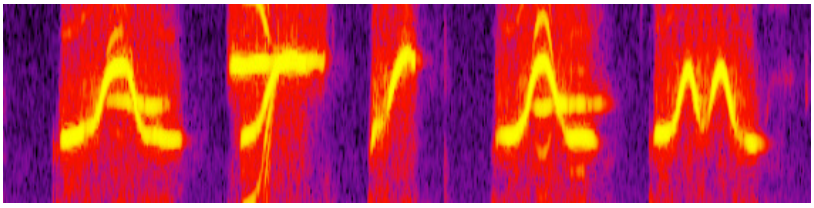
Antoine BONNEFOY
Stage de fin d'études Master ATIAM



Rapport de mi-parcours

Transcription Automatic de la Partie Percussive d'un Morceau de Musique.

Institutions IRCAM, UPMC
Telecom ParisTech

Lieu du stage IRCAM,
Equipe Analyse Synthèse

Encadrants Axel Röbel,
Mathieu Lagrange,
Geoffroy Peeters

Mercredi 8 Août 2012

Sujet du Stage

Introduction :

Un effort conséquent est actuellement mis en œuvre pour extraire de flux musicaux audionumériques des informations de type symbolique (partition). Pour ce faire de nombreux algorithmes sont disponibles pour l'analyse multi-pitch, la reconnaissance d'accords, etc. [1].

Un large panel d'approches est disponible pour ce faire [5], en fonction du type d'à priori disponible. Dans une approche aveugle, on cherche à décrire la partie percussive en fonction de ses propriétés spectro temporelles uniquement. Si on dispose d'informations d'une détection des débuts de notes, on peut alors prendre cette information en compte pour guider la transcription. Ensuite, on peut disposer des formes spectro temporelles à priori des différents instruments percussifs et éventuellement les adapter aux observations [3, 4].

Pour la mise en œuvre du système de transcription incorporant ces propriétés, le cadre de la Factorisation en Matrices Non-Négatives sera privilégié. On considérera également dans ce stage, le fait que nous disposons d'une estimation de la partie batterie (issue d'une séparation bien entendu imparfaite) grâce à un algorithme efficace développé dans l'équipe [6].

Contexte local :

L'équipe Analyse/Synthèse de l'Ircam poursuit des recherches portant sur l'analyse des signaux musicaux à des fins de transformation, de transcription et d'indexation. Le stage proposé sera réalisé dans ce groupe de recherche avec pour interlocuteurs principaux Mathieu Lagrange et Axel Roebel.

Travail à réaliser :

L'objectif du stage est de participer à une évaluation méthodologique de l'état de l'art dans le domaine de la transcription de la partie percussive d'un morceau de musique. Le travail comportera ces étapes :

Bibliographie et sélection des algorithmes de l'état de l'art les plus prometteurs Définition d'un protocole d'évaluation montrant les propriétés des algorithmes en terme de performance de transcription Mise en place des systèmes retenus Evaluation des résultats, identification des points faibles Réflexion sur les améliorations à apporter. Pré-requis : bonne acquisition des UE TSA et TSM ; maîtrise de Matlab et de la programmation impérative ; notions suffisantes en optimisation et algèbre linéaire ; des connaissances générales sur les algorithmes classiques et la méthodologie en modélisation du signal sonore et en séparation de sources seront appréciées.

Apports du stage : contexte applicatif stimulant, mise en œuvre pratique d'algorithmes, recherche pour l'amélioration des algorithmes, travail au sein d'une équipe dynamique. Stage pouvant éventuellement déboucher sur une thèse de doctorat. Stage rémunéré.

Bibliographie

[1] A. Klapuri et M. Davy , Signal processing methods for automatic transcription of music, Springer, New York, 2006. [2] Durrieu J.-L., Richard G. and David B., Singer melody extraction in polyphonic signals using source separation methods, in Proc. of ICASSP 2008. [3] Aymeric Zils, Francois Pachet, Olivier Delerue, Fabien Gouyon Automatic Extraction of Drum Tracks from Polyphonic Music Signals, in Proceedings of the 2nd International Conference on Web Delivering of Music(WedelMusic2002), Darmstadt, Germany, Dec. 9-11, 2002 [4] Kazuyoshi Yoshii , Masataka Goto, Hiroshi G. Okuno, Automatic Drum Sound Description For Real-World Music Using Template Adaptation And Matching Methods, in Ismir 2004 [5] Derry FitzGerald, Automatic Drum Transcription and Source Separation, Phd Thesis 2004 [6] F. Rigaud, M. Lagrange, A. Roebel, G. Peeters Drum Extraction From Polyphonic Music Based on a Spectro-Temporal Model of Percussive Sounds, Proc. of ICASSP, Prague, 2011

Table des matières

Table des figures

Chapitre I

Introduction

I.1 Motivation

La taille des bases de données musicales en contant agrandissement engendre de nouveau problème tant pour les professionnels que pour les particuliers. Comment accéder facilement à ces quantités d'informations ? Il est déja possible d'effectuer une recherche par artiste, titre ou genre grâce aux métadonnées contenus dans certains formats de fichiers (mp3, wma...). Mais les limites de ces métadonnées sont vite atteintes, pour retrouver un morceau il faut déjà connaitre les informations. Comment, par exemple, retrouver le morceau que j'ai dans la tête et que je chantonne depuis ce matin ? Ou comment retrouver cette reprise dub de *The wall* que j'ai entendu hier ? C'est à ces questions que l'indexation audio s'efforce de répondre. Le vrai challenge est bien sur de pouvoir extraire ces informations de haut niveau automatiquement à partir du signal lui même. Ces taches qui paraissent très simple à l'auditeur, même non musicien, peuvent se révéler d'une toute autre difficulté pour un ordinateur.

La description automatique de contenu musical est au cœur de la recherche liée aux "Music Information Retrieval". La norme MPEG-7 a été développée dans le but de normaliser les informations nécessaires à la description des contenus des documents multimédias pour l'indexation. Les méthodes de récupération automatique de contenus musicaux, permettent en effet la construction de bases de données musicales intelligentes non plus uniquement basées sur le titre et l'artiste du morceau mais aussi sur d'autres informations musicales, qui contiennent plus de sens. L'information primordiale qui contiendrait toutes les informations musicales que l'on aimerait idéalement obtenir, est la partition complète du morceau. Récupérer tout ce contenu sémantique est un des défis de l'indexation audio automatique, il s'agit de la transcription automatique.

L'étude de la structure rythmique d'un morceau apporte de nombreuses informations structurelles sur l'œuvres aidant à déterminer son genre, son tempo, une ambiance, un degré de similarité avec un autre morceau, etc... Par sa prédominance dans la musique populaire actuelle, la batterie offre une abondante source d'information rythmique. C'est pourquoi l'étude des motifs rythmiques est un outils robuste pour la détection de genre, effectivement d'un genre à l'autre les phrases de batterie varient beaucoup. Encore faut-il avoir la partition.

C'est donc le but de ce stage de réaliser la transcription automatique de la partie de batterie d'un morceau de musique, plus particulièrement par méthode NMF. Nous partons dans le cadre de ce stage d'un signal préalablement séparé par l'algorithme proposé au sein de l'équipe Analyse-Synthèse par François Rigaud dans [RLRG11]. Nous pourrons donc démarrer par l'étude de la transcription d'un signal de batterie pur : l'oracle, puis tester sur un même morceau séparé grâce à l'algorithme de l'équipe. Nous nous limiterons à transcrire en trois éléments différents Snare, Kick et Hi-hat comme il est fait dans la littérature et car ce sont ces trois éléments les plus utilisés et donc ce sont eux qui sont les plus intéressants du point de vu

sémantique.

Pour ce faire on pourra comparer les méthodes de l'Etat de l'art avec une méthode NMF propre. Nous présenterons donc en première partie une description d'une liste non exhaustive, mais représentative, des méthodes de transcription automatique de batterie. Nous reviendront un peu plus en détail sur la NMF en deuxième partie. Puis en troisième partie sera présenté une manière d'inclure dans la NMF des à priori sur la structure du dictionnaire et sur les fréquences des combinaisons de frappe. Enfin la transcription en elle même, ainsi qu'une réflexion sur les mesures de performance de transcription seront détaillés en quatrième partie. Une analyse des résultats est proposée en dernière partie.

I.2 Les éléments de la batterie

La batterie est incontestablement l'instrument de percussion le plus représenté dans la musique occidentale actuelle. Une batterie est un ensemble de percussions réunis en un même instrument. On appel aussi parfois "un kit" de batterie une des configurations de cet ensemble. Ces éléments constitutifs de la batterie peuvent être séparés en deux familles :

1. Les **membranophones** : constitués d'un fût cylindrique, qui peut être en bois ou en métal, aux extrémités duquel sont tendue deux membranes, originairement des peaux. Il arrive qu'une des deux extrémités ne soit pas fermé pour certains d'entre eux. On compte parmi les membranophones :
   - la *caisse claire* (en : snare drum) : dont le fût a un diamètre compris entre 25 et 35cm, et une profondeur entre 10 et 20cm. La caractéristique de la caisse claire vient de l'ensemble de ressort tendu au contact de la membrane inférieur appelé timbre, qui blanchit le son de la caisse et empêche les résonances.
   - la *grosse caisse* (en : bass drum) : son diamètre est nettement supérieur entre 45 et 65cm, et produit un son plus grave et plus sourd du fait du maillet utilisé pour frapper la peau.
   - les toms sont les seuls éléments de la batterie qui peuvent être accordés, généralement aux nombres de trois. On les distingue donc en *tom aigu* (en : high tom), le *tom medium* (en : medium tom), et le *tom basse* (en : low tom). Il arrive qu'ils ne soient pas fermés par une membrane arrière.
2. les **idiophones** : ce sont les instrument de musique dont le son est produit par le materiel vibrant lui même. Dans la batterie ce sont souvent des disques de métal :
   - Le *charleston* (en : Hi-hat) est constitué de deux cymbales dont on contrôle la position avec le pied. Il y a donc deux sons accessibles l'un fermé les deux cymbales se touchant il y a très peu d'oscillation et le son résultant est très court, l'autre ouvert ou les deux cymbales sont séparées.
   - les cymbales *ride, crash, chinoise, splash* se distinguent par leur diamètre, leur poids et leur dureté, elles produisent des sons variés.
   - D'autre éléments sont souvent ajoutés à la batterie dans des kit plus complets, tels que le *woodblock* ou la *cloche*, dans le kit de batterie considéré dans ce stage la cloche est incluse.

Chaque éléments à un sens musical différents. Par exemple la grosse caisse marque souvent les temps fort tandis que le charleston ou la ride définira le swing du morceau en s'établissant comme une sorte de métronome. La cloche par exemple est souvent utilisée dans les rythmes latinos pour marquer la clave.

Pour certain de ces éléments il existe différentes frappes possibles, qui vont apporter des sons bien différents. Par exemple pour la caisse clair on en compte au moins trois. Le son de base quand la baquette frappe la peau, le *Rim-shot* quand la baguette frappe en même temps la peau et le cerclage qui fait résonner des harmonique plus aigu du fut, moins sensible au *timbre*, ou encore le *Cross-stick* quand la baguette est posée sur la peau et frappe le cerclage, cette technique est particulièrement utilisée dans le reggae. Ceci induit de nombreuses sonorités différentes même en utilisant un même kit de batterie. Mais la variabilité des sons vient aussi du type de baguette utilisée (ballet, fagot) et des choix fait lors de la sonorisation (réverbérations, filtres, compressions...). Ces variabilités font qu'il est illusoire d'avoir un dictionnaire exhaustif contenant tous les sons de batterie.

Figure I.1 – Le Kit de batterie basique

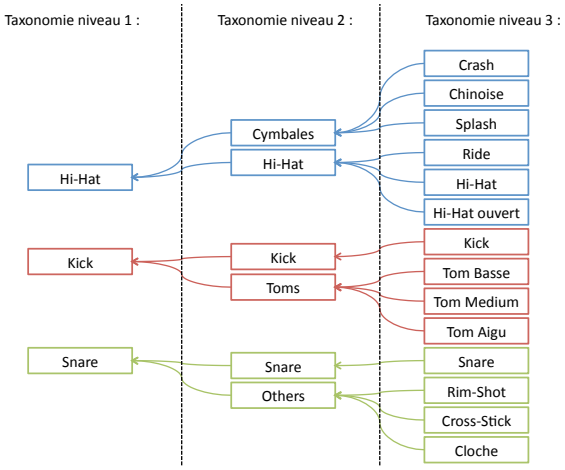


Figure I.2 – Différents niveau hiérarchique avec les taxonomie correspondante.

I.3 Taxonomies Hiérarchiques

Pour décrire les éléments de la batterie on peut tenter de les regrouper. La première taxonomie envisageable est bien sur simplement de prendre chaque élément de la batterie pour constituer une classe, on obtient ainsi une une taxonomie complète. Si l'on considère des regroupements sémantique, en réunissant des classes qui ont des rôles similaire dans la musique, on obtient une sorte de hiérarchie de taxonomie. Ces 3 taxonomies sont présentées dans I.2.

I.4 La transcription recherchée

Le but de ce stage est donc d'obtenir automatiquement à partir d'un signal une transcrition. Qu'est-ce qu'une transcription pour la batterie ? Une transcription peut être définit comme un ensemble d'onset où chaque onset est un couple $\{t_i, c_i\}$ contenant le temps d'apparition d'une frappe t_i et l'élément de la batterie utilisé c_i. C'est cette description, primaire que l'on espère obtenir grâce à la méthode développée. Cette description bien que partielle contient presque toute l'information nécessaire à analyser le rythme d'un morceau d'un point de vue musical. L'unique information manquante serait l'intensité de ces frappes.

Chapitre II

Etat de l'art

II.1 Etat de l'art de la transcription de batterie

Plusieurs méthodes de transcription automatique de batterie ont été proposées dans la littérature, certaines s'appliquent au signaux polyphoniques et d'autres à des signaux monophoniques contenant uniquement des parties de batterie. Ces méthodes peuvent être regroupées en trois approches principales, cette classificaion est proposée par Gillet [Gil07] : Segmenter et reconnaitre II.1.1, Mettre en correspondance et adapter II.1.2, Séparer et détecter II.1.3.

II.1.1 Segmenter et Reconnaitre

Fonctionnement

Le principe de fonctionnement de ces méthodes, qui fonctionnent principalement dans le cas monophonique, est de procéder en deux étapes :

- **Segmenter** le signal en une succession de frappes
- **Reconnaitre** l'instrument ou l'ensemble d'instruments composants le son du segment considéré à partir d'une représentation du timbre.

Segmentation puis Classification

Dans cette approche les deux étapes sont souvent séparées et l'étape de reconnaissance passe souvent par l'extraction de descripteurs (*features*) puis la classification de ces ensembles de descripteurs par des méthodes de classifications usuelles. Par exemple la méthode proposée par Gillet et Richard dans [GR04] se sépare en trois parties : (1) détection d'onset par décomposition en sous-bandes ; (2) extraction de descripteurs sur ces onsets ; (3) classification de ces vecteurs de descripteurs par différentes méthodes incluant un HMM (Hiden Markov Model), et un SVM (Suport Vector Machine). Les différences principales entre ces méthodes se situent au niveau des descripteurs choisis et de la méthode de classification allant du plus proche voisin (1-NN) à des méthodes combinant GMM (Gaussian Mixture Model) et SVM.

Segmentation régulière

D'autre solution ont été étudiées dans lesquels la segmentation est en fait un découpage de l'extrait en trames, à l'instar des systèmes de reconnaissance de parole. Une autre taxonomie HMM est alors utilisées, les trois états considérés sont pour chaque pairs de frappes à reconnaitre sont (décroissance de la frappe précédente ou silence ; attaque ; décroissance). La segmentation et la reconnaissance sont dans ce cas simultanées [Pau06][GR03].

Clustering

Pour s'affranchir du problème à la grande variété de timbres et la grande variété de bruits additifs succeptibles de modifier le signal dans les cas polyphoniques il serait appréciable de pouvoir apprendre des classifieurs très générales les methodes par clustering sont alors plus adéquates. Ainsi Ravelli et al. [BRB06] forment trois clusters à partir des segments détectés. Chacun de ces clusters est alors associé à une classe en fonction de la valeur du centroïd spectral calculée sur ce cluster (bas = grosse caisse ; medium =caisse claire ; haut= HiHat+Cymbals). L'inconvénient de cette methode est la généralisation forte qui réduit la taxonomie à peut de classe (2 ou 3)

II.1.2 Mettre en correspondance et Adapter

Méthode par corrélation

Proposée par Zils [ZPDG02] cette méthode démarre d'un son de percussion synthétique pour la grosse caisse et un pour la caisse claire. Il détecte les occurrences de ce son dans le signal par une méthode de corrélation. Enfin il génère un nouveau son de synthèse en moyennant les occurrences obtenues et itère le processus de détection/génération jusqu'à l'obtention d'un point fixe. La représentation temporelle n'étant pas adaptée à la compréhension des signaux car très peu robuste, fait que cette méthode ne peut obtenir de résultats satisfaisants.

Adaptation d'un modèle temps-fréquence

Sur un même principe mais avec une représentation temps fréquence Yoshii [YGO04] obtiens de bien meilleurs résultats. La distance spectrale, utilisée dans cet article pour choisir les occurrences des modèles construits, permet en effet une meilleur justification perceptive. Cependant ici encore la taxonomie est limitée à peu de classes.

II.1.3 Séparer et Détecter

Fonctionnement

Le principe de fonctionnement de cette approche est d'utiliser des systèmes de séparations de sources sur les signaux à transcrire pour créer un signal pour chaque instrument ou éléments de la batterie. Puis d'effectuer la transcription en faisant de la détection d'onsets sur les pistes séparées.

NMF + Classification

Heln et Virtanen proposent une méthode [HV05] pour séparer la partie percussive de la partie harmonique d'un morceau de musique polyphonique. Ils effectuent le calcul d'une NMF sur le signal puis une classification des bases en deux classes. Les bases qui contiennent de la percussion et les autres. Moreau et Flexer [MF07] prolongent cette méthode pour l'utiliser dans le cadre de la transcription automatique. Ils extraient dans leur technique un ensemble de descripteurs (temporels et fréquentiels) : centroïd spectral, Kurtosis, MFCC, centroïd temporel, périodicité... Ces descripteurs sont appris sur des sons polyphoniques annotés à la main. Puis chaque couple {bases + vecteurs d'activation}, résultant de la séparation NMF, sont classés en Kick, snare, hi-hat et non percussif, par une simple recherche du plus proche voisin sur les vecteurs de descripteurs appris sur les spectres (bases) extrais lors de la NMF. Les vecteurs d'activation sont alors passer dans un détecteur de pic pour obtenir les temps d'onsets.

Prior Subspace Analysis (PSA)

Une autre possibilité est d'utiliser des connaissances à priori, un exemple a été présentée par Fitzgerald dans [FLC03]. Dans cette méthode sont connus à priori les spectres des éléments percussifs que l'on s'attend à trouver (Snare,Kick et HiHat). Ils sont obtenus par moyennage des spectres sur des signaux connus.

Ensuite on réalise une analyse en sous-espace indépendant (ISA) qui s'appuie sur ces spectres pour obtenir les activations de chaque spectre connu à priori. Le problème de la mauvaise détection des onsets de Hi-hat est résolue en normalisant le spectrogramme par la densité spectrale de puissance calculée sur tout le signal. Ceci permet de ré-hausser les hautes fréquences d'autant plus que les basses fréquences sont présentes dans le signal et isole ainsi mieux les Hi-hat qui ont leur énergie en hautes fréquences.

NMF-PSA

Une variante de la PSA [PV05] est NMF-PSA, ici aussi les à priori sont contenus dans le spectre. Des bases sont apprises sur les instruments seuls par NMF avec un représentation ne comportant que 5 bandes de fréquences (20-180 Hz, 180-400 Hz, 400-1000 Hz, 1-10 kHz et 10-20 kHz). Uniquement les bases (les vecteurs de W) sont gardées. Les signaux à transcrire sont des signaux comportant uniquement des sons de batterie, une seconde NMF est réalisée pour décomposer ces signaux sur les bases précédemment apprises. Les résultats de cet algorithme sont très concluant car les bases calculées sont directement liées à la méthode de décomposition.

II.1.4 Performance de l'état de l'art

En 2005 une des taches du MIREX (Musical Information Retrieval Evaluation eXchange) était justement la transcription automatique de la partie percussive de morceau de musique (drum detection). Plusieurs algorithmes ont été proposée voici les résultats qu'il ont obtenus. On voit que les résultats dépendent beaucoup de la base de test utilisée, et les résultats présentés ici ne comportent que les F-mesure pour deux éléments de la batterie cependant cela donne une référence de ce qui a put être fait dans ce domaine.

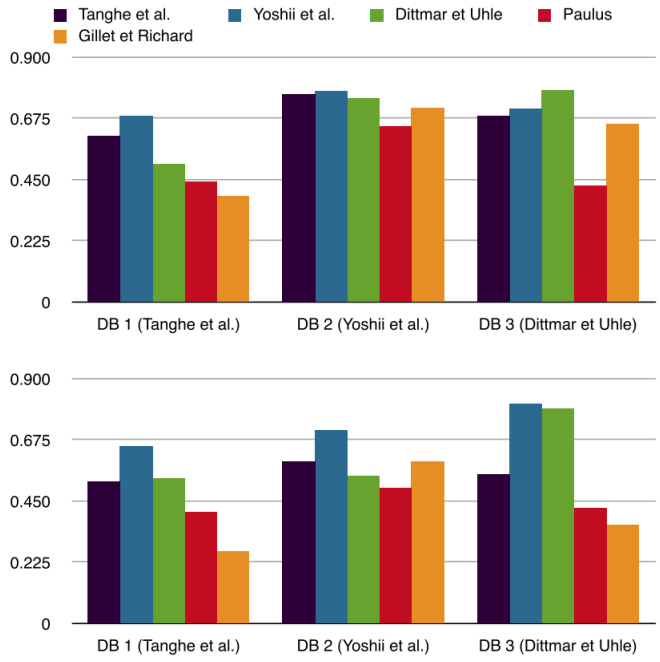


Figure II.1 – Résultats de la campagne d'évaluation MIREX 2005 des algorithmes de transcription de batterie. Détection de frappes de grosse caisse (F-mesure donnée en haut), et détection de frappes de caisse claire (F-mesure donnée en bas). D'après Gillet [Gil07]

II.2 Présentation

Parmi ces méthodes j'ai choisit de me pencher durant ce stage sur une méthode dérivée de la NMF-PSA en y incorporant des contraintes sur les caractéristiques des vecteurs d'activation recherchés (parcimonie et contituité) ainsi que des contraintes hiérarchiques utilisant les informations sémantiques des classes telles que les fréquences d'occurences des combinaisons de frappes. La première étape est l'apprentissage par NMF du dictionnaire sur lequel seront décomposés les signaux, puis la décomposition des signaux sur ce dictionnaire en NMF-PSA. La représentation des signaux choisie est le spectrogramme filtré par 60 bandes de MEL non normalisées. L'échelle de MEL semble appropriée pour plusieurs raisons, elle est un approximation de la perception humaine des fréquences, elle permet aussi une forte réduction de la dimension du spectrogramme ce qui est important dans le cas de la NMF qui requiert beaucoup de ressource de calcul. Les filtres ne sont pas normalisés ceci permet en effet de ré-hausser les hautes fréquences automatiquement, le hautes fréquences contiennent les contributions des idiophones qui sont en générale bien moins énergétique que les membranophones. La figure II.2 montre les différences entre le spectrogramme brut et le MEL-spectrogramme, et le banc de filtre utilisé. On observe que le nombre d'entrées du banc est supérieur au nombre de bandes du spectrogramme car on a effectué un fort zeros-padding lors du calcul de la TFCT (Transformée de Fourrier à Court Terme) pour augmenter la précision de la représentation fréquentielle. On évite ainsi le recouvrement temporel pour les filtres en basses fréquences, ils ont alors assez d'éléments non nuls pour être significatif II.2c. La TFCT de base a été calculée avec des fenêtre de 20ms, le coéficient de recouvrement est de 50% et on ajoute finalement un zeros-padding d'un facteur 16. Une taille de fenêtre plus petite ou un recouvrement plus grand a été envisagé, mais les coûts de calcul étaient vraiment prohibitifs.

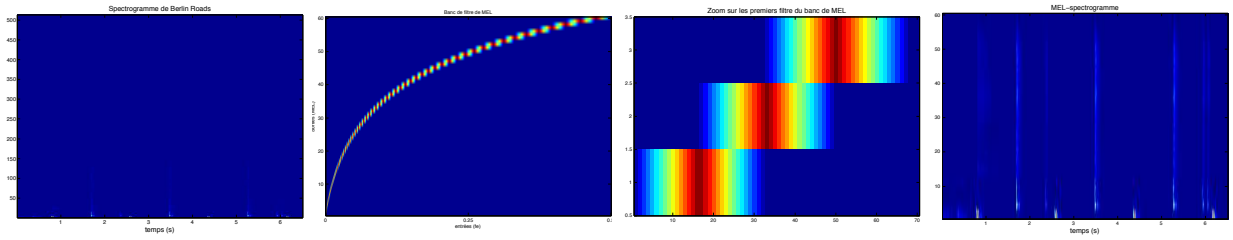


Figure II.2 – Le spectrogramme (a), le banc de filtre de MEL (b), sur le zoom on voit que le facteur 16 de zéros padding est nécessaire pour éviter le recouvrement fréquentiel en basses fréquences (c), Mel-spectrogramme (d)

Chapitre III

NMF Convolutive et contrainte

Le chapitre à venir présente avec plus de détails la NMF ses justifications probabiliste et ses variantes, en particulier l'amélioration de NMF convolutive proposée par P. Smaragdis [Sma04] qui est utilisée dans ma méthode. Y est ensuite présentée la mise en place des contraintes de parcimonie et continuité, applicables lors des mises à jour, par modification de la fonction de coût.

III.1 La NMF

III.1.1 La base

La NMF (Non-negative Matrice Factorization) est un algorithme de réduction de rang qui a pour but d'approcher une matrice V à valeurs positives de dimension (F*N) par le produit de deux matrices aussi à valeurs positives WH respectivement de dimension (F*K) et (K*N), de sorte que $F * N \gg (F + N) * K$. La contrainte de non-négativité sur les valeurs de V,W et H permet de reconstruire la matrice V en K composantes qu'on espère sémantiquement intéressantes. Les colonnes de W sont les mots du dictionnaire, appelés bases, qui servent à reconstruire V, et les lignes de H sont les vecteurs d'activation de ces bases. L'effet de chaque base est directement identifiable indépendamment des autres bases. En effet les vecteurs d'activation (H) étant eux même à valeurs positives, il ne peut y avoir de soustraction et la représentation est en conséquence strictement additive. Ce n'est pas le cas d'autres méthodes similaires telles que ICA, PCA ou la SVD.

Initialement proposée par Paatero dans [PJL96] puis par Lee & Seung dans [LS99] dans le cadre du traitement de l'image la méthode a été appliquée à l'audio par P. Smaragdis & Brown sur le signal audio.[SB03]

La matrice d'observation V est alors un spectrogramme, les mots du dictionnaire (colonnes de W) des spectres et les lignes de H les activations temporelles de ces spectres. Les dimensions de la matrice V sont alors t et f qui représentent respectivement la trame et la fréquence du point du spectrogramme. L'approximation est réalisée en minimisant l'erreur de reconstruction de V par WH. Les fonctions de coût servant à mesurer cette erreur sont détaillées dans la suite.

$$V \approx WH \Longleftrightarrow \left\{ \forall \{f,t\} \in [1..F] * [1..N], \quad V_{ft} \approx \sum_{r=1}^{K} W_{fr} H_{rt} \right\} \tag{III.1}$$

III.1.2 Les fonctions de coût

Les fonctions de coût sont exprimées grâce à une distance ou une divergence :

$$C_V(W,H) = D(V|WH) = \sum_{f,t} d([V]_{ft}|[WH]_{ft}) \tag{III.2}$$

Le but étant de minimiser l'erreur entre V (l'observation) et WH l'estimation. Les seules propriétés que doit vérifier la fonction de coût sont que $d(x|y)$ doit être croissante quand $|x-y|$ croit, et quelle soit nulle ssi $x=y$. Les trois principales fonctions de coût utilisées sont la distance euclidienne, la divergence de Kulback-Liebler, ou la divergence d'Itakura-Saito. Ces trois divergences sont généralisables par la β-divergence

$$\begin{cases} d_\beta(x|y) &= \frac{1}{\beta(\beta-1)}(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}) & \text{pour } \beta \in \mathbb{R}\backslash\{0,1\} \\ d_\beta(x|y) &= x\log(\frac{x}{y}) + y - x & \text{pour } \beta = 1 \\ d_\beta(x|y) &= \frac{x}{y} - \log(\frac{x}{y}) - 1 & \text{pour } \beta = 0 \end{cases} \tag{III.3}$$

On remarque que la β-divergence correspond à la distance euclidienne pour $\beta = 2$ et les cas limites $\beta = 1$ et $\beta = 0$ correspondent respectivement à la divergence de Kullback-Liebler (KL) et à la divergence d'Itakura-Saito (IS). L'influence du choix de β est étudié dans les sections suivantes.

III.1.3 Modèles generatifs

Dans le cas de l'audio la matrice V à approximer vient du spectrogramme X, obtenue par TFCT, qui n'est pas à valeurs positives. On choisit donc généralement sa valeur absolue élevée à une puissance α pour la matrice. Ceci implique une approximation bin à bin vérifiant

$$|[X]_{ft}|^\alpha \approx \sum_{r=1}^{K} |[X_r]_{ft}|^\alpha \tag{III.4}$$

où les X_r représentent les contributions des K éléments présents dans le signal audio à ce point temps-fréquence.

Une discussion sur le choix d'un exposant optimal [Hen11] conclue que dans le cas de deux composantes indépendantes (distribution de phase uniforme) l'α optimal se trouve autour de 1. Les exposants couramment utilisés sont $\alpha = 1$ et $\alpha = 2$ qui correspondent respectivement au spectre d'amplitude et au spectre de puissance. Le liens entre les fonctions de coût et cet exposant s'établit à travers deux modèles probabilistes de signal : le modèle gaussien est présenté dans la partie III.1.3, et le modèle de poisson dans la partie III.1.3.

Modèle gaussien

Ce modèle suppose une contribution de chaque source, ou chaque élément, comme étant une variable aléatoire gaussienne centrée dont la variance dépend de la source. En notant $\mathbf{x}_t$ les vecteurs colonnes du spectrogramme X cela s'écrit :

$$\mathbf{x}_t = \sum_{r=1}^{K} \mathbf{c}_{r,t} \tag{III.5}$$

Les $\mathbf{c}_{r,t}$ sont iid et suivent une loi Normale complexe centrée $\mathbf{c}_{r,t} \sim \mathcal{N}_c(0, h_{r,t} diag(\mathbf{w}_r))$. Où $\mathbf{w}_r$ est le r-ième vecteur colonne de W et $h_{r,t}$ l'activation de ce vecteur au temps t. En prenant $V = |X|^2$ comme matrice d'observation, il est montré dans [FBD09] que l'estimation du maximum de vraisemblance de W et H par maximisation de la Log-vraisemblance par rapport à W et H revient à la NMF entre V et l'estimation WH, en utilisant la divergence IS.

Dans ce cas là une estimation de la contribution des chaque mot du vocabulaire (ou source) est donnée par filtrage de Wiener :

$$\hat{c}_{r,ft} = \frac{w_{fr}h_{rt}}{\sum_{k=1}^{K} w_{fk}h_{kt}} x_{ft} \tag{III.6}$$

Ceci est très intéressant et très utilisé pour effectuer la séparation de source. On peut en effet reconstruire un signal temporel à partir des $\hat{\mathbf{C}}_r$ par méthode TFCT inverse. Prendre une puissance $\alpha = 2$ et $\beta = 0$ a donc une justification dans le cas d'un modèle de source gaussienne additive, et est particulièrement approprié dans le cadre de la séparation de source.

Modèle de Poisson

Une autre supposition générative est possible. Supposons maintenant que le spectrogramme d'amplitude est généré par :

$$|\mathbf{x}_t| = \sum_{r=1}^{K} |\mathbf{c}_{r,t}| \tag{III.7}$$

Où les $|\mathbf{c}_{r,ft}| \sim \mathcal{P}(w_{fr}h_{rt})$ sont iid. $\mathcal{P}$ est la distribution de poisson. $|x_{ft}|$ suit alors aussi une loi de Poisson : $|x_{ft}| \sim \mathcal{P}(\sum_{r=1}^{K} w_{fr}h_{rt})$ étant une somme de variable de Poisson. Dans ce cas en prenant $V = |X|$ l'estimation du maximum de vraisemblance revient à la NMF avec la divergence KL [Vir07].

Virtanen propose alors comme reconstruction :

$$\hat{c}_{r,ft} = w_{fr}h_{rt}arg(x_{ft}) \tag{III.8}$$

Cette reconstruction peut poser des problèmes au niveau de la conservation de l'énergie.

Comment choisir β

Il y a un autre argument pour le choix de β, il s'agit de la dépendance au changement d'échelle.

- **divergence d'Itakura Saito** (beta=0) : $d(x.A, x.B) = d(A, B)$ la divergence IS est dite invariante par homothétie, elle donnera autant d'importance à une point temps fréquence contenant des petites valeurs, typiquement les hautes fréquences, qu'à un points contenant beaucoup d'énergie (les basses fréquences). Par contre elle n'est pas convexe partout et a donc de nombreux minima locaux qui sont très problématiques en optimisation.
- **distance euclidienne** (beta = 2) : $d(x.A, x.B) = x^2.d(A, B)$; privilégie beaucoup les basses fréquences qui sont les plus énergétiques, elle a cependant l'avantage d'avoir moins de minima locaux, ce qui amène Bertin a proposer une NMF à β variable pour éviter les minima locaux (commencer par β grand) et prendre les avantages de la IS-NMF en même temps [BBF].
- **divergence de Kullback-Liebler** (beta = 1) : $d(x.A, x.B) = x.d(A, B)$, se situe donc entre les deux.

III.1.4 algorithmes : règles de mises à jour

L'algoryhtme de calcul de la NMF se fait de manière itérative, en mettant à jour une matrice tout en gardant fixe l'autre. Par exemple on fixe W puis on met à jour H et on inverse. Le calcul de la mise à jour s'effectue par descente de gradient. La méthode la plus utilisée, parfois appelée descente de gradient à pas adaptatif, permet une mise à jour multiplicative simple, elle a été proposée dans [LS99]. Cette méthode part de l'idée de séparer le gradient de la fonction de coût en une différence de deux partie strictement positive.

$$\frac{\partial C}{\partial \theta} = p_\theta - m_\theta \tag{III.9}$$

La mise à jour de θ est alors

$$\theta^* = \theta * \frac{p_\theta}{m_\theta} \tag{III.10}$$

Cette méthode assure la positivité, l'évolution de θ dans la même direction que la dérivée partielle, et que θ sera constant uniquement si on se trouve sur un point fixe.

Cette méthode donne dans le cas de la β-divergence les règles de mise à jour suivantes :

$$H = H \otimes \frac{W^T\Big[(WH)^{\beta-2} \otimes V\Big]}{W^T\Big[(WH)^{\beta-1}\Big]} \tag{III.11}$$

$$W = W \otimes \frac{\Big[(WH)^{\beta-2} \otimes V\Big]H^T}{\Big[(WH)^{\beta-1}\Big]H^T} \tag{III.12}$$

Ici le $\otimes$ représente le produit terme à terme, l'exposant est aussi la puissance terme à terme. Ces mises jour ont été précisées dans [FI11] en y ajoutant un exposant au facteur multiplicatif qui dépend de β permettant une adaptation du pas assurant mathématiquement la décroissance de la fonction de coût sur un intervalle de β plus grand que {0,2}.

III.1.5 Les améliorations

Des variantes ont été apporté à la première méthode de décomposition de spectre audio [SB03] en permettant des invariances par translation temporelle, fréquentielle ou les deux.

- la NMF convolutive [Sma04] permet des bases en 2D les mots du dictionnaire sont des sprectogrammes et permet donc une invariance par translation temporelle.
- la NMF2D [SMr] permet des éléments de la base W en 2D et des matrices d'activation aussi en 2D. Ceci est utile pour les instruments de musique non percussifs, chaque note étant considérée à une translation fréquentielle près. Voir aussi [Hen11]

NMF Convolutive

On ne détaillera que la NMF convolutive proposée par P. Smaragdis dans [Sma04] qui parait être l'amélioration la plus utile pour mon application à des sons percussifs. Chaque mot du dictionnaire sera alors un spectrogramme d'une durée 0.5s. l'équation d'approximation s'écrit alors ainsi :

$$V \approx R = \sum_{t=1}^{T-1} W_t \overset{t\rightarrow}{H} \tag{III.13}$$

L'operateur $\overset{t\rightarrow}{\mathrm{M}}$ est l'opérateur de translation, il décale toutes les colonnes de la matrice M de t indices vers la droite en remplissant de zéros les colonnes insérer. W_t est la t-ième tranche du tenseur W qui décrit la t-ième trame du spectrogramme de tous les K éléments de la base qui sont maintenant des matrices de dimension (F*T). T est donc le nombre de trames sur lequel on considère l'évolution des bases. On peut interpréter W comme un cube de données 3D de dimension (F*K*T). Cette méthode ne détecte que les éléments dont les occurrences sont très similaires. C'est le cas des instruments de percussion. Cette approche convolutive parait en conséquence la plus adaptée pour la transcription de batterie. Les vecteurs d'activation que l'on désire alors observer doivent être très parcimonieux, on ajoute alors souvent des contraintes de parcinomie sur H (cf partie III.2.1).

$$\begin{aligned}
H &= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix} \\
\overset{1\rightarrow}{H} &= \begin{bmatrix} 0 & 1 & 2 & 3 \\ 0 & 5 & 6 & 7 \end{bmatrix} \\
\overset{2\rightarrow}{H} &= \begin{bmatrix} 0 & 0 & 1 & 2 \\ 0 & 0 & 5 & 6 \end{bmatrix} \\
\overset{\leftarrow 2}{H} &= \begin{bmatrix} 3 & 4 & 0 & 0 \\ 7 & 8 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{III.14}$$

Pour la divergence KL, les règles de mise à jour sont pour chaque $t \in [0..T-1]$:

$$H = H \otimes \frac{W_t^T \overset{\leftarrow t}{\left[\frac{V}{R}\right]}}{W_t^T \mathbf{1}} \tag{III.15}$$

$$W_t = W_t \otimes \frac{\left[\frac{V}{R}\right] \overset{t\rightarrow T}{H}}{\mathbf{1} \overset{t\rightarrow T}{H}} \tag{III.16}$$

Les **1** sont des matrices remplies de uns. Pour les mises à jour de H vu la rapide convergence de l'algorithme il est fortement conseillé de moyenner H sur les T itérations.

III.2 Les contraintes

Pour améliorer la pertinence de la NMF il est fréquent d'y ajouter des contraintes dans le calcul des mises à jour. Elles sont en générale ajoutées sous forme d'une modification de la fonction de coût :

$$C_V(W,H) = D(V|WH) + \lambda C_c(W,H) \tag{III.17}$$

Les contraintes fréquemment utilisées pour les NMF sont :
- parcimonie des vecteur d'activation ou des mots du dictionnaire
- harmonisité des mots du dictionnaire
- continuité sur les vecteurs d'activation
- orthogonalité en H ou W

III.2.1 La parcimonie

Une mesure

Parmis ces contraintes les seules qui peuvent servir pour mon étude sont la parcimonie en H, on espère en effet avoir une impulsion pour chaque frappe, ainsi que la continuité des lignes de H qui aidera lors de la détection de pics sur les vecteurs d'activation.

Commensons par la contrainte de parcimonie. Une définition de la mesure de parcimonie est proposée dans [Hoy04], pour un vecteur de taille n :

$$sp(\mathbf{x}) = \frac{\sqrt{\mathbf{n}} - \frac{\sum_i |\mathbf{x_i}|}{\sqrt{\sum_i \mathbf{x_i^2}}}}{\sqrt{\mathbf{n}} - \mathbf{1}} \tag{III.18}$$

En d'autres termes la parcimonie augmente quand le rapport entre la norme L_1 et la norme L_2 augmente.

Opérateur de projection parcimonieuse

L'option choisit par Hoyer, est de forcer la valeur de parcimonie des lignes de H, il fixe donc la norme $L_2 = 1$ et règle la norme L_1 pour obtenir le degré de parcimonie désiré par un opérateur de projection. Dans son article Hoyer propose un algorithme de déscente de gradient avec projection sur l'espace contraint, le pas μ_w est optimisé à chaque itération pour être la plus grande valeur assurant la décroissance de la fonction de coût (dans la cas de Hoyer la divergence KL). La force de l'algorithme réside dans son opérateur de projection III.1.

Descente de gradient :

$$W = W - \mu_w(WH - V)H^T \tag{III.19}$$

fonctionnement de l'Algorithme
- **Résultats désiré** : le vecteur **s** non-negatif le plus porche de **x** (au sens enclidien), ayant une norme L_1 et une norme L_2 fixée
- **Procédure** :
  - commencer par projeter le vecteur sur l'hyperplan $\sum s_i = L_1$,
  - puis on projeter sur la sphère intersection de la contrainte L_1 et la contrainte L_2,
  - puis assigner la valeur 0 aux élément négatif de **s**
  - itérer jusqu'a ceux **s** soit à valeur non négative.

Dans notre application cette contrainte forte n'est pas très adapté, en effet la valeur de parcimonie désirée doit être connu à l'avance ce qui ne peut se faire dans notre cas. Cependant l'opérateur de projection sera réutilisé par la suite.

problem Given any vector x, find the closest (in the euclidean sense) non-negative vector **s** with a given L_1 norm and a given L_2 norm.
algorithm The following algorithm solves the above problem.

1. Set $s_i = x_i + \frac{L_1 - \sum x_i}{dim(x)}, \forall i$
2. Set $Z := \{\}$
3. Iterate
   (a) Set $m_i := \begin{cases} \frac{L1}{dim(x)-size(Z)} & \text{if i} \notin Z \\ 0 & \text{if i} \in Z \end{cases}$
   (b) Set $\mathbf{s} := \mathbf{m} + \alpha(\mathbf{s} - \mathbf{m})$, where $\alpha \geq 0$ is selected such that the resulting **s** satisfies the L_2 norm constraint. This requires solving a quadratic equation.
   (c) If all components of **s** are non-negative, return **s**, end
   (d) Set $Z := Z \cup \{i; s_i < 0\}$
   (e) Set $s_i := 0, \forall i \in Z$
   (f) Calculate $c := \frac{\sum s_i - L_1}{dim(x) - size(Z)}$
   (g) Set $s_i := s_i - c, \forall i \notin Z$
   (h) Go to (a)

Figure III.1 – Algorithme de projection sur un espace à parcimonie fixée d'après Hoyer [Hoy04]

Une référence pour la NMF parcimonieuse

Une autre option est bien entendu celle de modifier la fonction de coût en y introduisant un poids qui permettra d'augmenter la parcimonie itération par itération. Eggert propose donc "Sparse Coding and NMF" [EK04] dans lequel la fonction de coût est modifiée :

$$C_V(W,H) = D_{euc}(V|WH) + \lambda \sum_{i,j} g(H_{i,j}) \tag{III.20}$$

Le choix de g se porte sur la norme L_1, un point important est détaillée dans cet article, celui de la normalisation de la NMF. Sans normalisation les valeurs de la matrice H seraient diminuées au fur et à mesure des itérations en étant compensées par de grandes valeurs dans la matrice W. C'est pourquoi Eggert insiste sur le fait que pour avoir une augmentation de la parcimonie de H il faut normaliser W à chaque itération.

Les règles de mises à jour multipplicative pour cette méthode avec sont les suivantes :

$$\begin{aligned} W_k &= \frac{W_k}{\|W_k\|}, \forall k \\ H &= H \otimes \frac{W^T V}{W^T[WH]+\lambda} \\ W &= W \otimes \frac{VH^T + W \otimes ([WH]H^T \otimes W)}{[WH]H^T + W \otimes (VH^T \otimes W)} \end{aligned} \tag{III.21}$$

Intéressons-nous maintenant à la combinaison des deux contraintes.

III.2.2 Continuité et parcimonie

La seconde contrainte, la contituité a pour but de réduire les grandes variations entre deux échantillons. T. Virtanen avec la "NMF with temporal continuity and sparseness criteria" [Vir07] assigne donc un poids aux changements importants entre deux valeurs adjacentes activation.

Les contraintes ne sont appliquées que sur H. Une normalisation, rappelons le, est necessaire pour empécher la diminution du poids de la contrainte sans pour autant changer l'erreur de recontruction par changement d'échelle sur les matrices. Ici elle s'effectue sur H, le facteur de normalisation est $\sigma_i = \sqrt{\frac{1}{T}\sum_{j=1}^{T} h_{ij}^2}$ l'estimateur de l'écart type. Finalement le poids de parcimonie est la L_1 normalisée, et la contrainte de continuité est :

$$C_s(H) = \sum_{k=1}^{K} \frac{\|H_k\|_1}{\sigma_k} \qquad C_c(H) = \sum_{k=1}^{K} \frac{1}{\sigma_k} \sum_{t=2}^{T} (h_{t,k} - h_{t-1,k})^2 \tag{III.22}$$

La fonction de coût est finalement :

$$C_V(W,H) = D(V|WH) + \gamma C_c(H) + \lambda C_s(H) \tag{III.23}$$

En calculant les gradients de chaque partie de la fonction de coût et en la séparant en une difference de deux termes strictement positifs (cf descente de gradient à pas adaptatif) on obtient des règles de mise à jour multiplicative.

$$H = H \otimes \frac{\nabla C_V^-}{\nabla C_V^+} = H \otimes \frac{\nabla D^-(W,H) + \lambda \nabla C_s^-(H) + \gamma \nabla C_c^-(H)}{\nabla D^+(W,H) + \lambda \nabla C_s^+(H) + \gamma \nabla C_c^+(H)} \tag{III.24}$$

Cet algorithme n'assure malheuresement pas la décroissance de la fonction de coût, en pratique cependant cela fonctionne.

III.2.3 Convolution, Parcimonie et Continuité

Pour finalement appliquer cette descente de gradient dans le cadre de la NMF convolutive, on applique à partir de la mise à jour précédente les translations adéquates. Le protocole de calcul de la NMF convolutive contrainte en parcimonie et continuité est alors :

1. calcul des gradients (d'après [Vir07]) :

$$\forall f,t \in [1..F] * [1..N] \begin{cases} [\nabla C_s^+(H)]_{ft} &= \dfrac{1}{\sqrt{\frac{1}{T}\sum_{i=1}^{N} H_{fi}^2}} \\ [\nabla C_s^-(H)]_{ft} &= \dfrac{H_{ft}\sqrt{N}\sum_{i=1}^{N} H_{fi}}{(\sum_{i=1}^{N} H_{fi}^2)^{\frac{3}{2}}} \\ [\nabla C_c^+(H)]_{ft} &= \dfrac{4NH_{ft}}{\sum_{i=1}^{N} H_{fi}^2} \\ [\nabla C_c^-(H)]_{ft} &= 2N\left[\dfrac{H_{f,t-1}+H_{f,t+1}}{\sum_{i=1}^{N} H_{fi}^2} + \dfrac{H_{ft}\sum_{i=2}^{N}(H_{f,i}-H_{f,i-1})^2}{\left(\sum_{i=1}^{N} H_{fi}^2\right)^2}\right] \end{cases} \tag{III.25}$$

2. calcul des gradients de la divergence (partie convolutive),

$$\forall t \in [0..T-1] \begin{cases} \nabla D_t^+(W,H) &= W_t^T \mathbf{1} \\ \nabla D_t^-(W,H) &= W_t^T [\overleftarrow{\frac{V}{R}}^t] \\ H'_t &= H \otimes \dfrac{\nabla D_t^-(W,H) + \lambda \nabla C_s^-(H) + \gamma \nabla C_c^-(H)}{\nabla D_t^+(W,H) + \lambda \nabla C_s^+(H) + \gamma \nabla C_c^+(H)} \end{cases} \tag{III.26}$$

3. moyennage de la mise à jour de H sur les T trames de la convolution

$$H = \frac{\sum_{t=0}^{T-1} H'_t}{T} \tag{III.27}$$

4. mise à jour convolutive de W

$$\forall t \in [0..T-1], W_t = W_t \otimes \frac{\left[\frac{V}{R}\right] \overset{t\rightarrow}{H}^T}{\mathbf{1}\overset{t\rightarrow}{H}^T} \tag{III.28}$$

III.2.4 Poids des contraintes

Dans cet algorithme les contraintes sont appliquées avec en introduisant un poids supplémentaire dans la fonction de coût, ces poids sont pondérés par des réels positifs qui règlent le degré d'importance de chaque contrainte. Le choix de ces pondérations est important même si la question n'est pas souvent abordé dans la littérature. Prendre une valeur identique pour tous les signaux entraine de façon certaine des changements de représentation très fort en fonction par exemple de la taille du signal considéré. En supposant que toutes les parties de la fonction de coût sont décroissantes, on peut choisir λ ou γ pour que le poids des contraintes pèsent autant qu'une proportion m choisie de la divergence :

$$\lambda = m * \frac{D_{kl}(V|W_iH_i)}{C(H_i)} \qquad H_i \text{ et } W_i \text{ sont les matrices initiales de la NMF} \tag{III.29}$$

Cette solution empirique permet de prévoir que l'apprentissage de toutes les bases et que toutes les décompositions s'effectuent avec des contraintes similaires.

III.2.5 Choix des pondérations

Le choix des coefficients de pondérations a été fait en comparant les F-mesure maximale moyenne sur 100 itérations et tous les morceaux des bases de test avec différentes valeurs de λ et γ, la figure III.2 représente ces résultats de F-mesure par rapport à λ et γ. On observe que contrairement à ce à quoi on aurait pu s'attendre la valeur du poids donné à la continuité à plus d'importance que celui donné à la parcimonie. On choisit donc graphiquement les valeurs pour la suite $\lambda = 5.10^{-4}$ et $\gamma = 2.10^{-7}$.

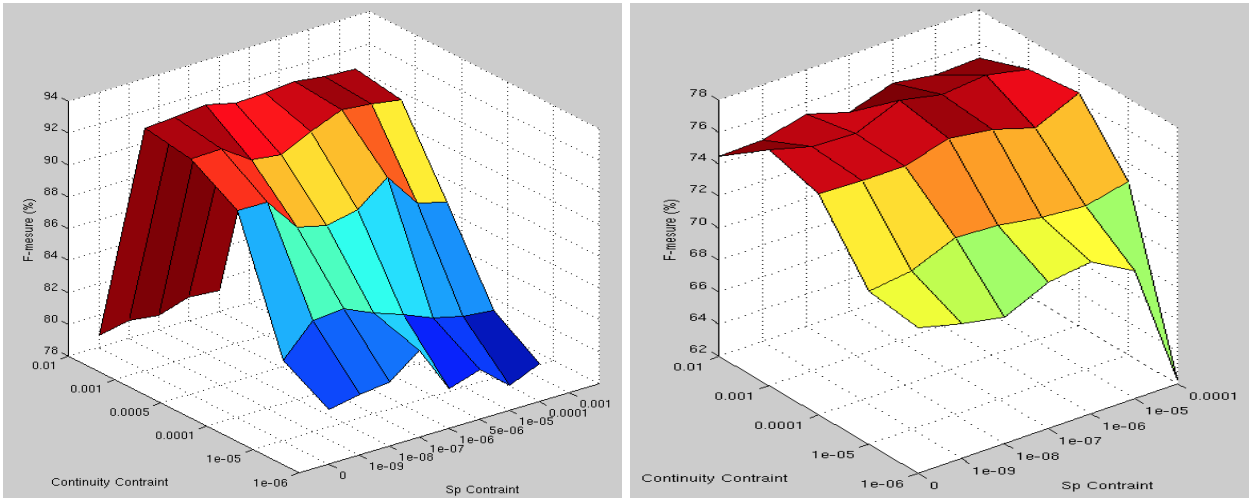


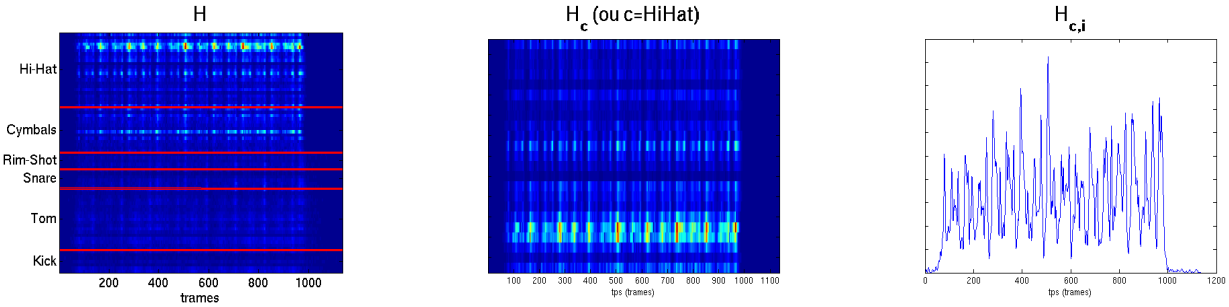
Figure III.2 – dans chacun des graphiques l'axe x (abscisse) représente les différentes valeurs de λ considérées, et l'axe y (profondeur) les valeurs de γ ; à gauche a) pour la base de test ENST, et à droite b) pour la MASS-GT

Chapitre IV

Des à priori : contraintes hiérarchiques

On peut ajouter des à priori de plus haut niveau à la parcimonie et la continuité. En effet on décode un morceau sur un dictionnaire 'sur-complet' il contient en effet plus de mots que le nombre de composantes que l'on espère extraire de la décomposition. Ce dictionnaire étant divisé en classe, qui sont les différents éléments d'un kit de batterie (cf I.2) on peut à partir de ces informations de haut niveau (à quelle classe appartient cet atome) appliquer de nouvelles contraintes pour donner plus de sens à la décomposition.

Il est nécessaire ici d'introduire quelques notations supplémentaires :



- C = nombre de classe (ici 6)
- K_c = nombre de bases dans la classe c (ici 23) (il s'agit bien de la même matrice H mais de bas en haut)
- H_c est la matrice construit par la concaténation des vecteurs d'activation de la classe c.
- N = nombre de trames du signal (ici 1135)

IV.1 La contrainte Intra-classe

Dans un seul morceau 'acoustique' on peut supposée qu'une seule batterie joue, on peut donc espérer trouver, pour chaque classe, quel sera l'atome le plus proche de l'élément du kit utilisé dans le morceau. C'est ce but que la contrainte est introduite. A chaque itération on renforcera le vecteur d'activation le plus pertinent tout en abaissant les vecteurs d'activation moins pertinents, de facon à tendre vers une décomposition où dans chaque classe on aura plus qu'un vecteur actif. On va utiliser pour cela la projection parcimonieuse de Hoyer définit plus tôt.

IV.1.1 Contrôle de la parcimonie

En utilisant de l'algorithme de projection définit par Hoyer on peut construire un vecteur V' à partir d'un vecteur V qui aura la même norme L_2 et qui sera le plus proche vecteur de V, au sens euclidien, ayant une valeur de parcimonie $\Theta \in [0, 1]$. Dans chaque classe on aimerait arriver à un seul vecteur actif, c'est donc un cas de parcimonie égal à un. On prendra donc un valeur Θ telle que :

$$\Theta > sp(V), \Theta \in [0, 1] \tag{IV.1}$$

Une fonction permet de trouver ce Θ avec un paramètre α servant à régler l'incrément de la parcimonie.

$$\Theta = \frac{log\big(1 + (\alpha - 1) * sp(V)\big)}{log(\alpha)} \tag{IV.2}$$

On définit ainsi l'opérateur $V' = \coprod(V, \alpha)$ La figure IV.1 montre l'effet de cet opérateur sur un vecteur.

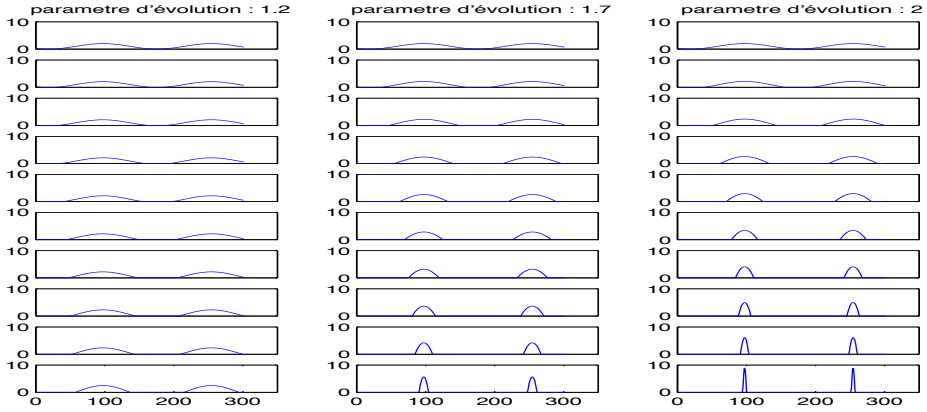


Figure IV.1 – Visualisation du contrôle de la parcimonie, l'opérateur $\coprod$ est appliqué itérativement pour trois valeurs différentes de α

IV.1.2 Modification de H

Nous allons donc utiliser cette projection pour renforcé les "bons" vecteurs d'activations au sein d'une classe. La question est alors de savoir discriminer les bons vecteurs des mauvais. Une première intuition serait de prendre comme mesure de pertinence pour chaque activation leur norme L_1 ainsi les atomes les plus actifs seront peu à peu mis en avant par rapport au moins actifs. L'inconvénient de cette approche est que l'on peut obtenir en sortie des vecteurs un peu actif partout mais sans véritable pic révélant une frappe sur l'élément associé. La mesure de parcimonie seule n'est pas non plus une bonne alternative car un vecteur nulle partout sauf en un point aura la plus grand valeur de parcimonie possible mais ne sera pas non plus important. C'est pourquoi on introduit une valeur de pertinence pour chaque vecteur comme étant une pondération des deux. Donc pour chaque classe $c \in [1..K]$ et chaque vecteur de la classe soit $i \in [1..K_c]$ on calcul une valeur de pertinence $S_c(i)$:

$$S_c(i) = R * sp(H_{c,i}) + (1 - R)\|H_{c,i}\| \tag{IV.3}$$

Au sein de chaque classe cette valeur va être utilisée pour savoir quels vecteurs doivent être renforcés et quels autres diminués. On va utilisée l'opérateur de contrôle de parcimonie, pour une classe c donnée, on calcule $S'_c = \coprod(S_c, \alpha)$ puis le rapport terme à terme de $\frac{S'_c}{S_c}$ nous fournit les information désirées : $\frac{S'_c(i)}{S_c(i)} > 1$

pour les atomes i pertinents au sein de la classe c et $\frac{S'_c(i_c)}{S_c(i)} < 1$ pour les autres. Cette contrainte peut donc aussi être dite globale car elle prend une valeur de pertinence globale sur chaque vecteur d'activation.

Finalement la contrainte Intra-classe ou globale s'applique sur les activations de cette manière :

$$\forall c \in [1..K], \forall i \in [1..K_c], \forall t \in [1..N], H_{c,i}(t) = \frac{S'_c(i)}{S_c(i)} H_{c,i}(t) \tag{IV.4}$$

Le choix de R a été obtenue de la même manière que λ et γ (cf :III.2.5). Finalement par la suite $R = 0.8$. Ainsi avec le même nombre d'itérations on obtient sans la contrainte la figure IV.2a et avec la contrainte la figure IV.2b.

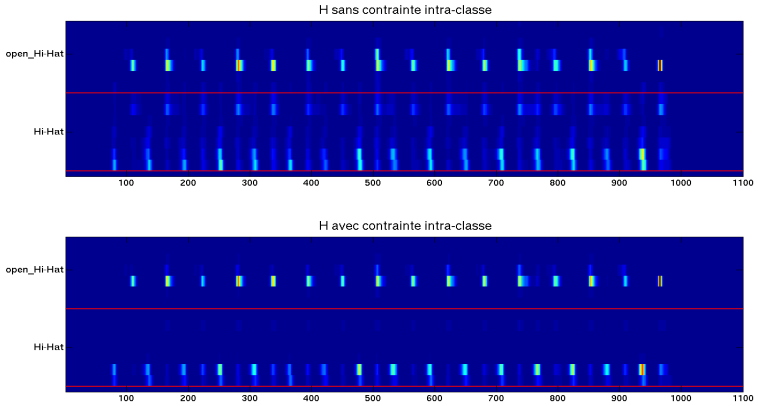


Figure IV.2 – Visualisation des vecteurs d'activations (lignes de la matrice) de deux classes, (a) sans la contrainte Intra-classe et (b) avec la contrainte.

IV.2 La contrainte Inter-classe

L'autre information que l'on peut obtenir des différentes classes est la proximité entre classe. Dans le dictionnaire appris, ont été différentiées des classes dont les atomes peuvent être différents mais qui ont une sémantique proche : il existe une classe Hi-Hat et une classe open-Hi-Hat ces deux classes sont issues du même élément du kit de batterie, et de plus ne peuvent être jouées simultanément.

En se basant sur ces considérations on peut mettre en place une deuxième contrainte qui proscrira les activations simultanées de deux classes issues du même élément ou ayant une faible probabilité d'être joués simultanément. Une étude sur les fréquences des frappes proposée dans [Gil07] est présentée dans le tableau IV.3. Ainsi les classes 'Snare', 'Rim-shot' et 'Cross-Stick' (les 3 techniques de production de son avec la caisse-claire) peuvent être regroupées dans un même groupe, ne pouvant être joués simultanément. De même que les toms avec la grosse caisse. On peut donc considérer une partition des classes $\mathcal{P}$ qui seront donc ces groupes.

Frappe	**Frèq.** (%)
$\{sd\}$	19.3
$\{hh\}$	17.3
$\{bd,hh\}$	13.5
$\{hh,sd\}$	9.5
$\{bd\}$	5.9
$\{bd,cym\}$	5.8
$\{cym\}$	5.5
$\{tom\}$	3.9
$\{db,sd\}$	3.6
$\{db,sd,hh\}$	3.3
$\{cym,sd\}$	2.6
$\{cym,hh\}$	1.9
$\{db,cym,hh\}$	1.6
...	

Figure IV.3 – Fréquence des combinaisons de frappes dans le corpus ENST-drums, en considérant la taxonomie $\{bd,\ sd,\ hh,\ cym,\ tom\}$, d'après [Gil07]

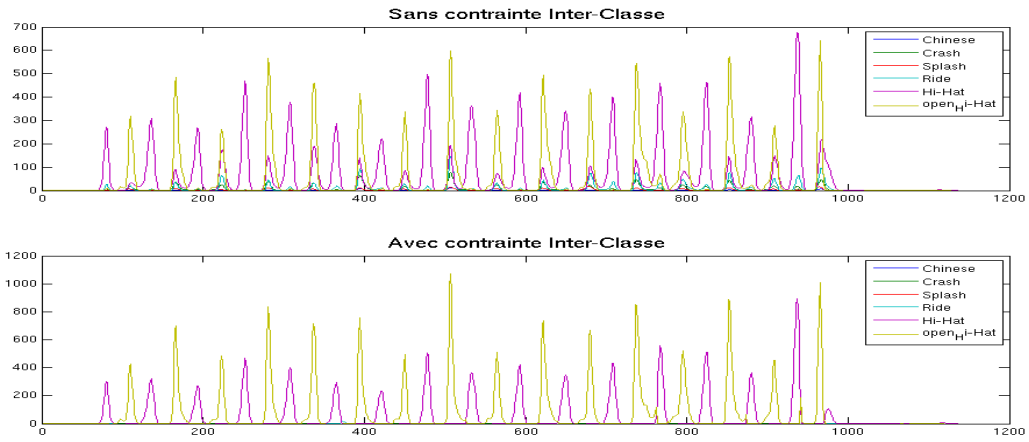


Figure IV.4 – Visualisation des activations totales pour les classes du groupe {*Chinese, Crash, Splash, HiHat, open-HiHat*}. (a) sans la contrainte inter-classe ; (b) avec la contrainte

ex : $G \in \mathcal{P}$où G={ snare, rimshot, cross-stick }. Cette partition va servir à mettre en place la contrainte inter-classe.

La contrainte va s'appliquer un peu de la même manière que la contrainte intra-classe, excepté que les valeurs de pertinences seront différentes et que la contrainte s'applique localement. De cette manière il y aura un vecteur de pertinence pour chaque colonne de la matrice H, et les valeurs de pertinences sont calculées simplement à partir de la somme des activations au sein de chaque classe. Formellement le calcul des vecteurs de pertinences, $\forall t \in [1..N]$:

$$\forall c \in [1..K], A_c(t) = \sum_{i=1}^{K_c} H_{c,i}(t) \tag{IV.5}$$

$$\forall G \in \mathcal{P}, \text{ où } G = \{c_1, c_2, .., c_n\}, S_G(t) = [A_{c_1}(t) \;\; A_{c_2}(t) \;\; ... \;\; A_{c_n}(t)]^T \tag{IV.6}$$

$$\text{Soit}, S'_G(t) = \coprod(S_G(t), \alpha)$$

Puis modification des vecteurs d'activation en renforçant localement les plus présents du groupe G :

$$\forall c \in G, \forall i \in [1..K_c], \quad H_{c,i}(t) = \frac{S'_{G,c}(t)}{S_{G,c}(t)} H_{c,i}(t) \tag{IV.7}$$

Les vecteurs d'activation total (par classe) ne pourront plus être actif simultanément, comme le montre la figure IV.4 pour le groupe composé des Cymbales et des Hi-Hat. Il est essentiel, pour que cette contrainte hiérarchique soit effective, que les vecteurs d'activation soient continus.

Les 3 groupes utilisés dans mon algorithme correspondent au trois couleurs dans I.2, les fils de chaque classes de niveau hiérarchique 1 sont supposé ne pas pouvoir être joué simultanément.

IV.3 Adaptation des atomes

La contrainte Intra-classe permet de se rapprocher d'une matrice d'activation H ou seulement un vecteur est non nul par classe, on espère alors avoir trouver dans notre dictionnaire le mot de la classe le plus proche de l'élément de batterie présent dans le morceau à transcrire. On peut espérer alors améliorer le résultat de la transcription en laissant l'algorithme choisir ce mot pour décrire totalement cette classe et le soumettre à une

adaptation modérée pour lui permettre de coller encore plus aux données. Une des variantes de l'algorithme consiste donc, à partir d'un certain nombre d'itérations de sélectionner pour chaque classe le mot le plus actif (plus grande norme L_1 du vecteur d'activation), puis limiter le dictionnaire à ces mots (1 mot pour chaque classe) et enfin de permettre les mises à jour de W (cf III.28). Cependant si l'on permet les mises à jour de W le problème auquel on est susceptible de se confronter est : chaque mot va-t-il rester représentatif de sa classe de départ ?
Une solution pour limiter les évolutions de W est de rajouter un exposant servant à tempérer ces évolutions. La mise à jour proposée est alors :

$$\forall t \in [0..T-1], W_t = W_t \otimes \left[\frac{\left[\frac{V}{R}\right] \overset{t\rightarrow T}{H}}{\overset{t\rightarrow T}{\mathbf{1}\,H}} \right]^{\frac{1}{1+\|W-W_i\|_2}} \tag{IV.8}$$

où W_i représente le mot extrait du dictionnaire sans adaptation, ainsi plus la différence entre le mot de base et le mot courant sera grande moins l'adaptation aura d'effets.

IV.4 Sous espace non contraint : pour les résidus harmoniques

Finalement lors de transcription de signaux polyphonique ou des signaux séparés par un algorithme, il faut prévoir une partie du dictionnaire pour décrire le bruit (sons ou résidu de instruments harmoniques). Le dictionnaire doit donc contenir une classe "poubelle" qui ne serais pas contrainte ni en parcimonie ni hiérarchiquement, et qui ne sera pas utilisée dans la détection.

Chapitre V

Détection et mesure de performance

A partir de la matrice H obtenue on doit pouvoir effectuer une détection sur les activations de chaque classe pour déterminer si oui ou non est présent à un instant telle ou telle élément. Une fois le morceau transcris il reste à établir une mesure de performance à partir de cette estimation de la transcription et de la vérité, annotation manuelle du morceau.

V.1 Détection des notes

La dernière partie de la transcription qui fournira les instants des frappes pour chaque classe est la détection de pics sur les lignes de la matrice H. Une détection de pic en elle même ne pose pas de problème mais dans ce cas le seuillage est crucial si l'on ne veut pas faire d'à priori excessifs. En effet j'avais au départ considérer d'utiliser la procédure décrite par Paulus [PV05] qui consiste à normaliser chaque vecteur d'activation, puis de réaliser une compression logarithmique du résultat et de faire enfin la détection de pic sur le résultats de la compression. L'à priori fort de cette méthode est que toute les classes sont présentes dans le signal, en effet l'information perdue lors de la normalisation fait que, à moins de prendre un seuil supérieur à un, on est obligé de détecter au moins une occurrence de chaque classe. Ce qui ne peut pas être le cas dans ma méthode vu le grand nombre de classe considéré (Paulus a une taxonomie à trois classe). Pour s'affranchir de cet à priori, la détection doit être combinée avec une préalable détection de classes actives, laquelle est très instable et peu robuste.

C'est pourquoi la détection de pic se fait directement sur les activations totales de chaque classe sans normalisation ni compression. Pour chaque classe le seuil optimal dépend du morceau et ne peuvent donc pas être calculé de manière absolue, les seuils choisis pour chaque classe sont donc assez arbitraires, on a prit finalement des seuils assez bas pour ne pas rater d'occurrence en espérant que les contraintes inter-classe et intra-classe éliminent assez les pics inopportuns.

Classes	{*Kick, Snare, HiHat, open-HiHat, Crash, Ride, Rim-Shot, Cross-Stick*}	{*Toms*}	{*CowBell*}
Seuil	30	100	150

La détection de seuil se fait par détection d'annulation de la dérivée avec dérivée seconde négative, seuls sont gardés les pics supérieurs au seuil. De même quand deux pics sont proches de moins de 30ms seul est gardé le plus grand des deux. Pour certaines classes telles que les cymbales libres les maxima du vecteurs d'activation ne correspondent pas avec l'instant de frappe et peuvent en être assez éloignés pour être comptés comme faux. Pour résoudre ce problème dans le cas des cymbales libre (Crash, Splash, Ride, open-HiHat)

je détecte l'instant du maximum de la dérivée de l'activation dans les 200ms précédant le pic cette valeur est prise comme instant de frappe, il est plus proche de la vérité que le pic.

V.2 Evaluation

Nous avons donc à ce point une estimation de la transcription pour un morceaux donné. Il reste à préciser la métrique servant à évaluer la qualité de la transcription. Prenons comme exemple la figure V.1

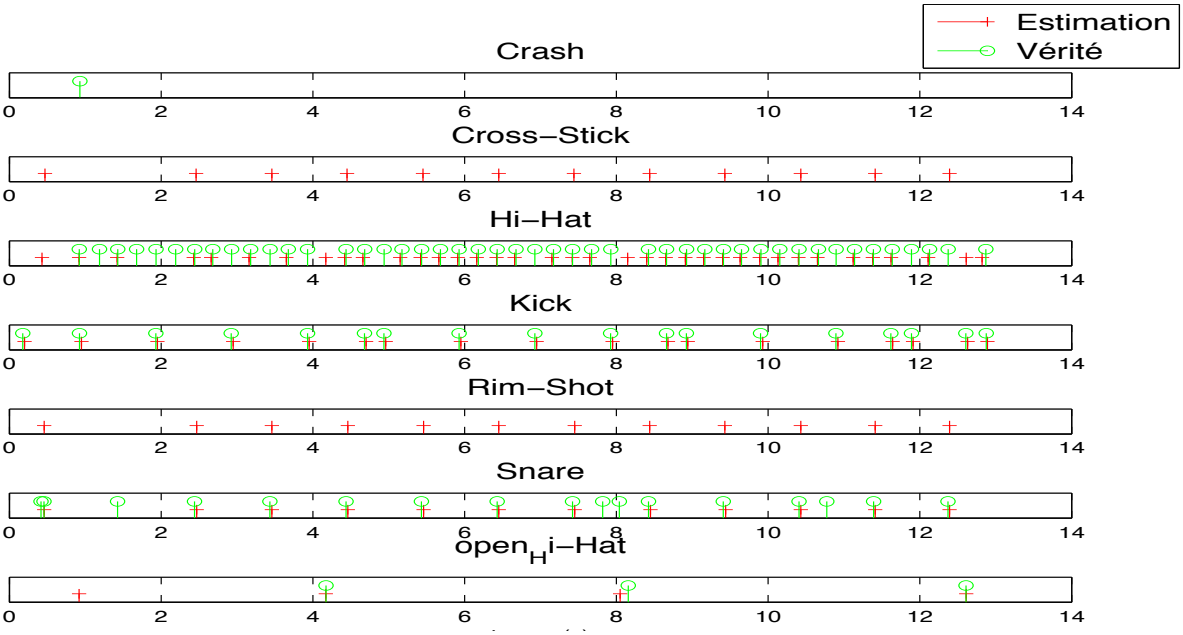


Figure V.1 – Résultat de la transcription, en vert est la verité et en rouge ma transcription

Soit K le nombre de classe (élément de la batterie) présentes dans l'estimation ou dans la vérité (*ground truth*). Soit E_k le nombre d'événements estimés dans la classe k $\forall k \in [1..K]$ de même T_k est le nombre d'événements présents en vérité dans la classe k. On définit alors les temps d'onset de la vérite :

$$t_{j,k} \in \mathbb{R}^+, \forall k \in [1..K] \text{ et } j \in [1..T_k] \tag{V.1}$$

Ainsi que les temps d'onset de l'estimation :

$$e_{i,k} \in \mathbb{R}^+, \forall k \in [1..K] \text{ et } i \in [1..E_k] \tag{V.2}$$

J'ai implémenté plusieurs solutions pour obtenir une évaluation de la transcription, à la fois les mesures standard de rappel, précision et F-mesure d'une part et la matrice de confusion d'autre par. Ces mesures sont possible en tout ou rien V.2.1 et V.2.1, c'est le cas binaire ou une note est soit bien estimée soit mal ; ou avec une tolérance gaussienne V.2.2 et V.2.2 dans quel cas une note tombant avec un petit décalage ne sera pas complètement mauvaise.

Soit m_t la marge de tolérance temporelle,si un événement estimé est distant de plus de cette valeur d'une vérité il est compté comme faux dans le cas binaire. La valeur choisit pour cette marge est 30ms, comme il est fait dans le MIREX 2005.

V.2.1 Mesure sans tolérance gaussienne

Précision, Rappel et F-mesure

Dans cette section est présenté la mesure la plus communément utilisée. Le but est de calculer les taux de rappel, de précision et la F-mesure qui combine ces deux résultats. Le taux de rappel sert à mesurer le nombre d'événements vrais bien détectés et la précision le nombre de bons événement estimés. Pour cela on définit le nombre de d'événements vrai non détectés les 'faux négatifs' (fn) et le nombre d'événement estimé sans correspondance avec un événement vrai, c'est à dire les événements détectés en trop, on les appelle les 'faux positifs' (fp). Le calcul de le précision et du rappel s'effectue alors pour une classe k :

$$\mathcal{P}_k = \frac{E_k - fp_k}{E_k} \tag{V.3}$$

$$\mathcal{R}_k = \frac{T_k - fn_k}{T_k} \tag{V.4}$$

Si $E_k = 0$ alors $\mathcal{P}_k = 1$ et si $T_k = 0$ alors $\mathcal{R}_k = 1$. La F-mesure propose une vision globale de ces deux taux :

$$\mathcal{F}^k_{mes} = \frac{2\mathcal{R}_k\mathcal{P}_k}{\mathcal{R}_k + \mathcal{P}_k} \tag{V.5}$$

On obtient donc trois score par classe, voici le tableau correspondant à la transcription V.1

	$Precision$	$Recall$	$F-meas$
$Crash$	100	0	0
$Cross-Stick$	0	100	0
$Hi-Hat$	89.2	71.7	79.5
$Kick$	100	100	100
$Rim-Shot$	0	100	0
$Snare$	100	70.6	82.8
$open_Hi-Hat$	50	66.7	57.1

Table V.1 – Précision, Rappel et F-mesure pour la transcription

Matrice de confusion

Une autre évaluation plus complète des résultats peut être obtenue en gardant l'information entre les classes, il est en effet probable qu'un faux positif dans une classe corresponde à un faux négatif dans une autre. On construit donc une matrice de confusion qui permet de visualiser ces confusions entre classes.

V.2.2 Mesure avec tolérance gaussienne

Précision, Rappel et F-mesure

Pour éviter le cas binaire on aimerait pouvoir donner une note à chaque événement qui ne soit pas 0 ou 1, mais un réel entre 0 et 1. Si par exemple on veut permettre une tolérance gaussienne sur les instants des événements détectés et que une estimation distante de m_t d'une vérité n'ai plus une note de 0 mais 0.5 par exemple. On propose donc une méthode de calcul de la F-mesure avec tolérance gaussienne.

On commence par calculer la matrice des différences des temps d'onsets entre les vérités et les estimations, puis on calcul la valeur de la loi gaussienne centré de variance proportionnelle à la marge de tolérance, la matrice obtenue est normalisée pour avoir des 1 dans le cas d'une estimation parfaite.

$$[M_k]_{i,j} = \mathcal{N}\Big((e_{i,k} - t_{j,k})|0, \sigma\Big) * \sigma\sqrt{2\pi} \tag{V.6}$$

Estimation \ Vérité	$Crash$	$Cr-Stick$	$Hi-Hat$	$Kick$	$Rim-Shot$	$Snare$	$open-HH$	NC
$Crash$	**0**	0	0	0	0	0	0	0
$Cross-Stick$	0	**0**	0	0	0	0	0	12
$Hi-Hat$	0	0	**33**	0	0	1	1	2
$Kick$	0	0	0	**18**	0	0	0	0
$Rim-Shot$	0	0	0	0	**0**	0	0	12
$Snare$	0	0	0	0	0	**12**	0	0
$open_Hi-Hat$	1	0	0	0	0	0	**2**	1
NC	0	0	13	0	0	3	0	0

Table V.2 – La matrice de confusion correspondant à la transcription V.1, les valeurs sont indiquées en nombre d'événements

On choisit le facteur de proportionnalité tel que la gaussienne soit à la moitié de son maximum pour un écart égal à la marge de tolérance $\sigma = 1.177 m_t$, une estimation distante de m_t d'une vérité aura donc bien une note de 0.5.

Ensuite la precision et le rappel sont obtenue $\forall k \in [1..K]$:

$$\mathcal{P}_k = \frac{\sum_{i=1}^{E_k} \underset{j}{max}([M_k]_{i,j})}{E_k} \tag{V.7}$$

$$\mathcal{R}_k = \frac{\sum_{j=1}^{T_k} \underset{i}{max}([M_k]_{i,j})}{T_k} \tag{V.8}$$

	$PrecisionG$	$RecallG$	$F-measG$
$Crash$	100	0	0
$Cross-Stick$	0	100	0
$Hi-Hat$	84.9	68.3	75.7
$Kick$	92.7	92.7	92.7
$Rim-Shot$	0	100	0
$Snare$	91.6	69.2	78.9
$open_Hi-Hat$	54.8	73	62.6

Matrice de confusion avec tolérance Gaussienne

Pour obtenir la matrice de confusion avec une tolérance gaussienne, je construis deux signaux de transcription qui représente la transcription, un pour la vérité et un pour l'estimation et un signal de pondération. Le premier indique les classes présentes (codage en base 2). On considère qu'une classe est présente à un instant s'il existe une gaussienne centré sur un de ses temps d'onset dont la valeur dépasse un seuil th, pour la vérité le "signal de classe" c_t est :

$$\begin{aligned} c_{t,k}(n) &= 1 \quad \text{si } \left[\sum_{j=1}^{T_k} \mathcal{N}(\tfrac{n}{fe}|t_{j,k},\sigma)\right] > th \\ &= 0 \quad \text{sinon} \end{aligned} \tag{V.9}$$

$$c_t(n) = \sum_{k=1}^{K} c_{t,k}(n) * 2^{k-1} \qquad \forall n \in [1..N] \tag{V.10}$$

Le taux d'échantillonnage fe importe peu, on prend par exemple $fe = \frac{1}{m_t} * 50$. Ce sur-échantillonnage sert à la précision des pondérations. Si fe est trop petit la précision sera moindre.

On construit de la même façon les signaux $c_{e,k}$ et c_e.

L'autre partie du signal est le poids à donner à chaque échantillon. On considère que les informations de $NC_{est} = NC_{tru}$ c'est à dire les 'Vrai Négatif' ne sont pas intéressantes, c'est pourquoi on donne du poids aux échantillons seulement lorsqu'il y à présence d'une estimation ou d'une vérité. Le poids donné aux échantillons est un composite des gaussiennes centrées autour des $e_{i,k}$ et des $t_{j,k}$.

$$w(n) = \frac{\sum_{k,j} \mathcal{N}(\frac{n}{fe}|t_{j,k}, \sigma) + \sum_{k,i} \mathcal{N}(\frac{n}{fe}|e_{i,k}, \sigma)}{max\Big(1, \sum_{k=1}^{K}[c_{t,k}(n) + c_{e,k}(n)]\Big)} * \frac{1}{fe} \tag{V.11}$$

On commence par construire une matrice de confusion intermédiaire en remplissant la matrice par la somme des poids.

$$M_{i,j} = \sum_{\{n \in [1..N] | c_e(n)=i, c_t(n)=j\}} w(n) \tag{V.12}$$

Cette matrice M est ensuite décodée pour retrouver les bases originales. Le choix de σ et th détermine les résultats obtenue dans la matrice de confusion. On souhaite qu'une note bien détecter ($e_{i,k} = t_{j,k}$) apporte une valeur de 1 à la matrice de confusion sur la diagonale, en choisissant le seuil $th = \mathcal{N}(3\sigma|0, \sigma)$, on s'assure de cette condition. En effet $[-3\sigma, 3\sigma]$ est l'intervalle de normalité à 99.7% ($\int_{-3\sigma}^{3\sigma} \mathcal{N}(x|0, \sigma)dx = 0.997$). Or la somme que l'on effectue est l'approximation rectangulaire de l'intégrale. (c'est ici qu'intervient la fréquence d'échantillonnage sur l'erreur d'approximation.)

On souhaite deuxièmement qu'une note détectée distante de m_t d'une note vrai apporte une contribution de 0,5 dans la matrice de confusion, on fixe donc $\sigma = \frac{m_t}{3}$. Le facteur 3 pour les calculs de th et σ peut être augmenté pour plus de précision en prenant garde d'augmenter aussi fe en conséquence.

Pour notre exemple les signaux de transcriptions sont présentés dans la figure V.2.

La matrice de confusion obtenue alors après décodage est, l'unité est aussi en nombre de note :

Estimation \ Vérité	$Crash$	$Cr-Stick$	$Hi-Hat$	$Kick$	$Rim-Shot$	$Snare$	$open-HH$	NC
$Crash$	0	0	0	0	0	0	0	0
$Cross-Stick$	0	0	2.64	0	0	0.41	0	9.18
$Hi-Hat$	0	0	31.1	0.09	0	0.23	0.99	4.27
$Kick$	0.01	0	0.12	16.9	0	0.01	0	0.72
$Rim-Shot$	0	0	0	0	0	0	0	12.1
$Snare$	0	0	0	0	0	11.3	0	0.657
$open_Hi-Hat$	0.99	0	0	0.01	0	0.99	2	0.02
NC	0.02	0	12.6	1.13	0	3.63	0.01	0.18

V.2.3 Matrice de confusion pour le rappel et la précision

Pour la visualisation des matrices de confusion le nombre de frappes n'est pas forcément la grandeur la plus représentative. En effet si l'on veut étudier l'efficacité d'un algorithme sur une base de test il faudra moyenner sur la base, or le nombre de frappes dans chaque classe dépend clairement de chaque morceau. Une normalisation est alors nécessaire à chaque morceau, on peut normaliser selon les lignes ou selon les colonnes. Dans le premier cas on retrouve dans la diagonal la mesure de la précision les autres valeurs non nulles de la ligne sont les classes vraiment présentes aux instants où l'on a détecté à tord la classe de la ligne, c'est la matrice de confusion pour la précision. Dans le deuxième cas on retrouve sur la diagonal les mesures de rappel et les autres valeurs non nulles de la colonne sont les classes qui ont été détectées à la place de la classe correspondant à cette colonne, c'est la matrice de confusion pour le rappel.

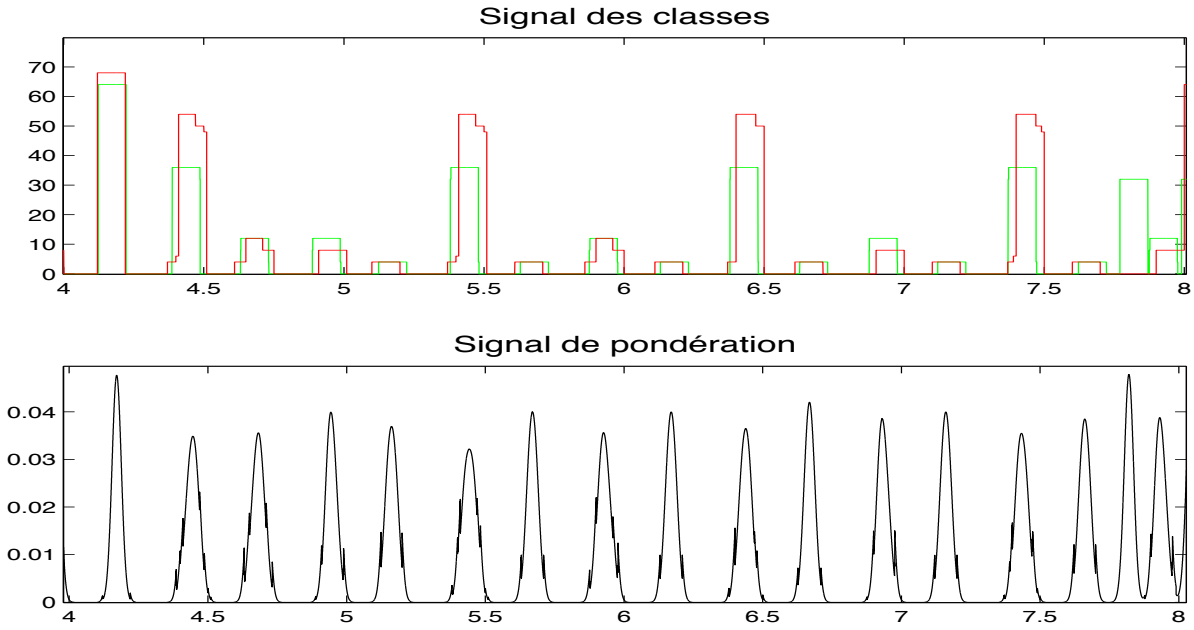


Figure V.2 – Zoom sur les signaux de transcription : en vert le signal de transcription de la vérité $\mathbf{c_t}$ et en rouge le signal de transcription estimé $\mathbf{c_e}$. Les poids attribués à chaque échantillon $\mathbf{w}$ est dans le graph du bas

V.2.4 Performance final

Le tout est maintenant de donner une note de performance globale à cette transcription. La F-mesure peut être adaptée pour la taxonomie de niveau 1, mais en tirant des informations de la matrice de confusion on pourrait avoir des valeurs plus intéressantes que la F-mesure pure avec la taxonomie de niveau 3. Par exemple en pénalisant plus fortement certaines confusions et en tolérant certaines autres. Mais une telle mesure de performance n'a pas été concrétisée durant le stage, car elle n'aurait pas permit de comparaison avec l'état de l'art. Don malgré les tentatives menées au cours de ce stage pour trouver une meilleure mesure, on utilise finalement la F-mesure. La note totale d'une transcription est alors la moyenne des F-mesures, pondérées par le nombre d'événements (vrais ou estimés) de chaque classe.

V.3 Récapitulatif

Pour résumer le processus de transcription en quelques points :

1. **L'apprentissage** : apprendre, à partir d'un ensemble de sons isolés de chaque éléments de la batterie, un dictionnaire de mot ou atomes qui servira à décomposer le signal. Ces atomes sont des spectrogrammes que l'on peut voir comme des matrices. Pour que les vecteurs d'activation ait un véritable sens il est important de normaliser ces atomes. La normalisation se fait donc par la norme de Frobenius, il s'agit d'une normalisation énergétique sur la matrice. $\|M\|_{fro} = \sqrt{\sum_{i,j} |M_{i,j}|^2}$.
2. **La décomposition** : à travers une NMF contrainte en continuité et parcimonie obtenir la matrice H des activation des atomes du dictionnaire, et appliquer à chaque itération de la NMF les contraintes sémantique sur H (contraintes intra et inter-classe).

3. **La détection** : en utilisant une détection de pics avec seuillage sur les activations totales de chaque élément, construire une transcription.

A cela peut être ajouté la **mesure de performance** qui, à travers le calcul de la F-mesure sur chaque classe pondérée par le nombre d'événements présents ou détectés, donne une note générale à la transcription.

Une vue d'ensemble est proposée en V.3.

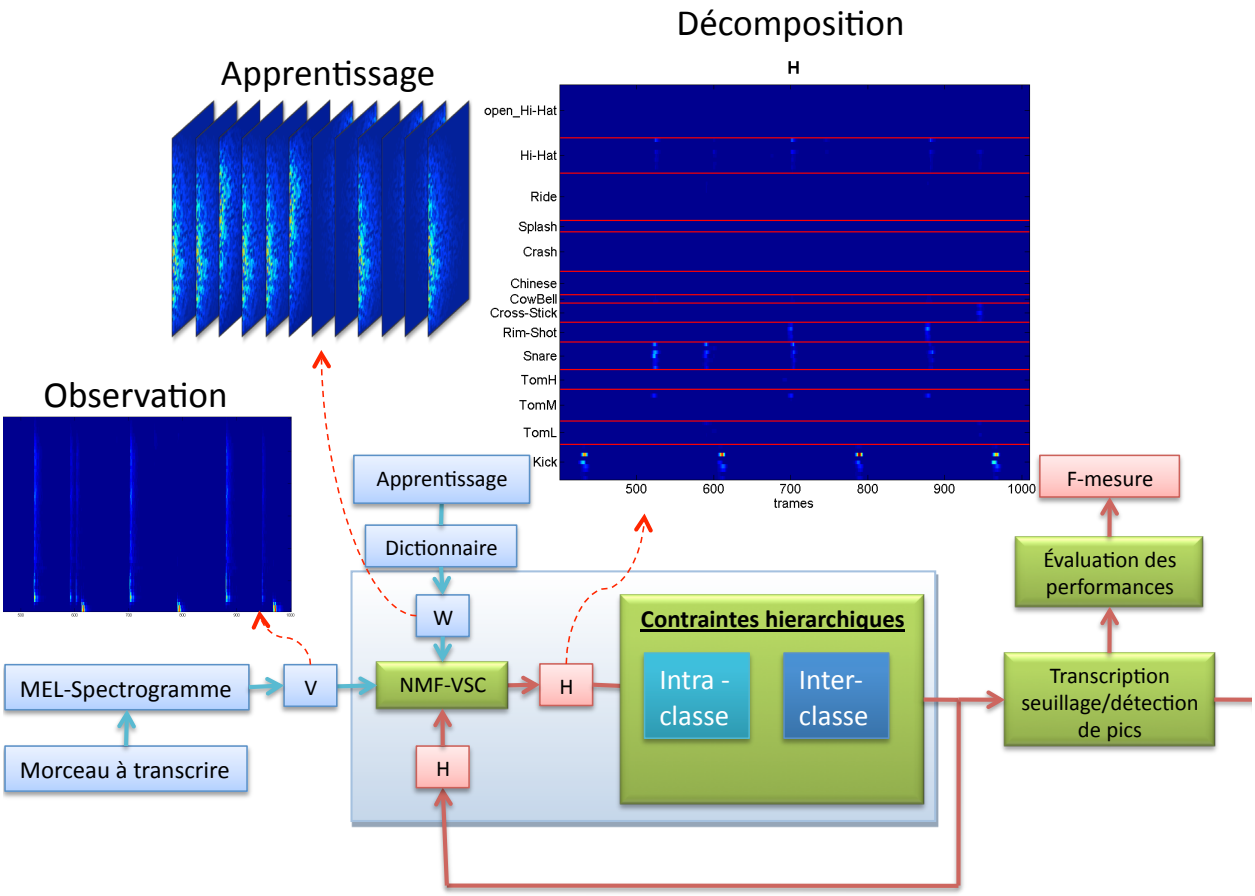


Figure V.3 – Vue d'ensemble du processus de transcription

Chapitre VI

Résultats et Analyse

VI.1 Bases de données

VI.1.1 Apprentissage du dictionnaire

Les mots du dictionnaire ont étés appris sur différentes bases. La ENST-public-drum, qui propose pour chaque éléments une séquence de frappes isolées et permet donc de les apprendre séparément. De même on a utilisé les échantillons de batterie acoustique de la base kontakt (Real Drum Kit 1,2 et 3) qui contient pour chaque éléments de chaque "kit" plusieurs échantillon à différentes intensité de frappes. Dans chaque cas une base est apprise pour chaque éléments, i.e. on effectue la NMF sur chaque signal avec $K = 1$ donc une seule base, soit un vecteur d'activation et un spectrogramme. La longueur des specrogramme choisit est de 500ms qui est une longueur intermédiaire entre les sons très longs des cymbales qui peuvent durer 3s et les sons bien plus court de Hi-Hat fermé par exemple.

NMF non contrainte

Pour apprendre les bases j'ai voulu comparer trois méthodes, la première est l'utilisation d'une NMF convolutive usuelle. Le résultats se trouve en figure VI.1. On observe une lente décroissance de la fonction de coût, les bases résultantes sont aussi très granuleuses ce qui rend la décomposition d'autant plus sensible au particularité de chaque frappe, ce qui ne permet aucune généralisation.

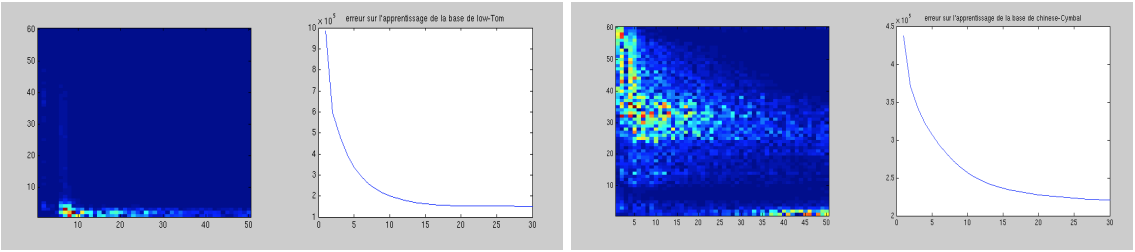


Figure VI.1 – Exemple de bases apprises pour (a) le tom basse, et (b) la cymbale chinoise. La courbe sur la droite des bases apprises est la valeur de la fonction de coût en fonction du nombre d'itération ici de 1 à 30 itérations

NMF-VSC

Cette fois-ci on utilise la NMF contrainte en parcimonie et continuité de Virtanen (NMF-VSC). La décroissance de la fonction est plus rapide mais cela ne résout pas le problème de granularité.

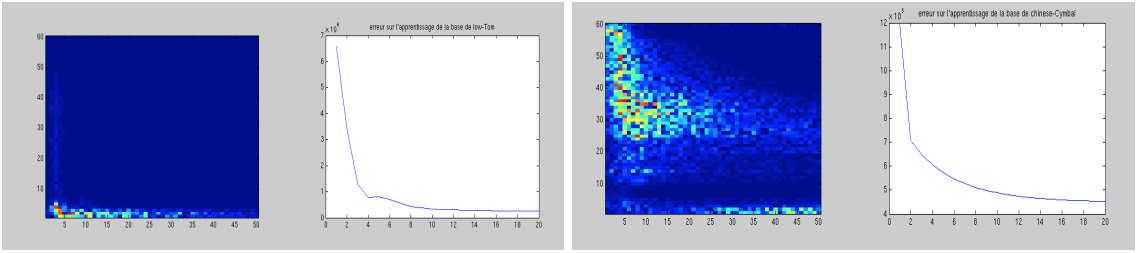


Figure VI.2 – Exemple de bases apprises pour (a) le tom basse, et (b) la cymbale chinoise. La courbe sur la droite des bases apprises est la valeur de la fonction de coût en fonction du nombre d'itération ici de 1 à 20 itération

NMF-VSC + initilisation mar moyennage sur les onsets

Enfin pour accélérer encore la NMF, j'utilise une initialisation pour W. Au lieu d'initialiser les deux matrice W et H aléatoirement comme dans le processus habituel, j'initie la matrice W en détectant les onsets sur la piste, puis la matrice W est initialisée à la moyenne des Mel-Spectrogrammes constitués des 500ms suivant chaque onset. La décroissante est très rapide et les mots du dictionnaires sont lissés. C'est donc la solution retenue.

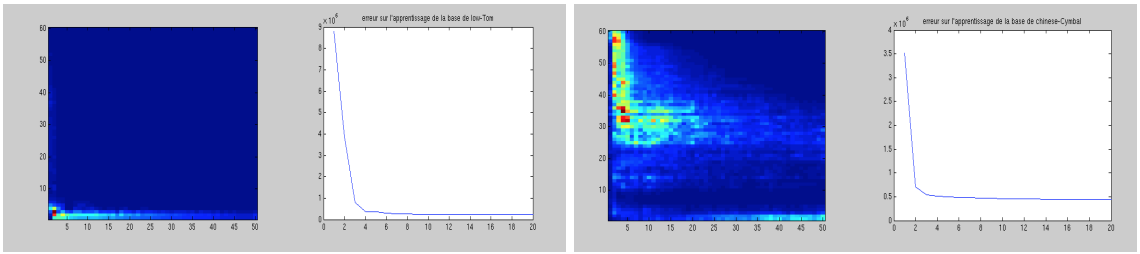


Figure VI.3 – Exemple de bases apprises pour (a) le tom basse, et (b) la cymbale chinoise. La courbe sur la droite des bases apprises est la valeur de la fonction de coût en fonction du nombre d'itération ici de 1 à 20 itération

L'ensemble du dictionnaire appris est présenté en annexe A.

VI.1.2 Bases de tests

Pour tester la méthode on a utilisé plusieurs bases de données qui ont des niveaux croissant de complexité.

ENST

Composés de 14 phrases tirées du corpus de ENST-drums. La caractéritique de cette base est que les atomes présents dans le signal à transcrire sont dans le dictionnaire, car ceux sont les mêmes éléments qui ont servit à l'apprentissage.

MASS-GT

Composée de 7 extraits de morceau annotés à la main. Dans la MASS-GT seules sont présentes les pistes de batteries donc il s'agit de musique monophonique dans le sens ou seul un instrument est présent. Mais dans ce cas la contrairement à l'ENST-public les atomes optimaux pour décrire le signal ne sont pas à priori dans le dictionnaire.

MASS-Sep

Cette base est composée des 7 mêmes extraits mais qui contiennent une séparation non parfaite de la partie percussive. Ceux sont les résultats de la séparation réalisée par l'algorithme de Rigaud [RLRG11]. Cette fois les bases apprises ne sont pas dans le signal et de plus le signal contient des artefacts harmoniques qui diminue la qualité de transcription.

VI.2 Résultats généraux

VI.2.1 L'aléatoire

Un premier repaire peut être fixée comme limite basse, le cas de génération aléatoire des transcriptions. Pour générer ces transcriptions aléatoires j'ai procédé de deux manière différentes, la première ou l'on suppose connues les classes présentes dans la vrai transcription ainsi que le nombre de frappes dans chaque classe (Prior-Alea) et l'autre ou l'on ne connait pas à priori les éléments présents (Full-Alea) ni leur nombre de notes. Les transcriptions sont générées en prenant aléatoirement N réels entre le début et la fin de l'extrait (distribution uniforme). Si deux événements générés sont proches à moins de 30ms seul un est gardé. Dans le cas Prior-Aléa N est égal au nombre d'événements de la vérité par classe, et dans le cas Full-Aléa pour chaque classe N est un entier prit aléatoirement entre 0 et 20 (distribution uniforme). Les résultats présentés dans le tableau VI.1 sont obtenus en moyennant, pour chaque morceau de chaque base, sur 100 générations de transcription aléatoire. Le résultat est finalement la moyenne sur chaque base.

F-mesure (%)	ENST	MASS
Prior-Alea	16.6	16.2
Prior-Alea-H1	20.7	19.3
Full-Alea	4.1	3.3
Full-Alea-H1	18.8	21.9

Table VI.1 – F-mesure moyenne pour chacune des bases de test, pour des transcriptions aléatoires

VI.2.2 L'Arrêt anticipé

La NMF utilisée respecte en pratique la décroissance de la fonction de coût, cependant la décroissance de la fonction de coût n'est pas corrélées à l'amélioration des performances de transcription. Ce problème est d'ailleurs un des désavantages de la NMF, l'impossibilité de relier la valeur de la fonction de coût à un valeur sémantique de la décomposition. L'algorithme à donc été réaliser sur 150 itérations et les graphique suivants montre que dans certain cas il est important de s'arrêter avant la fin de ces 150 itérations. Nous avons donc essayé de trouver une bonne valeur pour cet arrêt anticipé. Par exemple dans la figure

Ces graphiques [VI.4 ; VI.5 ; VI.6] montrent de manière claire la complexité croissante des bases de test, l'implication est directe sur la F-mesure. Les contraintes hiérarchiques (courbe bleu à partir de 10 itération et courbe rouge dès la première itération) apporte surtout un effet de vitesse sur la meilleur transcription obtenue. Par contre elles détériorent la qualité de la transcription si l'on les laisse tourner trop longtemps. Il est donc important de stopper l'algorithme pour obtenir une transcription de meilleur qualité. La valeur de l'early stop peut être différent pour les trois bases de test qui représentent 3 niveaux de complexité. Aux vues de ces figures nous prendrons pour la transcription final de l'ENST la valeur à la $25^{\text{ième}}$ itération qui à la meilleur F-mesure moyenne en appliquant les contraintes hiérarchiques seulement à partir de la $10^{\text{ième}}$ itération. Pour MASS-GT et MASS-Sep le "Early-stopping" est fixé à la $20^{\text{ième}}$ itération. Les résultats avec adaptations des atomes ne sont pas montré car leurs effet sont faibles et arrivent bien après les max de la F-mesure moyenne.

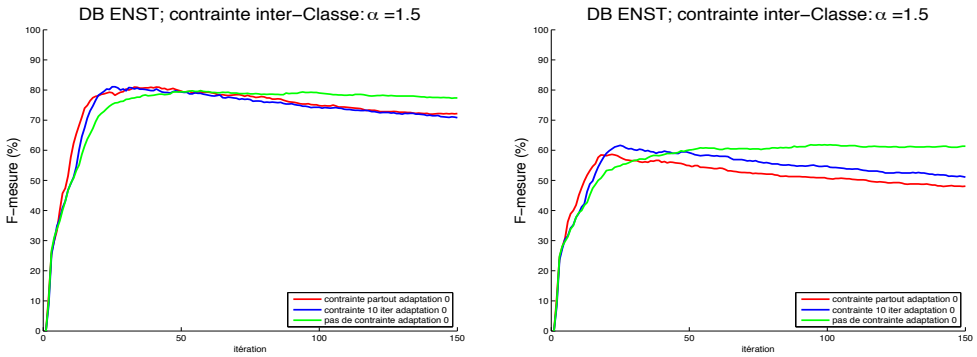


Figure VI.4 – Moyenne des Fmesure à chaque itération pour α =1.5, pour une taxonomie H1 (a) et H3 (b) sur l'ENST.

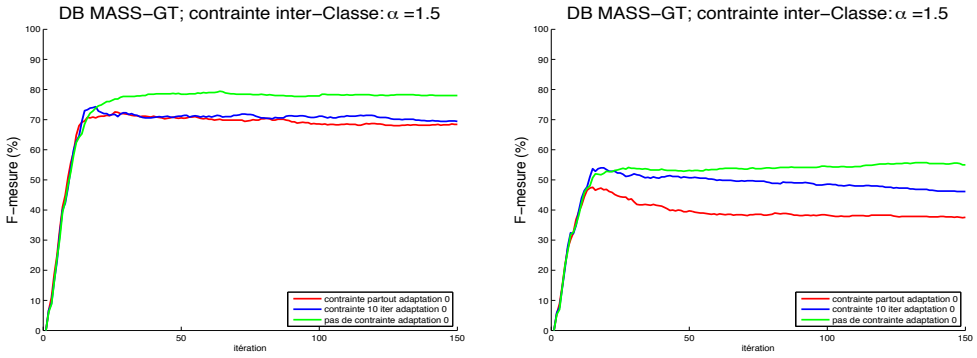


Figure VI.5 – Moyenne des Fmesure à chaque itération pour α =1.5, pour une taxonomie H1 (a) et H3 (b) sur l'MASS-GT.

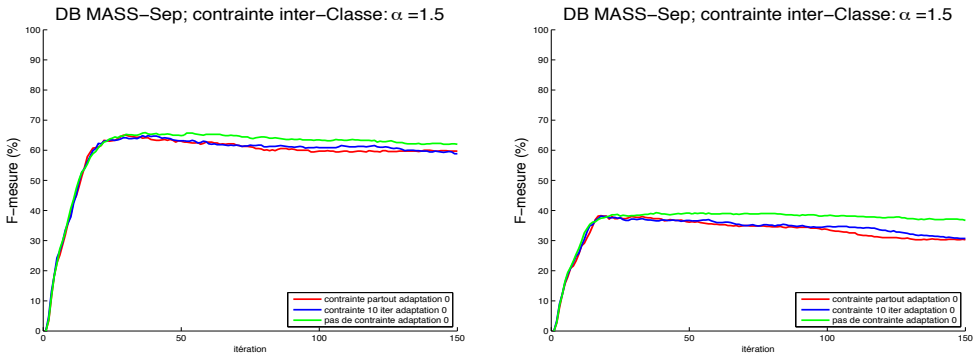


Figure VI.6 – Moyenne des Fmesure à chaque itération pour α =1.5, pour une taxonomie H1 (a) et H3 (b) sur l'MASS-Sep.

VI.2.3 Amélioration des contraintes

La contrainte inter-classe permet en pratique de s'affranchir un peu de la valeur de seuil à donner à chaque classe, en effet en ne gardant qu'une seule classe active par groupe à chaque instant elle limite l'influence du seuil des autre classes. Ceci est très intéressant car il est en fait impossible de trouver un seuil optimal par classe. J'ai effectué une détection avec seuil optimale c'est à dire que pour chaque morceau, chaque itération et chaque classe on trouve le seuil optimal (i.e. celui qui maximise la F-mesure de la classe). La figure VI.7 permet de mettre en avant l'avantage de cette contrainte quand ce seuil optimale ne peut être obtenue. On observe en effete que dans le cas avec seuil optimal si l'on ajoute la contrainte la F-mesure est détériorée, alors que si l'on ajoute la contrainte quand on ne connait pas le seuil optimal la performance moyenne est améliorée.

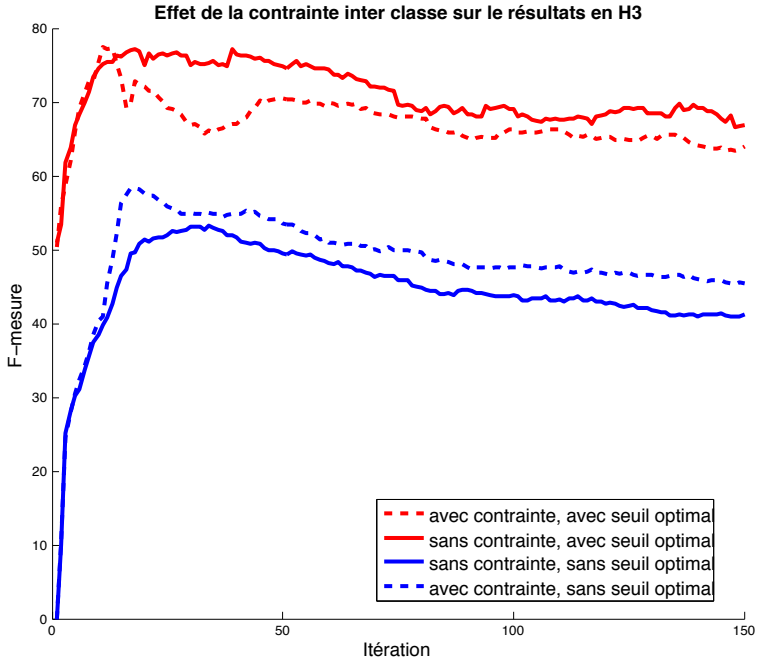


Figure VI.7 – Effet de la contrainte inter-classe sur les valeurs de seuil : F-mesure dans quatre cas différents.

VI.3 Résultats par Classe

Les résultats suivants sont obtenus en prenant la transcription pour la $20^{\text{ième}}$ itération. Avec le set de paramètre choisit :

- pour la NMF-VSC : parcimonie $\lambda = 5.10^{-4}$ et continuité : $\gamma = 2.10^{-7}$
- contrainte intra-classe : $\alpha = 2$
- contrainte inter-classe : $\alpha = 1.5$ les autres valeurs sont trop grande et ne laisse pas la NMF compenser les modification de la contrainte hiérarchique.
- les contraintes s'appliquent à partir de la $10^{\text{ième}}$ itération pour laisser le temps au éléments prédominant de "prendre le dessus".

On observe alors les résultats suivants pour chaque degré de complexité de la tache et chaque niveau de taxonomie, ils sont présentés dans VI.2. Chaque performance (précision, rappel ou F-mesure est moyennée sur toute la base. Cette moyenne explique que la F-mesure affichée ne corresponde pas au calcul à partir des précision et rappel affichés ($\mathcal{F} = \frac{2\mathcal{P}\mathcal{R}}{\mathcal{R}+\mathcal{P}}$). Pour le niveau 1 les résultats obtenus pour la MASS-GT sont relativement bien placés par rapport aux résultats de l'état de l'art II.1.

$Inst\backslash Perf$	$Precision$	$Recall$	$F-meas$
$Hi-Hat$	92	63	72
$Snare$	100	79	87
$Kick$	94	95	94
$Inst\backslash Perf$	$Precision$	$Recall$	$F-meas$
$Chinese$	100	91	91
$CowBell$	100	91	91
$Crash$	60	79	66
$Cross-Stick$	82	97	80
$Hi-Hat$	68	50	37
$Kick$	93	86	89
$Ride$	69	67	49
$Rim-Shot$	36	100	36
$Snare$	98	77	85
$Splash$	100	100	100
$TomH$	73	100	73
$TomL$	3	84	4
$TomM$	100	88	89
$open_Hi-Hat$	77	58	57

(ENST)

$Inst\backslash Perf$	$Precision$	$Recall$	$F-meas$
$Hi-Hat$	75	69	62
$Snare$	82	83	82
$Kick$	78	78	77
$Inst\backslash Perf$	$Precision$	$Recall$	$F-meas$
$Chinese$	100	100	100
$CowBell$	86	100	86
$Crash$	87	77	73
$Cross-Stick$	55	98	55
$Hi-Hat$	73	68	60
$Kick$	87	82	81
$Ride$	43	86	29
$Rim-Shot$	10	98	11
$Snare$	66	64	64
$Splash$	100	100	100
$TomH$	57	89	48
$TomL$	60	61	32
$TomM$	71	86	57
$open_Hi-Hat$	56	75	47

(MASS-GT)

$Inst\backslash Perf$	$Precision$	$Recall$	$F-meas$
$Hi-Hat$	58	73	58
$Snare$	79	56	58
$Kick$	63	83	70
$Inst\backslash Perf$	$Precision$	$Recall$	$F-meas$
$Chinese$	71	100	71
$CowBell$	86	100	86
$Crash$	43	62	21
$Cross-Stick$	21	84	22
$Hi-Hat$	68	60	48
$Kick$	65	90	71
$Ride$	3	95	5
$Rim-Shot$	7	90	5
$Snare$	70	52	54
$Splash$	43	100	43
$TomH$	10	91	7
$TomL$	57	46	29
$TomM$	14	86	0
$open_Hi-Hat$	17	64	4

(MASS-Sep)

Table VI.2 – Résultats pour l'ensemble de paramètre retenu par classe et niveau hiérarchique de taxnomie. En haut en niveau 1 et en bas niveau 3.

VI.4 Les confusions fréquentes

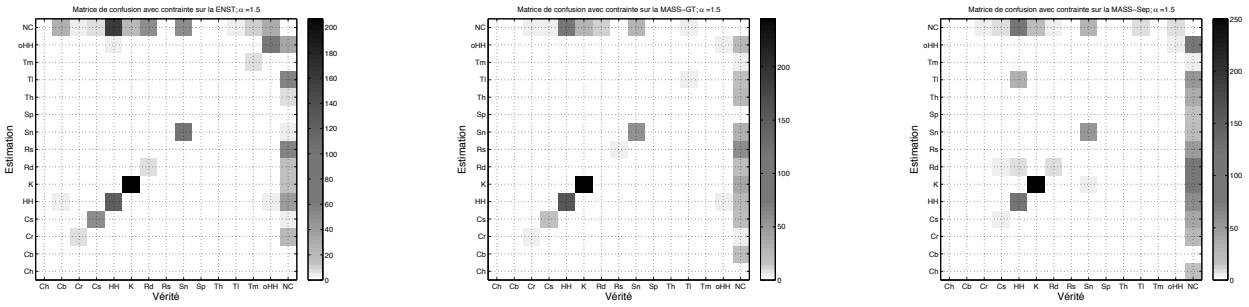
Les informations obtenues avec ces mesures de performances peuvent être complétées par les matrices de confusion VI.8 et VI.9.

$est\backslash truth$	$Hi-Hat$	$Snare$	$Kick$	NC
$Hi-Hat$	290	0	1	44
$Snare$	0	169	0	4
$Kick$	0	0	231	25
NC	190	61	17	8.38e+003
$est\backslash truth$	$Hi-Hat$	$Snare$	$Kick$	NC
$Hi-Hat$	221	1	1	46
$Snare$	0	87	0	42
$Kick$	0	2	264	74
NC	98	31	28	8.17e+003
$est\backslash truth$	$Hi-Hat$	$Snare$	$Kick$	NC
$Hi-Hat$	186	1	1	120
$Snare$	1	61	2	12
$Kick$	4	6	247	85
NC	128	53	43	8.13e+003

Figure VI.8 – Matrices de confusion en niveau 1, pour les trois bases de test. De haut en bas ENST, MASS-GT et MASS-Sep

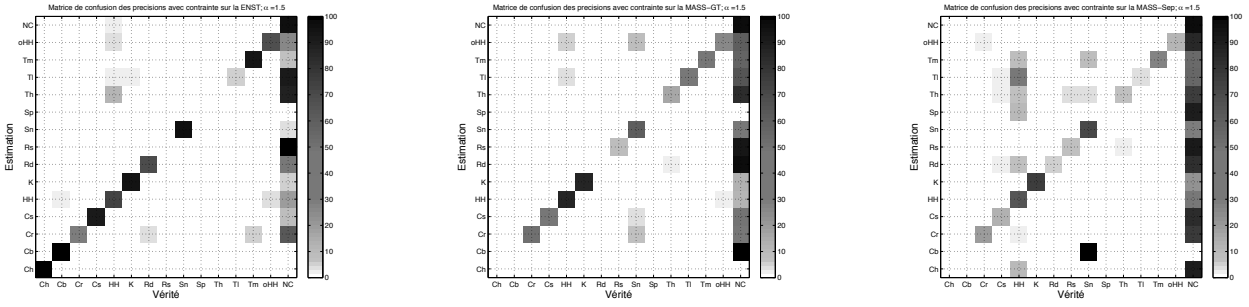
Comme la visualisation des matrices de confusion dépend trop du nombre de frappes, j'ai aussi calculé les normalisations selon les lignes et les colonnes de ces matrices (cf V.2.3). Elles permettent d'avoir des informations pour chaque classe, sur les éléments avec lesquels elles sont fréquemment confondues. Les figures VI.10 et VI.11 présentent ces matrices de confusion pour le rappel et pour la précision. Elles ont été calculées en pondérant chaque classe de chaque transcription par le nombre d'éléments présents en vérité ou détectés dans cette classe et cette transcription. Ainsi si dans un morceau il y a un seul coup de crash ce morceau aura moins de poids pour la crash qu'un morceau en contenant une dizaine.

L'observation de ces matrices montre que les confusions, bien que importantes pour certaines classes, ne sont pas les principaux problèmes. Les oublis et ajouts semblent en effet plus prédominants.



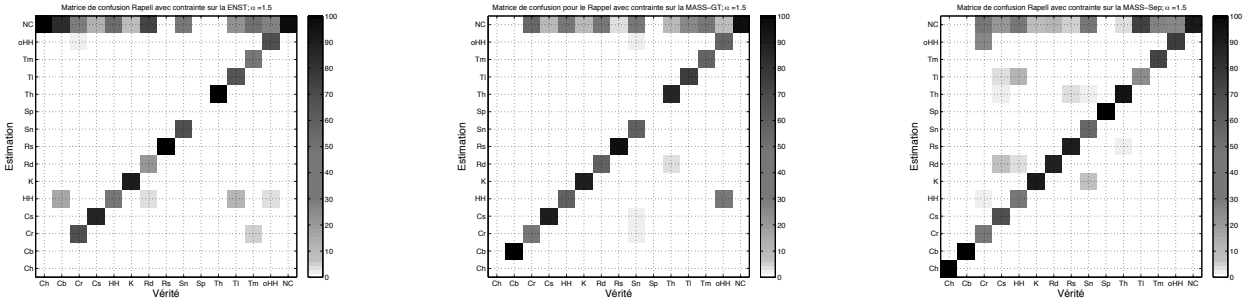
(ENST) (MASS-GT) (MASS-Sep)

Figure VI.9 – Les matrices de confusion pour les trois bases de test



(ENST) (MASS-GT) (MASS-Sep)

Figure VI.10 – Les matrices de confusion en Précision pour les trois bases de données.



(ENST) (MASS-GT) (MASS-Sep)

Figure VI.11 – Les matrices de confusion en Rappel pour les trois bases de données.

VI.5 Analyse

Il est étonnant qu'en ayant intégré autant d'à priori dans les contraintes, mais de manière relativement souple, le résultats reste si peu satisfaisant. L'avantage principal de l'ajout des contraintes hiérarchiques est qu'il accélère notoirement la croissance de la F-mesure par rapport au nombre d'itération, mais dans le même temps rend les résultats plus sensible au nombre d'itération effectuée. Ces analyses restent cependant très

dépendante de la base de test utilisée. Il faudrait pour pouvoir tirer de réelles conclusions, construire une base de test de plus grande envergure, mais la tache d'annotation à la mains des pistes nécessite beaucoup de temps, même avec les pistes séparées. Et celles-ci sont rarement accessibles dans la musique enregistrée à moins d'avoir les droits. On aurait pu générer une base de test à partir de boucle MIDI et ajouter une partie instrumentale, mais le contexte u MIDI n'apporte pas des conditions de variabilité sonore équivalentes à l'enregistrement. Pour obtenir enfin des résultats comparable à l'état de l'art il faudrait soit avoir eu a disposition la même base de test, soit ré-implémenter les différentes méthodes ce qui aurait demandé d'y consacrer beaucoup de temps.

Chapitre VII

Conclusion

L'approche par NMF convolutive pour la transcription automatique apporte des résultats intéressants. Les principaux atouts de cette méthode étant la perspective convolutive particulièrement importante dans le cas de la batterie.

D'autre part le fait de connaitre le dictionnaire à priori, appris par NMF sur des sons isolés, fournit les modèles spectro-temporel des instruments pouvant être présents dans les signaux à traiter et aide ainsi la transcription.

Grâce enfin aux contraintes appliquées sur la fonction de coût on arrive à faciliter l'étape de détection d'onset sur les pistes séparées (les activations par classe).

Le problème de ne pouvoir avoir un dictionnaire exhaustif, bien que sur-complet, est en parti résolu grâce à des considérations d'ordre sémantique, grâce auxquelles ont été introduites les contraintes intra et inter-classe.

L'évaluation des méthodes de transcriptions, du moins dans le cas de la batterie, reste assez basique dans la littérature, ce qui ne permet pas d'obtenir un score qui ait une valeur sémantique forte, c'est pourquoi sont proposées quelques alternatives à l'évaluation basique par F-mesure, mais n'ont malheureusement pas été finalisées.

Je tiens à remercier mes encadrants de stage, de m'avoir permit de travailler sur ce sujet, interressant dans ces applications et ces mises en place.

Bibliographie

[BBF] Nancy Bertin, Roland Badeau, and Cédric Févotte. A tempering approach for Itakura-Saito NMF, with application to music transcriptio. *Musicales*.

[BRB06] Juan P . Bello, Emmanuel Ravelli, and Mark B . Sandler. Drum sound analysis for the manipulation of rythm in drum loops. *ICASSP*, pages 8–11, 2006.

[EK04] Julian Eggert and Edgar Koerner. Sparse Coding and NMF. *Simulation*, 2(4) :2529–2533, 2004.

[FBD09] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence : with application to music analysis. *Neural computation*, 21(3) :793–830, March 2009.

[FI11] Cedric Févotte and Jerome Idier. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural Computation*, (3) :1–24, 2011.

[FLC03] Derry Fitzgerald, Bob Lawlor, and Eugene Coyle. Drum Transcription in the presence of pitched instruments using Prior Subspace Analysis. *ISSC*, (3), 2003.

[Gil07] Olivier Gillet. *Transcription des signaux percussifs. Application à l'analyse de scènes musicales audiovisuelles.* PhD thesis, 2007.

[GR03] Olivier K Gillet and Gaël Richard. Automatic Labelling of Tabla Signals. 2003.

[GR04] Olivier Gillet and Gaël Richard. Automatic transcription of drum loops. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2–5, 2004.

[Hen11] Romain Hennequin. *Décomposition de spectrogrammes musicaux informée par des modèles de synthèse spectrale.* PhD thesis, 2011.

[Hoy04] Patrik O Hoyer. Non-negative Matrix Factorization with Sparseness Constraints. *Journal of Machine Learning Research*, 5 :1457–1469, 2004.

[HV05] Marko Heln and Tuomas Virtanen. Separation of drums from polyphonic music using non-negative matrix factorisation and support vector machine. *Technology*, 1(1), 2005.

[LS99] D D Lee and H S Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755) :788–91, October 1999.

[MF07] Arnaud Moreau and Arthur Flexer. Drum transcription in polyphonic music using non-negative matrix factorization. *Artificial Intelligence*, 2007.

[Pau06] Jouni Paulus. Acoustic modelling of drum sounds with hidden markov models for music transcription. 2006.

[PJL96] P. Paatero, M. Juvela, and K. Lehtinen. The use of positive matrix factorization in the analysis of molecular line spectra. *Royal Astronomical Society*, pages p. 616 – 626, 1996.

[PV05] Jouni Paulus and Tuomas Virtanen. Drum transcription with non-negative spectrogram factorization. In *Proc. of the 13th EUSIPCO*, 2005.

[RLRG11] François Rigaud, Mathieu Lagrange, Axel Röbel, and Peeters Geoffroy. Drum Extraction from poylphonic music based on Spectro-temporal model of percussive sounds. In *ICASSP*, pages 381–384, 2011.

[SB03] Paris Smaragdis and Judith C. Brown. Non-Negative Matrix Factorization for Polyphonic Music Transcription. *Signal Processing*, (3) :3–6, 2003.

[Sma04] Paris Smaragdis. Non-negative Matrix Factor Deconvolution ; Extracation of Multiple Sound Sources from Monophonic Inputs. *International Congress on Independent Component Analysis and Blind Signal Separation (ICA)*, 2004.

[SMr] Mikkel N Schmidt and Morten Mø rup. Nonnegative Matrix Factor 2-D Deconvolution for Blind Single Channel Source Separation. *Source*.

[Vir07] Tuomas Virtanen. Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria. *Language*, 15(3) :1066–1074, 2007.

[YGO04] Kazuyoshi Yoshii, Masataka Goto, and Hiroshi G Okuno. Automatic drum sound description for real-worls music, using template adaptation and matching methods. *ISMIR*, (October) :184–191, 2004.

[ZPDG02] Aymeric Zils, François Pachet, Olivier Delerue, and Fabien Gouyon. Automatic Extraction of Drum Tracks from Polyphonic Music Signals. *Popular Music*, pages 7–11, 2002.

Annexe A

Le dictionnaire appris

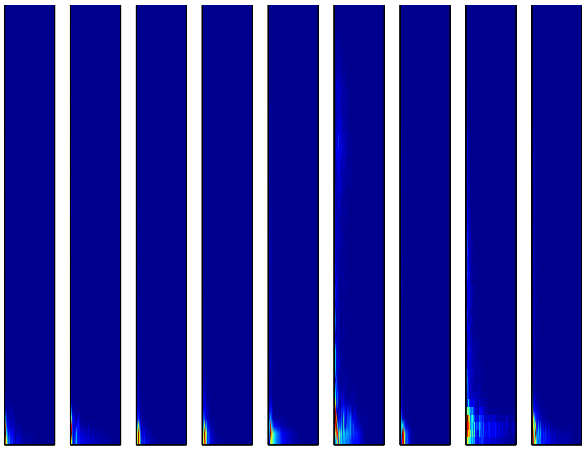


Figure A.1 – Les atomes appris pour la **Grosse caisse**

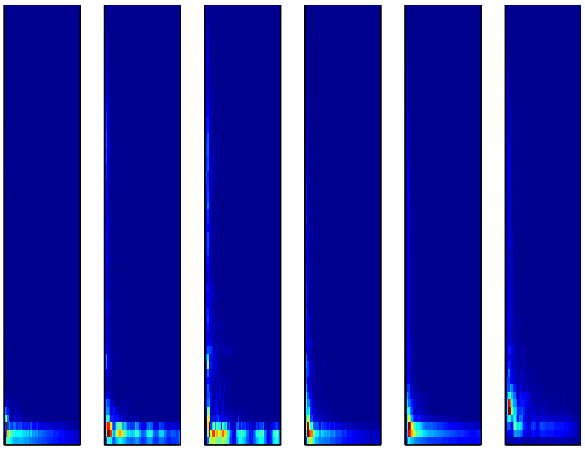


Figure A.3 – Les atomes appris pour le **Tom basse**

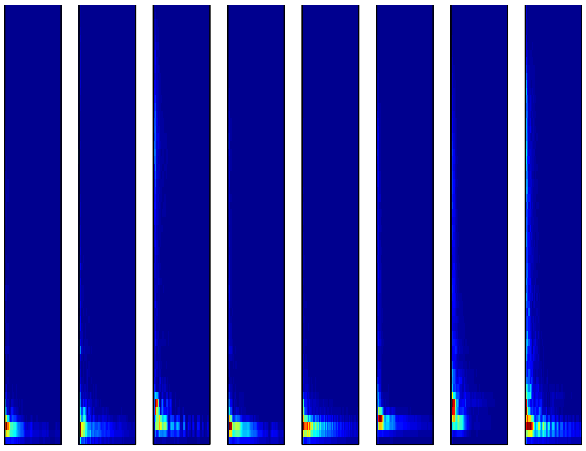


Figure A.2 – Les atomes appris pour le **Tom medium**

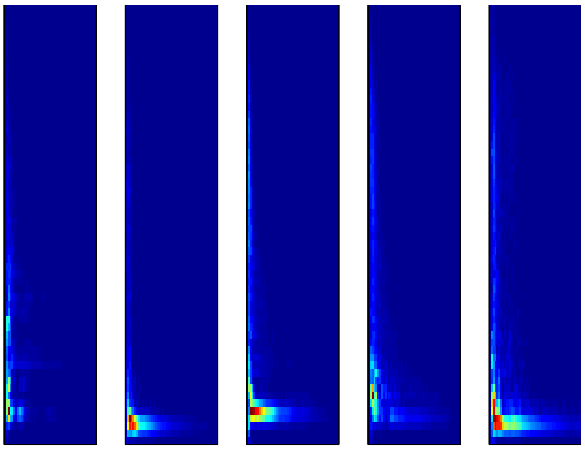


Figure A.4 – Les atomes appris pour le **Tom aigu**

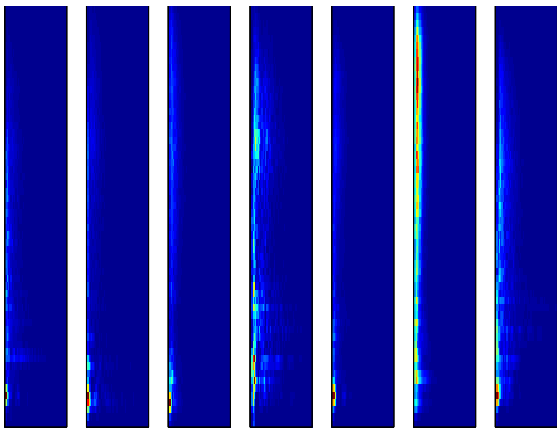


Figure A.5 – Les atomes appris pour la **Caisse claire**

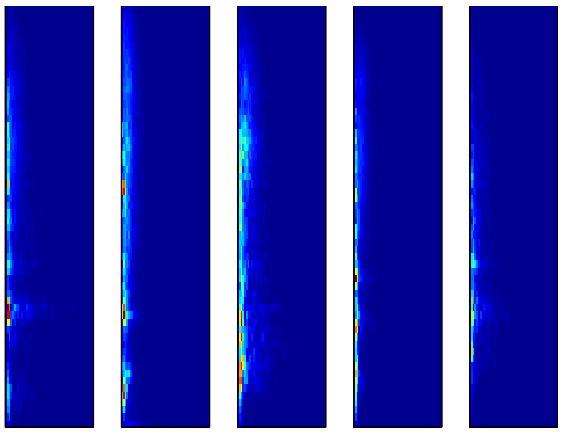


Figure A.7 – Les atomes appris pour le **Cross-stick**

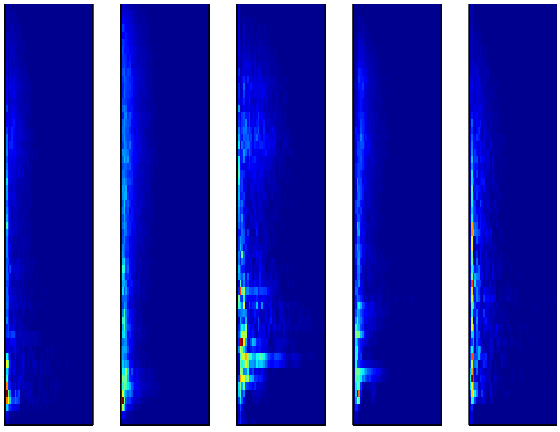


Figure A.6 – Les atomes appris pour le **Rim-shot**

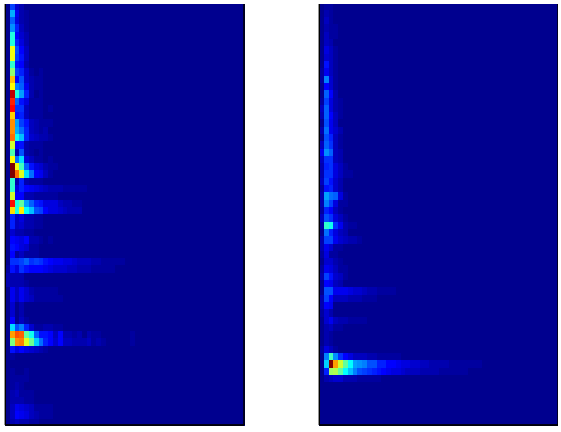


Figure A.8 – Les atomes appris pour la **Cloche**

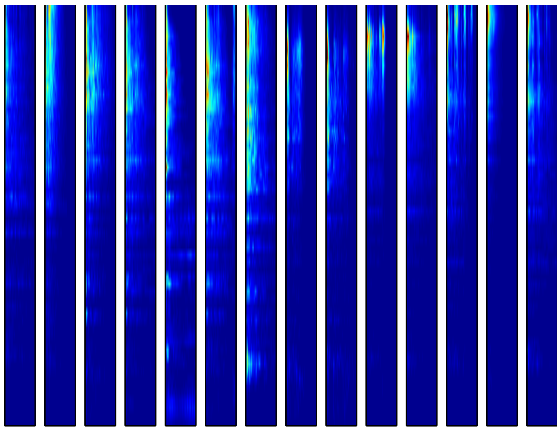


Figure A.9 – Les atomes appris pour le **Charleston ouvert**

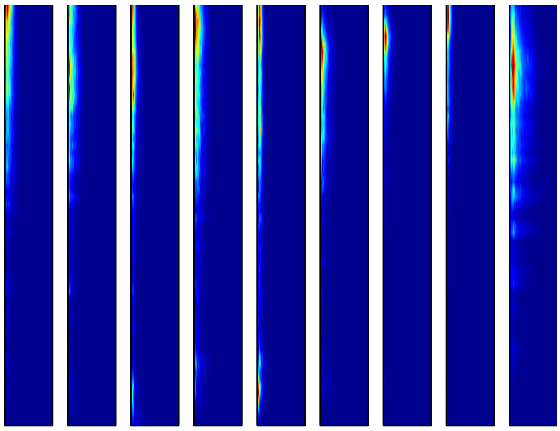


Figure A.10 – Les atomes appris pour le **Charleston**

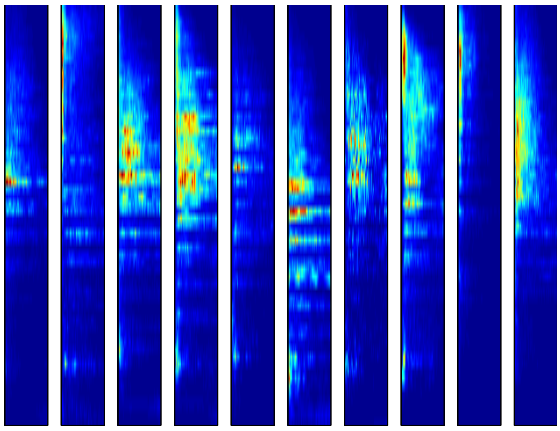


Figure A.11 – Les atomes appris pour la **Cymbale Crash**

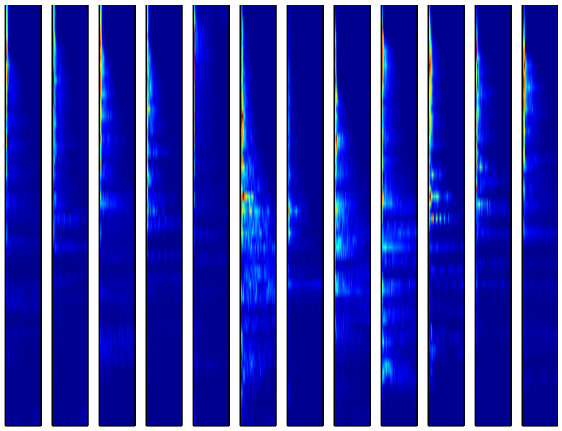


Figure A.12 – Les atomes appris pour la **Cymbale Ride**

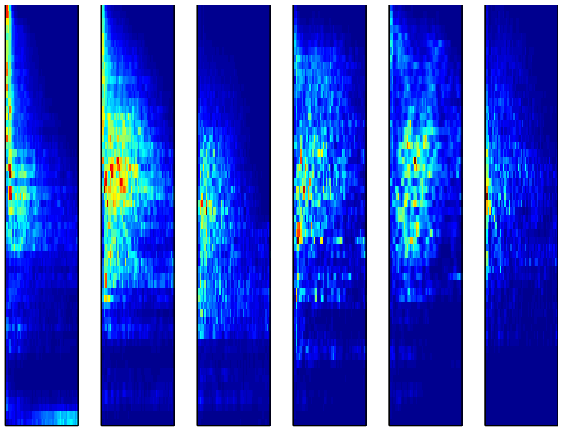


Figure A.13 – Les atomes appris pour la **Cymbale Chinoise**

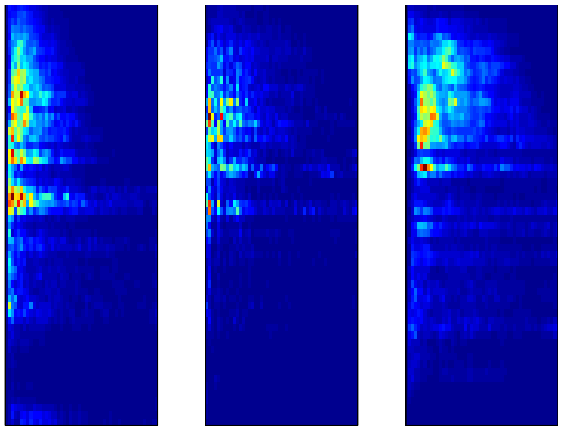


Figure A.14 – Les atomes appris pour la **Cymbale Splash**