Identificação de covers a partir de grandes bases de dados de músicas

Martha Dais Ferreira

Identificação de covers a partir de grandes bases de dados de músicas

Martha Dais Ferreira

Orientador: **Prof. Dr. Luis Gustavo Nonato**

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

USP – São Carlos
Julho de 2014

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

Ferreira, Martha Dais
F383i Identificação de covers a partir de grandes bases de dados de músicas / Martha Dais Ferreira; orientador Luis Gustavo Nonato. -- São Carlos, 2014.
84 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2014.

1. Identificação de músicas cover. 2. Aprendizado de máquina não-supervisionado. I. Nonato, Luis Gustavo, orient. II. Título.

Agradecimentos

Primeiramente gostaria de agradecer aos meus pais, Antônio e Dais, pelo incentivo nos estudos e por me mostrarem que o conhecimento sempre pode nos levar além do que esperamos. Quero agradecer ao meu irmão Arthur e ao meu namorado Guilherme por acompanharem o desenvolvimento do projeto, pelo apoio que me deram durante esses anos e pelas conversas criativas.

Agradeço aos amigos da "jogatina"(Thays, Cata, Lai, Luiz, Raul, Carol, Willian), que mesmo distantes, continuaram torcendo por mim durante essa etapa, como também, aos amigos de laboratório que me auxiliaram no desenvolvimento do projeto com discussões e comentários. Além disso, agradeço especialmente a Débora, que trabalhou comigo durante as pesquisas e que se tornou uma grande amiga.

Gostaria de agradecer ao meu orientador Gustavo, que sempre muito sorridente, me ajudou a desenvolver o projeto e me ensinou muito durante esse período, assim como o professor Rodrigo, que também acompanhou de perto o meu trabalho. Ambos me proporcionaram grandes aprendizagens sobre pesquisa científica como também sobre a vida acadêmica.

Por fim, agradeço a CAPES por possibilitar o desenvolvimento deste projeto.

Resumo

A crescente capacidade de armazenamento introduziu novos desafios no contexto de exploração de grandes bases de dados de músicas. Esse trabalho consiste em investigar técnicas de comparação de músicas representadas por sinais polifônicos, com o objetivo de encontrar similaridades, permitindo a identificação de músicas *cover* em grandes bases de dados. Técnicas de extração de características a partir de sinais musicais foram estudas, como também métricas de comparação a partir das características obtidas. Os resultados mostraram que é possível encontrar um novo método de identificação de *covers* com um menor custo computacional do que os existentes, mantendo uma boa precisão.

Abstract

The growing capacity in storage and transmission of songs has introduced a new challenges in the context of large music data sets exploration. This work aims at investigating techniques for comparison of songs represented by polyphonic signals, towards identifying cover songs in large data sets. Techniques for music feature extraction were evaluated and compared. The results show that it is possible to develop new methods for cover identification with a lower computational cost when compared to existing solutions, while keeping the good precision.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Siglas

BOF	Bag of Features
BT	Beat-Tracking
CRP	Cross Recurrence Plots
DCT	Discrete Cossine Transform
DFT	Discrete Fourier Transform
DP	Dynamic Programming
DPLA	Dynamic Programming Local Alignment
DTW	Dynamic Time Warped
EM	Expectation-Maximization
EPCP	Enhanced Pitch Class Profile
FFT	Fast Fourier Tranform
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
HPCP	Harmonic Pitch Class Profile
IR	Information Retrieval
MAP	Mean of Average Precision
MFCC	Mel-Frequency Cepstrum Coefficients
MIR	Music Information Retrieval
MIREX	Music Information Retrieval Evaluation eXchange
MMR	Mean Reciprocal Rank
OTI	Optimal Transposition Index
PAA	Piecewise Aggregate Approximation
PCP	Pitch Class Profile
RP	Recurrence Plots
RQA	Recurrence Quantification Analysis
STFT	Short Time Fourier Transform
SVM	Support Vector Machine

Capítulo

1

Introdução

A crescente capacidade de armazenamento e transmissão de dados introduziu novos desafios para a exploração de grandes bases de dados musicais. Coleções de músicas encontram-se amplamente disponíveis na Internet e em dispositivos móveis, em parte devido ao surgimento, e evolução de técnicas de compactação de áudio (como formatos .mp3), possibilitado o armazenamento e transmissão de grandes conjuntos de músicas [44]. Este contexto intensificou a importância de organizar, indexar, e processar esses dados, a fim de atender as necessidades dos usuários.

Uma forma inerente de organizar coleções de música consiste em estabelecer grupos de músicas que compartilham características (padrões) em comum. Entretanto, tarefas como a classificação de gêneros e a busca por conteúdo de acordo com padrões de similaridades, são cada vez mais desafiadoras em grandes coleções de músicas.

Nesse contexto, a área de Recuperação de Informação Musical, do inglês *Music Information Retrieval* (MIR), tem como objetivo o desenvolvimento de técnicas e ferramentas para a análise e extração de informações relevantes em sinais de áudio, de forma a estabelecer critérios adequados para que a recuperação de informação relevante seja realizada de forma eficiente. MIR é uma área de pesquisa interdisciplinar, que envolve disciplinas como, por exemplo, a musicologia, para o estudo de aspectos históricos e culturais envolvidos na música; psicoacústica, que estuda a resposta cognitiva e os comportamentos humanos em relação à música; processamento de sinais, para desempenhar a extração de características significativas; e aprendizado de máquina, para o estudo de sistemas que aprendem a partir de dados.

Dessa forma, a similaridade entre músicas configura um aspecto importante para o desenvolvimento de sistemas de recuperação musical. Entre as aplicações envolvendo o estudo de padrões similares em música, a identificação de músicas *covers* tem sido bastante explorada por pesquisadores em MIR.

Além disso, a identificação de *covers* é importante para identificação de plágios, em que os artistas tem suas composições, ou partes de suas composições, copiadas por outros artistas afetando os direitos autorais do compositor original. A identificação manual tem se tornando cada vez mais difícil de ser efetuada, frente ao volume de gravações

disponibilizadas em meios digitais. Ademais, mesmo um ouvinte treinado pode não conseguir encontrar todas as composições e fragmentos que tenham sido copiados em uma grande base de dados.

Nas aplicações MIR, a informação musical pode estar representada através de formatos simbólicos, como o MIDI (*Musical Instrument Digital Interface*), ou através de sinais áudio. O formato MIDI tipicamente possui qualidade sonora inferior devido à limitações nos sintetizadores e no controle de parâmetros [40].

Por outro lado, sinais de áudio são de maior interesse para aplicações MIR, pois refletem a característica da maioria das bases de músicas disponíveis. Sinais de áudio são geralmente polifônicos, visto que toda informação musical é gravada em alguns poucos canais. Em outras palavras, sinais polifônicos contem informação de vários instrumentos que executam melodias e ritmos simultaneamente. Assim, técnicas de comparação de músicas representadas por sinais polifônicos tem sido propostas na área de MIR para tarefas como reconhecimento de acordes, classificação de gêneros, identificação de artista, reconhecimento de emoções, identificação de *covers*, entre outras.

Na literatura encontram-se diversas abordagens visando a identificação de *covers*. Estas abordagens geralmente fazem uso de técnicas de processamento de sinais, aprendizado de máquina e estatística. Porém, grande parte dos métodos propostos não apresentam uma boa acurácia. Entre os métodos que manifestam bom desempenho [20, 51], o alto custo computacional e, consequentemente, o elevado tempo de processamento, os tornam inviáveis em aplicações reais que contemplam grandes conjuntos de dados.

Dessa forma, a proposta deste projeto de mestrado tem como intuito principal o estudo de novas abordagens para a identificação de músicas *covers* que possibilitem tempos de processamento inferiores quando comparados às abordagens encontradas na literatura.

O modelo de representação de dados empregado é o *Bag-of-Features*. Bastante adotado em tarefas de classificação de imagens, este modelo busca representar os dados de forma a criar um dicionário utilizando técnicas de agrupamento [33, 34]. Para a avaliação da abordagem proposta, duas métricas de precisão foram utilizadas: média das precisões médias (do inglês, *Mean of Average Precision*), bastante utilizada na literatura e em competições do MIREX *(Music Information Retrieval Evaluation eXchange)*; e a métrica Separabilidade, que avalia a precisão de acordo com a ordem de retorno dos músicas.

Os resultados obtidos pela abordagem desenvolvida neste projeto foram comparados com resultados de dois trabalhos na literatura que apresentaram as melhores precisões na competição MIREX. São eles: o trabalho de Ellis [16], e o trabalho de Serrà et al. [51]. Ambos são detalhadamente descritos no Capítulo 5.

1.1 Objetivos

Esse trabalho de mestrado visa estudar os métodos desenvolvidos para a identificação de músicas *covers* em conjunto de dados. Para isso, primeiramente é realizado um levantamento bibliográfico das técnicas de extração de características de sinais de áudio existentes na literatura, como também das técnicas de comparação dessas características. Em seguida, técnicas de aprendizado de máquina são estudadas dentro da perspectiva da aplicação proposta. Por fim, uma nova metodologia que utiliza técnicas de aprendizado de máquina é proposta e os resultados são comparados com os trabalhos anteriores.

Portanto, os principais objetivos podem ser citados da seguinte forma:

- Levantamento bibliográfico de técnicas de extração de informações de sinais musicais e de métodos de comparação de músicas polifônicas;
- Estudo e comparação dos trabalhos que alcançaram as melhores precisões na competição MIREX;
- Proposta de um novo método de identificação de *covers* utilizando técnicas de aprendizado de máquina;
- Validação do método desenvolvido através da comparação dos resultados obtidos com os resultados dos trabalhos estudados.

1.2 Organização da Monografia

O restante desta dissertação está estruturado da seguinte maneira:

- No Capítulo 2 é apresentado uma breve noção musical para melhor compreensão das técnicas de extração de características;
- No Capítulo 3 é realizada uma descrição conceitual das técnicas de extração e comparação de características de sinais brutos, assim como dos métodos de agrupamento de dados, e das métricas de avaliação.
- No Capítulo 4 é apresentado uma descrição das pesquisas realizadas nos últimos anos com a utilização de sinais musicais e suas várias áreas de atuação;
- No Capítulo 5 são descritos os métodos existentes na literatura que obtiveram bons resultados;
- No Capítulo 6 é apresentado a metodologia utilizada neste trabalho para o desenvolvimento do nova proposta de identificação de músicas *covers*,
- No Capítulo 7 são exibidos os resultados obtidos e a comparação do método proposto com os métodos existentes;
- Por fim, no Capítulo 8 são apresentadas as conclusões sobre os resultados obtidos e as perspectivas de trabalhos futuros.

Capítulo 2

Noções musicais

Música é considerada como a arte do som, sendo geralmente constituída de melodia, que se refere à execução de sons sucessivos; ritmo, que corresponde à intensidade e duração dos sons e das pausas; e harmonia, a qual é a combinação de sons executados simultaneamente. Instrumentos musicais são fontes sonoras utilizados para produzir sons, sendo classificado em 3 tipos: cordas esticadas (violão, piano, violino), colunas de ar (flauta, trompete, gaita) e membranas (tamborim, cuíca, bongô) [10, 5].

Sons são vibrações percebidas pela audição humana, sendo esses musicais e não musicais. As vibrações são medidas em Hertz (Hz), que representam uma unidade de frequência expressa em termos de ciclos por segundo, utilizada para descrever ondas senoidais. O ouvido humano é capaz de reconhecer sons entre 20 e 18 mil Hz aproximadamente. Os sons fundamentais se localizam na faixa aproximada de 32 a 4 mil Hz, nesse limite encontram-se os harmônicos que caracterizam o timbre dos instrumentos musicais [10, 5].

As propriedades do som são altura, intensidade, duração e timbre. Altura, também denominada frequência, representa o número de vibrações por segundo, caracterizando o som em grave, médio ou agudo. A diferença de frequência, entre dois sons é definido como intervalo, sendo que o semitom é o menor intervalo usado normalmente, e tom é o intervalo formado por dois semitons. Um intervalo pode ser melódico, que são sons ouvidos sucessivamente, ou harmônico, que são sons ouvidos simultaneamente [10, 5].

A intensidade é o volume do som, ou a amplitude da vibração, deixando o som forte ou fraco. A duração é o tempo em que se prolonga o som ou o intervalo de silêncio (pausa) entre dois sons. O timbre está relacionado com a série harmônica produzida pelo som emitido, o que caracteriza a origem do som, essa propriedade permite com que instrumentos possam ser identificados [10, 5].

A série harmônica é uma série de subvibrações geradas pela vibração do som principal tal como definido na Equação 2.1. Assim, o corpo vibra primeiramente em toda a sua extensão (primeiro termo da Equação 2.1), emitindo uma frequência denominada frequência fundamental, que caracteriza a percepção de altura de uma nota, também conhecida como primeiro componente harmônico, em seguida esse mesmo corpo vibra em duas metades

(segundo termo da Equação 2.1), um terço, um quarto de sua extensão e assim por diante [10].

Matematicamente, cada termo da Equação 2.1 representa o número de divisões do corpo submetido à vibração, neste caso, as divisões da equação representam o comprimento do corpo, onde 1 é a vibração do comprimento total do corpo. Conforme aumenta o número de divisões, i.e., termos seguintes na série harmônica, maior a frequência produzida pela vibração do corpo.

$$\sum_{n=1}^{\infty} \frac{1}{n} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \cdots \tag{2.1}$$

Outro ponto importante na teoria musical é a notação musical, onde figuras representam o valor da duração das notas e das pausas de acordo com a fórmula de compasso. O compasso é formado por um conjunto de figuras musicais que representam um valor determinado de duração do som ou do silêncio. A acentuação métrica dos compassos permite identificar as várias formas de compassos existentes, pois as batidas obedecem essa acentuação, sendo umas fortes e outras fracas, seguindo a divisão do compasso definindo. Com isso, é definido o contra-tempo que são as notas executadas no tempo fraco, de modo que o tempo forte seja preenchido com pausa [5]. Um tempo forte também é conhecido como tempo de ataque.

O andamento de uma música é o movimento lento, moderado e rápido dos sons de acordo com a proporção do tempo nos compassos, definindo as batidas por minuto (BPM) mencionadas na literatura, essas vão de 40 a 208 BPM. Séries de notas ouvidas sucessivamente, de modo a terminar na nota inicial uma oitava acima ou abaixo são denominadas escalas, podendo ser, ascendentes (som seguinte é mais agudo) ou descendentes (som seguinte é mais grave) [10, 5].

A escala mais utilizada no contexto de extração de atributos para identificação e classificação de músicas é a escala cromática, a qual considera a sequência de todos os 12 semitons, que correspondem às notas Dó, Dó#, Ré, Ré#, Mi, Fá, Fá#, Sol, Sol#, Lá, Lá#, Si, sendo que o sustenido (#) um sinal de alteração que eleva o som em um semitom e o bemol(b) diminui o som em um semitom. As notas podem ser representadas por cifras: C (Dó), D (Ré), E (Mi), F (Fá), G (Sol), A (Lá), B(Si) [10]. Em músicas orientais uma discretização mais fina das frequência é utilizada, denominada escala de quarto de tom, a qual adiciona notas entre as notas usuais da escala cromática. Notas com quarto de tom são notas com metade do intervalo do semitom, assim, dois quartos tons formam um semitom.

Notas quando tocadas em conjunto produzem acordes. Os acordes são tipicamente montados em cima de tríades, i.e., um conjunto de três notas simultaneamente tocadas, as quais produzem uma frequência fundamental. Acordes também adicionam outras notas às tríades com o intuito de produzir sensações de tristeza, tensão, alegria, etc. As transições entre acordes mais agradáveis ao ouvido humano são denominadas campos harmônicos. Um campo harmônico é tipicamente definido com base na escala adotada por uma música. Após selecionar a escala desejada, monta-se as tríades dos acordes e formula-se suas transições ao longo do tempo, da maneira mais adequada a atingir os objetivos da música [10].

O conjunto de sons de uma escala em relação a sua tônica (nota inicial) é conhecido como tonalidade. A transposição é a leitura ou escrita de uma música em uma tonalidade

diferente da composição original. O conceito de transposição é muito utilizado na identificação de músicas covers, porque algumas versões da mesma música podem ser tocadas em diferentes tonalidades.

Capítulo 3

Conceitos

Quando se trata de identificação de *covers* em sinais polifônicos, uma importante etapa é a representação do sinal através de um vetor ou vetores de características [58]. Outra etapa importante é a comparação dessas características extraídas, o que possibilita quantificar a relação de similaridade entre dois sinais. Esse capítulo tem o intuito de apresentar algumas formas de extração e comparação dessas característica, assim como técnicas de agrupamento que são utilizadas na metodologia deste trabalho de mestrado.

Este capítulo foi dividido em três partes. Inicialmente, detalha-se a descrição dos extratores de características de sinais musicais encontrados na literatura. A segunda parte é dedicada a descrição de alguns métodos de comparação utilizados. E por fim, está a descrição de técnicas de agrupamento que foram utilizadas neste trabalho.

3.1 Extração de Características de Sinais de Áudio

A análise de Fourier consiste na aproximação de uma função pela soma de termos envolvendo senos e cossenos resultando no que é conhecido como série de Fourier. No caso de uma aproximação utilizando um número finitos de termos, a série de Fourier é dada na Equação 3.1, onde, $t = 1, 2, ..., N$, N é o número de observações da série x e $p = 1, ..., (N/2) - 1$. [9].

$$x_t = a_0 + \sum_{p=1}^{(N/2-1)} [a_p \cos(2\pi pt/N) + b_p \sin(2\pi pt/N)] + a_{N/2} \cos \pi t \tag{3.1}$$

$$\begin{aligned} a_0 &= \bar{x} \\ a_{N/2} &= \sum(-1)^t x_t/N \\ a_p &= 2[\sum x_t \cos(2\pi pt/N)]/N \\ b_p &= 2[\sum x_t \sin(2\pi pt/N)]/N \end{aligned} \tag{3.2}$$

Os coeficientes da série de Fourier particionam a variabilidade dos dados em componentes de frequência $2\pi/N, 4\pi/N, ..., \pi$. A componente de frequência $\omega = 2\pi p/N$ é conhecida como a p-ésima harmônica, representada na Equação 3.3, para $p \neq N/2$ [9].

$$a_p \cos \omega_p t + b_p \sin \omega_p t = R_p \cos(\omega_p t + \phi_p) \tag{3.3}$$

onde, $R_p = \sqrt{a_p^2 + b_p^2}$ é a amplitude da p-ésima harmônica e $\phi_p = \tan^{-1}(-b_p/a_p)$ é a fase da p-ésima harmônica.

A partir da análise de Fourier surge a Transformada de Fourier, que é utilizada para obter as frequências existentes nos sinais de áudio, gerando uma representação do sinal no domínio das frequências.

Nesse contexto, o método utilizado como etapa inicial pelos extratores é a Transformada Discreta de Fourier, do inglês *Discrete Fourier Transform* (DFT), derivada da Transformada de Fourier. O objetivo da DFT é transformar o sinal, que está no domínio do tempo, para o domínio das frequências, possibilitando construir histogramas de frequência e histogramas de fase. Para a extração de características utiliza-se o histograma de frequência, pois esses mostram as frequências mais significativas que existem no sinal explorado [9]. A otimização da DFT é conhecida como Transformada Rápida de Fourier (FFT) [1].

A DFT pode ser representada, de maneira simplificada, pela Equação 3.4, onde N representa a quantidade de amostra, $x[n]$ o sinal no domínio do tempo e $X[k]$ o sinal no domínio das frequências. O processo inverso da DFT, conhecido como *Inverse Discrete Fourier Transform* (iDFT), transforma o sinal que está em domínio de frequência de volta ao domínio do tempo com a Equação 3.5 [41].

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{i2\pi}{N} kn}, k = 0, 1, ..., N-1 \tag{3.4}$$

$$x[n] = \frac{1}{N} \sum_{n=0}^{N-1} X[k] e^{\frac{i2\pi}{N} kn}, n = 0, 1, ..., N-1 \tag{3.5}$$

3.1.1 Transformada de Fourier de Curto Tempo

Na Transformada de Fourier de Curto Tempo, do inglês *Short-Time Fourier Transform* STFT, aplica-se a DFT em pequenas partes do sinal, chamados *frames* (fragmentos), que são multiplicados por funções janelas (*window*). Porém o janelamento introduz problemas na resolução das frequências, deixando-as menos precisas, pois quanto menor o números de amostras para a aplicação da DFT, maior a gama de frequências de sua saída [41, 11].

A sobreposição de janelas, pode amenizar o problema de precisão, mantendo a sincronia temporal [41]. Este método captura a transição do sinal e gera um espectrograma do mesmo. A STFT é dada pela Equação 3.6 [11]:

$$X(k, l) = \sum_{k=0}^{n-1} w(l-m) x(m) e^{-j\omega_k m}, l = 0, 1, 2, ... \tag{3.6}$$

[1] Maiores informações sobre a Transformada Rápida de Fourier no livro *The Analysis of Time Series: An Introduction* [9]

Onde, $X(k,l)$ é a matriz resultante da STFT, w é a função *window*, $x(m)$ é o sinal dado como entrada e $\omega_k = 2\pi k f_s/N$, sendo $k = 0, 1, ..., N-1$, f_s é a frequência de amostragem e N é o tamanho do fragmento do sinal [11]. Para gerar o espectrograma a partir da matriz resultante, utiliza-se a Equação 3.7, onde k e l representam os índices de frequência e tempo respectivamente:

$$S(k,l) = |X(k,l)|^2 \tag{3.7}$$

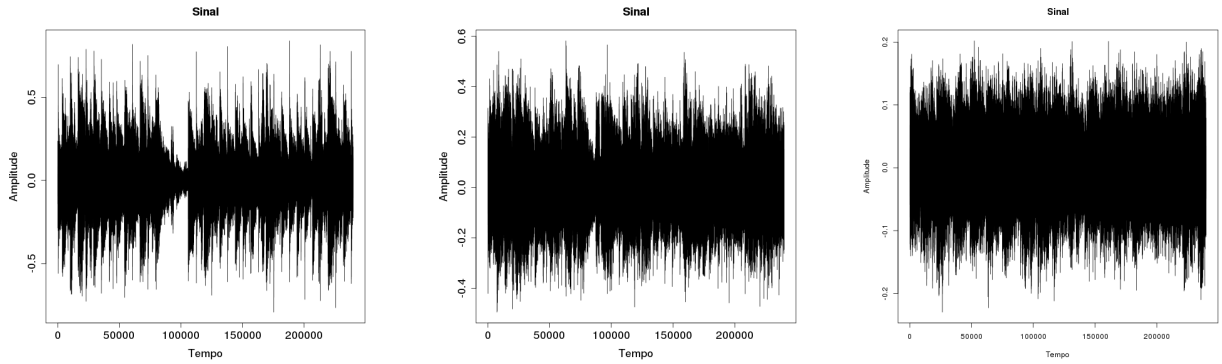
Neste projeto de mestrado adotou-se a função *Blackman Harris window* com 62 dB, a Equação 3.8 define o *L-term Blackman Harris*, onde $\alpha_0 = 0,44859$, $\alpha_1 = 0,49364$, $\alpha_2 = 0,05677$, tendo o nível igual a -62 dB, $N_{frame} = 4096$ amostras que equivale a 93 ms com a taxa amostral de 16 KHz [27].

$$w(n) = \frac{1}{N_{frame}} \sum_{l=0}^{L-1} \alpha_l \cdot \cos(\frac{2nl\pi}{N_{frame}}),\ n = 0, 1, ..., N_{frame} - 1 \tag{3.8}$$

A Figura 3.2 mostra um exemplo do espectrograma gerado para 3 músicas distintas, sendo duas *cover* e uma não-*cover*, todas com duração de 10 segundos, onde cada coluna do espectrograma representa a magnitude gerada para cada fragmento de tempo. Os sinais polifônico bruto que geraram as figuras podem ser visto na Figura 3.1, esses serão usados nos próximos exemplos.

Os sinais das imagens foram divididos em janelas de 16 ms, com uma sobreposição de 0,75%, e a quantidade de amostras por segundo (*sample rate*) foi de 16 KHz. Nos espectrogramas da Figura 3.2, o eixo x é o eixo do tempo de duração do sinal e no eixo y as frequências. Quanto mais vermelho, maior é a intensidade daquela frequência no tempo.

Nota-se que nas imagens as músicas *covers* apresentam similaridades quando comparadas com uma não-*cover* através de padrões em suas estruturas que podem ser visualizadas.

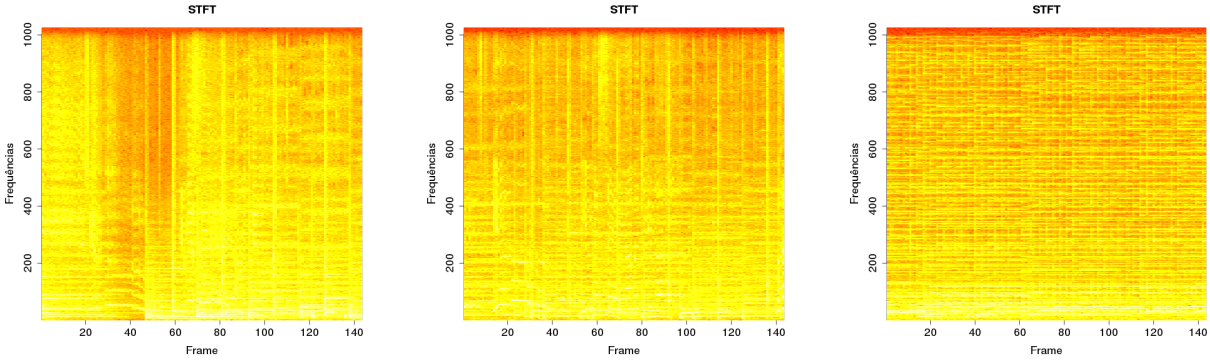


(a) Sinal polifônico bruto da música *Imagine* na versão por *John Lennon*.

(b) Sinal polifônico bruto da música *Imagine* na versão por *Diana Ross*.

(c) Sinal polifônico bruto da música *Book I Prelude* na versão por *Christiane Jaccotte*.

Figura 3.1: Sinais polifônicos brutos utilizados para gerar os atributos descritos neste capítulo.



(a) STFT aplicado na música *Imagine* na versão por *John Lennon*.

(b) STFT aplicado na música *Imagine* na versão por *Diana Ross*.

(c) STFT aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*.

Figura 3.2: Espectrogramas resultantes do método STFT aplicada em 10 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz.

3.1.2 Coeficientes Cepstrais da Frequência Mel

Um dos métodos mais utilizados para extração de características de sinais sonoros é o Coeficientes Cepstrais da Frequência Mel, do inglês *Mel-Frequency Cepstrum Coefficients* (MFCC) [57, 17, 45, 46, 50, 44]. Desenvolvido para reconhecimento de voz, o MFCC é bastante eficaz para representar o timbre [39] gerando um *chromagrama*, ou uma matriz de *chroma*[2]. Esta matriz é mais compacta quando comparada ao espectrograma resultante da STFT [44].

Um problema com o MFCC é a variedade de parâmetros envolvidos, sendo que cada implementação descrita na literatura utiliza configurações diferentes adaptadas a cada aplicação. Parâmetros como a configuração dos filtros passa-banda, e a quantidade de frequências utilizadas na escala mel, são largamente explorados [44]. Variações do MFCC como o *Human Factor Cepstral Coefficients* (HFCC) propõe configurações alternativas para o banco de filtros de modo a reduzir a banda de filtragem [25].

A Figura 3.3 apresenta a sequência de passos do MFCC [39], lembrando que esse processo pode ser modificado e existem várias formas de implementação.

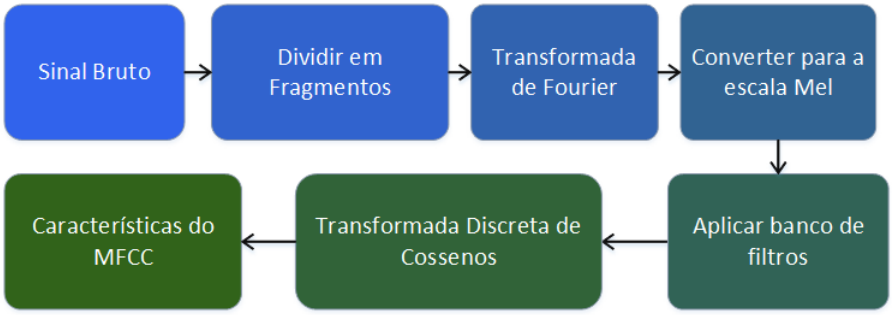


Figura 3.3: Diagrama do Processo de extração de Características do MFCC

[2]O chroma é baseado em escalas cromáticas da música ocidental, definido no Capítulo 2.

Na aplicação do STFT, o sinal é dividido em quadros, ou seja, o sinal é repartido em blocos de tempo, determinado em segundos ou milissegundos. Para isso, Aucouturier e Pachet [1] utiliza 50 milissegundos, enquanto que Ellis et al. [21] utiliza 16 milissegundos, além de utilizar potência de dois em seus parâmetros. Entretanto, Logan [39] propõe utilizar 20 milissegundos, para obter um balanço de tempo de processamento e qualidade de características.

Com espectrograma adquirido a partir do STFT, aplica-se uma escala logarítmica para adquirir a informação de sonoridade [39]. A escala mel definida pela Equação 3.9 [60] é utilizada para converter o espectrograma em uma escala logarítmica. Ela expressa a percepção humana das frequências relacionadas a voz. O nome Coeficientes Cepstrais da Frequência Mel, ou *Mel-Frequency Cepstrum Coefficients*, foi escolhido devido ao uso dessa escala no método.

$$Mel(f) = 2595 \log_{10}(1 + \frac{f}{700}) \tag{3.9}$$

Após a conversão do espectrograma para a escala mel, em que f é a escala de frequências, aplica-se um filtro passa-banda que permite destacar as frequências altas e baixas. Antes, devem ser decididos quais valores serão utilizados para a aplicação do filtro, ou banco de filtros, isto é, qual a variação de frequências que deverão passar, e também a quantidade de filtros que serão utilizados. As frequências são normalmente trabalhadas em Hz. Quanto maior o número de filtros, menor serão as taxas de erros [25].

Depois da aplicação dos filtros, é calculado a energia definida na Equação 3.10 [32], em que, $|y(k)|$ é o resultado da STFT, $\phi_{(i)}(k)$ é o resultado dos filtros utilizados, e $i = 1, 2, ..., Q$, sendo Q o número de filtros e N o número total de amostra em cada quadro (20 ms) do áudio.

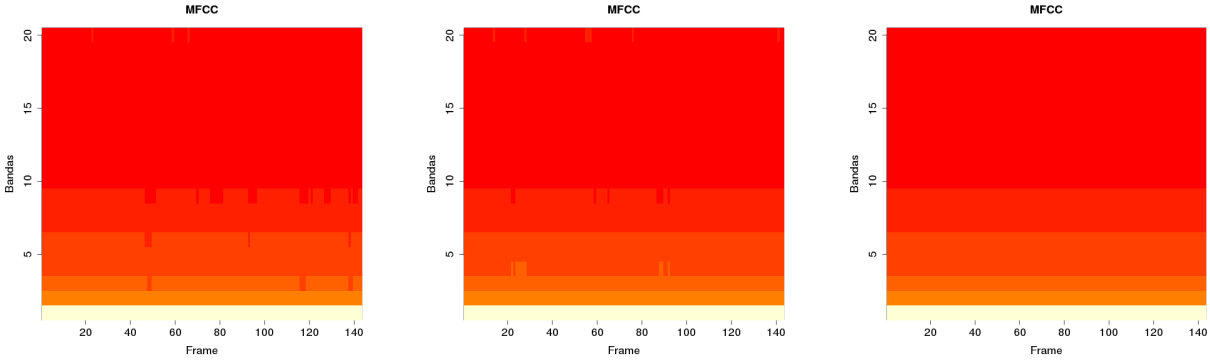
$$e(i) = \sum_{k}^{N} |y(k)|^2 \phi_{(i)}(k) \tag{3.10}$$

Em seguida, a energia resultante é utilizada no cálculo da Transformada Discreta de Cossenos, ou *Discrete Cosine Transform* (DCT), que retornará os coeficientes representantes das características do sinal como mostra a Equação 3.11 [32], onde $m = 1, 2, ..., R$, sendo R o número total de coeficientes.

$$C_m = \sqrt{\frac{2}{N}} \sum_{i=1}^{Q} (\log[e(i+1)]) \cdot \cos\left[m \cdot \left(\frac{2i+1}{2}\right) \cdot \frac{\pi}{Q}\right] \tag{3.11}$$

A Figura 3.4 apresenta exemplos de *chromagramas* gerado pelo MFCC utilizando 22 filtros e 20 coeficientes, 16 ms como passo na divisão dos fragmentos do sinal com uma sobreposição de 0,75%. Com 10 segundos centrais do sinal amostrados a 16 KHz de 3 músicas distintas, duas *covers* e uma não-*cover*. Nos *chromagramas* da Figura 3.4, o eixo x são os fragmentos do sinal e o eixo y são os coeficientes, e a cor representa a intensidade do coeficiente naquele fragmento.

Muelder et al. [44] utiliza o MFCC para gerar matrizes de similaridade entre os resultados de um conjunto de dados musicais. Na etapa final aplica-se uma visualização utilizando quatro métodos diferentes e faz uma análise dos resultados. Porém, outros



(a) MFCC aplicado na música *Imagine* na versão por *John Lennon*.

(b) MFCC aplicado na música *Imagine* na versão por *Diana Ross*.

(c) MFCC aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*.

Figura 3.4: Atributos resultantes do método *Mel-Frequency Cepstrum Coefficients* (MFCC) aplicado em 10 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz.

autores, como Lu e Zhang [40], dizem que o MFCC perde muita informação, então complementam essa característica com um espectro baseado em oitavas, como por exemplo, o método *Pitch Class Profiles* descrito a seguir.

3.1.3 Método de Perfil de Classes de Altura

O Método de Perfil de Classes de Altura, do inglês, *Pitch Class Profile* (PCP), é utilizado no reconhecimento de acordes musicais, sendo uma extensão do método *Simple Auditory Model* que utiliza 12 semitons para identificar as notas tocadas no áudio [23]. Similar ao método MFCC, ele recebe como entrada o sinal bruto da música, e tem como etapa inicial do processo a aplicação do STFT (subseção 3.1.1). Um diagrama do processo pode ser visto na Figura 3.5

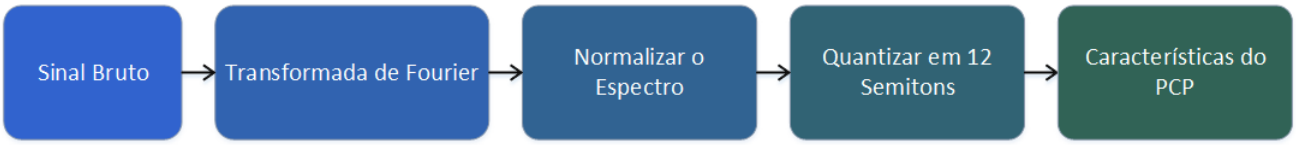


Figura 3.5: Diagrama do Processo de extração de Características do PCP

Dado um sinal bruto, aplica-se o STFT e faz a quantização nos 12 semitons. Essa quantização do espectrograma em 12 dimensões é feita de acordo com a Equação 3.12, onde $p = 1, 2, ..., 12$ representa os índices dos semitons obtidos e $X(l)$ é o espectrograma obtido com o método STFT [23].

$$PCP(p) = \sum_{l \text{ s.t. } M(l)=p} \| X(l) \|^2 \tag{3.12}$$

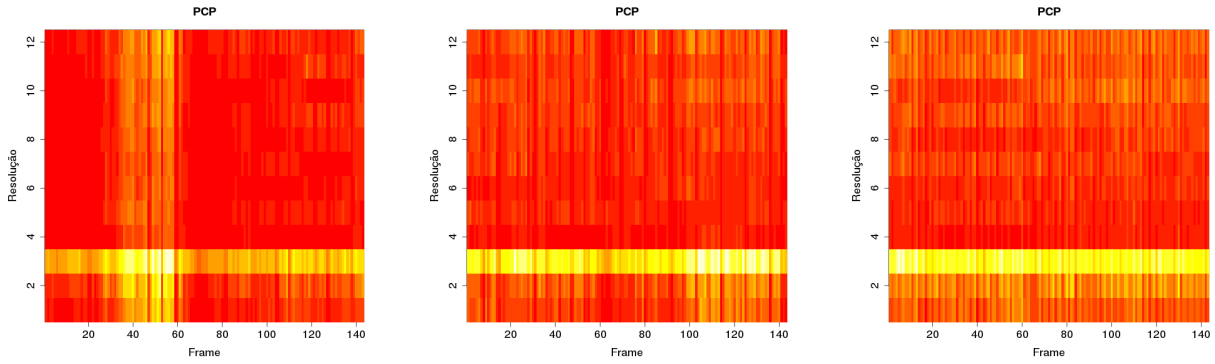
O $M(l)$ é um mapeamento das frequências do espectrograma resultante do STFT para a quantização nos doze vetores representantes dos semitons. Fujishima [23] define $M(l)$

como na Equação 3.13, em que f_{ref} representa a frequência de referência (normalmente adota-se o valor 440) e $f_s \cdot (\frac{1}{N})$ representa as frequências do espectrograma.

$$M(l) = \begin{cases} -1 & \text{para } l = 0 \\ round(12log_2((f_s \cdot (\frac{l}{N}))/f_{ref}))mod12 & \text{para } l = 1, 2, ..., N/2 - 1 \end{cases} \quad (3.13)$$

A Figura 3.6 mostra um exemplo do PCP gerado para 3 músicas distintas, sendo duas *cover* e uma não-*cover*, todas com duração de 10 segundos. Os sinais foram divididos em janelas de 16 ms, com uma sobreposição de 0,75%, e a quantidade de amostras por segundo (*sample rate*) foi de 16 KHz. O eixo x é o eixo do tempo de duração do sinal em fragmentos e no eixo y as frequências de acordo com os 12 semitons. Da mesma forma como nas figuras anteriores, quanto mais vermelho, maior é a intensidade daquela frequência no tempo.

Nota-se nas imagens que as frequências estão quantizadas em 12 semitons, em que cada um representa uma nota na escala cromática. Este tipo de representação auxilia em tarefas como reconhecimento de acordes e identificação de músicas *covers*, em que o MFCC e o STFT não foram muito eficientes.



(a) PCP aplicado na música *Imagine* na versão por *John Lennon*.

(b) PCP aplicado na música *Imagine* na versão por *Diana Ross*.

(c) PCP aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*.

Figura 3.6: Atributos resultantes do método *Pitch Class Profile* (PCP) aplicada em 10 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz.

Na literatura atual já existem derivações do método PCP, como o método *Harmonic Product Spectrum* (HPS), o *Enhanced Pitch Class Profile* (EPCP) e o *Harmonic Pitch Class Profile* (HPCP). Como o extrator HPCP é o estado da arte para identificação de *cover*, a próxima seção descreve o seu processo.

3.1.4 Método de Perfil de Classes de Altura Harmônica

O Método de Perfil de Classes de Altura Harmônica, do inglês, *Harmonic Pitch Class Profile* (HPCP), é uma derivação do PCP [50], que também tem como etapa inicial de seu processo a aplicação do STFT. Um diagrama do processo pode ser visto na Figura 3.7

De acordo com Gómez [27] as maiores modificações em relação ao PCP foram a introdução de pesos, a consideração de harmônicos e alteração do nível de quantização.

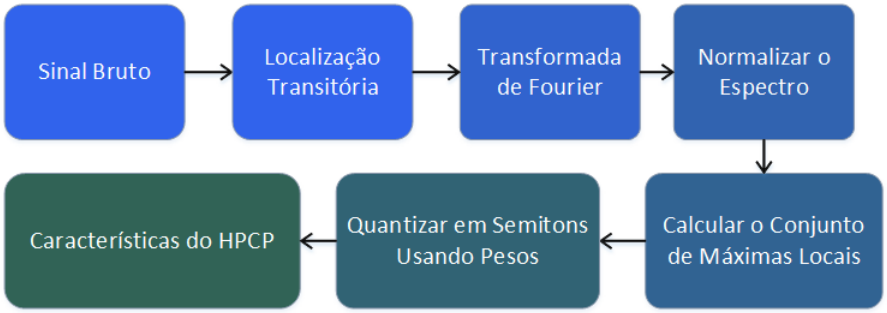


Figura 3.7: Diagrama do Processo de extração de Características do HPCP

Antes da aplicação do HPCP, é feito um pré-processamento denominado localização transitória (*transient location*) proposta por Bonada [6]. Este pré-processamento altera o tamanho do sinal, eliminando o ruído de regiões das estruturas harmônicas sem modificar características como tom e timbre, e, portanto, diminui o custo computacional do HPCP.

A localização transitória se inicia com a aplicação da Transformada de Fourier no sinal, depois analisa-se a energia de um banco de filtros, que contêm 42 bandas com frequências entre 40 Hz e 20 Hz, seguido da aplicação do MFCC, descrita na subseção 3.1.2. Então, obtêm-se a localização das mudanças rápidas que representam os ataques da música e, por fim, aplica-se a inversa da transformada de Fourier para reconstruir o sinal [3]. A Figura 3.8 mostra a sequência do processo, em que a FFT é aplicada em um sinal janelado, depois aplica-se um banco de filtros combinado ao extrator MFCC, e por fim o sinal é reconstruído com a inversa do FFT.

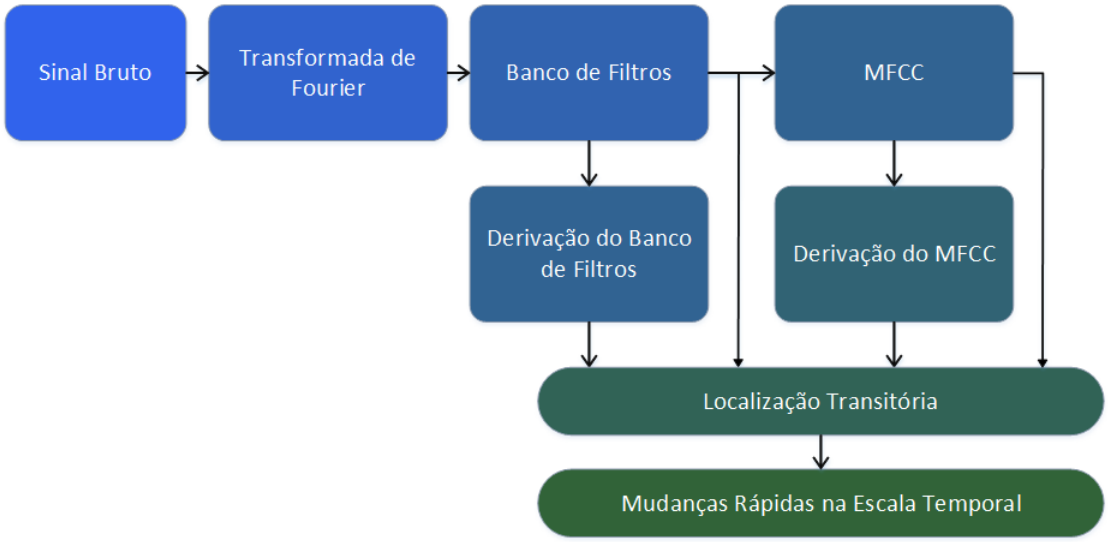


Figura 3.8: Diagrama do Processo de localização transitória

Após o pré-processamento com a localização transitória, aplica-se o método STFT, descrita na subseção 3.1.1, e com a aplicação de *zero-padding*.

Na próxima etapa é feito a localização dos picos sobre os fragmentos separados com o STFT. De acordo com Gómez [27], para extrair os picos da função resultante, deve-se saber que, perto desses picos, é formado uma parábola, dado pela Equação 3.14, em que p é o centro da parábola, a a medida da concavidade e b o deslocamento.

[3] Maiores detalhes sobre a localização transitória são encontrados em [6] e [27].

$$y(x) = a(x-p)^2 + b \tag{3.14}$$

Considerando K_β o máximo local, tem-se:

$$\begin{aligned} &y(-1) = \alpha = 20log_{10}|X(K_\beta - 1)| \\ &y(0) = \beta = 20log_{10}|X(K_\beta)| \\ &y(1) = \gamma = 20log_{10}|X(K_\beta + 1)| \\ &\alpha \leq \beta \geq \gamma \end{aligned} \tag{3.15}$$

Considerando que o centro da parábola p pode ser obtido interpolando o local do pico, como mostrado na Equação 3.16 e na Equação 3.17, obtêm-se a magnitude do pico, que é dado pela Equação 3.18 [27].

$$p = \frac{1}{2} \cdot \frac{\alpha - \gamma}{\alpha - 2\beta + \gamma} \tag{3.16}$$

$$k^* = k_{beta} + p \tag{3.17}$$

$$y(p) = 20 \cdot log_{10}|X(k^*)| = \beta - \frac{1}{4}(\alpha - \gamma)p \tag{3.18}$$

Após a detecção dos picos, as frequências computadas são limitadas no intervalo entre 100 a 5000 Hz e, então, aplica-se a quantização definida pela Equação 3.19.

$$HPCP(n) = \sum_{i=1}^{nPeaks} \omega(n, f_i) \cdot a_i^2, n = 1...size \tag{3.19}$$

Onde a_i é a magnitude linear do sinal (resultado da STFT), f_i é o valor do pico da frequência de número i, $nPeaks$ é a quantidade de picos que estão sendo considerados, n é o semitom, $size$ é a quantidade semitons (12, 24, 36,...) e $\omega(n, f_i)$ é o peso da frequência f_i no semitom n [27].

Para calcular o peso das frequência em cada semitom utiliza-se a Equação 3.20, onde l é o tamanho da janela definida de modo empírico, e d é a distância de semitons entre a frequência do pico f_i e a frequência do semitom central f_n. A variável d é dada pela Equação 3.21, onde m é o inteiro que minimiza $|d|$, e f_n é dado pela Equação 3.22. A variável l é 4/3 do $semitom = 12 \times log_2(h+1)$, em que $h = 1, 2, ..., nHarmonicos$ e $nHarmonicos = 8$ [27].

$$\omega(n, f_i) = \begin{cases} cos^2\left(\frac{\pi}{2} \cdot \frac{d}{0.5 \cdot l}\right) & \text{, se } |d| \leq 0.5 \cdot l \\ 0 & \text{, se } |d| > 0.5 \cdot l \end{cases} \tag{3.20}$$

$$d = 12 \cdot log_2 \frac{f_i}{f_n} + 12 \cdot m \tag{3.21}$$

$$f_n = f_{ref} \cdot 2^{\frac{n}{size}}, n = 1...size \tag{3.22}$$

O espectro é composto de vários harmônicos, onde as frequências são múltiplos da frequência fundamental, e cada nota aparece em frequências de harmônicos diferentes,

afetando os valores do HPCP. Para isso, a Equação 3.23 associa i_n com a n-ésima harmônica de cada nota, onde i_1 é a frequência fundamental. Por fim, uma normalização do HPCP é feita no pós-processamento, dado pela Equação 3.24 [27].

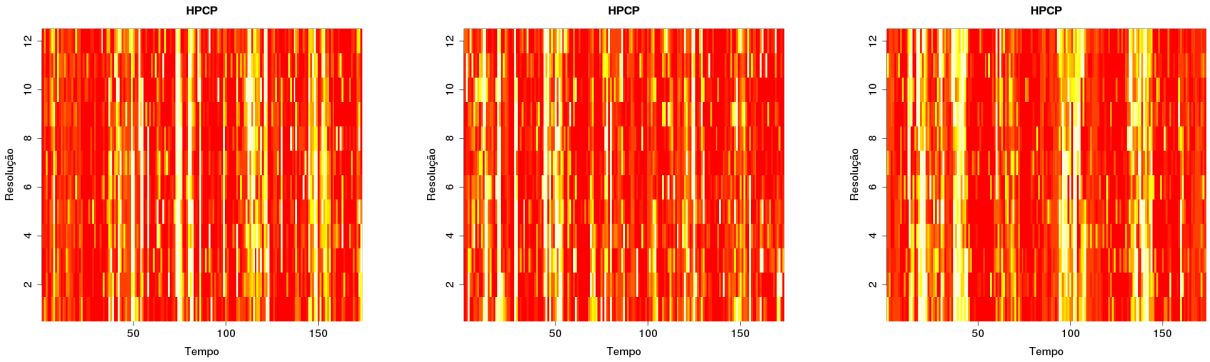
$$i_n = mod[(i_1 + 12 \cdot log_2(n)), 12] \tag{3.23}$$

$$HPCP_{norm}(n) = \frac{HPCP(n)}{Max_n(HPCP(n))}, n = 1...size \tag{3.24}$$

Nas Figuras 3.9 e 3.10 é possível ver o *chromagrama* gerado a partir de 3 músicas distintas, sendo duas *covers* e uma não-*cover*, todas com duração de 10 segundos, onde o eixo y representa os semitons e o eixo x o tempo de duração do sinal. A cor vermelha mostra as frequências de maior intensidade.

Os sinais foram divididos em janelas de 16 ms, com uma sobreposição de 0,75%, e a quantidade de amostras por segundo (*sample rate*) foi de 16 KHz. O eixo x representa o tempo de duração do sinal em fragmentos, e no eixo y, estão as frequências de acordo com a resolução, 12 para a Figura 3.9 e 36 para a Figura 3.10.

Nota-se que o HPCP apresentou uma melhor nitidez quando comparado com o PCP, mesmo quando executado com resolução 12. O HPCP permite visualizar padrões nas estruturas de modo que seja possível identificar músicas *cover* das não-*cover*.



(a) HPCP aplicado na música *Imagine* na versão por *John Lennon*.

(b) HPCP aplicado na música *Imagine* na versão por *Diana Ross*.

(c) HPCP aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*.

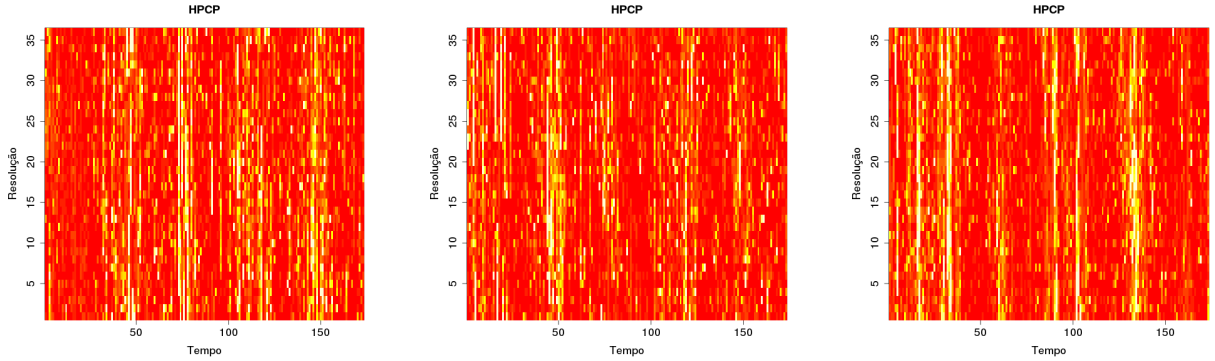
Figura 3.9: Atributos resultantes do método *Harmonic Pitch Class Profile* (HPCP) aplicada em 10 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz e com resolução 12.

Recentemente Bogdanov et al. [4] disponibilizaram uma biblioteca, denominada *Essentia*[4], que está implementada em python e em C++. Essa biblioteca contêm vários métodos de pré-processamento e extratores de características de um sinal de áudio, incluindo o código do HPCP, que foi desenvolvido a partir de [23, 27].

3.1.5 Rastreamento das Batidas

O método de Rastreamento das Batidas, ou *Beat Tracking* é importante em muitas aplicações, pois possibilita encontrar a posição temporal das notas. Um dos problemas

[4]Os códigos utilizados para reproduzir o extrator HPCP está disponível em http://essentia.upf.edu/



(a) HPCP aplicado na música *Imagine* na versão por *John Lennon*.

(b) HPCP aplicado na música *Imagine* na versão por *Diana Ross*.

(c) HPCP aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*.

Figura 3.10: Atributos resultantes do método *Harmonic Pitch Class Profile* (HPCP) aplicada em 10 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz e com resolução 36.

na aplicação da técnica *Beat Tracking* é que, em arquivos polifônicos, a quantidade de instrumentos gravados em um mesmo canal dificulta a identificação e a obtenção precisa das notas de ataque. Outra dificuldade é que algumas partes da música contêm picos de energia que não são exatamente relacionados com as batidas [30]. O processo de cálculo do método *Beat Tracking* está ilustrado na Figura 3.11 e tem 3 etapas básicas:

1. Tempo de Ataque (*Onset Time*)
2. Mudanças de acordes
3. Padrão da Percussão

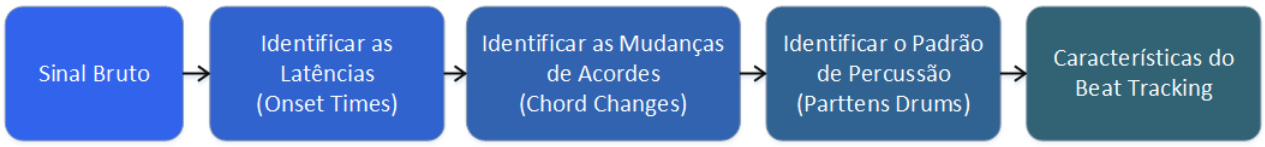


Figura 3.11: Diagrama do Processo de extração de Características do *Beat Tracking*

O tempo de ataque (ou *onset time*) inicia-se com STFT. Depois faz-se a extração dos componentes de ataque do sinal derivadas do espectro de frequências obtido. Nessa etapa, os componentes de frequências são relacionados com os componentes de ataque de acordo com a condição 3.25, onde $X(t, f)$ é a potência do espectro de frequência f no tempo t obtido com o STFT [30].

$$\begin{cases} X(t,f) > pp \\ np > pp \end{cases} \tag{3.25}$$

$$pp = max(X(t-1,f), X(t-1,f \pm 1), X(t-2,f)) \tag{3.26}$$

$$np = min(X(t+1,f), X(t+1,f \pm 1)) \tag{3.27}$$

As equações 3.26 e 3.27 mostram o cálculo de *pp* e *np* para extrair os componentes de frequência onde houve um aumento de potência. O grau do ataque é dado pela Equação 3.28. A partir das componentes extraídas encontra-se o tempo dos ataques que é dado pela soma dos graus de ataque $D(t)$ ao longo do tempo como mostra a Equação 3.29. Porém, antes de calcular o $D(t)$ é feito um alinhamento linear por convolução de Kernel [30].

$$d(t, f) = X(t, f) - pp + max(0, X(t + 1, f) - X(t, f)) \tag{3.28}$$

$$D(t) = \sum_{f} d(t, f) \tag{3.29}$$

Todas essas equações que definem o grau de ataque $d(t, f)$ podem ser simplificadas utilizando a condição 3.30, para que posteriormente o processo da análise das frequências encontrando o tempo dos ataques ($D(t)$) seja feito [28].

$$d(t, f) = \begin{cases} max(X(t, f), X(t + 1, f)) - PrevPow(min(X(t, f), X(t + 1, f)) > PrevPow), \\ 0, \text{caso contrário} \end{cases} \tag{3.30}$$

$$PrevPow = max(X(t - 1, f), X(t - 1, f \pm 1)) \tag{3.31}$$

Até agora, tudo o que foi apresentado faz parte do tempo de ataque (*onset time*) que é a primeira etapa do método *Beat Tracking*. Após concluída esta etapa, deve-se detectar as mudanças de acordes do sinal. Essa detecção é feita sem que se tenha um conhecimento prévio das notas e dos acordes. Nos componentes de frequência é possível encontrar as notas dos acordes como também a harmônica, assim, se todos os componentes de frequência forem considerados, existirá uma tendência em ocorrer uma mudança significativa no momento de mudança de acorde e uma estabilidade relativa quando não existe uma mudança [28].

Os componentes de frequência são difíceis de serem encontradas corretamente a partir do sinal bruto, então, para identificar os componentes de frequência dominantes, é feito um histograma inicial com as componentes existentes. Isso é feito utilizando o STFT no sinal. Considerando o mapeamento de ω (frequência central ou frequência média de um filtro STFT) para a sua frequência instantânea, e tendo $\lambda(\omega, t)$ como sua saída, existirá uma frequência ψ, que se colocada em um ponto fixo do mapeamento, as frequências exatamente ao redor de ψ ficarão constantes no mapeamento. Então existirá um conjunto $\Psi_f^{(t)}$ de frequências instantâneas candidatas a serem componentes de frequência as quais são extraídas usando a seguinte equação [29]:

$$\Psi_f^{(t)} = \{\psi | \lambda(\psi, t) - \psi = 0, \frac{\delta}{\delta\psi}(\lambda(\psi, t) - \psi) < 0\} \tag{3.32}$$

O conjunto Ψ é chamado de espectro de potência. A função de distribuição desse espectro de potências é dado pela equação 3.33, sendo que esta formará um histograma de frequências do sinal [29].

$$\Psi_p^t(\omega) = \begin{cases} |X(\omega, t)| \text{se } \omega \in \Psi_f^{(t)} \\ 0, \text{caso contrário} \end{cases} \tag{3.33}$$

Com isso, obtém-se o histograma das frequências. Para a identificação das mudanças do acorde é necessário dividir o espectro de frequências em faixas de acordo com o tempo de batida provisório encontrado. Esse tempo de batida provisório é estimado utilizando o histograma de componentes de frequência. A mudança é obtida quando comparado os componentes dominantes de frequências das faixas adjacentes. Esse processo tem a vantagem de não necessitar da identificação dos acordes, o que é uma tarefa geralmente complexa [28].

Passado a etapa das latências e da mudança de acordes, a última etapa, denominada Padrão de Percussão, é feita para identificar o bumbo e a caixa da bateria. O bumbo é detectado utilizando a latência dos componentes, e a caixa é detectada utilizando o ruído dos componentes. A maioria das detecções de latência desta etapa é feita a partir deste tempo de batida provisório [28].

Primeiramente, é feita a detecção do bumbo. Como não se sabe nada a priori, uma aprendizagem da característica da frequência é feita examinando a latência dos componentes extraídas anteriormente $d(t, f)$. O bumbo é representado pelos picos de frequência grave do histograma, e a caixa pelos picos de frequência aguda. Assim, o bumbo e a caixa são encontrados quando um ataque é detectado ou quando um pico de frequência coincide com os picos de frequência do bumbo e da caixa [30].

No caso, a caixa é tipicamente os componentes de ruido $n(t, f)$ distribuídos ao longo do tempo dado pela Equação 3.34, onde *HighFreqAve* e *LowFreqAve* representam as médias locais das regiões de altas e baixas frequências respectivamente [28].

$$n(t,f) = \begin{cases} X(t,f)\ (min(HighFreqAve, LowFreqAve) > \frac{1}{2}X(t,f)), \\ 0, \text{ caso contrário} \end{cases} \tag{3.34}$$

$$HighFreqAve = \frac{1}{4}(X(t, f+2) + \sum_{i=-1}^{1} X(t+i, f+1)) \tag{3.35}$$

$$LowFreqAve = \frac{1}{4}(X(t, f-2) + \sum_{i=-1}^{1} X(t+i, f-1)) \tag{3.36}$$

Finalmente, calcula-se o *Beat Tracking* utilizando programação dinâmica, assumindo inicialmente que o sinal musical tem um ritmo constante. A finalidade principal é gerar uma sequência de batidas regulares, que corresponde à percepção de ataques do sinal. Assim, define-se uma função objetiva [19]:

$$C(\{t_i\}) = \sum_{i=1}^{N} O(t_i) + \alpha \sum_{i=2}^{N} F(t_i - t_{i-1}, \tau_p) \tag{3.37}$$

Onde, t_i é a sequência de N instantes de batidas encontrados pelo rastreador, $O(t)$ é um *onset strength envelop*, derivado do sinal de áudio, que corresponde a etapa de *Onset Time*, α é um peso para balancear a importância de dois termos, e $F(\Delta t, \tau_p)$ é função da medida de consistência entre um intervalo entre batidas δt e um espaço ideal de batidas τ_p definido por um ritmo alvo.

Para F utiliza-se uma simples função de erros quadráticos aplicada à uma razão logarítmica do atual e do ideal espaçamento de tempo (Equação 3.38), o qual tem como valor

máximo 0 quando $\Delta t = \tau$, e começa a diminuir quando existe divergência entre ambos, e é simétrico no eixo temporal logarítmico quando $F(k\tau, \tau) = F(\frac{\tau}{k}, \tau)$ [19].

$$F(\Delta t, \tau) = -\left(\log \frac{\Delta t}{\tau}\right)^2 \tag{3.38}$$

A propriedade chave da função objetivo é a sequência do tempo de melhor pontuação $C^*(t)$, que pode ser obtida recursivamente pela relação definida na Equação 3.39. Esta é baseada nas observações da melhor pontuação da força de ataque do tempo t, somadas a melhor pontuação da batida temporal precedente τ, para maximizar a soma da melhor pontuação e do custo de transição do tempo corrente. Nessa etapa também é armazenado a batida precedente que resultou em uma melhor pontuação como mostra a Equação 3.40, onde $\tau = t - 2\tau_p ... t - \frac{\tau_p}{2}$ [19].

$$C^*(t) = O(t) + \max_{\tau=0,\ldots,t} \{\alpha F(t - \tau, \tau_p) + C^*(t)\} \tag{3.39}$$

$$P^*(t) = \operatorname*{argmax}_{\tau=0,\ldots,t} \{\alpha F(t - \tau, \tau_p) + C^*(t)\} \tag{3.40}$$

Dado um envelope de ataque para a função objetivo, calcula-se C^* e P^* para todo o tempo t, começando do tempo inicial 0. Assim, encontra-se o maior valor de C^*, o qual está no intervalo de τ_p ate o final do alcance temporal, resultando no instante final de batidas t_N, onde N é o número total de batidas. Fazendo o *backtrace* através de P^*, encontrando a batida precedente $t_{N-1} = P^*(t_N)$, e assim sucessivamente até o inicio do sinal, isso resultará em uma sequência de batidas ótima $\{t_i\}^*$.

O intervalo de tempo ideal τ_p é considerado 120 BPM, que é conhecido como sendo o tempo de percepção humana. Porém, esse valor pode ser estimado aplicando pesos perceptivos na autocorrelação do sinal para diminuir a periodicidade dos picos, distanciando do viés de 120 BPM. A força do período do tempo é dado pela Equação 3.41, onde $W(t)$ é uma função de peso Gaussiana (Equação 3.42), τ_0 é o centro do viés do período do tempo, e σ_τ controla a curva de ponderação. [19].

$$TPS(\tau) = W(\tau) \sum_t O(t)O(t - \tau) \tag{3.41}$$

$$W(t) = \exp\left\{-\frac{1}{2}\left(\frac{\frac{\log_2 \tau}{\tau_0}}{\sigma_\tau}\right)^2\right\} \tag{3.42}$$

Assim, o primeiro tempo estimado é o τ que resultar o maior $TPS(\tau)$. Depois, outras duas funções são calculadas, Equação 3.43 e Equação 3.44, para re-amostrar o TPS de modo a obter metade e um terço de seu tamanho original. Os novos TPS são sobrepostos ao TPS original, então é escolhido o maior pico que cruza essas novas sequências como período do tempo, ou ritmo [19].

$$TPS2(\tau) = TPS(\tau) + 0.5TPS(2\tau) + 0.25TPS(2\tau - 1) + 0.25TPS(2\tau + 1) \tag{3.43}$$

$$TPS3(\tau) = TPS(\tau) + 0.33TPS(3\tau) + 0.33TPS(3\tau - 1) + 0.33TPS(3\tau + 1) \tag{3.44}$$

Em que, os valores de τ_0 e σ_τ são 0.5 segundos (120 BPM) e maior ou igual a 0.9 oitavas, respectivamente [19].

3.2 Técnicas de Comparação

Esta seção apresenta alguns métodos de comparação de características utilizados na identificação de músicas covers.

3.2.1 Correlação Cruzada

A Função de Correlação Cruzada, do inglês *Cross-Correlation Function* (CCF), é utilizada para calcular a similaridade entre duas séries temporais [38]. Para entender a função de correlação cruzada é necessário conhecer a função de autocorrelação, ou *autocorrelation function* (ACF), que é uma abordagem intuitiva para encontrar periodicidade no sinal, o que equivale a encontrar a frequência fundamental, pois as harmônicas são igualmente espaçadas no espectro de magnitude e a frequência fundamental é igual a distância de harmônicas vizinhas [38].

Com a ACF é possível caracterizar sequências de estruturas no domínio do tempo, pois a distribuição da amplitude permite encontrar uma amostra individual em vários níveis. Essa função retorna a dependência temporal entre as observações da série apresentando sua estrutura no domínio do tempo. O cálculo da ACF tem relação com a Função de Autocovariância, do inglês *Autocovariance Function* (ACV) [47].

A ACV mostra a covariância de uma série entre uma observação e outra com um atraso de k observações entre elas. Ou seja, para uma série Y a autocovariância entre duas observações de atraso k será a autocovariância entre uma observação Y_t no tempo t e uma observação Y_{t-k} no tempo $t-k$, dado na Equação 3.45, onde μ_y representa a média de Y [61].

$$ACV(k) = E(Y_t Y_{t-k}) = \sum_{t=1}^{n-k} (Y_t - \mu_y)(Y_{t-k} - \mu_y) \tag{3.45}$$

A Equação 3.46 define a função de autocorrelação, onde $E(\cdot)$ é o valor esperado, k é um atraso entre as observações, μ_y representa a média de Y e σ_y representa o desvio padrão.

$$\begin{aligned} ACF(k) &= \frac{ACV(Y_t, Y_{t-k})}{\sigma_Y \times \sigma_Y} \\ &= \frac{\sum_{t=1}^{n-k}(Y_t - \mu_y)(Y_{t-k} - \mu_y)}{\sum_{i=1}^{n}(Y_t - \mu_y)^2/n} \\ &= \frac{E(Y_t Y_{t-k})}{\sigma_Y^2} \end{aligned} \tag{3.46}$$

A Função de Correlação Cruzada (CCF) é essencialmente similar à função de autocorrelação, porém esta é computada entre duas séries distintas, com o objetivo de identificar estruturas similares. As séries devem ter mesmo tamanho, e caso não tenham, completa-se a menor com zeros para deixá-las do mesmo tamanho. Para calcular a CCF é necessário

conhecer a Função de Covariância Cruzada, ou *Cross-Covariance Function* (CCV) definida na Equação 3.47, que utiliza a covariância dos dados, em que X e Y representam séries distintas [9].

$$CCV(k) = \text{Cov}(X_t, Y_{t+k}) = \text{E}[(X_t - \mu_x)(Y_{t+k} - \mu_y)] \tag{3.47}$$

A CCF, então, é definida pela Equação 3.48, onde $\lambda_{xy}(k)$ é a CCV, σ_x é o desvio padrão da série X e σ_x é o desvio padrão da série Y [9].

$$CCF_{xy}(k) = \frac{\text{Cov}(X_t, Y_{t+k})}{\sqrt{\text{Cov}(X_t, X_t)\text{Cov}(Y_t, Y_t)}} = \frac{\text{Cov}(X_t, Y_{t+k})}{\sigma_x \sigma_y} \tag{3.48}$$

Esta função é assimétrica, significando que quando for positivo a série X tem um avanço em relação a Y, e quando for negativo a série X tem um atraso em relação à Y [61]. Além disso essa função tem as seguintes propriedades:

- $CCF_{xy}(k) = CCF_{xy}(-k)$
- $|CCF_{xy}(k)| \leq 1$

Em que $CCF_{xx}(0) = CCF_{yy}(0) = 1$, porém $CCF_{xy}(0)$ não é necessariamente 1, pois as séries podem ser não correlatas.

3.2.2 Análise de Quantificação Recorrente

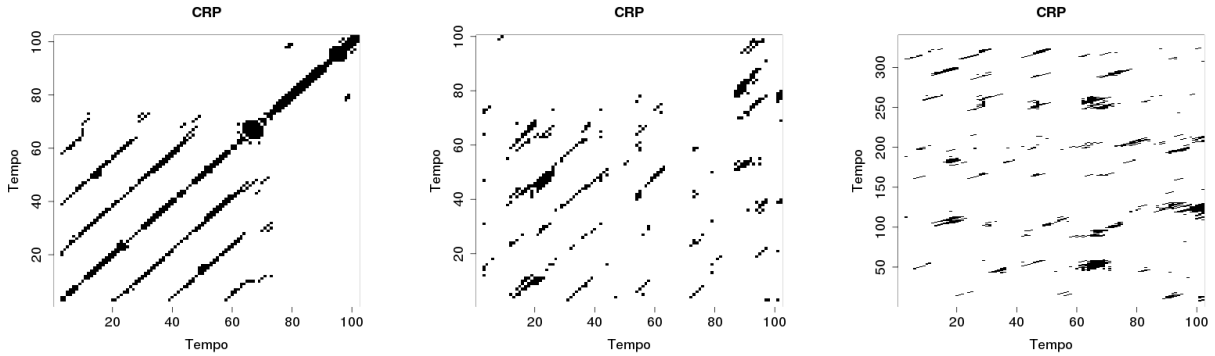
A Análise de Quantificação Recorrente, ou *Recurrence Quantification Analysis* (RQA), é um critério quantitativo para avaliar a similaridade entre duas séries. Para isso é necessário abordar o método *Cross Recurrence Plots* (CRP), uma generalização do *Recurrence Plots* (RP), pois a RQA será aplicada sobre a matriz retornada pelo CRP.

O método RP é uma maneira de visualizar as similaridades entre um sinal e ele mesmo em diferentes instantes de tempo, retornando uma matriz quadrada, em que cada coordenada (i,j) representa se existe similaridade entre os termos i e j. Sendo assim, o 0 significa que não há similaridade e 1 que há similaridade. Em todo caso, a diagonal principal é o caminho que representa a similaridade do sinal [51].

A construção do CRP é feita da mesma forma que o RP, porém a matriz pode ser retangular, e não unicamente quadrada. Além disso, permite destacar equivalências de estado entre dois sistema em diferentes posições no tempo. Quando é usado para caracterizar sistemas distintos, qualquer caminho seguido de valores 1 representa a similaridade dos dois sistemas [51]. A matriz resultante do CRP está ilustrada na Figura 3.12, onde a cor preta mostra onde houve similaridade e a cor branca onde não houve. A matriz é computada utilizando a Equação (3.49), onde pode-se notar que o valor resultante será 1 somente se x_i e y_i forem similares.

$$R(i,j) = \Theta(\epsilon_i^x - \| x_i - y_j \|)\Theta(\epsilon_j^y - \| x_i - y_j \|) \tag{3.49}$$

Para $i = 1, ..., N_x$ e $j = 1, ..., N_y$, onde x_i e y_j são duas séries distintas, $\theta(\cdot)$ é uma *heaviside step function* mostrada na equação 3.50, ϵ_i^x e ϵ_j^y são duas distâncias limitantes distintas, e $\| \cdot \|$ é uma norma, no caso, a norma euclidiana [51].



(a) CRP da música *Imagine* na versão por *John Lennon* com ela mesma.

(b) Matriz de Recorrência da música *Imagine* na versão por *John Lennon* com a versão por *Diana Ross*.

(c) CRP da música *Imagine* na versão por *John Lennon* com música *Book I Prelude* na versão por *Christiane Jaccotte*.

Figura 3.12: Matrizes de Recorrência resultantes do método *Cross Recurrence Plots* (CRP) aplicada em 38 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 8 kHz, com dimensão embutida $m = 10$ e atraso temporal de $d = 1$, para 464 ms com 75% de sobreposição, resolução 12 e fração de vizinhos em 0.1.

$$\theta(v) = \begin{cases} 0 \text{ se } v < 0 \\ 1 \text{ caso contrário} \end{cases} \tag{3.50}$$

Os limitantes ϵ_i^x e ϵ_j^y são a máxima porcentagem dos vizinhos k aplicados em ambos x_i e y_i. Com isso, ϵ deve ser estimado a partir dessa porcentagem de vizinhos, que foi definido em uma pré-analise por Serrà et. al. [51], como sendo o valor 0.1. Assim, para cada ponto da série x são selecionados 0.1 vizinhos do total de pontos da série y, e ϵ_i^x será a distância entre o ponto até seu vizinho k, sendo este o raio da vizinhança do ponto de x. Esse processo é feito da mesma forma para os pontos da série y que terá como vizinhos os pontos da série x. A quantidade de entradas não zeros das linhas e colunas não ultrapassa kN_y e kN_x [51].

Com a matriz do CRP calculada para as duas séries, é aplicado um critério quantitativo para determinaro nível de similaridade entre elas. Para isso, existem 3 métricas diferentes de RQA. Os estudos mostram que a similaridade pode ser definida pelo tamanho das diagonais produzidas no CRP [51].

A primeira métrica é a L_{max} que pode ser expressa com o máximo valor de uma matriz acumulativa L. Essa métrica é calculada utilizando a equação (3.51) [51].

$$L_{i,j} = \begin{cases} L_{i-1,j-1} + 1, \text{ se } R_{i,j} = 1 \\ 0, \text{ se } R_{i,j} = 0 \end{cases} \tag{3.51}$$

Para $i = 2, ..., N_x$ e $j = 2, ..., N_y$, com a inicialização $L_{1,j} = L_{i,1} = 0$ e $L_{max} = \max L_{i,j}$, ambos para $i = 1, ..., N_x$ e $j = 1, ..., N_y$. Além disso, ela aplicada recursivamente. Esse método permite identificar passagens de um sinal inseridos em outro, pois ela considera as diagonais independente de suas posições o que representam as mudanças de estruturas. Apesar disso, a medida L_{max} não considera as mudanças temporais [51].

Por conta dessa limitação, a equação 3.51 foi estendida para quantificar os traços de curva no tempo para isso é computada a matriz acumulativa S a partir do CRP como mostra a equação 3.52 [51].

$$S_{i,j} = \begin{cases} \max(S_{i-1,j-1}, S_{i-2,j-1}, S_{i-1,j-2}) + 1, \text{ se } R_{i,j} = 1 \\ 0, \text{ se } R_{i,j} = 0 \end{cases} \tag{3.52}$$

Para $i = 3, ..., N_x$ e $j = 3, ..., N_y$, com a inicialização $S_{1,j} = S_{2,j} = S_{i,1} = S_{i,2} = 0$ e $S_{max} = \max S_{i,j}$, ambos para $i = 1, ..., N_x$ e $j = 1, ..., N_y$. S_{max} corresponde ao tamanho do traço de curva mais longo do CRP. Esse método não considera caminhos verticais e horizontais, e mesmo identificando os desvios no tempo, o S_{max} não identifica os caminhos de similaridade caso a série tenha curtas interrupções [51].

Por conta dessa limitação a equação 3.52 foi estendida para o Q_{max}. O Q_{max} também é calculado a partir do CRP como mostra a equação 3.53 [51].

$$Q_{i,j} = \begin{cases} \max(Q_{i-1,j-1}, Q_{i-2,j-1}, Q_{i-1,j-2}) + 1, \text{ se } R_{i,j} = 1 \\ \max(0, (Q_{i-1,j-1} - \gamma(R_{i-1,j-1})), (Q_{i-2,j-1} - \gamma(R_{i-2,j-1}), (Q_{i-1,j-2} - \gamma(R_{i-1,j-2}))), \text{ se } R_{i,j} = 0 \end{cases} \tag{3.53}$$

Para $i = 3, ..., N_x$ e $j = 3, ..., N_y$, podendo inicializar $Q_{1,j} = Q_{2,j} = Q_{i,1} = Q_{i,2} = 0$ e $Q_{max} = \max Q_{i,j}$, ambos para $i = 1, ..., N_x$ e $j = 1, ..., N_y$. Também é aplicado recursivamente, onde γ é expresso pela equação 3.54, onde γ_o representa a interrupção no tempo e γ_e representa a interrupção na extensão. [51].

$$\gamma(z) = \begin{cases} \gamma_o, \text{ se } z = 1 \\ \gamma_e, \text{ se } z = 0 \end{cases} \tag{3.54}$$

O zero na segunda cláusula impede que a penalidade seja negativa. Se γ_o e γ_e tenderem a infinito, a Equação 3.53 se torna a Equação 3.52, e se γ_o e γ_e forem zero, $Q_{i,j}$ se torna acumulativo e representa a similaridade global das duas séries temporais, que começa na amostra 0 e termina na amostra i e j, respectivamente. Serrà et. al. utiliza $\gamma_o = 5$ e $\gamma_e = 0.5$ [51].

3.3 Técnicas de Agrupamento

Esta seção apresenta conceitos de técnicas de agrupamento utilizadas na metodologia deste trabalho de mestrado. O objetivo principal do uso dessas técnicas é agrupar características extraídas dos sinais de modo a encontrar similaridades entre os dados. Em outras palavras, busca-se encontrar grupos de músicas de forma que as que fazem parte do mesmo grupo sejam similares entre si; e objetos de grupos diferentes sejam dissimilares entre si.

3.3.1 K-médias

A técnica K-médias, do inglês *K-means*, é uma das técnicas de agrupamento mais utilizadas na literatura. Esta técnica é classificada como particional ou não-hierárquica, resultando em operar uma partição dos dados (grupos disjuntos), sendo que cada grupo

terá um centro chamado centroide ou média (*mean*). O algoritmo de K-médias resultará em k grupos, onde k é um valor fixo e fornecido como entrada [24].

O procedimento inicia-se com um número k de grupos pré-determinado, os dados vão sendo alocados aos grupos, de acordo com as proximidades ao centroides dos grupos, ou seja, o dado é atribuído a um grupo se ele está mais próximo do centroide deste grupo do que dos demais centroides. Centroides vão sendo reajustados para a média dos elementos do grupo, sendo os membros do grupos realocados de acordo com suas proximidades dos novos centroides. Esse processo de realocação acontece repetidas vezes de acordo com uma função de erro, e se repete enquanto o erro for significativo [24].

No algoritmo convencional, dado D o conjunto com n instâncias, e $C_1, C_2, ..., C_k$ os k grupos disjuntos formados em D, a função de erro é definida como a Equação 3.55, onde $\mu(C_i)$ é centroide do grupo C_i, $d(x, \mu(C_i))$ é a distância entre x e $\mu(C_i)$, tipicamente a distância Euclidiana [24].

$$E = \sum_{i=1}^{k} \sum_{x \in C_i} d(x, \mu(C_i)). \tag{3.55}$$

O Algoritmo 1 pode ser dividido em duas partes. A primeira fase é a inicialização, onde o algoritmo faz o primeiro agrupamento associando os dados com os centros de k grupos selecionados aleatoriamente, e a segunda fase é a iteração, onde o algoritmo calcula as distâncias entre todos os elementos do conjunto D e a média de cada grupo k, associando o elemento ao grupo mais próximo [24].

Algoritmo 1: Algoritmo convencional da técnica K-médias

Entrada: Conjunto de Dados D, Número de grupos k, Dimensão d
Saída: Resultado do Agrupamento
inicio
  C_i é o i-ésimo grupo;
  Fase de Inicialização;
  $(C_1, C_2, ..., C_k)$ = partição inicial de D;
  repita
    d_{ij} = distância entre o elemento i e o grupo j;
    $n_i = \text{argmin}_{1 \leq j \leq k} d_{ij}$;
    Associe o elemento i ao grupo n_i;
    Recalcule o centro dos grupos se houve mudança de grupos;
  até *que nenhuma mudança aconteça nos membros dos grupos em uma iteração completa*;
fin

Algumas propriedades da técnica são [24]:

- É eficiente em grandes bases de dados pois sua complexidade é linearmente proporcional ao tamanho dos dados;
- Frequentemente termina em um ótimo local;
- Os grupos tem formas convexas;

- O desempenho é dependente da inicialização dos centros.

Dentre as limitações está a inicialização aleatória dos centros, que pode não resultar no melhor agrupamento dos dados. Além disso, esta técnica não é eficiente quando os dados estão em dimensão muito alta. Porém existem variações desta técnica que otimizam o processo [24]. A Imagem 3.13 ilustra um agrupamento de pontos utilizando K-médias para 3 valores de k.

3.3.2 Ligação Simples

A técnica de agrupamento Ligação Simples, do inglês, *Single-Linkage* é uma técnica de agrupamento hierárquico aglomerativo, também conhecida, como método dos vizinhos próximos (*nearest neighbors method*), método de mínimos (*minimum method*), e o método de conectividade (*connectedness method*) [24].

Para medir a similaridade entre dois grupos essa técnica utiliza a distância de vizinhos próximos. Assim, dado três grupos C_i, C_j e C_k de pontos, a distância entre o grupo C_k e os grupos $C_i \cup C_j$ podem ser obtidas através da fórmula 3.56 de Lance-Williams [24]. A distância entre 2 grupos será a distância entre 2 elementos, um de cada grupo, mais próximos entre si.

$$D(C_k, C_i \cup C_j) = \frac{1}{2}D(C_k, C_i) + \frac{1}{2}D(C_k, C_j) - \frac{1}{2}|D(C_k, C_i) - D(C_k, C_j)| \tag{3.56}$$

$$D(C, C') = \min_{x \in C, y \in C'} d(x, y) \tag{3.57}$$

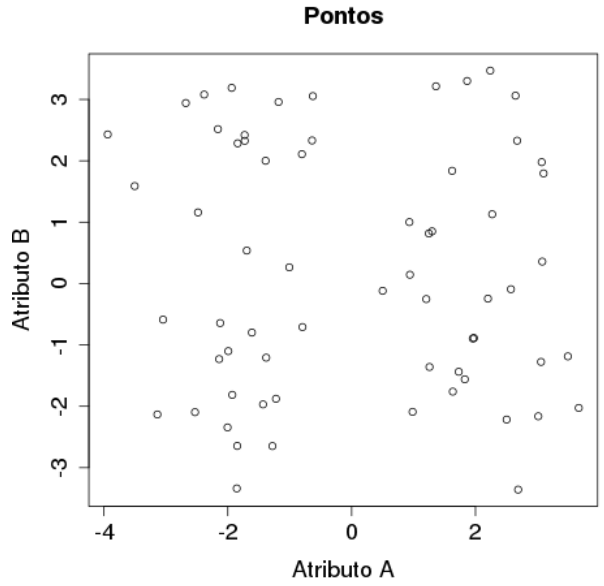
Onde $D(\cdot, \cdot)$ é a distância entre dois grupos, C e C' são dois grupos não vazios e sem sobreposição, e $d(\cdot, \cdot)$ é uma função de distância. A *Single-Linkage* é classificada em cinco tipos diferentes:

- Algoritmo de conectividade;
- Algoritmo baseado em uma transformação ultra métrica;
- Algoritmo estimação de densidade probabilística;
- Algoritmo aglomerativo;
- Algoritmo baseado em árvore mínima;

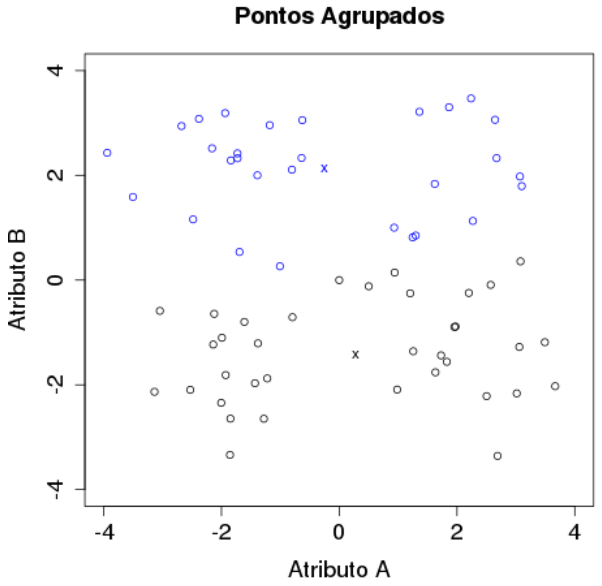
O algoritmo de conectividade é baseado na teoria dos grafos, onde cada ponto dos dados representa um vértice de um grafo, e as arestas desse grafo é a distância entre os pontos. Um par (i, j) de pontos é conectado somente se a distância entre i e j for $d_{i,j} \leq \Delta$. Os grupos da *Single-Linkage* no nível Δ corresponde ao conjunto pontos conectados do grafo. Esse algoritmo requer um esforço computacional considerável [24].

Para ilustrar o funcionamento do *Single-Linkage* com conectividade, considere a matriz de dissimilaridade da Tabela 3.1. Essa matriz foi calculada utilizando a distância Euclidiana para os pontos $x_1 = (1, 2)$, $x_2 = (1, 2.5)$, $x_3 = (3, 1)$, $x_4 = (4, 0.5)$, $x_5 = (4, 2)$ [24].

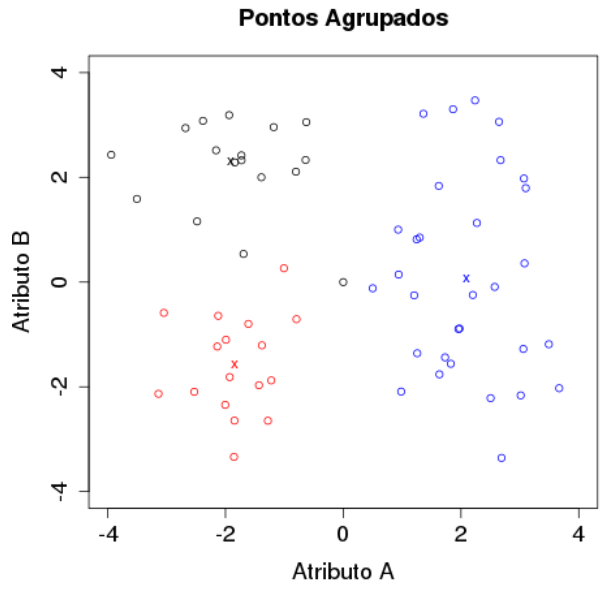
No inicio todos os pontos formam grupos individuais e a técnica vai unindo os pontos de modo que no final exista apenas um único grupo com todos os pontos. Assim, aplicando



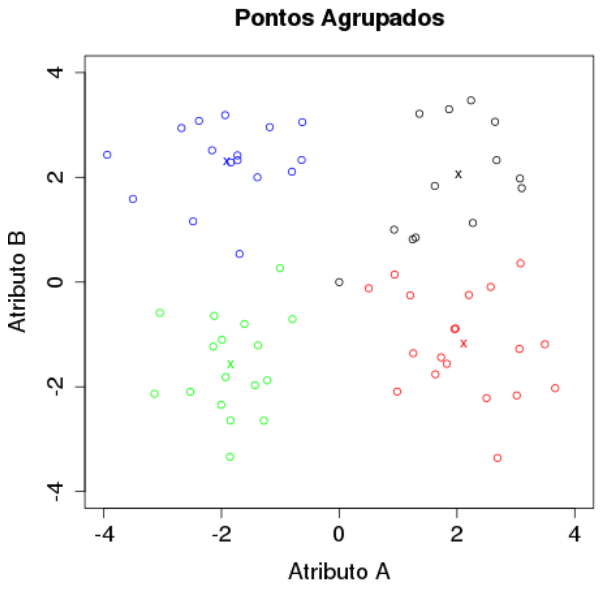
(a) Pontos no espaço de 2 dimensões.



(b) Agrupamento dos pontos da imagem *a* com 2 grupos.



(c) Agrupamento dos pontos da imagem *a* com 3 grupos.



(d) Agrupamento dos pontos da imagem *a* com 4 grupos.

Figura 3.13: Agrupamento dos pontos da imagem **a** utilizando a técnica K-médias com $k = 2$ (imagem **b**),com $k = 3$ (imagem **c**) e com $k = 4$ (imagem **d**), limitado a 25 iterações. As cores indicam o grupo do qual aquele ponto pertence e os centros do grupo estão representados pelo "x".

a *Single-Linkage* tem-se como primeiro estágio do algoritmo, a união dos pontos x_1 e x_2 em um grupo, por serem os mais próximos entre si. Desta forma recalcula-se uma nova matriz de dissimilaridade (Tabela 3.2), a qual obtém-se a distância entre x_1, x_2 e x_3, x_4 e x_5 [24].

Tabela 3.1: Matriz de Dissimilaridade do primeiro estágio do algoritmo da *Single-Linkage*, onde cada entrada (i, j) é a distância Euclidiana entre o ponto x_i e o ponto x_j

	x_1	x_2	x_3	x_4	x_5
x_1	0	0.5	2.24	3.35	3
x_2	0.5	0	2.5	3.61	3.04
x_3	2.24	2.5	0	1.12	1.41
x_4	3.35	3.61	1.12	0	1.5
x_5	3	3.04	1.41	1.5	0

$$\begin{aligned} D(\{x_1, x_2\}, x_3) &= \min\{d(x_1, x_3), d(x_2, x_3)\} = 2.24 \\ D(\{x_1, x_2\}, x_4) &= \min\{d(x_1, x_4), d(x_2, x_4)\} = 3.35 \\ D(\{x_1, x_2\}, x_5) &= \min\{d(x_1, x_5), d(x_2, x_5)\} = 3 \end{aligned} \tag{3.58}$$

Tabela 3.2: Matriz de Dissimilaridade do segundo estágio do algoritmo da *Single-Linkage*, onde cada entrada (i, j) é a distância Euclidiana entre os grupos

	x_1, x_2	x_3	x_4	x_5
x_1, x_2	0	2.24	3.35	3
x_3	2.24	0	1.12	1.41
x_4	3.35	1.12	0	1.5
x_5	3	1.41	1.5	0

No segundo estágio do algoritmo, os pontos que serão unidos na formação do grupo são os pontos x_3 e x_4 por terem a menor distância. Com isso a matriz de dissimilaridade é atualizada novamente gerando, a Tabela 3.3 pela seguinte equação [24]:

$$\begin{aligned} &D(\{x_3, x_4\}, \{x_1, x_2\}) \\ &= \min\{d(x_1, x_3), d(x_2, x_3), d(x_1, x_4), d(x_2, x_4)\} \\ &= \min\{D(\{x_1, x_2\}, x_3), D(\{x_1, x_2\}, x_4)\} = 2.24 \end{aligned} \tag{3.59}$$

Tabela 3.3: Matriz de Dissimilaridade do terceiro estágio do algoritmo da *Single-Linkage*, onde cada entrada (i, j) é a distância Euclidiana entre os grupos

	x_1, x_2	x_3, x_4	x_5
x_1, x_2	0	2.24	3
x_3, x_4	2.24	0	1.41
x_5	3	1.41	0

No terceiro estágio o ponto x_5 será acrescentado ao grupo $\{x_3, x_4\}$:

No ultimo estágio os dois grupos serão unidos formando um único grupo com todos os pontos [24].

O dendrograma é uma representação utilizada em agrupamentos hierárquicos, também conhecido como árvore de valores. O dendrograma é uma n-árvore (*n-tree*), onde cada nó interno é associado há uma altura que satisfaça a condição 3.60 para todo o subconjunto de pontos A e B se $A \cap B = \Phi$, onde $h(A)$ e $h(B)$ são as alturas respectivas de A e B, e Φ é um conjunto vazio.

Tabela 3.4: Matriz de Dissimilaridade do quarto e último estágio do algoritmo da *Single-Linkage*, onde cada entrada (i, j) é a distância Euclidiana entre os grupos

	x_1, x_2	x_3, x_4, x_5
x_1, x_2	0	2.24
x_3, x_4, x_5	2.24	0

$$h(A) \leq h(B) \Leftrightarrow A \subseteq B \tag{3.60}$$

A altura do nó interno h_{ij} será o menor valor da alta similaridade entre os pontos x_i e x_j, representando o menor grupo, o qual x_i e x_j pertencem. A ilustração do dendrograma para o exemplo anterior pode ser visto na Figura 3.14.

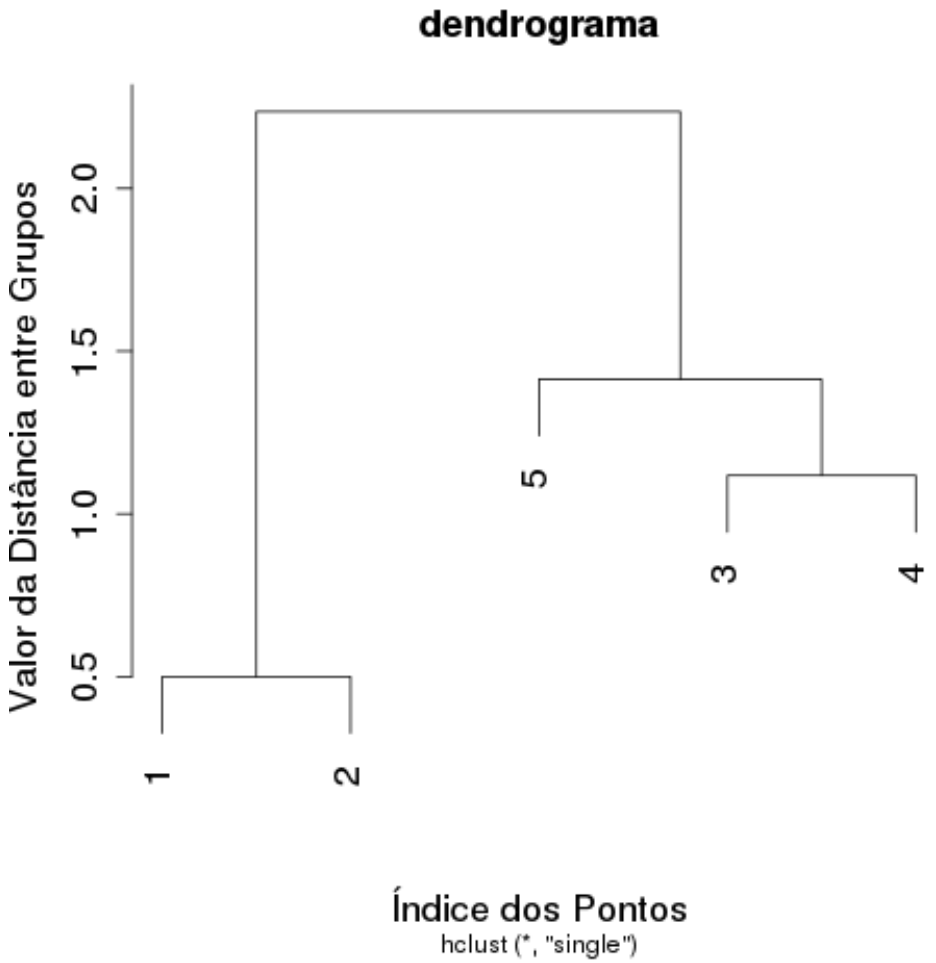


Figura 3.14: Dendrograma do resultado obtido com a técnica *Single-Linkage* para o exemplo apresentado.

O resultado final do agrupamento é definido pela quantidade de grupos desejados, que pode ser obtida através de um corte no dendrograma.

3.4 Métricas

Neste capítulo será descrito as métricas utilizadas para avaliar o método proposto, como também, para comparar os seus resultados com os resultados dos métodos apresentado no capítulo 5.

3.4.1 Média das Precisões Médias (MAP)

Esta medida é muito utilizada na área de recuperação de informação (IR) e na de recuperação de informações de músicas (MIR) [51], como também na avaliação de trabalhos de identificação de músicas cover da competição MIREX *(Music Information Retrieval Evaluation eXchange)*, que visa a avaliação formal de trabalhos na área de MIR.

Dado um conjunto de músicas D e a matriz de dissimilaridade desse conjunto, calcula-se a medida MAP denotada por Ψ que será a média da precisão média de cada música ψ_q. A equação 3.61 mostra o cálculo da média da precisão de cada música. A matriz de dissimilaridade é utilizada para criar uma lista Δ_q, sendo esta, uma lista de músicas ordenadas de forma crescente a respeito da música analisada q [51].

$$\psi_q = \frac{1}{C_q} \sum_{r=1}^{D-1} P_q(r) I_q(r), \tag{3.61}$$

onde $C_q = D - 1$, (quantidade total de músicas menos a música analisada), $P_q(r)$ é a precisão da lista Δ_q na posição r, como mostra 3.62, e $I_q(r)$ é uma função de relevância dado pela equação 3.63 [51]:

$$P_q(r) = \frac{1}{r} \sum_{l=1}^{r} I_q(l), \tag{3.62}$$

$$I(u) = \begin{cases} 1 & \text{se a música } u \text{ for cover de } q \\ 0 & \text{caso contrário} \end{cases} \tag{3.63}$$

3.4.2 Separabilidade

A outra medida utilizada é a Separabilidade, que mostra qual a precisão de acerto do método de acordo com o grau de busca. Dado uma matriz de dissimilaridade dos conjunto de dados D, onde cada linha da matriz representa a distância de uma música para todas as outras.

Ao retirar uma linha dessa matriz, e ordenar de forma crescente, obtemos uma classificação de músicas similares R_q, onde q será a música verificada. A primeira música da lista R_q será a música mais próxima da música verificada q, portanto esta deverá ser a cover. Porém a própria música deve ser retirada desta lista, pois, caso contrário, a primeira música da lista será ela própria.

A equação 3.64 mostra como é feito esse cálculo, onde p é a posição na lista R_q, D a quantidade de músicas da base, I_q representa se houve ou não um acerto (conforme a Equação 3.65) e S_p é a precisão de acertos no grau p da lista.

$$S_p = \frac{\sum_{q=1}^{D} I(q, R_q(p))}{D-1} \tag{3.64}$$

$$I(q,u) = \begin{cases} 1 & \text{se a música } u \text{ for cover de } q \\ 0 & \text{caso contrário} \end{cases} \tag{3.65}$$

Dessa forma, S_p representa a precisão de acerto do método para cada grau de busca, sendo este grau definido pela posição na lista R_q.

Capítulo 4

Trabalhos Relacionados

Os primeiros trabalhos voltados para recuperação de informação de música (MIR) faziam uso de arquivos MIDI. Como esses arquivos armazenam separadamente cada instrumento ou voz, a extração de características e a comparação de gravações se tornam tarefas um pouco mais simples, porém o formato MIDI restringe o problema de identificação de *covers* a subconjunto limitado de usuários, pois são difíceis de serem encontrados.

Assim, explorar a extração de características de sinais polifônicos de diversos gêneros musicais é um caminho natural neste contexto. Sinais polifônicos de áudio são compostos de várias instrumentos que executam melodias e ritmos diferentes simultaneamente. Esses são utilizados para fins de classificação de gêneros, rotulação automática, estimativa de acordes, composição de músicas, identificação de *covers* entre outros [14, 36, 48, 43, 59, 51, 20, 49].

A maioria das pesquisas da área de MIR são focadas no tratamento e na comparação de sinais, sendo a escolha do extrator de características apropriado para o problema alvo e o alto custo computacional envolvido nas etapas de processamento, questões importantes a serem tratadas [54, 52].

Os métodos de extração de características a partir de sinais polifônicos são diversos, e tentam cada vez mais se aproximar à percepção humana. Como mencionado anteriormente, uma das técnicas de processamento de sinal mais utilizada nas extrações de características é o *Short-Time Fourier Transform*, que é obtido através da transformada de Fourier aplicadas a janelas de tempo definidas sobre o sinal. Com ele é possível obter as frequências de acordo com a variação temporal e identificar instrumentos através do timbre [3, 11].

Sabe-se que as 12 notas da escala cromática correspondem aos logaritmos de base 2, portanto, seguem uma escala logarítmica. O fato dessas medidas logarítmicas mais se aproximarem da percepção humana, impulsionou o desenvolvimento de novas técnicas de extração de características. Uma uma das primeiras técnicas a utilizar escalas logarítmicas foi a Transformada Constant-Q, do inglês *Constant-Q Transform* (CQT), que foi desenvolvida a partir dos estudos sobre harmonia musical [7]. Atualmente, esta técnica é utilizada somente na avaliação de técnicas recentes ou como etapa de pré-processamento de técnicas mais sofisticada [2, 50].

Com base em medidas logarítmicas, desenvolveu-se também pesquisas em reconhecimento de voz utilizando a escala mel, que consegue representar o sinal de voz de modo bem próximo a percepção humana. No contexto de MIR, o extrator *Mel-Frequency Cepstrum Coefficients* (MFCC) tem sido bastante utilizado [39], mostrando resultados satisfatórios em aplicações específicas [1, 57, 52, 17, 53, 45, 3, 50, 46, 54, 44, 21].

O MFCC pode ser utilizado tanto para arquivos MIDI quanto para sinais polifônicos [57]. Aucouturier e Pachet [1] verificaram que a técnica pode ser utilizada para encontrar o timbre não apenas da voz, mas de instrumentos, porém a técnica sozinha não pode descrever o sinal com precisão. O MFCC é considerado uma característica de *chroma* do sinal bruto, mas além dela existem outras como o *IF-based chroma* e o *P-chroma* [18].

Este extrator tem várias maneiras de ser implementado, porque contém muitas variáveis que podem ser modificadas de acordo com a necessidade do problema tratado. Existem pesquisas mostrando como os parâmetros podem interferir na qualidade das características extraídas, como por exemplo, os relacionados aos filtros propostos por Ganchev et al. [25].

Ellis [17] propõe a utilização de apenas uma parte do método MFCC, tendo como objetivo adquirir características que possam discernir uma coleção de dados musicais. Porém o autor conclui que o sistema auditivo humano treinado ainda é bem mais eficiente. Em outro trabalho, Ellis [18], a partir de fragmentos do sinal, utiliza o MFCC combinado com outros extratores e consegue identificar fragmentos idênticos quando os sinais são comparados. Além disso, o MFCC também pode ser utilizado para identificar os rastros da batida de uma música [46].

Entretanto, a técnica utilizada para o rastreamento das batidas do sinal é chamada de *Beat Tracking* e tem sido utilizada por muitos pesquisadores como uma característica adicional para a classificação do conjunto de dados de sinais musicais [19, 49, 2, 31, 55]. Esta técnica tem sido aprimorada, o que a tornou cada vez mais complexa, porém, sua precisão ainda deixa a desejar [18].

Um dos pioneiros na utilização da *Beat Tracking* foi Dixon e Cambouropoulos [15] e Goto e Muraoka [30], com o objetivo de identificar a batida em sons acústicos e com poucos instrumentos. Posteriormente, ela foi utilizada para sinais com vários instrumentos, inclusive sem um instrumento de percussão [28]. Davies e Plumbley [13] e Degara e Rua [14] melhoraram o rendimento da técnica, tornando-a mais rápida, e assim possibilitando o uso desta em tempo real.

Outro método de extração explorado é o *Pitch Class Profile* (PCP), que busca reduzir o espectrograma a 12 bins correspondendo aos 12 semitons musicais. Este método foi desenvolvido por Fujishima [23] e posteriormente aperfeiçoado por Gómez [27], dando origem ao *Harmonic Pitch Class Profile* (HPCP), que atualmente é o extrator considerado o estado da arte.

O HPCP emprega conceitos da escala cromática, apresentada no Capítulo 2, e é utilizado para capturar o descritor tonal dos sinais [50]. Baseado no HPCP, Lee [37] criou um método chamado *Enhaced Pitch Class Profile* (EPCP), não superando porém o HPCP no reconhecimento de *covers*. A maior dificuldade de se trabalhar com HPCP se dá por sua complexidade, pois requer longos estudos para sua compreensão, e pela quantidade de parâmetros que devem ser definidos.

A literatura também traz trabalhos como a recriação do sinal original dado o espectrograma extraído. Cychowski [12] desenvolveu o *Inverse Constant-Q* que recria o sinal original tendo as características extraídas do *Constant-Q Transform*. Goto [28] chama a

atenção para a complexidade da técnica *Inverse Beat-Tracking* que seria o processo inverso da *Beat Tracking*, que utiliza o MFCC na sua execução e por isso contém o inverso deste.

Um trabalho muito citado na área é o de Tzanetakis e Cook [58], pois este contém um resumo das técnicas mais utilizadas para extração de características de sinais polifônicos, como também as técnicas mais utilizadas para comparar essas características. Além disso, faz uma análise das técnicas apresentadas e de seus resultados. Nessa mesma linha, Blume et al. [3], faz uma análise de várias técnicas de extração, comparação e classificação.

A classificação dos sinais em uma base de dados é feita a partir da comparação entre características extraídas de sinais. A distância euclidiana é comumente utilizada para esse fim, porém, quando se trata de músicas, a questão temporal seria um problema para essa distância, pois os sinais poderiam ser idênticos contendo apenas um deslocamento temporal. Isso ocorre quando o próprio artista toca a mesma composição em velocidades diferentes, ou faz uma pequena modificação.

Para minimizar este problema, *Dynnamic Time Warp* foi utilizada por muitos pesquisadores para comparar características extraídas [57, 45, 50, 2]. Por ser uma técnica de alto custo computacional outros se dedicam a otimizá-la [35], mesmo assim seu custo computacional ainda é alto.

A técnica *Support Vector Machine* (SVM) é uma das mais empregadas e tem sido utilizada para comparação, onde existe uma base de treinamento a partir da qual a classificação é feita manualmente, para criação de um modelo de aprendizagem. Raviuri e Ellis [49] utilizam SVM para identificar *covers* em grandes bases de dados, comparando-a a outras técnicas de identificação. Eles concluem que SVM apresenta bons resultados, e mesmo quando o treinamento é conduzido utilizando poucas músicas, o resultado ainda se mostra robusto.

A maioria dos trabalhos desenvolvidos na área de MIR são submetidos na competição MIREX *(Music Information Retrieval Evaluation eXchange)* para uma comparação formal. Essa competição é um esforço internacional para avaliação de trabalhos que visam identificar músicas cover, classificação por gênero, reconhecimento de gênero, reconhecimento de emoções, entre ouros [20].

O trabalho de Serrà et. al. [50, 51] obteve melhor acurácia na competição MIREX, sendo o seu método o estado da arte. Serrà faz uso do HPCP e da métrica Q_{max} em seus métodos de identificação de músicas *cover*. O resultado da acurácia obtido através do HPCP mostra que esta técnica permite obter uma boa precisão. Como dito anteriormente, a maior dificuldade de se trabalhar com o HPCP é a complexidade na compreensão da técnica e a quantidade de parâmetros que devem ser modulados, além do alto custo computacional.

A técnica de Ellis [16] também mostrou ter uma boa acurácia na competição MIREX, através do uso do conceito de rastreamento das batidas combinado com o extrator MFCC. Além disso, Ellis [16] obteve um tempo de processamento menor quando comparado com a técnica proposta por Serrà et al. [51], porém o método de Ellis [16] ainda tem um alto custo computacional.

Capítulo 5

Métodos de Identificação de Covers por Ellis e Serrà

Neste capítulo serão descritas as etapas dos métodos de dois trabalhos que resultaram em uma boa acurácia no MIREX. Eles serão utilizados como comparação para a avaliação da proposta deste projeto de mestrado. O primeiro foi proposto por Ellis [16, 20] e o segundo por Serrà et al. [51].

5.1 Método proposto por Ellis

Este método refere-se ao artigo de Ellis [16, 20][1], sendo este uma melhoria do método proposto em [20]. A novidade desses métodos é a combinação de 2 extratores de características: *Beat Tracking* e MFCC. A imagem 5.1 mostra a sequência de etapas utilizada no processo.

O método proposto por Ellis e Poliner [20] tem como etapa inicial a combinação da técnica *beat tracking*, descrita na seção 3.1.5, com uma matriz de *chroma* quantizada em 12 semitons, que é obtida a partir da técnica MFCC, descrita na seção 3.1.2.

Com ambas características extraídas, a matriz de *chroma* é segmentada e ajustada de acordo com a possível sequência de batidas encontradas pelo *beat tracking* [19], gerando uma nova matriz de *chroma*. Esta combinação de características é denominada pelo autor como *beat-synchronous*.

Para comparar as características extraídas dos sinais, o autor aplica a técnica de correlação cruzada (descrita na seção 3.2.1) entre as linhas da matriz de *chroma* obtida com o *beat-synchronous*. É nesse estágio que a transposição é incluída fazendo 12 comparações. As linhas da matriz de características do sinal buscado são deslocadas de modo circular; e a cada deslocamento as duas matrizes de *chroma* são comparadas. A transposição da

[1]Os códigos utilizados para reproduzir esse método está disponível em http://www.ee.columbia.edu/ln/rosa/matlab/

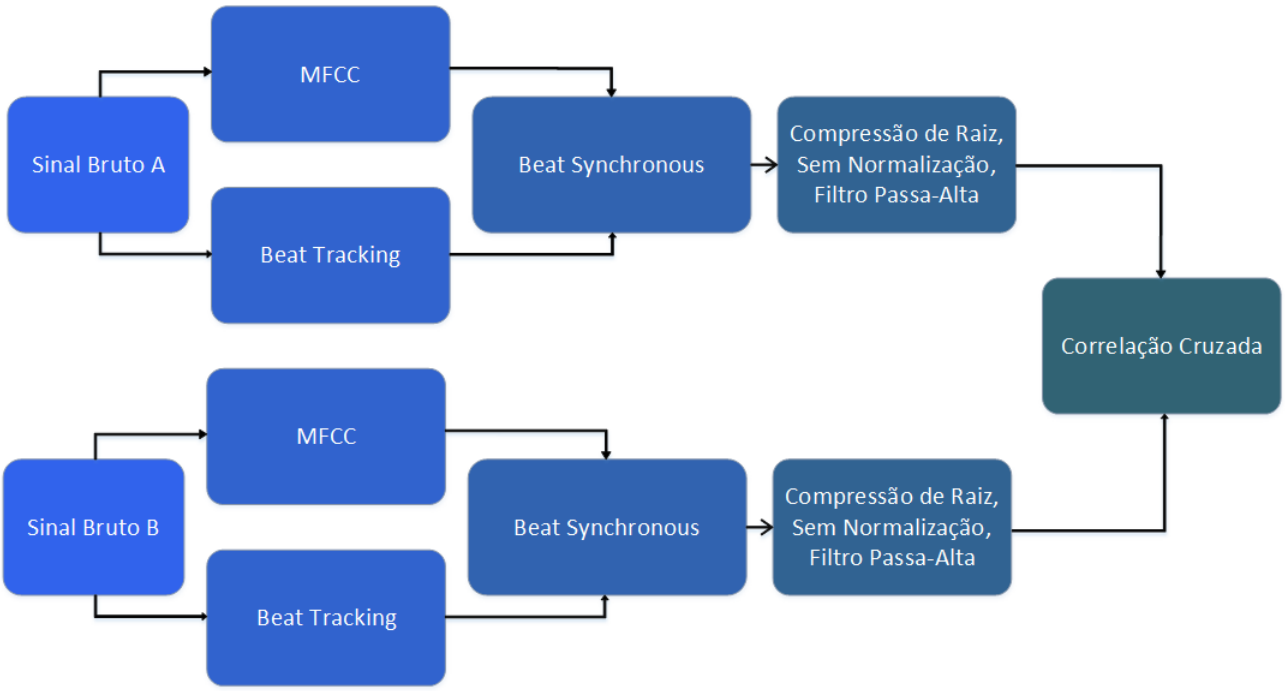


Figura 5.1: Diagrama do método de identificação de músicas *cover* proposto por Ellis e Poliner [20], Ellis [16]

matriz de *chroma* referente ao sinal buscado é a que resultar o maior valor da correlação dos picos, e este valor é utilizado para gerar a matriz de similaridade entre os sinais.

Essa descrição é do método proposto em Ellis e Poliner [20], o qual obteve melhorias descritas no artigo [16], como a incorporação de uma compressão com raiz quadrada, a retirada da normalização de correlação cruzada e a aplicação de um filtro passa alta antes da aplicação da correlação cruzada.

Ellis [16] utiliza 93 ms no STFT, que é umas das etapas da extração de características de *chroma*, e que só considera as frequências de 400 Hz para cima. Na matriz obtida com a *beat-synchronous* é aplicado uma compressão na magnitude com uma raiz quadrada. A base *Covers80* foi utilizada para avaliar o método aperfeiçoado e a acurácia aumentou visivelmente nos resultados [16].

5.2 Método proposto por Serrà et al.

Este método é referente ao artigo Serrà et al. [51], sendo este uma melhoria do método proposto em Serrà et al. [50]. A novidade desses métodos é a utilização de medidas provenientes da Analise de Quantificação Recorrente (RQA) e da técnica Plotagem de Recorrência Cruzada (CRP), que é uma generalização da Plotagem de Recorrência (RP). A imagem 5.2 mostra a sequência de etapas utilizada no processo.

Esse método tem como etapa inicial a extração de características do sinal bruto através da técnica HPCP, descrita na seção 3.1.4. O STFT é aplicado com uma janela de 93 ms, com 50% de sobreposição, e uma normalização da amplitude utilizando o filtro de branqueamento. O intervalo de frequências consideradas é entre $[50, 5000]$ Hz e a frequência de referência estimada é em torno de 440 Hz, e a resolução adotada é de 12 semitons.

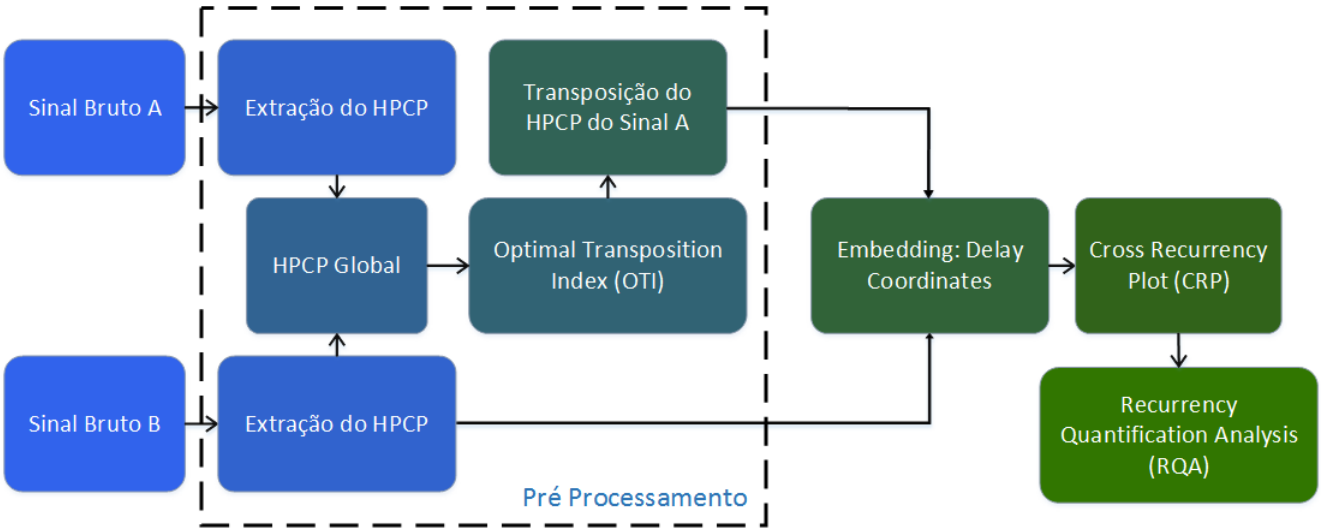


Figura 5.2: Diagrama do método de identificação de músicas *cover* proposto por Serrà et al. [51].

Logo após a obtenção dos HPCPs, um HPCP global é calculado para encontrar a chave de tranposição de duas músicas. O HPCP global é a média das linhas do HPCP divido pelo valor máximo, de modo a obter um vetor de tamanho 12. A chave de transposição é encontrada utilizando a técnica *Optimal Transposition Index* (OTI), que representa o número de vezes que o HPCP deve sofrer um deslocamento circular para a direita. Desse modo, apenas um dos HPCP sofre a transposição.

A OTI é definida na Equação 5.1, onde $\vec{h}_a$ e $\vec{h}_b$ são os HPCPs globais da música A e B, N_H é a quantidade de dimensões, no caso 12, e $circShift_R(\vec{h}, i)$ é uma função que faz um deslocamento circular de i posições para a direita do vetor $\vec{h}$. A transposição do HPCP da música A, $HPCP_A^{Tr}$, é o deslocamento circular para a direita, em que a quantidade de posições deslocadas é igual a chave obtida a partir da OTI (Equação 5.2)[50].

$$OTI(\vec{h}_A, \vec{h}_B) = \underset{0 \leq i \leq N_H - 1}{\text{argmax}}\{\vec{h}_A \cdot circShift_R(\vec{h}_B, i)\} \tag{5.1}$$

$$HPCP_A^{Tr} = circShift_R(HPCP_A, OTI) \tag{5.2}$$

A próxima etapa é a aplicação do Teorema de Imersão[2], proposto por Takens [56], que transforma o espaço de características em um espaço de coordenadas de atraso, ou *delay coordinates*. Este Teorema é utilizado para extrair informações do HPCP, isolando a parte essencial da música, importante para a percepção das melodias. Essa técnica permite representar séries temporais em um espaço euclidiano de m dimensões, que define o número de eixos do espaço de coordenadas de atraso. Quanto maior a dimensão m, maior seu desdobramento facilitando a sua compreensão, pois, a cada desdobramento, um novo comportamento pode ser analisado [42].

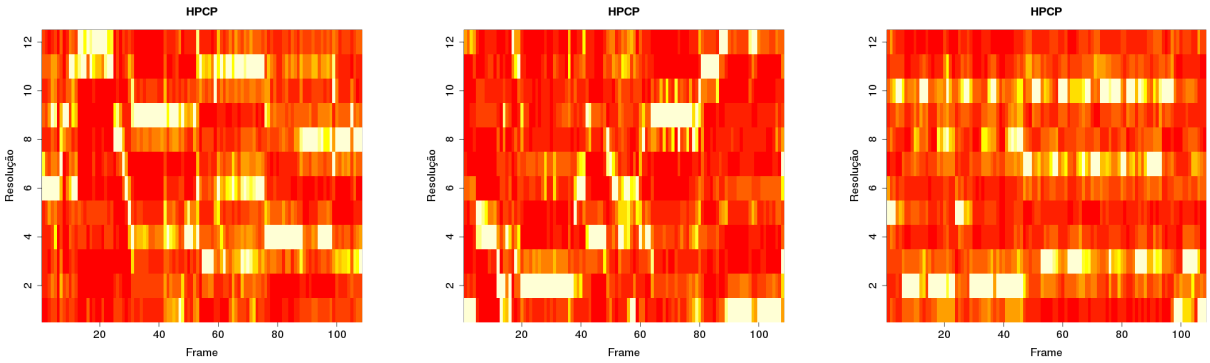
Nesse contexto, o autor fez algumas adaptações do Teorema de Imersão para se trabalhar com o HPCP, de modo que a Equação 5.3 define a transformação do espaço adquirido com o HPCP em um espaço de coordenadas de atraso. Utilizou-se $m = 10$ e $\tau = 1$. Considerando a evolução temporal de cada dimensão da resolução H, é construído uma série

[2]Maiores detalhes sobre esse Teorema de Imersão são encontrados em [56].

temporal no espaço embutido (*State Space Embedding*) de vetores $x = x_i$ para $i = 1, 2, ..., N_x$, com $N_x = N_X^* - (m-1)\tau$, onde N_X^* é a quantidade de quadros no tempo [51].

$$x_i = (x_{1,i}, x_{1,i+\tau}, ..., x_{1,i(m-1)\tau}, x_{2,i}, x_{2,i+\tau}, ..., x_{2,i(m-1)\tau}, ..., x_{H,i}, x_{H,i+\tau}, ..., x_{H,i(m-1)\tau}) \tag{5.3}$$

Com as informações obtidas pelo Teorema de Imersão, calcula-se o CRP, da forma descrita na seção 3.2.2. Nesse caso, x_i e y_j são as representações do espaço das janelas i e j para as músicas X e Y, respectivamente [51].



(a) HPCP aplicado na música *Imagine* na versão por *John Lennon*

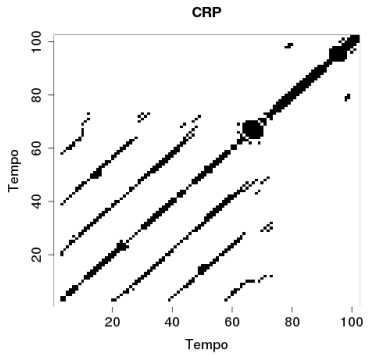
(b) HPCP aplicado na música *Imagine* na versão por *Diana Ross*

(c) HPCP aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*

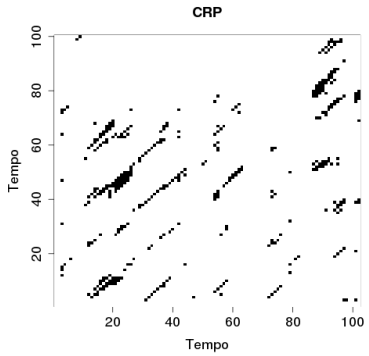
Figura 5.3: Atributos resultantes do método *Harmonic Pitch Class Profile* (HPCP) aplicada em 20 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz, com resolução 12, com fragmentos de 464 ms e sem sobreposição.

O estudo de Serrà et al. [51] mostra que a definição de cover e não cover pode ser definida pelo tamanho das diagonais produzidas no CRP, e que a melhor métrica de RQA para avaliar essa definição é o Q_{max} (descrita na seção 3.2.2), pois o L_{max} não identifica mudanças no tempo apesar de identificar passagens similares, e o S_{max} não considera as semelhanças caso exista algumas curtas interrupções na música. Essa interrupções podem ser causadas pela falta de alguns acordes ou parte da melodia. A Imagem 5.4 ilustra um exemplo de matrizes geradas com a CRP e seus respectivos Q_{max}.

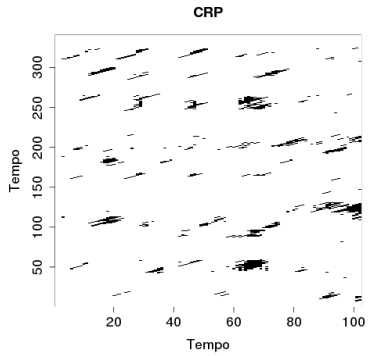
A matriz de similaridade é criada a partir do valor Q_{max} de cada combinação de músicas. O resultados apresentados no artigo [51] foram gerados utilizando a base do MIREX 2007, que contém 330 músicas com 30 composições com 11 covers cada.



(a) CRP da música *Imagine* na versão por *John Lennon* com ela mesma. Com o valor de $Q_{max} = 100$.



(b) Matriz de Recorrência da música *Imagine* na versão por *John Lennon* com a versão por *Diana Ross*. Com o valor de $Q_{max} = 20$.



(c) CRP da música *Imagine* na versão por *John Lennon* com música *Book I Prelude* na versão por *Christiane Jaccotte*. Com o valor de $Q_{max} = 11$.

Figura 5.4: Matrizes de Recorrência resultantes do método *Cross Recurrence Plots* (CRP) aplicada em 38 segundos de 3 sinais polifônicos distintos, sendo **b** *cover* de **a**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 8 kHz, com dimensão embutida $m = 10$ e atraso temporal de $d = 1$, para 464 ms com 75% de sobreposição, resolução 12 e fração de vizinhos em 0.1.

Capítulo 6

Metodologia

Na literatura existem muitos trabalhos voltados para extração de características e suas comparações, porém são poucos os trabalhos que apresentam as técnicas de modo claro para o leitor, como também suas mudanças para obter melhores resultados. Assim, um dos primeiros objetivos deste projeto de mestrado foi fazer um levantamento bibliográfico dos métodos de extração de características de sinais musicais polifônicos mais utilizados (Seção 3.1), como também métodos de comparação dessas características (Seção 3.2). Posteriormente, foram estudados os métodos de identificação de *covers*, que obtiveram melhores acurácias na competição MIREX. Esses métodos serviram de base para comparação e avaliação do método proposto nesta dissertação.

A presente pesquisa consiste em identificar músicas covers em uma base de dados, comparando-as de modo a agrupar os sinais de acordo com suas semelhanças. Dentro desse contexto, este projeto visa desenvolver um novo método para a comparação de similaridade entre sinais de áudio, através da identificação de músicas *cover*, utilizando as técnicas de aprendizado de máquina descritas na Seção 3.3.

Apesar da utilização destas técnicas já serem empregadas na literatura, a ideia de trabalhar com as matrizes de chroma como imagens e criar um dicionário musical, ainda não foi explorada. Com esse novo método verificou-se a possibilidade de identificar músicas *cover* em uma base de dados, com tempos de processamentos inferiores em comparação as abordagens descritas no Capítulo 5.

Seguindo nesse caminho, aplicamos o modelo *Bag of Features* encontrado na literatura para reconhecimento de imagens, possibilitando a criação de assinaturas musicais a partir de técnicas de agrupamento [34]. A Imagem 6.1 mostra o processo da metodologia utilizada.

6.1 Representação dos dados

Para realizar a proposta descrita foi necessário implementar e validar os extratores de características desenvolvidos de acordo com a literatura. Cada extrator resulta em

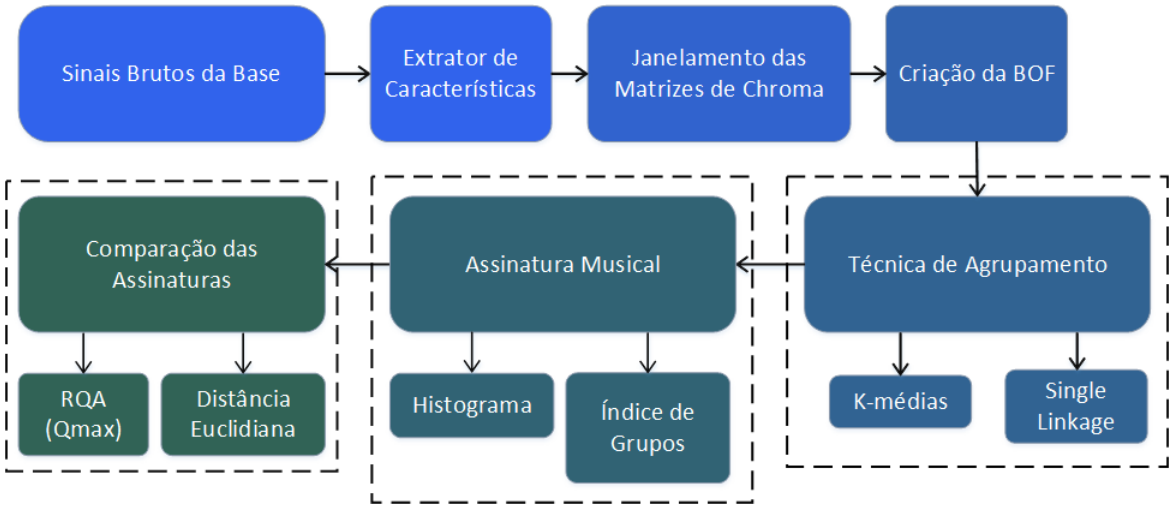


Figura 6.1: Diagrama do Processo da metodologia criada.

uma matriz de *chroma*, que determina a intensidade de cada frequência para cada período de tempo. Os extratores utilizados neste trabalho foram: STFT, MFCC, HPCP, *Chroma* Simples, sendo que o STFT, o MFCC e o *Chroma* Simples foram implementados pelo grupo de pesquisa, e a implementação do HPCP utilizada está disponível no pacote *Essentia*.

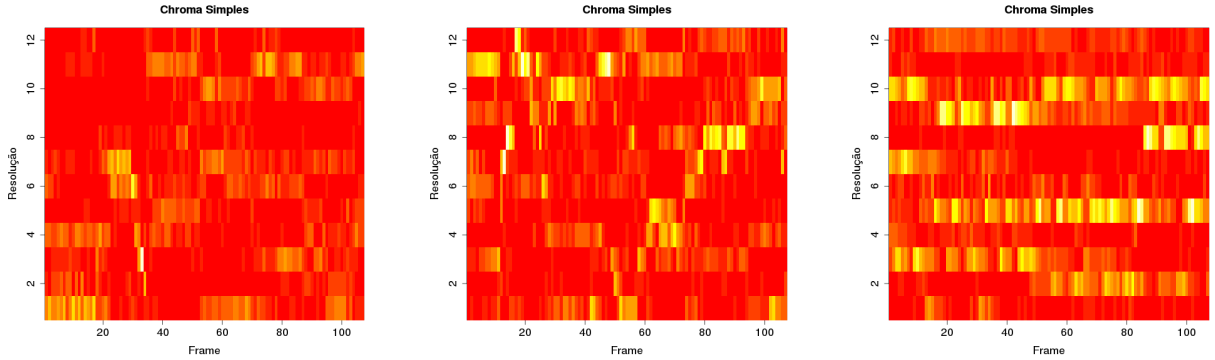
O extrator de *Chroma* Simples é uma simplificação do extrator HPCP. Este inicia-se com o STFT, depois faz-se a detecção dos picos em cada fragmento do sinal utilizado no STFT. Por fim, este é quantizado em 12 semitons. O diagrama da Figura 6.2 ilustra esse processo.

Figura 6.2: Diagrama do Processo do extrator de *Chroma* Simples.

As Figuras 6.3 e 6.4 apresentam os resultados do *Chroma* Simples para 3 músicas distintas, sendo duas *covers* e uma não-*cover*, todas com duração de 10 segundos, e com sinal dividido em janelas de 16 ms, com uma sobreposição de 0,75%, e a quantidade de amostras por segundo (*sample rate*) de 16 KHz. O eixo x é o eixo do tempo de duração do sinal em fragmentos e no eixo y estão as frequências de acordo com a resolução. Quanto mais vermelho, maior é a intensidade daquela frequência no tempo.

As características resultantes de um sinal polifônico podem ser analisados como imagens, permitindo a aplicação de técnicas de aprendizado de máquina para a comparação entre duas imagens obtidas. Para a representação das imagens é utilizado o modelo de *Bag-of-Features* (BOF), que é um conjunto de vetores de características de todas as imagens, e que pode ser feito de diversas maneiras [8]. Esse modelo foi adaptado para trabalhar com imagens resultantes das características extraídas dos sinais.

Para criar o BOF, é necessário percorrer a matriz de chroma de cada música da base de dados através de uma janela de tamanho fixo, denominada máscara. Assim, para cada música tem-se um conjunto de sub-matrizes adquiridas pela máscara, com ou sem sobreposição. Essas matrizes são convertidas em vetores, onde une-se as colunas das

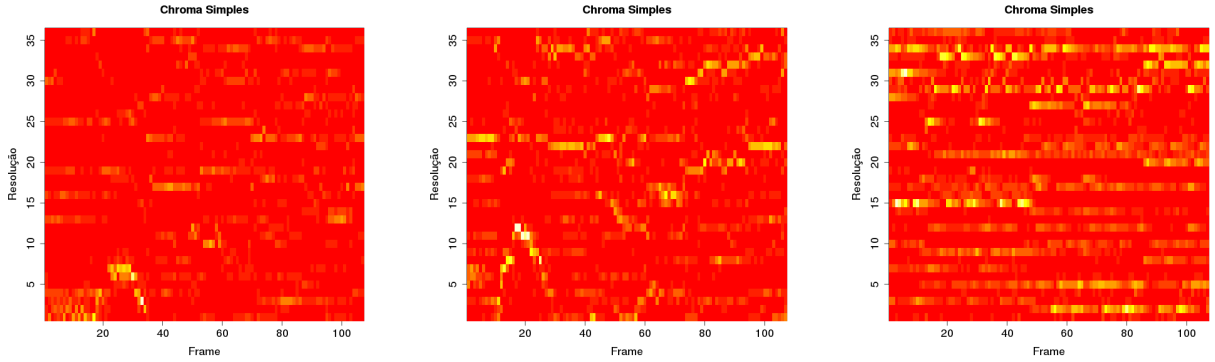


(a) *Chroma* Simples aplicado na música *Imagine* na versão por *John Lennon*

(b) *Chroma* Simples aplicado na música *Imagine* na versão por *Diana Ross*

(c) *Chroma* Simples aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*

Figura 6.3: Atributos resultantes do método *Chroma* Simples aplicada em 10 segundos de 3 sinais polifônicos distintos, sendo **a** *cover* de **b**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz e com resolução 12.



(a) *Chroma* Simples aplicado na música *Imagine* na versão por *John Lennon*

(b) *Chroma* Simples aplicado na música *Imagine* na versão por *Diana Ross*

(c) *Chroma* Simples aplicado na música *Book I Prelude* na versão por *Christiane Jaccotte*

Figura 6.4: Atributos resultantes do método *Chroma* Simples aplicada em 10 segundos de 3 sinais polifônicos distintos, sendo **a** *cover* de **b**, e **c** não-*cover* de **a** e **b**. Esses sinais foram amostrados a 16 kHz e com resolução 36.

matrizes para criar um vetor de característica. O Algoritmo 2 mostra como efetuar esse processo.

Algoritmo 2: Algoritmo da Representação dos Dados

Entrada: Matriz de *chroma M*, número de linhas da matriz de janelamento *mL*, número de colunas da matriz de janelamento *mC*, passo na linha *pL*, passo na coluna *pC*
Saída: *Bag of Features*
inicio
  $i = 0$;
  enquanto $i < M - mL$ **faça**
    $parte = M[i : (i + mL), :]$;
    $j = 0$;
    enquanto $j < M - mC$ **faça**
      $janela = parte[:, j : (j + mascaraColuna)]$;
      Faz a conversão da *janela* para um vetor de características;
      Acrescenta o vetor na BOF;
      $j = j + pC$;
    fim
    $i = i + pL$
  fim
  Retorna BOF
fin

A música é caracterizada pelo respectivo conjunto de vetores de características em dimensão R^n adquirido com as máscaras, onde n é o tamanho da máscara. A BOF é a união de todos os conjunto de vetores de características de todas as músicas da base. A imagem 6.5 ilustra uma sub-matriz obtida a partir da máscara 3×3 retirada da matriz de *chroma*. A máscara 3×3 percorre toda a matriz de *chroma*, e a cada iteração, a sub-matriz obtida com a máscara é convertida em um vetor de características posteriormente acrescentado a BOF.

6.2 Técnicas de agrupamento

Após o processamento dos sinais e a criação da *Bag-of-Features* (BOF), agrupa-se os vetores de características de acordo com suas similaridades. As técnicas de agrupamento descritas na Seção 3.3 são utilizadas para agrupar vetores de características similares e separar os vetores de características distintos existentes na BOF. Assim, todos os vetores extraídos de todas as músicas da base de dados serão agrupados.

A partir da formação dos grupos pode ser gerado 2 tipos de assinatura musical para cada música. A primeira delas é a assinatura pelos índices dos grupos, criada de acordo com os grupos em que os vetores de características dessa música foram incluídos. Supondo que uma música têm um conjunto de 10 vetores de características, e que esses vetores estejam inclusos nos grupos 1, 4, 2, 5, 1, 4, 4, 5, 1 e 2. A assinatura musical do sinal será o índice dos grupos na ordem dos vetores de características resultando no vetor $(1, 4, 2, 5, 1, 4, 4, 5, 1, 2)$.

O segundo tipo de assinatura musical é um histograma. O histograma é calculado a partir da quantidade de vetores que caíram em cada grupo, gerando o vetor $(3, 2, 0, 3, 2, 0)$, para o exemplo descrito anteriormente. O tamanho da assinatura musical pelo histograma, será a quantidade de grupos, e o tamanho da assinatura musical de todas as

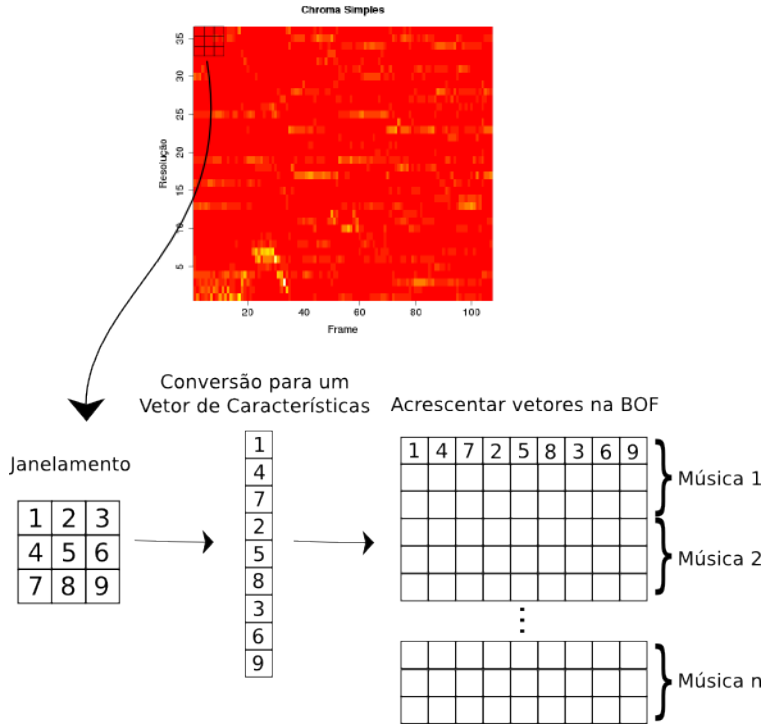


Figura 6.5: Ilustração do processo de janelamento utilizando uma matriz 3×3 e criação da *Bag of Features*. Na imagem é retirada da matriz de *chroma* um submatriz 3×3, depois essa é convertida em um vetor de características e incluída na *Bag of Features* que deve conter todos os vetores de características de todas as músicas.

músicas será fixo. Esse processo de criação do histograma também é chamado de quantização [34].

A Figura 6.6 exemplifica um conjunto de vetores de características de uma música, assim como os grupos gerados pelo agrupador seguindo das duas possíveis assinaturas musicais. Para facilitar a ilustração, os vetores de características contêm apenas 2 atributos (diferente da Figura 6.5 em que o vetor de características contêm 9 atributos). Dessa forma, a assinatura é criada a partir do agrupamento feito com a BOF.

6.3 Comparação das músicas

A comparação das assinaturas musicais é feita utilizando uma métrica que calcula a distância entre elas. Neste projeto de mestrado foram utilizados a medida de Análise de Quantificação Recorrente (RQA), descrita na Seção 3.2, e a distância euclidiana [26, 22].

As medidas são aplicadas entre as assinaturas musicais das músicas da base de dados. A distância Euclidiana retorna a distância entre pares de músicas, assim quanto menor a distância mais semelhantes serão os sinais dessa base. Isso permite a criação de uma matriz de dissimilaridade, onde o índice da matriz representa as músicas da base e os valores são as distâncias entre as músicas. Uma ilustração da matriz de dissimilaridade pode ser vista na Figura 6.1.

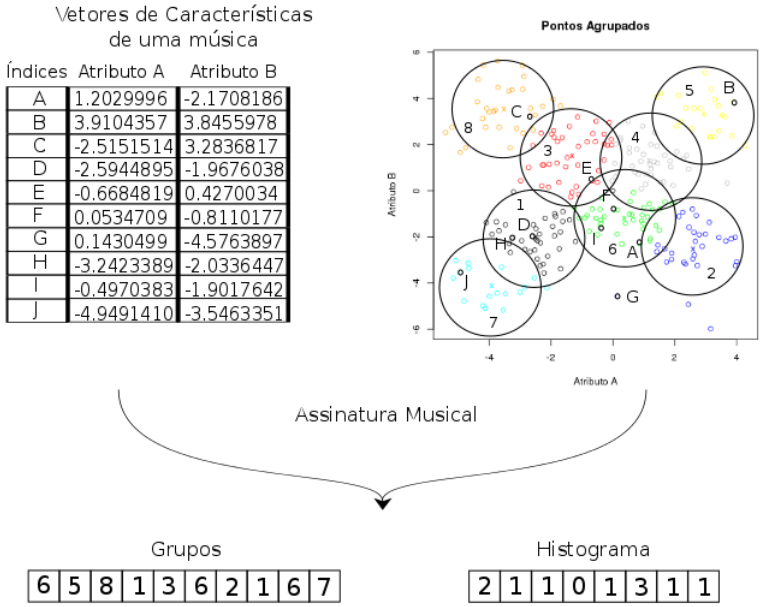


Figura 6.6: Ilustração do processo de criação de ambas as assinaturas musicais, onde existe um exemplo de atributos de uma música obtido com o janelamento (1×2), e a BOF agrupada em 8 grupos pelo k-médias. Resultando em 2 assinaturas musicais, sendo a primeira o índice dos grupos e a segunda pelo histograma.

$$\left[M_{dist}(D) = \begin{pmatrix} 0 & d_{12} & \cdots & d_{1n} \\ d_{21} & 0 & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & 0 \end{pmatrix} \right] \tag{6.1}$$

No caso da RQA, o valor retornado na comparação entre as assinaturas musicais gera uma matriz de similaridade, pois quanto maior o valor retornado, maior é a semelhança entre as músicas. Ambas são convertidas para matrizes de dissimilaridade para facilitar a avaliação de precisão. Essa avaliação tem como objetivo verificar a precisão do método na identificação das *covers* em uma base de dados.

Capítulo 7

Resultados

Com o intuito de verificar o desempenho do método proposto, comparamos os resultados obtidos com os resultados dos métodos descritos no Capítulo 5. Para poder efetuar a comparação utilizamos a mesma base de dados e as mesmas métricas.

Os métodos foram executados em cima de 3 bases dados: a *Covers80*, criada em [16] que contém 80 composições com 1 cover cada, totalizando 160 músicas[1]; e uma coleção pessoal de músicas com 28 composições com 9 *covers* cada e mais 1 composição com 8 *covers*, totalizando 289 músicas. Uma sub-base da base da coleção pessoal também foi utilizada, sendo composta por 5 composições com 10 *covers* de cada, totalizando 50 músicas. A avaliação da precisão do método é efetuada utilizando as métricas descritas na Seção 3.4.

Todos os métodos foram implementados na linguagem R, com exceção do extrator *Harmonic Pitch Class Profile* (HPCP), que está implementado em *python* no pacote *Essentia*[2]. Além disso, o código utilizado para executar o método proposto por Ellis [20] está implementado em Matlab, conforme disponibilização dos autores[3].

7.1 Método por Ellis

Este resultado é referente ao método descrito na Seção 5.1, proposto por Ellis e Poliner [20], Ellis [16]. Todos os resultados foram executados em toda a extensão do sinal.

Para a sub-base de coleção pessoal obteve-se o Gráfico 7.1, onde o eixo x representa a ordem de retorno de músicas corretamente encontradas dado uma música de busca e o eixo y a contagem de acertos de músicas retornadas que são *covers*. Assim a primeira barra representa a contagem de acertos quando recuperamos a música *cover* como sendo a primeira música retornada para toda a base, a segunda, representa a contagem de acertos

[1] http://labrosa.ee.columbia.edu/projects/coversongs/covers80/

[2] Os códigos utilizados para reproduzir o extrator HPCP está disponível em http://essentia.upf.edu/

[3] Os códigos utilizados para reproduzir esse método está disponível em http://www.ee.columbia.edu/ln/rosa/matlab/

quando recuperamos a música *cover* como sendo a segunda música retornada para toda a base, e assim por diante.

Das 50 músicas buscadas 39 *covers* são recuperadas como a primeira da lista. Quando considerado a segunda da lista, apenas 3 das músicas são *covers* e para a terceira da lista 3 músicas são *cover*. Esta execução demorou 396.3259 segundos.

Para a base *Covers80*, mostrado no Gráfico 7.2, nota-se que das 160 músicas buscadas 61 das músicas *covers* são recuperadas como a primeira da lista. Para a segunda da lista, apenas 3 das músicas são *covers* e para a terceira da lista nenhuma é *cover*. Esta execução demorou 2527.459 segundos.

Para a base de coleção pessoal, mostrado no Gráfico 7.3, das 289 músicas buscadas 149 das músicas *covers* são recuperadas como a primeira da lista. Para a segunda da lista, 86. Essa execução demorou 5240.512 segundos.

A Tabela 7.1 resume os resultados para a Base *Covers80*, para a sub-base e Base de coleção pessoal, obtidos através do método proposto por Ellis e Poliner [20], Ellis [16]. A Tabela 7.1 indica a precisão da métrica Separabilidade dado as três primeiras músicas retornadas e também o resultado da métrica MAP.

Bases	SEP [1]	SEP [2]	SEP [3]	MAP
Sub-base da Coleção Pessoal	0.7959	0.0612	0.0612	0.5531
Covers80	0.3836	0.0188	0	0.5221
Coleção Pessoal	0.5173	0.2986	0	0.4990

Tabela 7.1: Precisão do método proposto por Ellis e Poliner [20] para ambas as bases, utilizando ambas as métricas.

7.2 Método por Serrà et. al.

Os resultados a seguir referem-se ao método descrito na Seção 5.2, proposto por Serrà et al. [51]. O código foi implementado e validado de acordo com as descrições dos artigos [51, 27, 50]. Para os resultados utilizou-se uma parte da base de coleções pessoais, com 5 composições com 10 *covers* de cada, totalizando 50 músicas, e apenas 60 segundos do sinal. O alto tempo de processamento foi um dos motivos de utilizar apenas a menor base de dados e como também o corte do sinal.

O código foi executado utilizando a implementação do HPCP disponível no pacote *Essentia*, com janelas de 93 ms e com 50% de sobreposição. Neste mesmo experimento foi utilizado $m = 10$ para a dimensão embutida, $d = 1$ como atraso temporal, e 0.1 fração de vizinhos.

Para a sub-base de coleção pessoal, mostrado no gráfico 7.4, das 50 músicas buscadas 24 das músicas retornadas na primeira posição da lista são *covers*, para a segunda da lista, 7 e para a terceira da lista 5. Para essa execução, o tempo de processamento foi de 1339042 segundos.

A Tabela 7.2 resume os resultados para a sub-base de coleção pessoal, obtidos através do método proposto por Serrà et al. [51]. A Tabela 7.2 indica a precisão da métrica Separabilidade dado as três primeiras músicas retornadas e também o resultado da métrica MAP.

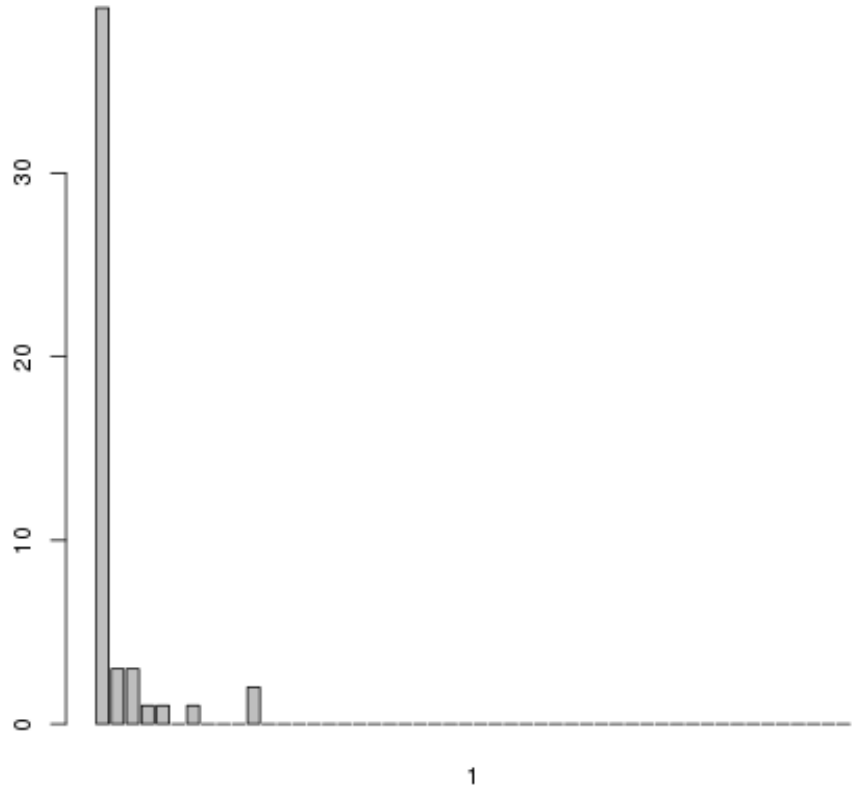


Figura 7.1: Gráfico da métrica de Separabilidade para a sub-base de coleções pessoais, com a utilização do método proposto por Serrà et al. [51].

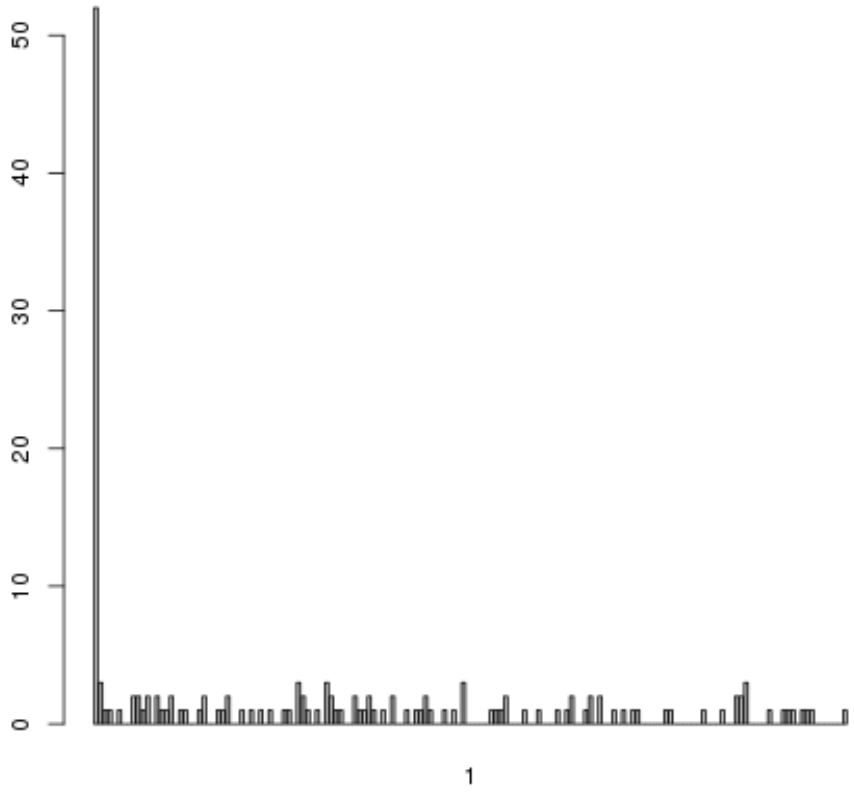


Figura 7.2: Gráfico da métrica de Separabilidade para a base *Covers80*, com a utilização do método proposto por Ellis e Poliner [20].

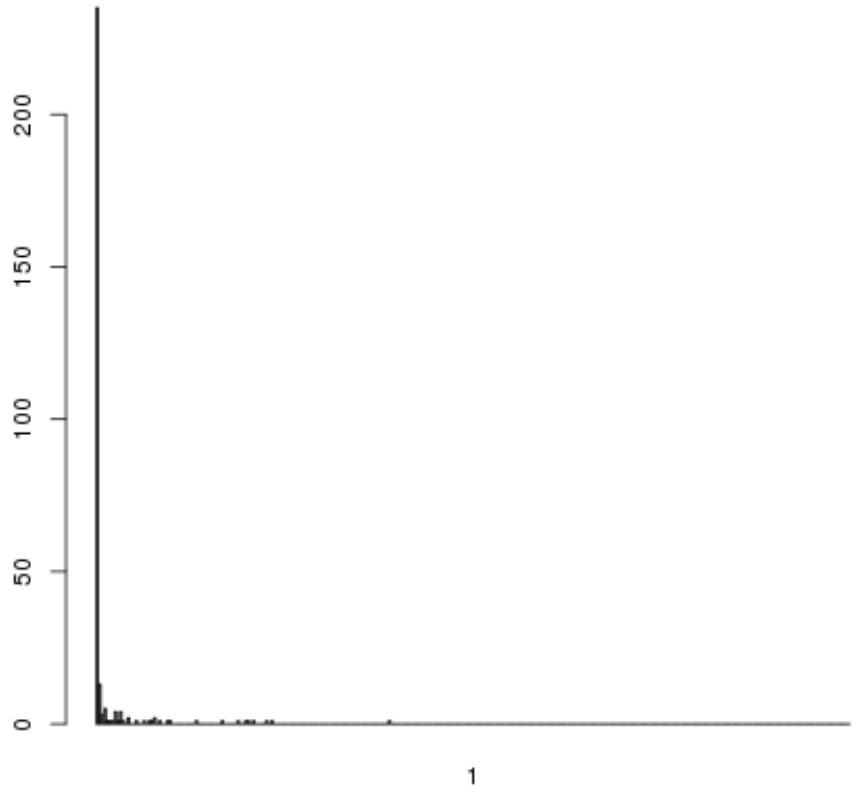


Figura 7.3: Gráfico da métrica de Separabilidade para a base de coleções pessoais, com a utilização do método proposto por Ellis e Poliner [20].

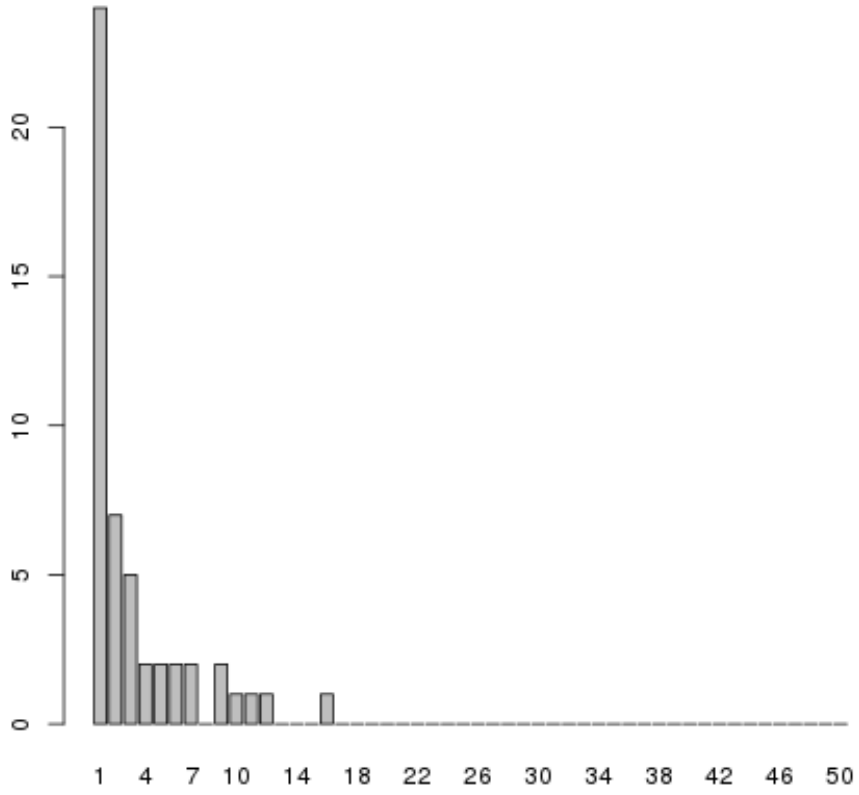


Figura 7.4: Gráfico da métrica de Separabilidade para a sub-base de coleções pessoais, com a utilização do método proposto por Serrà et al. [51].

Bases	SEP [1]	SEP [2]	SEP [3]	MAP
Sub-base da Coleção Pessoal	0.4897	0.1428	0.1020	0.3508

Tabela 7.2: Precisão do método proposto por Serrà et al. [51] para a sub-base de coleção pessoal, utilizando ambas as métricas.

O método proposto por Serrà et al. [51] não foi executado para as demais bases por causa do seu alto tempo de processamento. Além disso, os resultados obtidos não se aproximam dos resultados descritos na literatura.

7.3 Bateria de Testes do Método Proposto

Inicialmente, foi feito uma bateria de teste para descobrir qual o tamanho mais indicado para a máscara na etapa de representação dos dados (descrita na Seção 6.1) e qual a quantidade de grupos necessária na etapa de agrupamento (descrita na Seção 6.2) para obter uma boa precisão. Quais as técnicas de agrupamento que tiveram melhor desempenho para a tarefa de identificação de músicas *cover*. Esses testes foram efetuados em 60 segundos dos sinal e foi utilizado a sub-base de coleção pessoal que contém 50 músicas.

Na execução dos teste, o número de grupos foram variados entre 500 a 5000 com passo de 300, e o tamanho das janelas foram variados entre 3×12 até 12×21 com um passo de 3 nas linhas e nas colunas individualmente, em que a linha representa o tempo e a coluna as frequências do extrator (linha×coluna). Em todos os casos foram utilizado como extrator de caraterísticas o *Chroma* Simples com resolução 12, tamanho da janela de 93 ms e sem sobreposição. Este extrator foi escolhido para essa bateria de teste por ser o extrator com menos custo computacional.

A Tabela 7.3 mostra os 5 melhores resultado e a Tabela 7.4 os 5 piores resultados do MAP, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelos índices do grupo e a medida Q_{max} da Análise de Quantificação Recorrente (RQA).

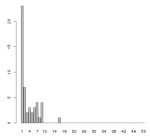
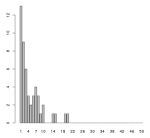
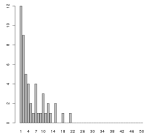
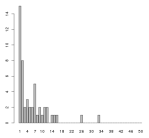
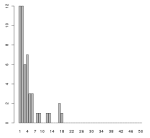
	1º	2º	3º	4º	5º
№Grupos	2600	2600	800	1700	2300
Janela	18×3	12×9	15×9	18×6	18×3
MAP	0.2968	0.2949	0.2927	0.2837	0.2835
SEP					

Tabela 7.3: Os 5 melhores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelos índices do grupo e a comparação através da medida Q_{max}. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

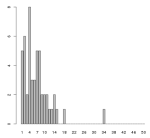
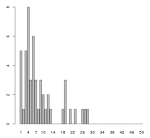
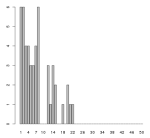
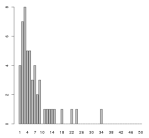
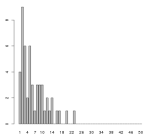
	1º	2º	3º	4º	5º
№Grupos	1700	3200	5000	2600	1100
Janela	21×6	18×3	12×6	18×6	15×9
MAP	0.2246	0.2209	0.2206	0.2181	0.2166
SEP					

Tabela 7.4: Os 5 piores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelos índices do grupo e a comparação através da medida Q_{max}. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

A Tabela 7.5 mostra os 5 melhores resultados e a Tabela 7.6 os 5 piores resultados do MAP, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelos índices do grupo e a comparação através da distância Euclidiana.

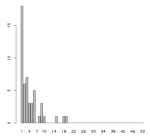
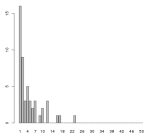
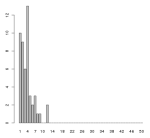
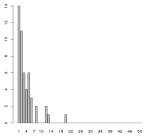
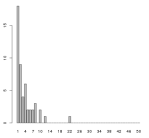
	1º	2º	3º	4º	5º
№Grupos	1400	5000	2900	2000	3800
Janela	15×12	12×9	18×6	15×6	12×6
MAP	0.2849	0.2779	0.2763	0.2336	0.2743
SEP					

Tabela 7.5: Os 5 melhores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelos índices do grupo e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

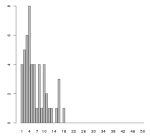
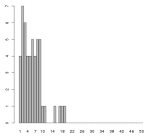
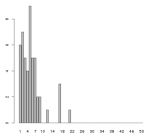
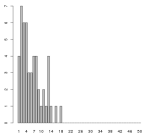
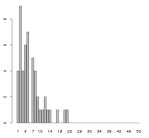
	1º	2º	3º	4º	5º
№Grupos	2900	1400	1100	2900	2300
Janela	12×6	18×9	15×6	18×9	15×3
MAP	0.2183	0.2163	0.2156	0.2142	0.2072
SEP					

Tabela 7.6: Os 5 piores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelos índices do grupo e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

A Tabela 7.7 mostra os 5 melhores resultados e a Tabela 7.8 os 5 piores resultados do MAP, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelo histograma e a comparação através da distância Euclidiana.

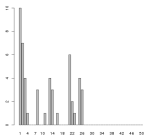
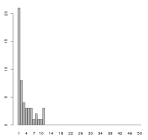
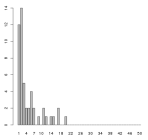
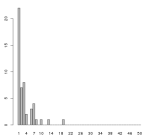
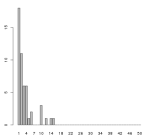
	1º	2º	3º	4º	5º
№Grupos	1700	500	2300	500	800
Janela	18×12	15×9	12×3	18×9	12×9
MAP	0.2894	0.2863	0.2847	0.2818	0.2814
SEP					

Tabela 7.7: Os 5 melhores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelo histograma e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

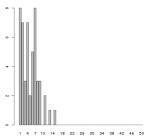
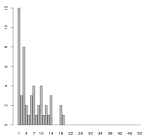
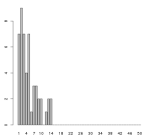
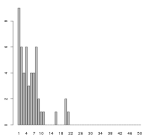
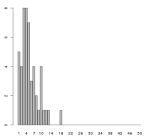
	1º	2º	3º	4º	5º
№Grupos	2900	1400	1100	2600	4400
Janela	21×3	12×3	18×6	21×9	12×6
MAP	0.2379	0.2374	0.2365	0.2353	0.2345
SEP					

Tabela 7.8: Os 5 piores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento K-médias com 25 iterações, a assinatura pelo histograma e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

A Tabela 7.9 mostra os 5 melhores resultados e a Tabela 7.9 os 5 piores resultados do MAP, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelos índices do grupo e a comparação através da medida Q_{max} da Análise de Quantificação Recorrente (RQA).

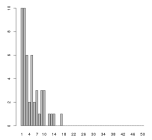
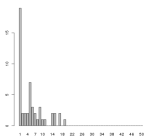
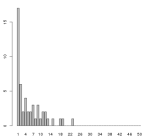
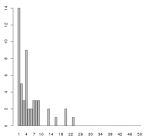
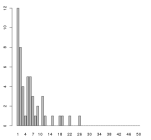
	1º	2º	3º	4º	5º
№Grupos	4100	4400	2600	3200	800
Janela	21×3	12×6	21×6	18×6	12×3
MAP	0.2749	0.2733	0.2719	0.2709	0.2691
SEP					

Tabela 7.9: Os 5 melhores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelos índices do grupo e a comparação através da medida Q_{max}. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

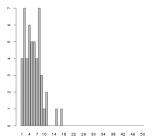
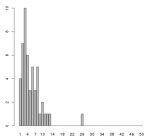
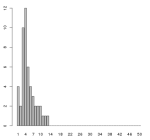
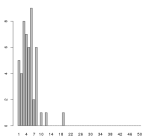
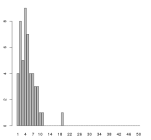
	1º	2º	3º	4º	5º
№Grupos	2300	1700	1700	1100	2000
Janela	12×6	12×9	15×3	15×6	15×3
MAP	0.2272	0.2261	0.2247	0.2241	0.2226
SEP					

Tabela 7.10: Os 5 piores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelos índices do grupo e a comparação através da medida Q_{max}. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

A Tabela 7.11 mostra os 5 melhores resultados e a Tabela 7.12 os 5 piores resultados do MAP, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelos índices do grupo e a comparação através da distância Euclidiana.

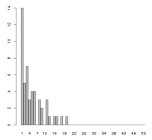
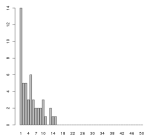
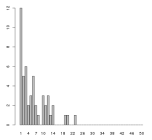
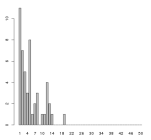
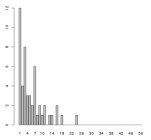
	1º	2º	3º	4º	5º
№Grupos	2300	2600	4700	2000	4100
Janela	12×12	12×12	21×3	12×12	12×6
MAP	0.2820	0.2675	0.2630	0.2622	0.2622
SEP					

Tabela 7.11: Os 5 melhores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelos índices do grupo e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

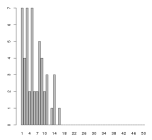
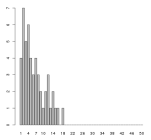
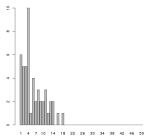
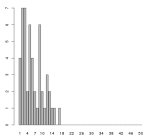
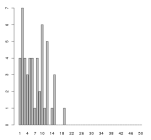
	1º	2º	3º	4º	5º
№Grupos	1400	2300	2000	2000	1700
Janela	21×3	21×3	18×3	21×3	21×3
MAP	0.2245	0.2244	0.2241	0.2216	0.2192
SEP					

Tabela 7.12: Os 5 piores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelos índices do grupo e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

A Tabela 7.13 mostra os 5 melhores resultados e a Tabela 7.14 os 5 piores resultados do MAP, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura feita com o histograma e comparada utilizando a distância Euclidiana.

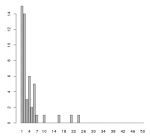
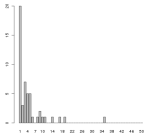
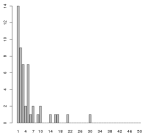
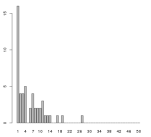
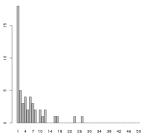
	1º	2º	3º	4º	5º
№Grupos	4100	4700	500	4700	500
Janela	15×6	12×6	12×6	15×3	21×6
MAP	0.3087	0.3083	0.2908	0.2891	0.2875
SEP					

Tabela 7.13: Os 5 melhores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelo histograma e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

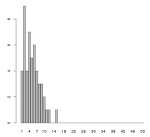
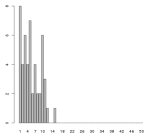
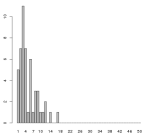
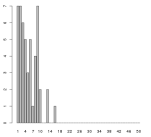
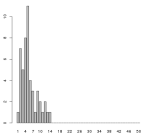
	1º	2º	3º	4º	5º
№Grupos	2000	1700	1100	3800	2000
Janela	12×3	15×6	18×3	12×9	21×6
MAP	0.2233	0.2233	0.2225	0.2185	0.2173
SEP					

Tabela 7.14: Os 5 piores resultados utilizando o extrator *Chroma Simples* com resolução 12, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelo histograma e a comparação através da distância Euclidiana. Os resultados são comparados em função da métrica MAP e da métrica Separabilidade

A execução com assinatura como histograma e a métrica RQA não foi utilizada por ter um alto custo computacional pela quantidade de grupos utilizados nos testes. Nesse caso, o tamanho da assinatura feita a partir dos índices do grupo é bem menor que o tamanho da assinatura como histograma.

7.4 Resultados Finais do Método Proposto

Através dos resultados apresentados nota-se que a técnica de agrupamento que mostrou melhores resultados foi a técnica *Single-Linkage*. A Tabela 7.15 apresenta os resultados obtidos com esta técnica de agrupamento para a janela 15×6, com o número de grupos definido em 4100, uma vez que esta configuração de parâmetros apresentou melhor desempenho. Os resultados apresentados na Tabela 7.15 expressam a variação das técnicas de extração com janela de 93 ms. Esses resultados são para analisar os extratores de características, além disso utilizou-se a base de 50 músicas com apenas 60 segundos do sinal.

Além da apresentação da métrica MAP, os valores SEP[·] representa o quantidade de *covers* recuperadas, divido pelo total de músicas na base, em que [·] representa a posição na lista de músicas recuperadas.

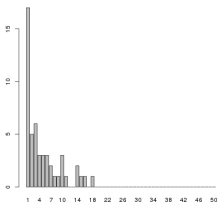
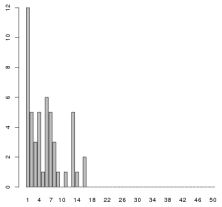
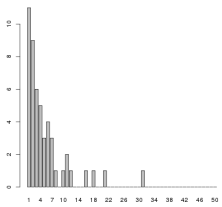
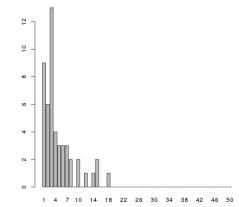
	MFCC	PCP	HPCP(12)	Chroma Simples(12)
Tempo (s)	109	72	136	174
MAP	0.2620	0.2883	0.2554	0.3087
SEP [1]	0.3469	0.2448	0.1632	0.3061
SEP [2]	0.1020	0.1020	0.1836	0.2857
SEP [3]	0.1224	0.0612	0.2244	0.0612
SEP				

Tabela 7.15: Resultados utilizando 4 extratores diferentes, em que o *Chroma Simples* e o *Harmonic Pitch Class Profile* tem resolução 12, utilizando a técnica de agrupamento *Single-Linkage*, a assinatura pelo índice dos grupos e a comparação através da distância Euclidiana.

Com os mesmos parâmetros, a Tabela 7.16 apresenta os resultados para o *Harmonic Pitch Class Profile* e o Chroma Simples, ambos com resolução 36. Além disso, foi utilizado o Chroma Simples, com resolução 36, combinado com o *Beat Tracking* da mesma forma como proposta por Ellis [20].

Por fim, o método proposto neste projeto foi executado para a base *Covers80* e utilizou-se o extrator *Chroma* Simples com 93 ms e sem sobreposição, e a técnica de agrupamento hierárquica *Single-Linkage*, com 4100 grupos para a base *Covers80*. A assinatura de cada música foi indicada a partir do histograma, e a distância Euclidiana foi adotada como métrica de comparação. Essa execução demorou 1024 segundos.

A Tabela 7.18 apresenta o tempo de processamento em segundos e o MAP dos métodos apresentados e do método proposto. Todos os métodos foram executados na base de 50 músicas.

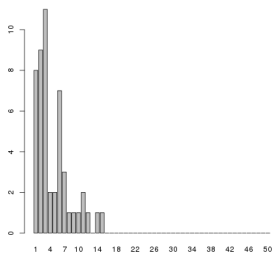
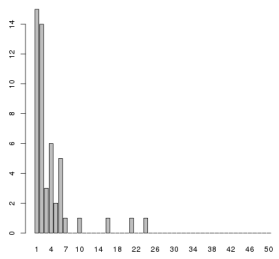
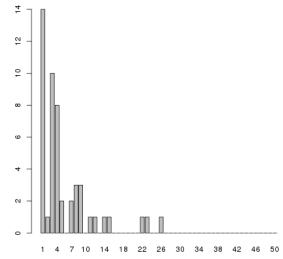
	HPCP(36)	Chroma Simples(36)	Chroma Simples/BT (36)
Tempo (s)	232	240	251
MAP	0.3228	0.2461	0.3286
SEP [1]	0.2244	0.1836	0.2857
SEP [2]	0.1836	0.1224	0.0204
SEP [3]	0.1224	0.2653	0.2040
SEP			

Tabela 7.16: Resultados utilizando 4 extratores diferentes, em que o *Chroma Simples* e o *Harmonic Pitch Class Profile* tem resolução 36. O *beat-synchronous* usado é o proposto por Ellis [20], porém o *Beat Tracking* é combinado ao Chroma Simples, ao invés do MFCC. Foi utilizado a técnica de agrupamento *Single-Linkage*, a assinatura pelo histograma e a comparação através da distância Euclidiana.

Bases	SEP [1]	SEP [2]	SEP [3]	MAP
Covers80	0.0061	0.0122	0.0122	0.0481

Tabela 7.17: Precisão do método proposto para a base *Covers80*, utilizando as métricas de avaliação MAP e Separabilidade. Esta execução foi feita utilizando o Chroma Simples, o *Single-Linkage*, a assinatura feita a partir do histograma e a distância Euclidiana.

Abordagens	MAP	Tempo de Processamento
Ellis	0.5531	396.3 (s)
Serrà et al.	0.3508	1339042 (s)
Proposta	0.3286	255 (s)

Tabela 7.18: Precisão e tempo de processamento dos métodos da literatura e do método proposto, utilizando a métrica MAP

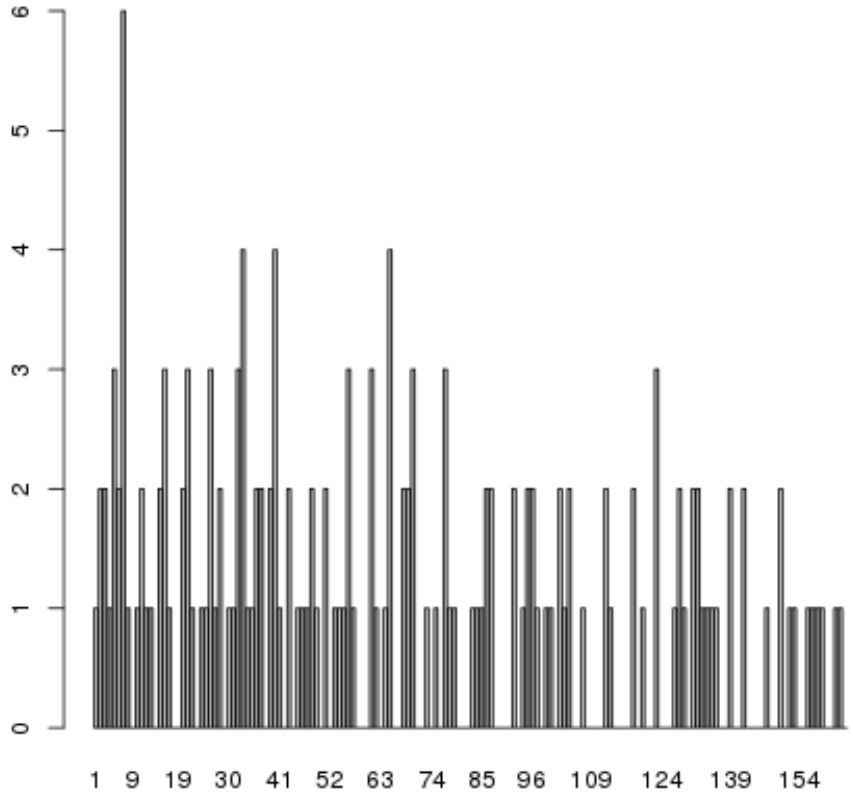


Figura 7.5: Gráfico da métrica de Separabilidade para a base *Covers80*, com a utilização do Chroma Simples com 93 ms, sem sobreposição, o *Single-Linkage* com 4100 grupos, a assinatura feita a partir do histograma e a distância Euclidiana. Serrà et al. [51].

Capítulo 8

Conclusão

As execuções dos métodos descritos no Capítulo 5, apresentaram um alto custo computacional, e consequentemente, um alto tempo de processamento, tornando inviáveis em aplicações reais. Isso motivou a criação de um novo método que adota-se técnicas de aprendizado de máquina para criar um dicionário a partir dos sinais de áudio e adotá-lo para toda a base.

Os resultados da proposta deste trabalho, ofereceram uma melhora no tempo de processamento quando comparado aos métodos de melhor precisão na área de MIR para identificação de *covers*. Para as bases empregadas a técnica de agrupamento adotadas gerou bons resultados, porém, tende a ser mais custosa para bases muito grandes, exigindo um estudo mais aprofundado. Apesar disso, os resultados sugerem que as técnicas de agrupamento podem contribuir na melhora da precisão dos métodos de identificação de *covers*.

Os resultados da bateria de testes, mostraram que a técnica de agrupamento *Single-Link* foi a mais efetiva. Isso ocorreu por conta da indexação dos grupos formados, de modo que os índices do grupo são colocados de acordo com a proximidade existente entre eles.

Os resultados também sugerem que a janela definida em 12 na dimensão da resolução abrange todos os 12 semitons da escala (2). O uso de outras janelas pode implicar em uma transposição errada na comparação entre músicas. Assim, ao usar extratores com resolução 36, a janela deverá ter também o tamanho 36 a fim de se manter essa resolução.

Além disso, os resultados obtidos com a proposta de [51] apresentaram divergência com os resultados apresentados na literatura, o que pode representar que a base de dados utilizada não é uma base bem comportada como a utilizada no MIREX.

8.1 Trabalhos Futuros

Ainda que a maioria dos resultados da bateria de testes (Seção 7.3) não alcançaram a precisão dos métodos que obtiveram a melhor precisão no MIREX, alguns resultados se aproximam dos resultados obtidos com a metologia de [51], porém a proposta deste

trabalho obteve um melhor tempo de processamento. Assim, um próximo caminho seria o estudo detalhado sobre a técnica *Single-Link*, de modo a encontrar um agrupador mais eficiente.

Além disso, o estudo feito sobre os extratores de característica, juntamente como método proposto por Serrà et al. [51], mostrou que o espaço gerado pelas informações extraídas do sinal ainda não foi totalmente explorado. Assim, será efetuado um estudo aprofundado desse espaço no Doutorado, de modo a obter uma melhora na precisão do método de identificação de *covers*.

Referências

[1] Aucouturier, J. J.; Pachet, F. Music similarity measures: What's the use? In: *Proceedings of 3rd International Conference on Music Information Retrieval*, 2002, p. 157–163.

[2] Bertin-Mahieux, T.; Ellis, D. P. W. Large-scale cover song recognition using hashed chroma landmarks. In: *WASPAA*, 2011, p. 117–120.

[3] Blume, H.; Haller, M.; Botteck, M.; Theimer, W. M. Perceptual feature based music classification - a dsp perspective for a new type of application. In: *ICSAMOS*, 2008, p. 92–99.

[4] Bogdanov, D.; Wack, N.; Gómez, E.; Gulati, S.; Herrera, P.; Mayor, O.; Roma, G.; Salamon, J.; Zapata, J.; Serra, X. Essentia: an open-source library for sound and music analysis. In: *ACM International Conference on Multimedia (MM'13)*, 2013.

[5] Bona, P. *Método de teoria e solfejo*. 2009.

[6] Bonada, J. Automatic technique in frequency domain for near-lossless time-scale modification of audio. In: *Proceedings of International Computer Music Conference*, 2000, p. 396–399.

[7] Brown, J. Calculation of a constant q spectral transform. *Journal of the Acoustical Society of America*, v. 89, p. 425–434, 1991.

[8] Cao, Y.; Wang, C.; Li, Z.; Zhang, L.; Zhang, L. Spatial-bag-of-features. In: *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010.

[9] Chatfield, C. *The analysis of time series: an introduction*. 2004.

[10] Chediak, A. *Harmonia & improvisaçao - vol. 1*. Harmonia & improvisaçao: 70 músicas harmonizadas e analisadas : viol˜ao, guitarra, baixo, teclado. 1986.

[11] Chen, Q.; Ai, L.; Wei, W.; Ma, W.; Xie, P.; Zhang, M. Analysis of music representations of vocal performance based on spectrogram. In: *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*, 2010, p. 1–4.

[12] Cychowski, D. F. C. T. Towards an inverse constant q transform. In: *Audio Engineering Society Convention 120*, 2006.

[13] Davies, M. E. P.; Plumbley, M. D. Context-dependent beat tracking of musical audio. *Trans. Audio, Speech and Lang. Proc.*, v. 15, p. 1009–1020, 2007.

[14] Degara, N.; Rua, E. A. Reliability-informed beat tracking of musical signals. In: *EEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING*, 2012.

[15] Dixon, S.; Cambouropoulos, E. Beat tracking with musical knowledge. In: *in ECAI 2000: Proceedings of the 14th European Conference on Artificial Intelligence*, 2000, p. 626–630.

[16] Ellis, Daniel P. W. Cotton, C. V. The 2007 labrosa cover song detection system. *MIREX 2007*, 2007.

[17] Ellis, D. P. W. Extracting information from music audio. *Commun. ACM*, v. 49, p. 32–37, 2006.

[18] Ellis, D. P. W. Identifying 'Cover Songs' with Beat-Synchronous Chroma Features. In: *MIREX 2006 System Abstracts*, 2006.

[19] Ellis, D. P. W. Beat tracking by dynamic programming. *J. New Music Research*, p. 51–60, 2007.

[20] Ellis, D. P. W.; Poliner, G. E. Identifying 'Cover Songs' with Chroma Features and Dynamic Programming Beat Tracking. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, 2007.

[21] Ellis, D. P. W.; Zeng, X.; McDermott, J. H. Classifying soundtracks with audio texture features. In: *ICASSP*, 2011, p. 5880–5883.

[22] Feng, J.; Ni, B.; Tian, Q.; Yan, S. Geometric lp-norm feature pooling for image classification. In: *CVPR'11*, 2011, p. 2697–2704.

[23] Fujishima, T. Realtime chord recognition of musical sound: a system using common lisp music. *Ann Arbor, MI: MPublishing, University of Michigan Library*, 1999.

[24] Gan, G.; Ma, C.; Wu, J. *Data clustering - theory, algorithms, and applications.* I-XXII, 1-466 p., 2007.

[25] Ganchev, T.; Fakotakis, N.; Kokkinakis, G. Comparative evaluation of various mfcc implementations on the speaker verification task. In: *in Proc. of the SPECOM-2005*, 2005, p. 191–194.

[26] Gemert, J.; Veenman, C.; Smeulders, A.; Geusebroek, J.-M. Visual word ambiguity. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, v. 32, p. 1271 –1283, 2010.

[27] Gómez, E. *Tonal description of music audio signals*. Tese de Doutoramento, Universitat Pompeu Fabra, 2006.

[28] Goto, M. An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research*, v. 30, p. 159–171, 2001.

[29] Goto, M.; Hayamizu, S. A real-time music scene description system: Detecting melody and bass lines in audio signals. In: *Speech Communication*, 1999, p. 31–40.

[30] Goto, M.; Muraoka, Y. A beat tracking system for acoustic signals of music. In: *In Proc. of the Second ACM Intl. Conf. on Multimedia*, 1994, p. 365–372.

[31] Holzapfel, A.; Davies, M.; Zapata, J.; Oliveira, J.; Gouyon, F. Selective sampling for beat tracking evaluation. *IEEE Transactions on Audio Speech and Language Processing*, 2012.

[32] Hossan, M.; Memon, S.; Gregory, M. A novel approach for mfcc feature extraction. In: *Signal Processing and Communication Systems (ICSPCS), 2010 4th International Conference on*, 2010, p. 1–5.

[33] Jégou, H.; Douze, M.; Schmid, C. Improving bag-of-features for large scale image search. *Int. J. Comput. Vision*, 2010.

[34] Jurie, F.; Triggs, B. Creating efficient codebooks for visual recognition. In: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2005.

[35] Keogh, E. Exact indexing of dynamic time warping. In: *Proceedings of the 28th international conference on Very Large Data Bases*, VLDB '02, 2002, p. 406–417 (*VLDB '02*, v.).

[36] Kotsifakos, A.; Papapetrou, P.; Hollmén, J.; Gunopulos, D.; Athitsos, V. Genre classification of symbolic music with smbgt. In: *PETRA*, 2013, p. 5.

[37] Lee, K. Automatic chord recognition from audio using enhanced pitch class profile. In: *ICMC Proceedings*, 2006.

[38] Lerch, A. *An introduction to audio content analysis: Applications in signal processing and music informatics*. Wiley-IEEE Press, 2012.

[39] Logan, B. Mel frequency cepstral coefficients for music modeling. In: *In International Symposium on Music Information Retrieval*, 2000.

[40] Lu, L.; Zhang, H. Automated extraction of music snippets. In: *Proceedings of the eleventh ACM international conference on Multimedia*, 2003, p. 140–147.

[41] Mckay, C. *Automatic music classification with jmir*. Tese de Doutoramento, McGill University, 2010.

[42] Mello, R. F. *Sistemas dinâmicos e técnicas inteligentes para a predição de comportamento de processos: Uma abordagem para otimização de escalonamento em grades computacionais*. Tese de Doutoramento, Universidade de São Paulo - (USP), 2010.

[43] Merwe, B.; Schulze, W. Music generation with markov models. *IEEE MultiMedia*, v. 18, p. 78–85, 2011.

[44] Muelder, C.; Provan, T.; Ma, K. Content based graph visualization of audio data for music library navigation. In: *Proceedings of the 2010 IEEE International Symposium on Multimedia*, 2010, p. 129–136.

[45] Paulus, J.; Klapuri, A. Music structure analysis by finding repeated parts. In: *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*, 2006, p. 59–68.

[46] Paulus, J.; Klapuri, A. Drum sound detection in polyphonic music with hidden markov models. *EURASIP J. Audio Speech Music Process.*, 2009.

[47] Proakis, J. G.; Manolakis, D. K. *Digital signal processing (4th edition)*. 43-150 p., 2006.

[48] Rebelo, A.; Capela, G.; Cardoso, J. S. Optical recognition of music symbols - a comparative study. *IJDAR*, v. 13, p. 19–31, 2010.

[49] S., R.; D.P.W., E. Cover song detection: From high scores to general classification. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, 2010, p. 65–68.

[50] Serrà, J.; Gómez, E.; Herrera, P.; Serra, X. Chroma binary similarity and local alignment applied to cover song identification. *IEEE Transactions on Audio, Speech & Language Processing*, p. 1138–1151, 2008.

[51] Serrà, J.; Serra, X.; Andrzejak, R. G. Cross recurrence quantification for cover song identification. *New Journal of Physics*, v. 11, p. 093017, 2009.

[52] Shao, X.; Xu, C.; Kankanhalli, M. S. Unsupervised classification of music genre using hidden markov model. In: *In IEEE International Conference on Multimedia and Expo*, 2004, p. 2023–2026.

[53] Sigurdsson, S.; Petersen, K. B.; Lehn-Schioler, T. Mel frequency cepstral coefficients: An evaluation of robustness of mp3 encoded music. In: *IN PROCEEDINGS OF THE INTERNATIONAL SYMPOSIUM ON MUSIC INFORMATION RETRIEVAL*, 2006.

[54] SillaJr., C. N.; Koerich, A. L.; Kaestner, C. A. A. A feature selection approach for automatic music genre classification. *Int. J. Semantic Computing*, p. 183–208, 2009.

[55] Stark, A. M.; Plumbley, M. D. Performance following: Real-time prediction of musical sequences without a score. *IEEE Transactions on Audio, Speech e Language Processing*, p. 190–199, 2012.

[56] Takens, F. Detecting strange attractors in turbulence. In: *Dynamical systems and turbulence*, p. 366–381, 1980.

[57] Turetsky, R. J.; Ellis, D. P. W. Ground-truth transcriptions of real music from force-aligned midi syntheses. In: *ISMIR*, 2003.

[58] Tzanetakis, G.; Cook, P. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, v. 10, p. 293–302, 2002.

[59] Viro, V. Peachnote: Music score search and analysis platform. In: *ISMIR*, 2011, p. 359–362.

[60] Xu, M.; Duan, L.-Y.; Cai, J.; Chia, L.-T.; Xu, C.; Tian, Q. Hmm-based audio keyword generation. In: *PCM (3)*, 2004, p. 566–574 (*Lecture Notes in Computer Science*, v.3333).

[61] Yaffee, R. A.; McGee, M. *Introduction to time series analysis and forecasting: With applications of sas and spss*. 2000.