-shot learning. Cela dit, les contenus dans une langue étrangère (comme le français) ou du moins jamais vus en *train* sont très mal imités pour les baselines. Il est possible que le démêlage avec IDE-VC réduise ce problème, mais l'article reste très obscur sur l'implémentation.

[1] Chou, J., Yeh, C., Lee, H., "One-shot VC by Separating Speaker and Content Representations with Instance Normalization", 2019.

[2] Qian et al., "AUTOVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss", 2019.

