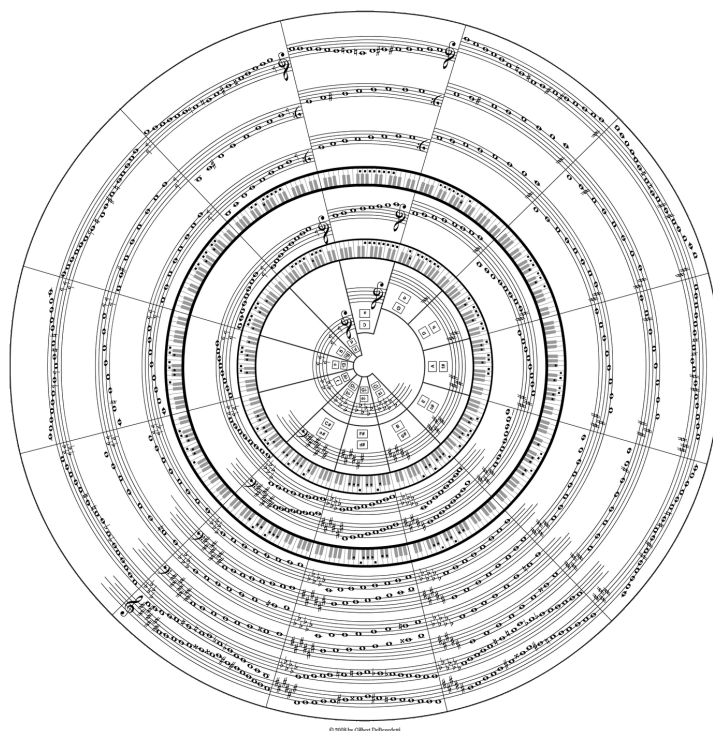


DÉPARTEMENT GÉNIE MATHÉMATIQUE

Projet de transcription de musique

RAPPORT D'AVANCEMENT DU PROJET SEMESTRIEL



Rand ASSWAD
Ergi DIBRA
Yuge SUN

A l'attention de :
Mme. Natalie FORTIER

5 mai 2018

Contents

Introduction	3
Idée du projet	3
Idée de traitement	3
Le traitement	3
Cas initial	3
Partie théorique	3
Partie appliquée	3
Détection des fréquences fondamentales	3
Analyse de notes	4
Introduction du problème	4
Gammes et intervalles	4
Nomenclature	4
Partie mathématique	5

Introduction

Idée du projet

Le but du projet est de créer un logiciel de transcription automatique de morceaux de musique, dans le cadre du projet semestriel on vise créer un version capable traiter des morc monophoniques.

Idée de traitement

On part du principe qu'un signal sonore est une série harmonique, ce qui sera expliqué en détails dans le rapport final du projet.

A partir du signal harmonique on extrait la fréquence fondamentale en fonction du temps, par la suite on transforme les fréquences en notes. Par la suite, on analyse les notes pour obtenir une suite de notes associées avec des durées.

L'étape finale fait appelle à la théorie de musique, elle consiste à extraire le *tempo* à partir des durées des notes, et de reconnaître la *gamme* du morceau à partir des notes obtenues.

Le traitement

Cas initial

Partie théorique

Le signal obtenu en entrée est un signal harmonique. C'est-à-dire qu'une note simple de durée T seconde est donnée par un signal

$$x(t) = \sum_{k \in \mathbb{N}} A_k \cdot \cos(2\pi k f_0 t), \forall t \in [0, T]$$

où f_0 est la fréquence fondamentale du signal sur $[0, T]$.

Un morceau musical peut être vu comme une suite de notes où chaque note est caractérisée par sa fréquence et sa durée. En divisant le morceau en N parties, on définit la suite $(t_i)_{i=0, \dots, N-1}$ telle que $t_i = i \cdot \frac{T}{N}$, le signal devient

$$x(t) = \sum_{i=0}^{N-1} \sum_{k \in \mathbb{N}} A_{k,i} \cdot \cos(2\pi k f_{0,i} t) \cdot \mathbb{1}_{[t_i, t_{i+1}[}(t)$$

Partie appliquée

Le stockage de signaux sonore consiste à prélever des valeurs à intervalles définis. En général, l'échantillonnage est régulier, souvent fixé à 44100 Hz (valeur par seconde).

% add python code showing an example

Détection des fréquences fondamentales

Ils existent plusieurs algorithmes de détection de fréquences fondamentales, il y'en a deux types : applications sur le domaine temporel et sur le domaine fréquentiel. Chaque type présente des avantages et des inconvénients, dans le cas d'un signal monophone les algorithmes temporel sont meilleur d'où notre choix.

Après de nombreuses recherches on a décidé d'appliquer l'algorithme de **YIN** (*Kawahara et de Cheveigné, 2002*) car il est rapide, efficace et produit des erreurs moins importantes comparé avec d'autres algorithmes. Son principe se base sur la fonction d'autocorrélation.

Les étapes de l'algorithme : 1. La méthode d'autocorrélation 2. La fonction de différences 3. La fonction de la moyenne normalisée cumulée 4. Le seuil absolu 5. L'interpolation parabolique 6. La meilleure estimation locale

On expliquera la méthode en détails dans le rapport final du projet, et on étudiera l'erreur de cette méthode. Voyons une l'application de l'algorithme sur l'exemple précédent.

% add python code showing an example

Analyse de notes

Introduction du problème

L'espace de notes est un espace linéaire discret, mais l'espace de fréquences est continue non-linéaire. Le problème consiste à trouver une fonction qui associe les fréquences fondamentales obtenues avec des valeurs entières.

Gammes et intervalles

En acoustique, un **intervalle** désigne le rapport de fréquences de deux sons. Or, en musique chaque intervalle est caractéristique d'une échelle musicale, elle-même varie selon le type de musique. En musique, une **gamme** est une suite de notes conjointes où la fréquence de la dernière est le double de celle de la première. Une gamme se caractérise par sa première note et la suite d'intervalles qui séparent les notes conjointes.

Pour simplifier, on va considérer la théorie de la musique occidentale basée sur l'accord tempéré (depuis le XVIII^e siècle). Dans ce cas, l'intervalle séparant la première et la dernière note d'une gamme est dite *octave*, une octave se divise en 12 écarts égaux appelés *demi-tons*. La dernière note porte le même nom de la première dans la gamme.



Figure 1: Les intervalles sur un piano

Nomenclature

Ils existent plusieurs systèmes de nomenclature de notes de musique, en France et dans beaucoup de pays on adopte le noms en termes de *Do-Re-Mi-Fa-Sol-La-Si*. Un système très répandu est celui basé sur l'alphabet latin : *C-D-E-F-G-A-B*.

Vu que les noms des notes se répètent au bout d'un octave, il faut distinguer une note *LA* de fréquence $440Hz$ d'une autre de fréquence $220Hz$ ou $880Hz$.

Le système de notation scientifique **Scientific Pitch Notation** identifie une note par sont nom alphabetique avec un nombre identifiant l'octave dans laquelle elle se situe, où l'octave commence par une note *C*. Par exemple la fréquence $440Hz$ représente A_4 sans ambiguïté, et les fréquences $220Hz$ et $880Hz$ représentent les notes A_3 , A_5 respectivement.

Dans le protocole **MIDI**, les notes sont représentées par un nombre entier, il permet de coder plus de 10 octave en partant de la note C_{-1} .

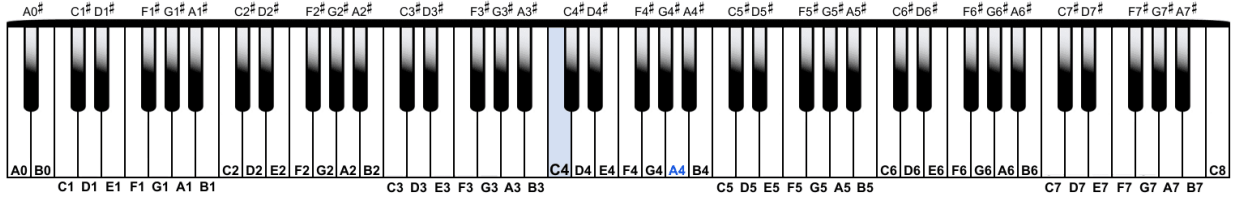


Figure 2: La notation scientifique des notes sur un piano

Partie mathématique

Un demi-ton est l'écart entre deux touches voisines sur un piano. On voudrais savoir le rapport r de fréquences associé à un demi-ton, sachant que l'octave double la fréquence on peut conclure facilement :

$$r^{12} = 2 \Rightarrow r = 2^{1/12}$$

On souhaite ramener l'espace de fréquences $F = (\mathbb{R}, \times)$ à l'espace $(\mathbb{N}, +)$ tel que $\boxed{\text{demi-ton} \equiv \pm 1}$. On définit donc une bijection

$$\forall f \in]0, \infty[, f \mapsto 12 \log_2 f$$

En arrondissant le résultat à la valeur entière la plus proche, on obtient un espace linéaire discret correspondant aux notes.

Il sera convenient d'obtenir les mêmes notes du protocole **MIDI** vu qu'il est très bien établi et très utilisé. Pour cela, on effectue une petite translation, en partant de la note de référence $A4 \equiv \text{MIDI}(69) \equiv 440\text{Hz}$.

$$\begin{cases} \varphi : f \mapsto 12 \log_2 f + C_{\text{ref}} \\ \varphi(440) = 69 \end{cases} \Rightarrow \varphi(f) - \varphi(440) = 12 \log_2 \left(\frac{f}{440} \right)$$

$$\varphi : f \mapsto 12 \log_2 f + (69 - 12 \log_2 440)$$

Références

- [1] Alain de Cheveigné et Hideki Kawahara.
YIN, a fundamental frequency estimator for speech and music, 2002.
- [2] Felix A. Gers, Nicol N. Schraudolph, et Jürgen Schmidhuber.
Learning Precise Timing with LSTM Recurrent Networks, 2002.