
Visualização de similaridades em bases de dados de música

Jorge Henrique Piazzentin Ono

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Jorge Henrique Piazzentin Ono

Visualização de similaridades em bases de dados de música

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

Área de Concentração: Ciências de Computação e
Matemática Computacional

Orientador: Prof. Dr. Luis Gustavo Nonato

USP – São Carlos
Julho de 2015

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

P584v Piazzentin Ono, Jorge Henrique
 Visualização de similaridades em bases de dados
 de música / Jorge Henrique Piazzentin Ono;
 orientador Luis Gustavo Nonato. -- São Carlos, 2015.
 86 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, 2015.

1. Visualização Computacional. 2. Recuperação de
Informação de Música. I. Nonato, Luis Gustavo,
orient. II. Título.

Jorge Henrique Piazzentin Ono

Visualization of similarities in song data sets

Master dissertation submitted to the Instituto de Ciências Matemáticas e de Computação - ICMC-USP, in partial fulfillment of the requirements for the degree of the Master Program in Computer Science and Computational Mathematics. *FINAL VERSION.*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Luis Gustavo Nonato

USP – São Carlos
July 2015

Agradecimentos

Agradeço primeiramente a Deus.

Ao meu orientador, professor Dr. Luis Gustavo Nonato, a orientação, paciência, ajuda e amizade. Obrigado por acreditar em mim e me incentivar a buscar objetivos cada vez mais desafiadores.

À minha família, em especial à minha mãe Eliana, o amor e apoio incondicional.

Às amigas e parceiras de pesquisa, Débora e Martha, que me introduziram à área de música computacional.

Aos meus amigos do VICG. Obrigado por tornarem a universidade um lugar mais divertido. Estudar, ver filmes e jogar com vocês foram experiências inesquecíveis.

Aos meus amigos de longa data, Camilla e Mateus. Obrigado por estarem presentes durante toda a minha vida acadêmica e não acadêmica, apesar da distância.

Ao Instituto de Ciências Matemáticas e de Computação ICMC-USP, seus professores e funcionários.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), processo nº 132239/2013-2, e a Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), processo nº 2012/24801-0.

A todos que de certa forma contribuíram para a realização deste trabalho.

Resumo

Coleções de músicas estão amplamente disponíveis na internet e, graças ao crescimento na capacidade de armazenamento e velocidade de transmissão de dados, usuários podem ter acesso a uma quantidade quase ilimitada de composições. Isso levou a uma maior necessidade de organizar, recuperar e processar dados musicais de modo automático. Visualização de informação é uma área de pesquisa que possibilita a análise visual de grandes conjuntos de dados e, por isso, é uma ferramenta muito valiosa para a exploração de bibliotecas musicais. Nesta dissertação, metodologias para a construção de duas técnicas de visualização de bases de dados de música são propostas. A primeira, Grafo de Similaridades, permite a exploração da base de dados em termos de similaridades hierárquicas. A segunda, RadViz Concêntrico, representa os dados em termos de tarefas de classificação e permite que o usuário altere a visualização de acordo com seus interesses. Ambas as técnicas são capazes de revelar estruturas de interesse no conjunto de dados, facilitando o seu entendimento e exploração.

Palavras-chave: Visualização Computacional, Recuperação de Informação de Música.

Abstract

Music collections are widely available on the internet and, leveraged by the increasing storage and bandwidth capability, users can currently access a multitude of songs. This leads to a growing demand towards automated methods for organizing, retrieving and processing music data. Information visualization is a research area that allows the analysis of large data sets, thus, it is a valuable tool for the exploration of music libraries. In this thesis, methodologies for the development of two music visualization techniques are proposed. The first, Similarity Graph, enables the exploration of data sets in terms of hierarchical similarities. The second, Concentric RadViz, represents the data in terms of classification tasks and enables the user to alter the visualization according to his interests. Both techniques are able to reveal interesting structures in the data, favoring its understanding and exploration.

Keywords: Computational Visualization, Music Information Retrieval.

Sumário

1	Introdução	1
1.1	Contribuições	2
1.2	Organização da dissertação	2
2	Conceitos básicos sobre MIR	3
2.1	Pré-processamento	4
2.1.1	Redução de canais de áudio	4
2.1.2	Remoção CC	4
2.1.3	Normalização	5
2.1.4	Redução da taxa de amostragem	5
2.2	Análise espectral	6
2.2.1	Centroide espectral	8
2.2.2	Fluxo espectral	8
2.2.3	<i>Mel-Frequency Cepstral Coefficients</i>	8
2.3	Análise tonal	10
2.3.1	<i>Pitch Class Profile</i>	11
2.3.2	<i>Harmonic Pitch Class Profile</i>	12
2.4	Métricas de similaridade	13
2.4.1	<i>Dynamic Time Warping</i>	14
2.4.2	<i>Cross Recurrence Plot</i>	14
2.5	Segmentação estrutural	17
2.5.1	Segmentação baseada em novidade	19
2.5.2	Segmentação baseada em homogeneidade	19
2.5.3	Segmentação baseada em repetição	20
2.5.4	Segmentação hierárquica	20
2.6	Considerações finais	23

3 Visualização de músicas	25
3.1 Visualização de música individual	25
3.2 Visualização de coleções de músicas	29
3.3 Considerações finais	34
4 Grafo de Similaridades	35
4.1 Conceitos básicos sobre interpolações polinomiais para o desenho de curvas	36
4.2 Método	38
4.2.1 Pré-processamento da base de dados	38
4.2.2 Visualização da base de dados	39
4.3 Aplicação	43
4.4 Considerações finais	48
5 RadViz Concêntrico	49
5.1 Conceitos básicos sobre RadViz	51
5.1.1 Algoritmo e propriedades	51
5.1.2 Extensões	52
5.2 Método	55
5.2.1 RadViz Concêntrico	55
5.2.2 Ponderação sigmoidal	57
5.3 Validação	59
5.4 Aplicações	62
5.4.1 Base de músicas <i>Dortmund</i>	63
5.4.2 Base de filmes IMDB-F	66
5.5 Considerações finais	68
6 Conclusões	69
6.1 Discussões, limitações e trabalhos futuros	70
6.1.1 Grafo de Similaridades	70
6.1.2 RadViz Concêntrico	70
Apêndices	73
A Listagem de Músicas da Base <i>Covers Youtube</i>	75

Lista de Figuras

2.1	Processamento do sinal em janelas de tamanho \mathcal{H} e intervalo \mathcal{K}	7
2.2	Espectrograma da música “Abracadabra”, interpretada por Steve Miller Band.	7
2.3	Banco de filtros passa-banda triangular.	9
2.4	MFCC da música “Abracadabra”, interpretada por Steve Miller Band.	10
2.5	Uma oitava do teclado de um piano (Lerch, 2012).	10
2.6	Visualização da percepção de tom.	11
2.7	PCP da música “Abracadabra”, interpretada por Steve Miller Band.	12
2.8	HPCP da música “Abracadabra”, interpretada por Steve Miller Band.	14
2.9	Execução do algoritmo de DTW.	15
2.10	<i>Cross Recurrence Plot</i> da música “Day Tripper” por The Beatles com (a) uma música <i>cover</i> interpretada por Cheap Trick e (b) “I Can’t Get No Satisfaction”, por Rolling Stones.	16
2.11	Possíveis segmentações para a música “Dança Húngara N.5”, por Johannes Brahms.	18
2.12	Segmentação de músicas por novidade com o algoritmo de Foote (2000)	20
2.13	Exemplo de transformação de SSM para MTD.	21
3.1	<i>Piano roll</i> da música “With You Friends”, por Skrillex.	26
3.2	Visualização gerada pela <i>Music Annotation Machine</i>	26
3.3	Partitura condensada da música “Quinteto em Lá maior para clarinete, K. 581, por W. A. Mozart”.	27
3.4	Visualização <i>Shape of Song</i> da música “All The Small Things”, por Blink 182.	28
3.5	Visualização <i>Infinite Jukebox</i>	28
3.6	Visualização de coleções de músicas com meta-dados.	30

3.7	Base de músicas representada com <i>MusicNodes</i>	31
3.8	Representação de um conjunto de músicas com <i>Islands of Music</i>	31
3.9	Exemplo de execução da ferramenta <i>MusicBox</i>	32
3.10	Organização de uma base de dados da banda The Beatles com curvas de preenchimento de espaço.	33
3.11	Sistema de criação de Playlists com a projeção multidimensional PLP.	33
4.1	Curva gerada por uma interpolação Bézier quadrática.	36
4.2	Curva gerada por uma interpolação Hermite cúbica.	37
4.3	Metodologia para a criação do Grafo de Similaridades.	38
4.4	Visualização da coleção de músicas <i>CoversYoutube</i> com seis técnicas de redução de dimensionalidade.	40
4.5	Esquema de curvas Hermite conectando duas músicas no Grafo de Similaridade Global.	42
4.6	Esquema de curvas Bézier conectando duas músicas no Grafo de Similaridade Local.	42
4.7	Visualização da base <i>CoversYoutube</i> com o Grafo de Similaridades Global.	44
4.8	Exploração da base de dados com o Grafo de Similaridades Local e Global.	45
4.9	Reprodução de um segmento de música na visualização local.	46
4.10	Visualização de quatro músicas com o Grafo de Similaridades Local.	46
4.11	Visualização da base de dados <i>Covers80</i> com o Grafo de Similaridades Global.	47
4.12	Visualização de duas versões de “Let it be” com o Grafo de Similaridades Local.	47
5.1	Visualização RadViz da base de dados Iris.	51
5.2	RadViz aplicado no contexto de classificação de bases de imagens.	53
5.3	VRV aplicado ao agrupamento de micro-vetores de DNA.	53
5.4	RadVizS, extensão do RadViz tradicional em três dimensões.	54
5.5	<i>Sphereviz</i> , extensão tridimensional do RadViz.	55
5.6	Base de dados de formas geométricas visualizada com RadViz e RadViz Concêntrico	56
5.7	Visualização de uma tarefa de classificação com RadViz.	58
5.8	RadViz com ponderação sigmoidal.	59
5.9	Visualização da base de dados <i>Dortmund</i> com o Radviz Concêntrico	64
5.10	Exemplo de interação: busca por músicas eletrônicas cantadas por mulheres.	65
5.11	Radviz Concêntrico da base de filmes IMDB-F.	67

Lista de Siglas

AD	Âncora Dimensional
CC	Corrente Contínua
CRP	<i>Cross Recurrence Plot</i>
DFT	<i>Discrete Fourier Transform</i>
DTW	<i>Dynamic Time Warping</i>
GD	Grupo de Dimensões
GS	Grafo de Similaridades
GSG	Grafo de Similaridades Global
GSL	Grafo de Similaridades Local
HPCP	<i>Harmonic Pitch Class Profile</i>
LAMP	<i>Local Affine Multidimensional Projection</i>
MAP	<i>Mean of Average Precisions</i>
MDS	<i>Multidimensional Scaling</i>
MFCC	<i>Mel-Frequency Cepstral Coefficients</i>
MIDI	<i>Musical Instrument Digital Interface</i>
MIR	<i>Music Information Retrieval</i>
MIREX	<i>Music Information Retrieval Evaluation eXchange</i>
MP	Média das Precisões
MTD	Matriz de Tempo-Deslocamento
MTFL	<i>Multi-task Facial Landmark</i>
PCA	<i>Principal Component Analysis</i>
PCP	<i>Pitch Class Profile</i>
PV	Preservação de Vizinhança
RC	Radviz Concêntrico
RP	<i>Recurrence Plot</i>
RQA	<i>Recurrence Quantification Analysis</i>

SE	Segmentação Estrutural
SOM	<i>Self-Organizing Map</i>
SSM	<i>Self Similarity Matrix</i>
STFT	<i>Short Time Fourier Transform</i>
SVM	<i>Support Vector Machine</i>
t-SNE	<i>t-Distributed Stochastic Neighbor Embedding</i>

Introdução

Recentemente, visualização computacional vem deixando o papel de coadjuvante e surge como área de pesquisa básica, onde são desenvolvidas metodologias integradas para análise, interação e mineração de grandes conjuntos de dados multimodais e distribuídos. Tal fato pode ser comprovado tomando como base os avanços alcançados pelas ferramentas de visualização que estão sendo empregadas na análise e exploração de grandes coleções de documentos (Cui et al., 2011) e na representação visual de redes dinâmicas (Hadlak et al., 2011; Withall et al., 2007).

Existem setores, porém, onde as ferramentas de visualização ainda não alcançaram um nível de desenvolvimento satisfatório, como é o caso da exploração visual de grandes bases de dados de música: embora existam diversas visualizações disponíveis na literatura, as técnicas propostas falham em prover metodologias para a exploração dos vários aspectos musicais.

Coleções de músicas estão amplamente disponíveis na internet: serviços de *streaming*, por exemplo “Spotify” e “Deezer”, oferecem mais de 30 milhões de músicas a seus usuários (Spotify, 2015; Deezer, 2015). A crescente capacidade de armazenamento e transmissão de dados criou uma dificuldade no processo de entendimento e exploração das bases disponíveis. Nesse contexto, ferramentas de visualização podem auxiliar no processo de descoberta de informação e organização de coleções musicais. De modo geral, essas ferramentas podem ser úteis tanto a usuários comuns, sem conhecimento de teoria musical, quanto a especialistas, que trabalham com a análise e edição de conteúdo.

1.1 Contribuições

Com base no contexto apresentado, o objetivo deste trabalho foi desenvolver visualizações capazes de representar diferentes aspectos de coleções musicais, complementando o arcabouço de ferramentas disponíveis na literatura. Duas metáforas visuais para a análise de músicas foram projetadas. A primeira, Grafo de Similaridades, visa apresentar a coleção de músicas por dois pontos de vista: similaridades entre pares de músicas e similaridades entre trechos musicais.

A segunda técnica, RadViz Concêntrico, foi desenvolvida para possibilitar a exploração de bases de música no contexto de resultados de classificação, isto é, cada música é primeiramente classificada em termos de diferentes aspectos musicais e, posteriormente, projetada no espaço visual, tomando em consideração a classificação executada e a interpretação do usuário. O RadViz Concêntrico também foi utilizado no contexto de bases de dados de imagens e de filmes, o que mostrou a versatilidade da visualização.

Dois trabalhos foram gerados a partir desta dissertação de mestrado: o RadViz Concêntrico será publicado nos *proceedings* do Sibgrapi 2015 (Ono et al., 2015a) e o Grafo de Similaridades, no Workshop de Visualização do Sibgrapi 2015 (Ono et al., 2015b).

1.2 Organização da dissertação

Esta dissertação está estruturada da seguinte maneira:

- No Capítulo 2, conceitos básicos sobre os principais algoritmos de recuperação de informação musical baseado em conteúdo serão apresentados;
- No Capítulo 3, é feita uma revisão sobre os principais trabalhos relacionados à visualização de músicas;
- No Capítulo 4, a técnica Grafo de Similaridades é proposta e aplicada no contexto de bases de dados de músicas *cover*;
- No Capítulo 5, o RadViz Concêntrico é introduzido e validado em bases de dados de músicas, imagens e filmes;
- Por fim, no Capítulo 6, uma revisão geral da dissertação é realizada e discutem-se trabalhos futuros.

As opiniões, hipóteses e conclusões ou recomendações expressas neste material são de responsabilidade do autor e não necessariamente refletem a visão da FAPESP.

Conceitos básicos sobre MIR

Music Information Retrieval (MIR) é uma linha de pesquisa interdisciplinar que trata da descrição automática, entendimento, pesquisa, recuperação e organização de conteúdos musicais (Orio, 2006). Abrange o processamento de sinais de áudio, análise de formatos de música simbólicos, como o MIDI (*Musical Instrument Digital Interface*), e meta-dados, por exemplo, título, letra e compositor de uma música (Lerch, 2012).

O MIR baseado em conteúdo está diretamente relacionado com a análise de conteúdo de áudio, uma área mais específica que trata da extração de informações de sinais de áudio, como por exemplo, gravações de músicas armazenadas em formato digital. Deste modo, a base de qualquer tipo de sistema de análise de áudio é a extração de características, isto é, o cálculo de uma representação numérica compacta que pode descrever o sinal (Tzanetakis et al., 2002).

Neste capítulo, os principais algoritmos de pré-processamento e extração de características de áudio serão apresentados e discutidos. Uma breve revisão sobre comparação de sinais e segmentação automática de músicas também será realizada. As técnicas abordadas foram utilizadas como base para duas metodologias de visualização de coleções musicais propostas neste trabalho: Grafo de Similaridades (Capítulo 4) e RadViz Concêntrico (Capítulo 5).

2.1 Pré-processamento

De acordo com Lerch (2012), o sinal de áudio é frequentemente pré-processado antes da etapa de extração de características, reduzindo-se assim a quantidade de dados a serem analisados. Os algoritmos para pré-processamento podem ser agrupados em duas categorias:

- Algoritmos para tempo real, em que são conhecidas apenas as amostras atuais e as anteriores do sinal;
- Algoritmos *offline*, em que todas as amostras são conhecidas no tempo do processamento.

A seguir serão apresentados brevemente os principais algoritmos de pré-processamento.

2.1.1 Redução de canais de áudio

Com a popularização dos equipamentos eletrônicos com suporte à áudio multicanais, existem situações em que o número de canais do arquivo ou das caixas de som são limitados. Quando isso ocorre, é necessário aumentar (*up-mixing*) ou reduzir (*down-mixing*) o número de canais para se adequar ao agente limitante. Um estudo detalhado sobre essas operações pode ser encontrado em Bai et al. (2007).

No contexto da análise de conteúdo de áudio, muitas vezes a informação de interesse pode ser representada por um único canal. A forma mais simples de *down-mixing* é a média aritmética dos canais, dada por:

$$x'(i) = \frac{1}{C} \sum_{c=0}^{C-1} x_c(i) \quad (2.1)$$

, em que C é o número de canais de áudio e x_c é o canal de índice c .

2.1.2 Remoção CC

Um deslocamento de corrente contínua (CC), mais conhecido como *DC offset*, ocorre quando a média aritmética do sinal é muito diferente de zero, o que pode ser prejudicial na etapa de extração de características (Lerch, 2012). Deseja-se, portanto, remover esse deslocamento de sinal.

Se o processamento *offline* é viável, subtrai-se a média aritmética do sinal x de tamanho n de todas as amostras, como ilustrado na equação a seguir (Lerch, 2012):

$$x'(i) = x(i) - \frac{1}{n} \sum_{i=0}^{n-1} x(i) \quad (2.2)$$

Em um sistema de tempo real, não é possível calcular a média aritmética de todo o sinal de áudio. Uma maneira de remover o deslocamento CC neste caso é por meio de um filtro passa alta, por exemplo:

$$x'(i) = (1 - \alpha)(x(i) - (i - 1)) + \alpha x'(i - 1) \quad (2.3)$$

, em que α é o parâmetro do filtro passa-baixas que atenua o impacto do diferenciador nas frequências mais altas.

2.1.3 Normalização

Durante a amostragem do áudio, as amplitudes do sinal são quantizadas, isto é, arredondadas para valores de amplitude pré-definidos. O sinal é usualmente normalizado antes da etapa de extração de características, de modo que o processamento ocorra independentemente da quantização utilizada. A normalização *offline* é realizada dividindo-se todas as amostras pelo módulo da maior amostra no sinal:

$$x'(i) = \frac{x(i)}{\max(|x(i)|)} \quad (2.4)$$

, onde x é o sinal original (Lerch, 2012).

De acordo com Lerch (2012), a normalização de sinais em tempo real é uma tarefa difícil. Para o fazer, pode-se utilizar algoritmos de controle de ganho automático, ou compressores e limitadores, que monitoram características instantâneas do sinal.

2.1.4 Redução da taxa de amostragem

Taxas de amostragens comumente encontradas em sinais de áudio e voz são: 8kHz para telefonia, 32kHz para rádio digital e 44.1kHz para gravações de CD. O *down-sampling* modifica a quantidade de amostras, f_s , para uma taxa mais baixa, f_d , em um fator l . Mais especificamente, a taxa de amostragem do sinal subamostrado será igual a $\frac{f_s}{l}$. Uma maneira simples de realizar o *down-sampling* é apresentada (Müller, 2007):

$$x'(i) = x(l \cdot i) \quad (2.5)$$

No contexto de MIR, músicas são usualmente re-amostradas para 8kHz antes da etapa de extração de características, o que reduz consideravelmente o tempo de processamento dos algoritmos. De acordo com o teorema de Nyquist, é possível reduzir a taxa de amostragem de um sinal x sem muita perda de informação se x não contém frequências acima da metade da taxa de re-amostragem (Müller, 2007).

2.2 Análise espectral

De acordo com Chen et al. (2010), a forma de onda de uma música no domínio do tempo não revela muito sobre o seu conteúdo. Entretanto, a distribuição de frequências desse sinal carrega informações mais relevantes para a análise musical e pode ser calculada por meio da decomposição do espectro de frequências. Dada uma sequência $\{x(n), x \in \mathbb{R}, n \in \mathbb{Z}\}$, a transformada de Fourier decompõe o sinal em uma soma ponderada de componentes senoidais. A ponderação para a senoide de frequência ω é dada por:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-i\omega n} \quad (2.6)$$

Uma variante da transformada de Fourier muito utilizada para capturar transições de eventos em música é a *Short Time Fourier Transform* (STFT), uma transformada tempo-frequência. A STFT realiza a transformada de Fourier em pequenas janelas do sinal, assumindo, nessas janelas, que o sinal é estacionário (pode ser caracterizado por sua média e desvio padrão). Divide-se a música em quadros sequenciais (possivelmente com sobreposição), multiplica-se cada quadro por uma função janela e analisa-se o sinal resultante com a transformada de Fourier. Desta forma, obtém-se um gráfico chamado spectrograma, cujos eixos representam *tempo* \times *frequência*, e a cor, a magnitude das frequências. A transformada é dada por:

$$X(n, i) = \sum_{m=0}^{\mathcal{K}-1} x[n+m] w[m] e^{-j\omega_i m} \quad (2.7)$$

, em que n corresponde ao tempo, i , ao índice da frequência, e w , à função janela de suporte compacto. A frequência ω_i é dada por $2\pi i f_s / N$, em que f_s é a taxa de amostragem e N é o tamanho do sinal avaliado com a transformada de Fourier a cada janela (Oppenheim et al., 1999; Chen et al., 2010).

O processamento do sinal é, portanto, baseado em blocos. Divide-se o sinal em janelas de tamanho \mathcal{K} e distância \mathcal{H} entre os inícios de blocos consecutivos. A Figura 2.1 ilustra esse processo.

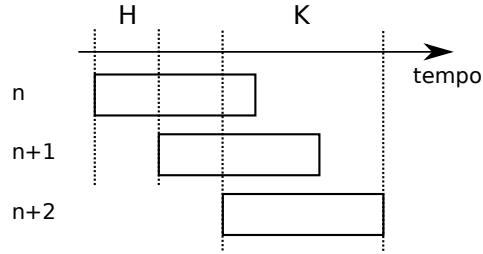


Figura 2.1: Processamento do sinal em janelas de tamanho \mathcal{H} e intervalo \mathcal{K} , adaptado de (Lerch, 2012).

A Figura 2.2 apresenta o spectrograma da música Abracadabra, interpretada por Steve Miller Band (base de dados *Covers80* (Ellis, 2007)), com parâmetros $\mathcal{H} = 512$, $\mathcal{K} = 512$ e função janela *Blackman-Harris*.

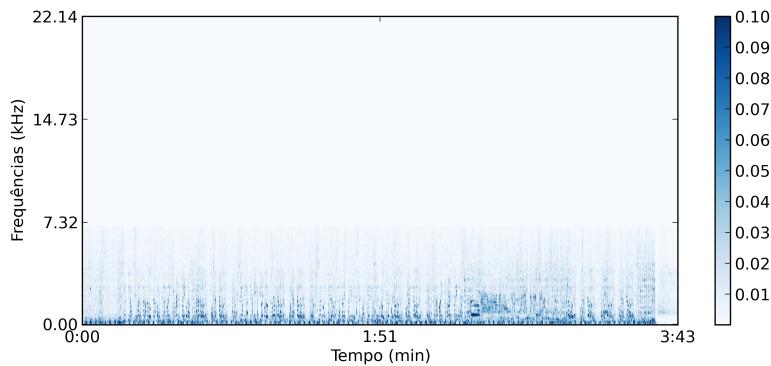


Figura 2.2: Espectrograma da música “Abracadabra”, interpretada por Steve Miller Band. Base de dados *Covers80* (Ellis, 2007).

A magnitude $|X(n, i)|$ e a fase $\phi(n, i)$ do spectrograma são dadas pelas Equações 2.8 e 2.9, descritas a seguir (Gomez, 2006).

$$|X(n, i)| = \sqrt{\text{Re}(X(n, i))^2 + \text{Im}(X(n, i))^2} \quad (2.8)$$

$$\phi(X(n, i)) = \arctan \frac{\text{Re}(X(n, i))}{\text{Im}(X(n, i))} \quad (2.9)$$

Características tipicamente utilizadas para representar o conteúdo de um áudio musical estão relacionadas ao timbre do sinal. O timbre pode ser descrito como a “cor” do som, sua qualidade ou textura. As características timbrais são baseadas na STFT e, portanto, calculadas para cada janela de som (Tzanetakis et al., 2002; Lerch, 2012). A seguir serão descritas três características pertencentes a esta categoria.

2.2.1 Centroide espectral

Centroide espectral é uma medida que representa o centro de gravidade da magnitude do espectro da STFT. Pode ser calculado por meio de:

$$C_t = \frac{\sum_{n=1}^N M_t[n]n}{\sum_{n=1}^N M_t[n]} \quad (2.10)$$

, onde $M_t[n]$ é a magnitude da transformada de Fourier no quadro t e frequência n . O centroide é uma medida de forma espectral, em que valores maiores correspondem à frequências mais altas (Tzanetakis et al., 2002).

2.2.2 Fluxo espectral

O fluxo espectral é definido como a diferença ao quadrado entre as magnitudes normalizadas de sucessivas distribuições espetrais. O cálculo do fluxo espectral é apresentado a seguir:

$$F_t = \sum_{n=1}^N (N_t[n] - N_{t-1}[n])^2 \quad (2.11)$$

$N_t[n]$ é a magnitude normalizada no tempo t e frequência de índice n . Essa medida representa a quantidade de variações na forma espectral. Valores baixos de fluxo espectral indicam um estado estacionário no sinal ou valores de entrada muito pequenos. Picos no fluxo espectral ocorrem quando há mudanças de tom ou no início de uma nova nota (Lerch, 2012).

2.2.3 Mel-Frequency Cepstral Coefficients

A percepção do som pelo homem é logarítmica. Essa não linearidade é atrelada à resolução de frequências perceptíveis pela cóclea humana (Lerch, 2012). Stevens et al. (1937) propuseram a escala Mel, um modelo que aproxima a percepção sonora, desenvolvido por meio de experimentos psicológicos. O modelo de conversão de frequências para a escala Mel elaborado por O'Shaughnessy (1987) é dado por:

$$m(f) = 1127 \cdot \log_{10} \left(1 + \frac{f}{700Hz} \right) \quad (2.12)$$

Mel-Frequency Cepstral Coefficients (MFCC) são características que tentam incorporar propriedades do sistema auditivo humano na representação do sinal. Essa técnica foi

criada no contexto de reconhecimento automático de voz por Hunt et al. (1980) e fornece uma representação mais compacta para o áudio em comparação com a STFT.

Primeiramente, calcula-se a STFT do sinal de áudio. Em seguida, o espectro é recodificado de acordo com um banco de filtros passa-banda triangulares, igualmente espaçados na escala Mel. A Figura 2.3 ilustra o banco de filtros na escala linear de frequências (Figura 2.3a) e na escala Mel (Figura 2.3b). Os MFCCs são obtidos por meio da equação abaixo:

$$c_n = \sum_{k=1}^K (\log S_k) \cos[n(k - 0.5) \frac{\pi}{K}] \quad (2.13)$$

, em que K é o número de filtros no banco de filtros, S_k é a potência de saída do k º filtro e n é o índice do bin MFCC (Han et al., 2006).

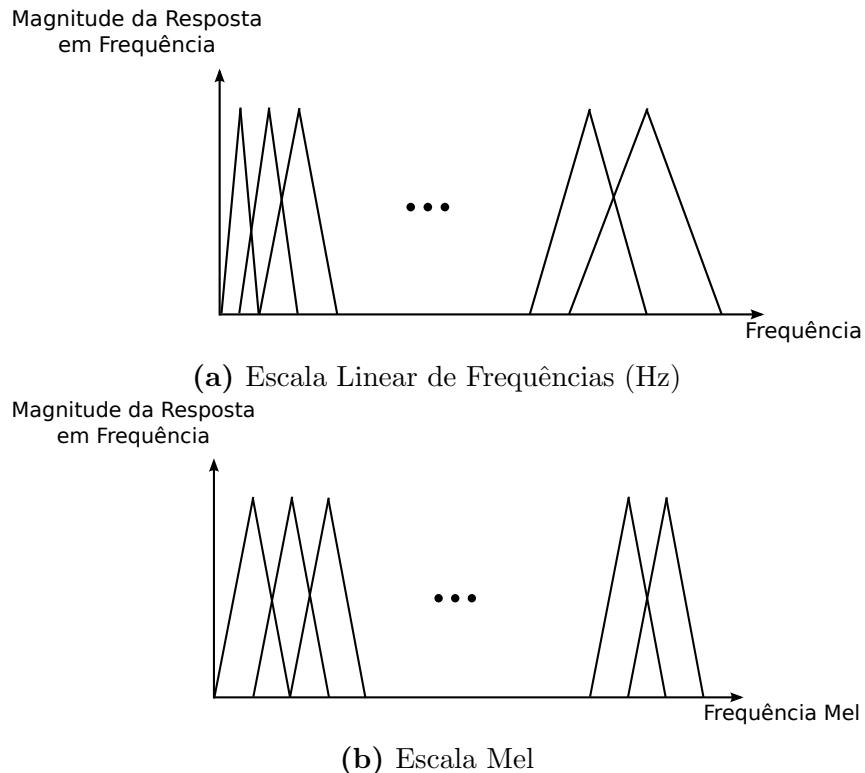


Figura 2.3: Banco de filtros passa-banda triangular. Figura adaptada de (Han et al., 2006)

A Figura 2.4 apresenta o MFCC da música Abracadabra, com $\mathcal{H} = 1024$, $\mathcal{K} = 2048$ e função janela *Blackman-Harris*.

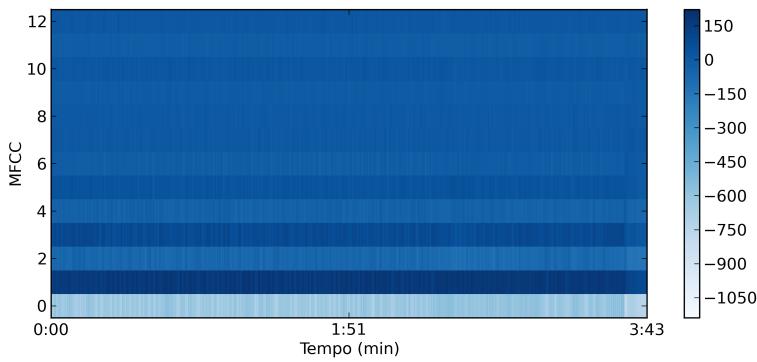


Figura 2.4: MFCC da música “Abracadabra”, interpretada por Steve Miller Band.
Base de dados *Covers80* (Ellis, 2007).

2.3 Análise tonal

Aspectos tonais são muito importantes para a análise de músicas. A altura (*pitch*) está diretamente relacionada com a melodia, a harmonia e a assinatura tonal de uma composição musical. A percepção humana de altura está relacionada com a frequência de um sinal, sendo que frequências mais altas levam a percepção de tons mais agudos (Lerch, 2012).

Instrumentos melódicos de espectro harmônico produzem sons que podem ser aproximados por uma combinação linear de componentes senoidais de frequências f_0 , $2f_0$, $3f_0$, ..., nf_0 , em que a frequência fundamental f_0 determina como o tom será percebido.

Na escala temperada, as notas são nomeadas usando as primeiras 7 letras do alfabeto, indicadores de acidentes (bemol \flat e sustenido \sharp) e um número que indica a posição da nota em oitavas. Uma oitava é o intervalo de um C até o próximo C (12 semitonos). A Figura 2.5 ilustra uma oitava de um teclado de piano, com as notas anotadas sobre as teclas. Em um piano, o primeiro C (mais a esquerda) é o C1 e o último C (a direita) é o C8 (Kostka et al., 1995).

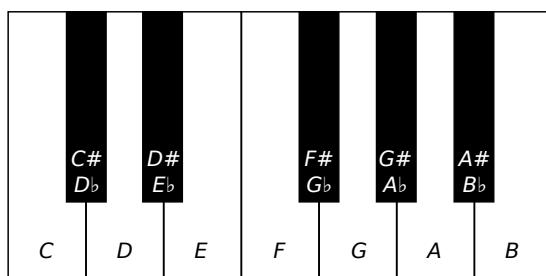


Figura 2.5: Uma oitava do teclado de um piano (Lerch, 2012).

Diversas estruturas são formadas a partir de notas musicais. Escalas consistem em arranjos de notas, seus intervalos de tons e semi-tonos. Cada escala é associada a uma assinatura tonal, isto é, um conjunto de zero a sete acidentes arranjados em uma determinada ordem na partitura. O conjunto de notas em relação à escala define a tonalidade da música e um conjunto de notas tocadas simultaneamente correspondem a acordes. Por fim, campos harmônicos correspondem a um conjunto de acordes formados a partir de uma determinada escala (Kostka et al., 1995; Chediak, 1986).

Sons cujas frequências têm uma razão de potência de 2 ($f_0, 2f_0, 4f_0, 8f_0, \dots$) são percebidos como similares. Esse fenômeno é conhecido como percepção de *chroma*. O termo *chroma*, ou classe de nota (*pitch class*), é utilizado para agrupar todas as notas que tem o mesmo tom, exceto pela diferença de uma ou mais oitavas. A Figura 2.6 ilustra esse fenômeno como um gráfico em hélice, em que a frequência cresce monotonicamente no eixo Z (Lerch, 2012). Pontos com as mesmas coordenadas em X e Y compartilham de uma razão de frequência em potência de 2, ou seja, possuem a mesma classe de nota. Na música ocidental baseada na escala temperada, as classes de notas são: *C, C \sharp /D \flat , D, D \sharp /E \flat , E, F, F \sharp /G \flat , G, G \sharp /A \flat , A, A \sharp /B \flat , B* (Kostka et al., 1995).

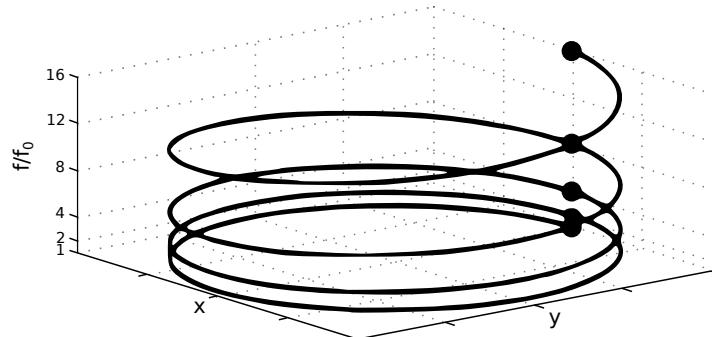


Figura 2.6: Visualização da percepção de tom. Os eixos X e Y simbolizam as classes de notas e o eixo Z, as frequências. Figura adaptada de (Lerch, 2012).

Diversos métodos foram desenvolvidos para identificar automaticamente quais classes de notas são tocadas numa janela de áudio. Nesta seção, duas técnicas serão apresentadas.

2.3.1 Pitch Class Profile

O algoritmo *Pitch Class Profile* (PCP) foi desenvolvido por Fujishima (1999) com o intuito de reconhecer acordes musicais. Neste processo, calcula-se a transformada discreta de Fourier (DFT) de janelas do áudio (*Short Time Fourier Transform*). Em seguida, deriva-se o PCP como sendo um vetor de doze dimensões que representa as intensidades de doze classes de semitonos em cada janela. Cada vetor PCP é calculado por meio da

Equação 2.14, sendo M definida pela Equação 2.15 e X a STFT em uma janela (Fujishima, 1999).

$$PCP(p) = \sum_{l|M(l)=p} ||X(l)||^2 \quad (2.14)$$

$$M(l) = \begin{cases} -1 & l = 0 \\ round(12log_2((f_s \cdot \frac{1}{N})/f_{ref}))mod12 & l = 1, 2, \dots, N/2 - 1 \end{cases} \quad (2.15)$$

A tabela M mapeia o espectro resultante da DFT para o espectro dos doze vetores que representam os semitonos. A frequência de referência (tom em que os instrumentos foram afinados) é dada por f_{ref} , cujo valor padrão é 440 Hz (nota A4).

A Figura 2.7 apresenta o PCP da música “Abracadabra”, interpretada por Steve Miller Band.

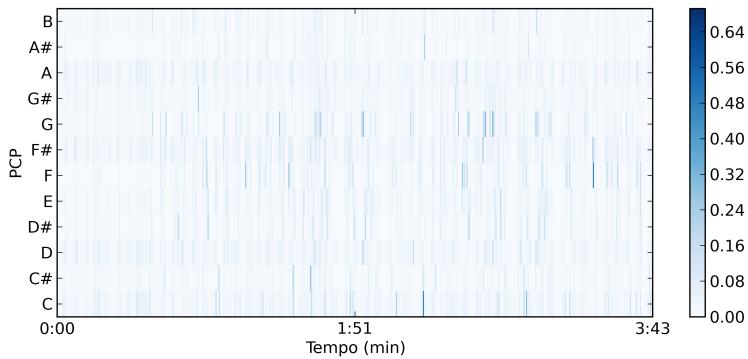


Figura 2.7: PCP da música “Abracadabra”, interpretada por Steve Miller Band. Base de dados *Covers80* (Ellis, 2007).

2.3.2 Harmonic Pitch Class Profile

Harmonic Pitch Class Profile (HPCP) (Gomez, 2006) é uma característica de distribuição de tons baseada no PCP que considera a presença de harmônicos para o cálculo do vetor de *chroma*. Primeiramente, é feito um pré-processamento do sinal com a técnica *transient location* (Bonada, 2000) para remover regiões de ruído da música. A STFT do sinal filtrado é calculada e, em seguida, os picos (máximos locais da magnitude do espectro) com frequências entre 40 e 5000 Hz são encontrados. Para aumentar a resolução do sinal na etapa de detecção de picos, são utilizadas interpolações quadráticas em pontos próximos do máximo local (Gomez, 2006).

O vetor HPCP é definido com a Equação 2.16, em que a_i é a magnitude e f_i é a frequência do pico de índice i . $nPeaks$ é o número de picos encontrados, n é o índice do *bin* HPCP (*chroma*), *size* é o tamanho do vetor HPCP (12, 24, 36, ...) e $w(n, f_i)$ é o peso associado à frequência f_i , *bin* n (Gomez, 2006).

$$HPCP(n) = \sum_{i=1}^{nPeaks} w(n, f_i) \cdot a_i^2, n = 1 \dots size \quad (2.16)$$

O peso $w(n, f_i)$ é dado pela Equação 2.17, em que l é o tamanho da janela de ponderação w ($\frac{4}{3}$ de um semitom) e d é a distância em semitons entre o pico da frequência f_i e a frequência central do bin n , f_n .

$$w(n, f_i) = \begin{cases} \cos^2\left(\frac{\pi}{2} \cdot \frac{d}{0,5 \cdot l}\right) & , |d| \leq 0,5 \cdot l \\ 0 & , |d| > 0,5 \cdot l \end{cases} \quad (2.17)$$

A distância d é dada pela Equação 2.18, em que f_n é a frequência central do *bin* n e m é o inteiro que minimiza o módulo da distância $|d|$.

$$d = 12 \cdot \log_2 \frac{f_i}{f_n} + 12 \cdot m \quad (2.18)$$

A frequência central do *bin* n (f_n) é calculada por meio da Equação 2.19, em que f_{ref} é a frequência de referência global da música. A metodologia para o cálculo de f_{ref} está disponível em (Gomez, 2006).

$$f_n = f_{ref} \cdot 2^{\frac{n}{size}}, n = 1 \dots size \quad (2.19)$$

A Figura 2.8 apresenta o HPCP da música “Abracadabra”, com $\mathcal{H} = 1024$, $\mathcal{K} = 2048$ e função de janelamento *Blackman-Harris*. 12 *bins* HPCP foram calculados, cada um representando uma classe de nota distinta.

2.4 Métricas de similaridade

Dadas as características de um sinal de áudio, pode-se definir métricas de similaridade para comparar músicas. Tais métricas são importantes para a identificação de versões *cover* de músicas, definidas por Serrà et al. (2009) como versões alternativas de uma música previamente gravada.

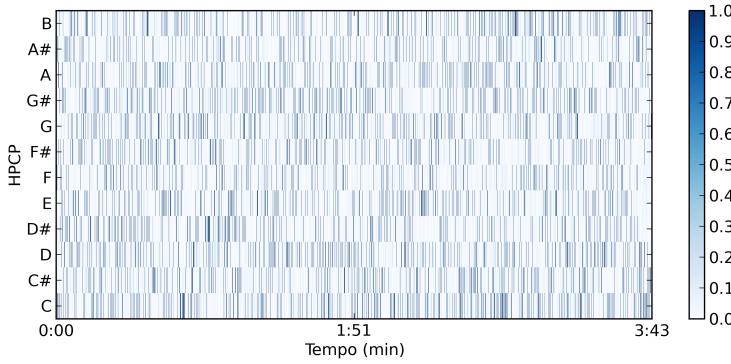


Figura 2.8: HPCP da música “Abracadabra”, interpretada por Steve Miller Band.
Base de dados *Covers80* (Ellis, 2007).

2.4.1 Dynamic Time Warping

O *Dynamic Time Warping* (DTW) (Keogh et al., 2005) é uma técnica que permite definir uma métrica de distância entre séries temporais, por exemplo, características de áudio. Dadas duas séries Q e C de tamanhos n e m , respectivamente, cria-se uma matriz D , $n \times m$, onde o elemento $d_{i,j}$ é a distância entre o elemento q_i e c_j .

A partir da matriz D , o algoritmo procura por um caminho mínimo W , onde $w_k = (i, j)_k$, que respeite as seguintes condições:

- Comece em $(1, 1)$ e termine em (n, m) ;
- Percorra apenas índices adjacentes;
- Percorra espaçamentos iguais no tempo.

O caminho mínimo é encontrado por meio de um algoritmo de programação dinâmica de complexidade $O(nm)$ (Keogh et al., 2005). A métrica de dissimilaridade DTW é dada por:

$$DTW(Q, C) = \sqrt{\sum_{k=1}^K D[w_k]} \quad (2.20)$$

, em que K é o tamanho do caminho e $D[w_k]$ é o valor da entrada $(i, j)_k$ na matriz D . A Figura 2.9 ilustra a execução do algoritmo DTW.

2.4.2 Cross Recurrence Plot

O trabalho de Serrà et al. (2009) propõe o uso de *Cross Recurrence Plots* (CRP) como métrica de similaridade entre músicas. Antes de definir CRP, é necessário introduzir

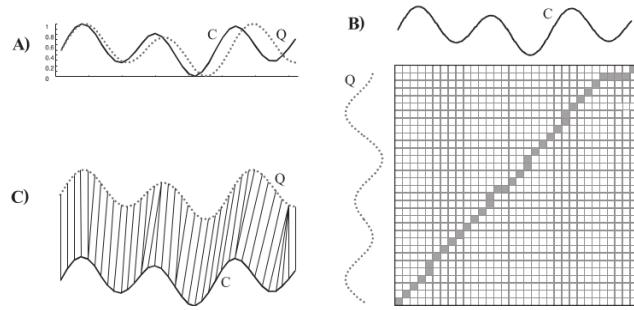


Figura 2.9: Execução do algoritmo de DTW. (A) Séries C e Q. (B) Matriz de distâncias. (c) Alinhamento das duas séries com o caminho mínimo. Figura adaptada de (Keogh et al., 2005).

o conceito de *Recurrence Plot* (RP). RP é uma ferramenta utilizada para visualizar recorrências em uma série temporal, ou seja, regiões onde a órbita da série passa perto de um estado previamente visitado. Mais especificamente, RP é uma matriz quadrada preenchida com zeros e uns, que indicam se há ou não recorrência, ou seja, se o estado no tempo i é similar ao estado do tempo j (Eckmann et al., 1995; Alligood et al., 2000). A diagonal principal de um RP é, portanto, composta por uns. CRPs são construídos da mesma maneira que RPs, mas cada eixo corresponde a uma série temporal diferente e a matriz resultante não é quadrada.

Primeiramente, o algoritmo extrai a característica HPCP (Gomez, 2006) de duas músicas, resultando em séries temporais de $H = 12$ variáveis. Dados os vetores HPCP da música \mathbf{x} e da música \mathbf{y} , calcula-se a transposição de \mathbf{y} de modo que ela fique na mesma tonalidade de \mathbf{x} . A transposição ocorre rotacionando-se o vetor HPCP de \mathbf{y} em k posições, por meio da técnica *Optimal Transposition Index*, proposta em (Serra et al., 2008).

A seguir, calcula-se o *embedding* das duas músicas em um espaço de fase, isto é, um espaço onde as recorrências do sinal podem ser obtidas. Considere que o HPCP \mathbf{x} tem N_x^* janelas. O *embedding* de \mathbf{x} é dado por $\mathbf{x}' = \{x_i\}$, para $i = 1, \dots, N_x$, $N_x = N_x^* - (m-1)\tau$, em que x_i é calculado com:

$$\begin{aligned} x_i = & (x_{1,i}, x_{1,i+\tau}, \dots, x_{1,i+(m-1)\tau}, \\ & x_{2,i}, x_{2,i+\tau}, \dots, x_{2,i+(m-1)\tau}, \\ & \dots \\ & x_{H,i}, x_{H,i+\tau}, \dots, x_{H,i+(m-1)\tau}) \end{aligned} \quad (2.21)$$

Os autores estimaram os valores ótimos de m e τ para o reconhecimento de músicas *cover*, por meio da divisão de uma base de dados em conjunto de treinamento e teste: os parâmetros encontrados foram $m = 10$ e $\tau = 1$.

Serrà et al. (2009) utilizam a Equação 2.22 para calcular o CRP, em que $\Theta(\cdot)$ é a função degrau tipo Heaviside ($\Theta(v) = 0$ se $v < 0$ e $\Theta(v) = 1$ caso contrário), ϵ_i^x e ϵ_j^y são limiares de distâncias e $\|\cdot\|$ é a norma Euclidiana. No artigo, os autores calculam os limiares dinamicamente, de modo que 10% dos vizinhos de cada entrada sejam considerados semelhantes.

$$R_{i,j} = \Theta(\epsilon_i^x - \|\mathbf{x}_i - \mathbf{y}_j\|) \Theta(\epsilon_j^y - \|\mathbf{x}_i - \mathbf{y}_j\|) \quad (2.22)$$

Em geral, pares de músicas diferentes não exibem nenhum padrão evidente no CRP e pares de músicas *cover* apresentam estrutura de linhas longas. A Figura 2.10 ilustra o CRP da música “Day Tripper”, por The Beatles, com (2.10a) uma música *cover* interpretada por Cheap Trick e (2.10b) com “I Can’t Get No Satisfaction”, por Rolling Stones. Observa-se que quando as músicas comparadas são *covers*, longas diagonais são formadas no CRP e caso contrário, tais padrões não ocorrem.

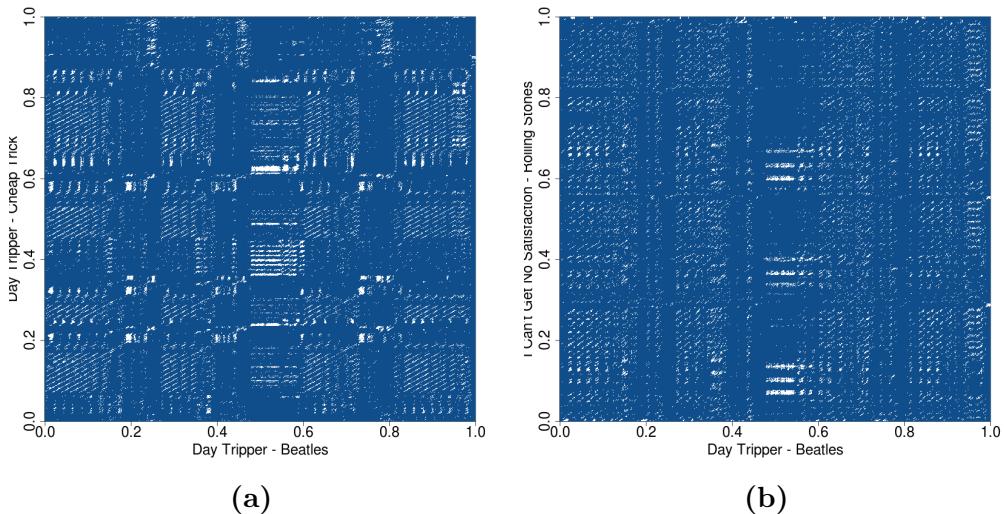


Figura 2.10: *Cross Recurrence Plot* da música “Day Tripper” por The Beatles com (a) uma música *cover* interpretada por Cheap Trick e (b) “I Can’t Get No Satisfaction”, por Rolling Stones. Base de dados Covers80 (Ellis, 2007).

Com base no CRP de duas músicas, o trabalho de Serrà et al. (2009) propõe o uso do maior comprimento das diagonais formadas na matriz como métrica de similaridade. A medida Q_{max} é definida como o maior comprimento das diagonais na matriz CRP, considerando possíveis variações no tempo da música (que correspondem a curvaturas nos traços) e na melodia (pequenas rupturas). Para o cálculo do Q_{max} , define-se primeiramente a matriz Q , $Q_{1,j} = Q_{2,j} = Q_{i,1} = Q_{i,2} = 0$ para $i = 1, \dots, N_x$ e $j = 1, \dots, N_y$, e constrói-se o restante da matriz por meio de:

$$Q_{i,j} = \begin{cases} \max\{Q_{i-1,j-1}, Q_{i-2,j-1}, Q_{i-1,j-2}\} + 1 & , R_{i,j} = 1 \\ \max\{0, Q_{i-1,j-1} - \gamma(R_{i-1,j-1}), Q_{i-2,j-1} - \gamma(R_{i-2,j-1}) \\ \quad , Q_{i-1,j-2} - \gamma(R_{i-1,j-2})\} & , R_{i,j} = 0 \end{cases} \quad (2.23)$$

$i = 3, \dots, N_x$
 $j = 3, \dots, N_y$

A função de penalidade γ é dada por:

$$\gamma(z) = \begin{cases} \gamma_0 & z = 1 \\ \gamma_e & z = 0 \end{cases} \quad (2.24)$$

, em que γ_0 e γ_e são penalidades para irregularidades nas diagonais da matriz. No trabalho, são usadas as constantes $\gamma_0 = 5$ e $\gamma_e = 0.5$.

Por fim, a medida Q_{max} é dada pelo valor máximo das entradas da matriz Q :

$$Q_{max} = \max\{Q_{i,j}\}, i = 1, \dots, N_x \text{ e } j = 1, \dots, N_y \quad (2.25)$$

A partir da medida Q_{max} , é possível identificar grupos de música *cover* em uma base de dados. Quanto maior o Q_{max} entre duas instâncias, maior a probabilidade de elas serem interpretações diferentes da mesma música.

O trabalho de Serrà et al. (2009) obteve acurácia de 0.661 na competição MIREX 2008 (*Music Information Retrieval Evaluation eXchange*)¹ de identificação de músicas *cover*, ficando em segundo lugar. O primeiro lugar da competição foi uma variação da mesma técnica que utiliza aprendizado não supervisionado para agrupar as músicas. Até hoje, Q_{max} é considerado o estado da arte em métricas para identificação de músicas *cover*. Sua principal desvantagem é seu alto custo computacional, $O(mn)$, similar ao DTW, em que m e n são o número de janelas das duas músicas.

2.5 Segmentação estrutural

Músicas podem ser definidas como sequências de eventos sonoros inter-relacionados. A interpretação desses eventos é hierárquica, podendo levar em consideração desde o nível de detalhe mais fino, como notas individuais, até estruturas mais complexas, como

¹www.music-ir.org/mirex

progressões de acordes e segmentos mais longos (Paulus et al., 2010). A segmentação estrutural de músicas (SE) é uma linha de pesquisa que tem como objetivo dividir uma gravação de áudio em segmentos, agrupando-os em categorias significativas no contexto musical, por exemplo, introdução, refrão, verso e conclusão.

Esta é uma área muito ativa em MIR, sendo que nos últimos três anos, 24 submissões foram feitas à competição MIREX. Assim, não é o objetivo desta seção fazer uma revisão completa do tópico, mas sim prover ao leitor uma ideia geral de como os algoritmos operam, embasando parte da metodologia de visualização proposta no Capítulo 4. Uma análise detalhada das técnicas de segmentação estrutural pode ser encontrada em (Paulus et al., 2010).

A Figura 2.11 ilustra dois possíveis resultados de segmentação da música “Dança Húngara N.5”, de Johannes Brahms. Observe que a SE pode ter mais de uma solução correta. A sequência de trechos “A-B-B” da segmentação 1 é agrupada para formar o trecho “E” da segmentação 2. Essa pluralidade de soluções torna a validação dos resultados uma tarefa árdua, principalmente porque a maioria das técnicas de SE retorna como resposta uma única segmentação.

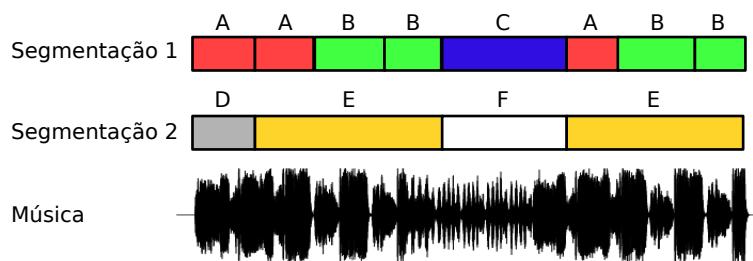


Figura 2.11: Possíveis segmentações para a música “Dança Húngara N.5”, por Johannes Brahms. As segmentações foram feitas manualmente.

A maior parte dos algoritmos de SE têm início com o cálculo de uma matriz de auto-similaridade, do inglês *Self Similarity Matrix* (SSM), avaliada com base em características janeladas do áudio. Entre as características mais utilizadas, pode-se citar o espectrograma, MFCC e HPCP do sinal (McFee et al., 2014). Dependendo do que se deseja segmentar, uma característica pode ser mais adequada do que outra. Por exemplo, se o usuário quer encontrar trechos com instrumentação semelhantes, um descritor de timbre deve ser utilizado. Por outro lado, se o objetivo é comparar segmentos em termos de harmonia e melodia, descritores de *chroma* são mais apropriados. A matriz é obtida por meio da aplicação de uma métrica de dissimilaridade, por exemplo, cosseno, entre os descritores janelados de áudio (Paulus et al., 2010).

A partir da SSM, os algoritmos podem abordar o problema de três modos conceitualmente distintos, buscando estruturas que indiquem novidade, homogeneidade ou repetição

na música (Paulus et al., 2010). As três abordagens serão descritas nas subseções 2.5.1, 2.5.2 e 2.5.3, respectivamente. Na subseção 2.5.4, um algoritmo de segmentação musical hierárquico será brevemente descrito. Essa técnica será utilizada como base para a construção da visualização de coleções musicais proposta no Capítulo 4.

2.5.1 Segmentação baseada em novidade

As abordagens baseadas em novidade buscam por pontos onde mudanças ocorrem na música. Um algoritmo bastante conhecido foi proposta por Foote (2000): dada uma SSM $S_{n,n}$, o algoritmo busca por pontos de novidade correlacionando S com uma matriz *kernel* $K_{m,m}$, $m < n$. O *kernel* é uma matriz diagonal por blocos, composta por dois blocos quadrados de tamanho $\frac{n}{2}$. Os blocos na diagonal são preenchidos com 1 e os blocos fora da diagonal, com -1. Um *kernel* de tamanho $m = 4$ é apresentado abaixo:

$$K_{4,4} = \begin{vmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 \end{vmatrix} \quad (2.26)$$

O *kernel* é então correlacionado com a matriz S na direção da diagonal principal. Isso gera uma função correlação, cujos picos são indicadores de que ocorreu uma novidade no sinal. Usando MFCCs para o cálculo da matriz, esses picos são bons indicadores de mudanças no timbre ou instrumentação da música (Foote, 2000). A Figura 2.12 apresenta a operação de correlação de um *kernel* em formato tabuleiro e da matriz SSM. Variando-se o tamanho do *kernel*, o algoritmo pode detectar novidades com uma granularidade maior ou menor.

2.5.2 Segmentação baseada em homogeneidade

Técnicas de segmentação baseadas em homogeneidade são extensões de segmentações baseadas em novidade. De modo geral, elas executam, primeiramente, uma detecção de novidades e em seguida, o agrupamento dos segmentos encontrados (Paulus et al., 2010). Essa abordagem foi introduzida por Cooper et al. (2003), onde, após a segmentação com o algoritmo de Foote (2000) e *kernel* de tamanho 256×256 , o conteúdo de cada segmento é modelado com uma distribuição normal multivariada. Calcula-se uma matriz de dissimilaridade entre os segmentos utilizando a divergência Kullback-Leibler e, finalmente, os trechos são agrupados por meio de um agrupamento espectral.

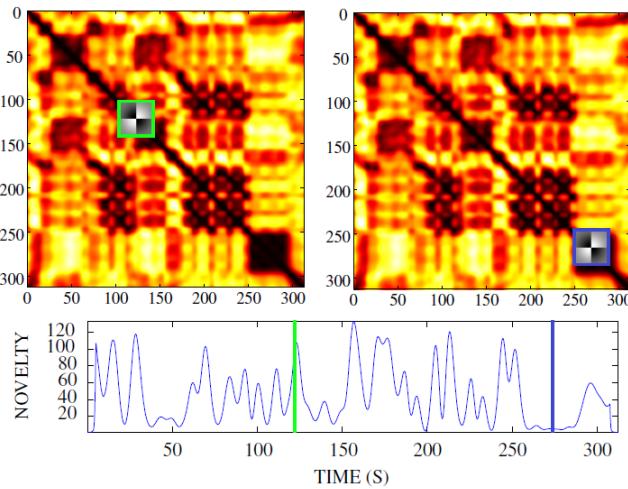


Figura 2.12: Segmentação de músicas por novidade com o algoritmo de Foote (2000). Acima: SSM usando MFCC da música “Tuonelan koivut”, de Kotiteollisuus. O *kernel* tabuleiro, mostrado em duas posições na matriz esquerda (verde) e direita (azul), é correlacionado na diagonal principal. Abaixo: Correlação resultante. Figura adaptada de (Paulus et al., 2010).

2.5.3 Segmentação baseada em repetição

Abordagens baseadas em repetição buscam por linhas paralelas à diagonal principal da SSM, que indicam a repetição de um trecho no arquivo de áudio (Paulus et al., 2010). O primeiro algoritmo desta categoria foi o de (Goto, 2003), que realiza uma transformação na SSM de modo que as diagonais sejam mais facilmente encontradas. A primeira etapa do algoritmo é a filtragem da SSM, por meio de um filtro passa baixa, que realça as estruturas diagonais. Em seguida, a SSM é binarizada e transformada em uma Matriz de Tempo-Deslocamento (MTD). A entrada i, j da MTD contém a similaridade entre a janela i e a janela $i + j$. Isso resulta na seguinte transformação: linhas paralelas à diagonal principal na SSM são convertidas em linhas horizontais na MTD.

A Figura 2.13 apresenta um exemplo de como a SSM pode ser transformada em uma matriz de deslocamento temporal. A Figura 2.13a contém uma SSM: é possível identificar que o segmento de início em tempo $t = 4$ se repete em $t = 6$ e $t = 8$. Na Figura 2.13b, as repetições são representadas por duas linhas horizontais, com início em tempo $t = 4$ e deslocamento $d = 2$ e $d = 4$. A região cinza da matriz não é utilizada.

2.5.4 Segmentação hierárquica

A maior parte dos algoritmos de SE têm como saída uma única segmentação, portanto não são capazes de representar as estruturas presentes em uma música de forma hierár-

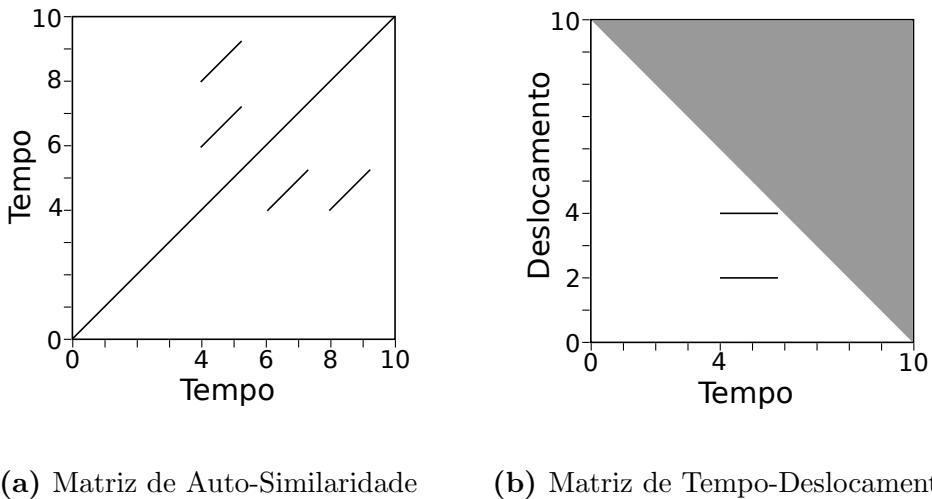


Figura 2.13: Exemplo de transformação de SSM para MTD.

quica (McFee et al., 2014). Isto é claramente uma deficiência da área, dado que podem existir mais de uma solução válida para a segmentação, com variações no nível de detalhe da análise. Este fato foi exemplificado no início desta seção, na Figura 2.11.

Recentemente, McFee et al. (2014) desenvolveram uma técnica de segmentação hierárquica de músicas, obtendo resultados competitivos com o estado da arte. O algoritmo segue a abordagem baseada em repetição, buscando por linhas paralelas à diagonal principal da SSM. O diferencial da técnica é o uso de teoria espectral de grafos para manipular a SSM.

Dado um vetor de características $X \in \mathbb{R}^{d \times n}$, a matriz de auto-similaridade R é calculada e binarizada, de modo que a Equação 2.27 seja satisfeita. O número de vizinhos k é definido como $1 + \lfloor 2 \log_2 n \rfloor$, onde n é a quantidade de batidas da música (previamente calculado com um algoritmo de *beat tracking*). Essa definição de SSM é similar à da matriz RP, definida na seção 2.4.

$$R_{ij} = \begin{cases} 1 & \text{se } x_i \text{ e } x_j \text{ são } k\text{-vizinhos mais próximos mutualmente} \\ 0 & \text{caso contrário} \end{cases} \quad (2.27)$$

R é então filtrada de modo a ressaltar as linhas diagonais. Um filtro moda janelado (maior número de votos) é aplicado à R , por meio de:

$$R'_{ij} = \text{moda}\{R_{i+t,j+t} | t \in -w, -w+1, \dots, w\} \quad (2.28)$$

R' pode ser interpretada como uma matriz de adjacência de um grafo não direcionado, onde arestas correspondem a sequências repetidas. Com uma normalização apropriada,

R' caracteriza um processo de Markov, onde R'_{ij} indica a probabilidade de um estado i ir para o estado j no próximo passo de um percurso aleatório no grafo.

A matriz de adjacência R' é então aumentada com a matriz Δ (Equação 2.29), por meio da Equação 2.30, para que vértices correspondendo a segmentos sucessivos $(i, i + 1)$ e $(i, i - 1)$ sejam conectados.

$$\Delta_{ij} = \begin{cases} 1 & \text{se } |i - j| = 1 \\ 0 & \text{caso contrário} \end{cases} \quad (2.29)$$

$$R^*_{ij} = \mu R'_{ij} + (1 - \mu) \Delta_{ij} \quad (2.30)$$

μ é o peso associado à ponderação que combina as matrizes R' e Δ , estimado de modo que em um percurso aleatório no grafo, um segmento i tenha a mesma probabilidade de ir para um segmento consecutivo $i + 1$, ou para um segmento similar aleatório no grafo. Isso é garantido por meio da seguinte restrição:

$$\mu \sum_j R'_{ij} \approx (1 - \mu) \sum_j \Delta_{ij} \quad (2.31)$$

Finalmente, com a R^* calculada, pode-se identificar os segmentos repetidos na música por meio de um agrupamento espectral. Sendo D a matriz diagonal contendo os graus de todos os vértices do grafo em R^* e I a matriz identidade, a Laplaciana L de R^* é dada por:

$$L = I - D^{-1/2} R^* D^{-1/2} \quad (2.32)$$

O agrupamento espectral é então calculado por meio da decomposição espectral da matriz L . Seja $Y \in \mathbb{R}^{n \times m}$ os auto-vetores associados aos m menores auto-valores de L , calcula-se o agrupamento *k-means* nas linhas de Y , com $k = m$. Mais especificamente, n é o número de janelas temporais da música e m é o número de segmentos que o algoritmo vai encontrar, desconsiderando repetições. Variando-se m , é possível controlar a granularidade da segmentação resultante, o que gera uma hierarquia de segmentações.

A hierarquia gerada pelo algoritmo é interessante para propósitos de visualização, onde o usuário pode estar interessado em identificar segmentos em diferentes níveis de detalhes. No Capítulo 4, uma visualização de coleções de músicas utilizando segmentação hierárquica é proposta.

2.6 Considerações finais

Neste capítulo, foram apresentados conceitos fundamentais sobre três tarefas da área de recuperação de informação musical: a extração de características de sinais de áudio, a comparação de séries temporais e a segmentação estrutural de músicas.

Após a recuperação de informação musical, as informações obtidas devem ser analisadas e interpretadas. A visualização de informações é uma importante ferramenta para análise, exploração e entendimento dos dados. No próximo capítulo, será dada continuidade à fundamentação teórica deste trabalho e os principais métodos de visualização de músicas da literatura serão descritos.

Visualização de músicas

A visualização de informações é uma ferramenta que permite a apresentação e a rápida interpretação de uma grande quantidade de dados. Além disso, possibilita a identificação de padrões e problemas no método de coleta de dados (Ware, 2013). No contexto de visualização de músicas, pode-se agrupar os métodos existentes com relação à quantidade de dados analisada: visualização de música individual e visualização de coleção de músicas. Neste capítulo, serão apresentados os principais trabalhos nessas duas vertentes.

3.1 Visualização de música individual

A visualização de composições musicais tem como objetivo facilitar a sua interpretação, oferecendo informações sobre diversas propriedades, por exemplo, tons, notas, acordes, harmonia e contexto.

A visualização *Piano Roll* é derivada dos rolos perfurados de programação de pianolas, pianos que permitem a execução automática de uma música gravada em um rolo de programação. Nesta representação, as notas são marcadas num plano 2D, onde o eixo Y representa a nota tocada e o eixo X, o tempo. A Figura 3.1 apresenta a música “With You Friends”, de Skrillex, com a representação de *piano roll*. O software utilizado para gerar a visualização foi o MIDITrail (Yknk, 2012).

No *piano roll*, pode-se visualizar as notas que serão tocadas, progressões e padrões no tempo. Entretanto, cabe ao usuário a interpretação da interação entre as notas e o seu contexto na música como um todo.

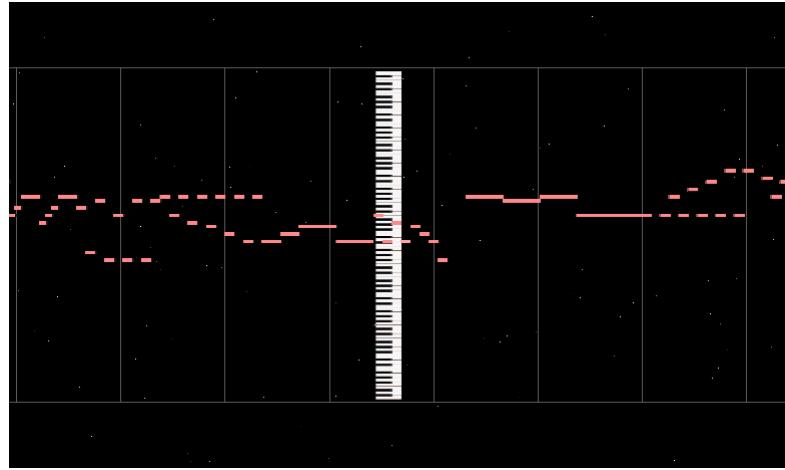


Figura 3.1: *Piano roll* da música “With You Friends”, por Skrillex (Yknk, 2012).

Malinowski (2013) desenvolveu uma variação para o *piano roll* original chamada *Music Annotation Machine*, em que cada faixa (instrumento) da música MIDI é representada por uma forma geométrica diferente e a cor indica a tonalidade (*chroma*). A Figura 3.2 apresenta a visualização gerada para a música “A Sagração da Primavera”, de Stravinsky. Para tornar a visualização mais agradável, foram utilizados efeitos de brilho, indicando as notas tocadas no momento.

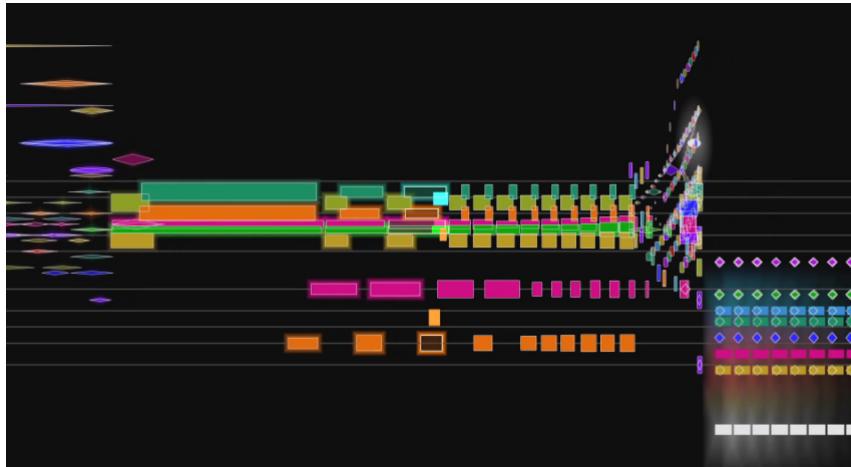


Figura 3.2: Visualização gerada pela *Music Annotation Machine* para a música “A Sagração da Primavera”, de Stravinsky (Malinowski, 2013).

Um conjunto de técnicas foram desenvolvidas para permitir a visão de uma música como um todo. Essas visualizações respeitam o princípio de foco + contexto e evitam exibições variantes no tempo. O princípio de foco + contexto foi definido por Ware (2013) como o problema de encontrar e explorar detalhes em um contexto maior. As principais visualizações dessa categoria são descritas a seguir.

Hiraga et al. (2002) utilizaram a técnica *fish eye* (Furnas, 1986) para criar a Partitura Condensada, uma notação reduzida de partituras inspirada no pentagrama musical (conjunto de cinco linhas horizontais que descrevem uma música). A representação utiliza barras verticais e variações de intensidade de cinza para transmitir uma ideia geral da composição. A Figura 3.3 ilustra a composição “Quinteto em Lá maior para clarinete, K. 581”, por W. A. Mozart, com a Partitura Condensada. A armadura da clave (parte inicial à esquerda) é preservada para prover informações de clave e tonalidade.

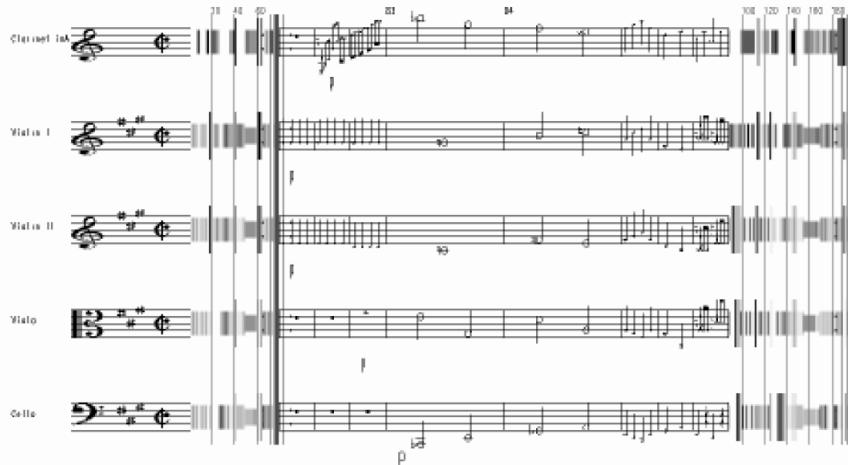


Figura 3.3: Partitura condensada da composição “Quinteto em Lá maior para clarinete, K. 581”, por W. A. Mozart (Hiraga et al., 2002).

A visualização *Shape of Song* (Wattenberg, 2002) busca por estruturas que se repetem em músicas MIDI, definindo um limiar de aceitação para correspondências quase perfeitas. A tarefa é realizada buscando-se sequências diretas de notas que aparecem em outras regiões, de forma similar a uma comparação de strings. Seções que se repetem são ligadas por um semi-círculo. A Figura 3.4 mostra uma execução do *software* para a música “All The Small Things”, da banda Blink 182.

A ferramenta *Infinite Jukebox* (Lamere, 2012) foi criada para visualizar estruturas de repetição dentro de um sinal de áudio. Ela permite que o usuário crie representações de músicas de uma base de dados pré-definida ou envie músicas para o sistema *online*. A API *Echonest* (Echonest, 2013) é utilizada para segmentar o sinal em tempos (batidas) e extrair o tom, timbre e altura de cada um dos segmentos de áudio. Assim como em *Shape of Song*, busca-se por segmentos que se repetem e representa-se essa estrutura como um grafo com *layout* circular. O aplicativo permite que se crie uma versão “infinita” da música por meio de um percurso aleatório nos caminhos descritos no grafo. A Figura 3.5 apresenta essa visualização. As cores dos nós representam o timbre da música na janela de áudio.



Figura 3.4: Visualização *Shape of Song* da música “All The Small Things”, por Blink 182 (Wattenberg, 2002).

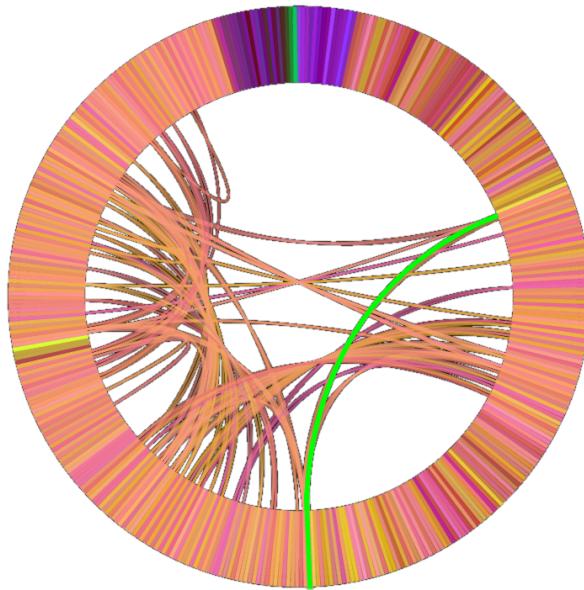


Figura 3.5: Visualização *Infinite Jukebox* da música “Lights”, por Ellie Goulding (Lamere, 2012).

A tabela 3.1 apresenta uma comparação de técnicas de visualização de composições musicais. As técnicas são classificadas de acordo com o tipo de dado analisado, metáfora visual e informações representadas: notas, timbre/fonte sonora, harmonia (combinação de notas) e contexto.

Tabela 3.1: Comparaçao de técnicas de visualização de composições musicais

Técnica	Tipo de dado	Metáfora visual	Notas	Timbre	Harmonia	Contexto
<i>Piano Roll</i>	Partitura digital	Fluxo temporal	✓		✓	
<i>Music Annotation Machine</i>	Partitura digital	Fluxo temporal	✓	✓	✓	
Partitura Condensada	Partitura digital	Pentagrama musical	✓		✓	✓
<i>Shape of Song</i>	Partitura digital	Grafo				✓
<i>Infinite Jukebox</i>	Sinal de áudio	Grafo		✓		✓

3.2 Visualização de coleções de músicas

Diversos trabalhos foram desenvolvidos no contexto de visualização de coleções de músicas, com o objetivo de facilitar a exploração de grandes bibliotecas de áudio. As técnicas apresentadas a seguir mapeiam o conjunto de músicas para o espaço visual e permitem a interação do usuário, a identificação de grupos e, em sua maioria, a criação de *playlists*.

O trabalho de Torrens et al. (2004) propõe a exploração de coleções de músicas por meio de meta-dados. Três visualizações foram desenvolvidas: disco (Figura 3.6a), retângulo (Figura 3.6b) e *treemap* (Figura 3.6c). As características utilizadas provém de meta-dados (*tags*) das músicas, contendo informações sobre o artista, compositor, ano, álbum, gênero, quantidade de reproduções e classificação.

A visualização disco é baseada nos gráficos de disco, que possibilitam a percepção de porcentagem e proporção. O disco é dividido em diferentes setores que são associados aos gêneros da biblioteca. O seu tamanho é proporcional à quantidade de músicas do gênero. Cada setor é dividido em sub-setores, que representam os artistas no determinado gênero. *Glyphs* para músicas são posicionados em seus respectivos setores e sub-setores.

Disco e retângulo têm basicamente as mesmas funcionalidades: permitem a exploração da coleção de músicas e a criação de *playlists*. A diferença está no formato de apresentação, que no retângulo consiste em colunas representando o gênero e linhas, os artistas.

A representação *treemap* apresenta três níveis de detalhe da biblioteca: gênero, sub-gênero artista. Entretanto, não permite a criação de *playlists*, pois músicas individuais não dispostas na tela.

Um segundo trabalho que permite visualizar, explorar e gerenciar um conjunto de músicas baseado em meta-dados é o *MusicNodes* (Dalhuijsen et al., 2010). A ferramenta possibilita que o usuário atribua uma porcentagem de semelhança a diferentes gêneros, por exemplo, 40% rock e 60% eletrônica, e crie novos meta-dados para classificar as músicas. Nesta representação, cada ponto simboliza um álbum, que é colorido de acordo com um mapeamento gênero-cor. Os álbuns são posicionados no espaço bidimensional por meio de um sistema baseado em forças: gêneros musicais atraem álbuns que pertencem

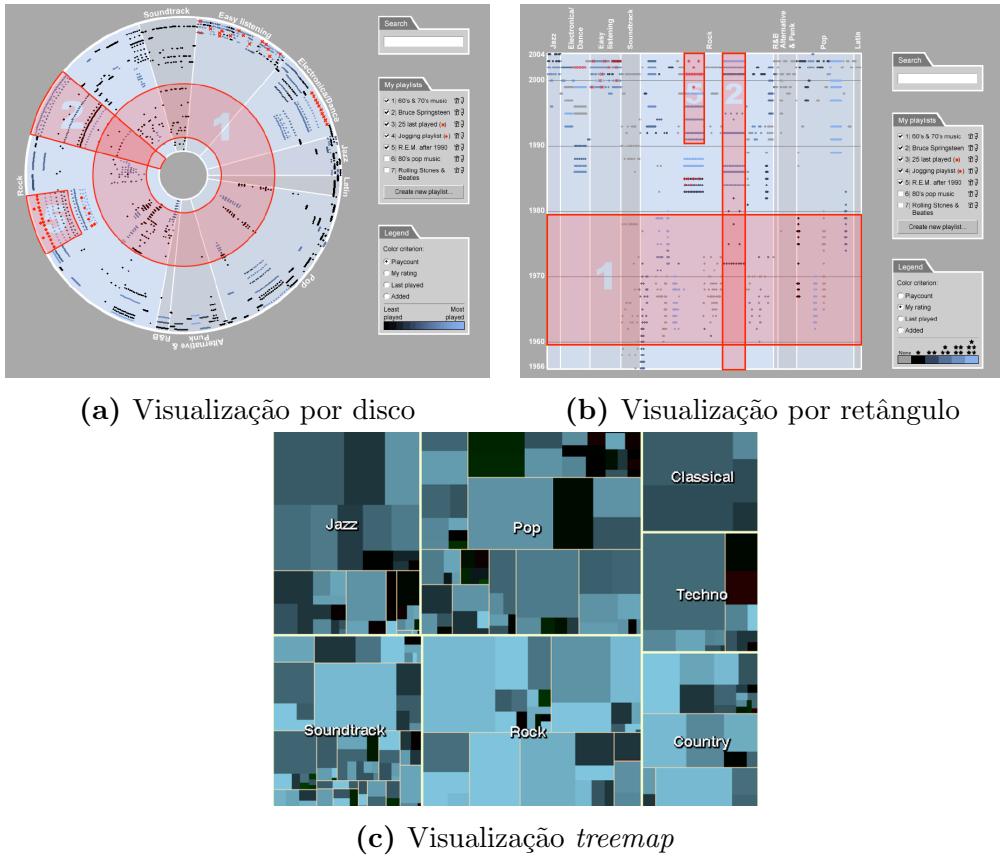


Figura 3.6: Visualização de coleções de músicas com meta-dados. Figura adaptada de (Torrens et al., 2004).

a sua classe, enquanto cada álbum afasta um pouco seus vizinhos. O sistema permite que usuários façam seleção e exportem *playlists*, por meio de buscas textuais e seleções visuais. A Figura 3.7 apresenta uma base de músicas com a ferramenta *MusicNodes*.

O trabalho de (Pampalk et al., 2002) propõe a visualização *Islands of Music*. Nela, não são considerados os meta-dados dos arquivos de áudio, mas sim o seu conteúdo. Cada música é avaliada por um *pipeline* de extração de características de ritmo que resulta em uma matriz de 20x60 coeficientes. As características do conjunto de músicas são levadas ao algoritmo *Self-Organizing Map* (SOM), que organiza a coleção de dados no espaço visual. A representação *Islands of Music* é gerada como um mapa de densidade dos pontos no espaço 2D. A Figura 3.8 ilustra a execução desta técnica em um conjunto de setenta e sete músicas.

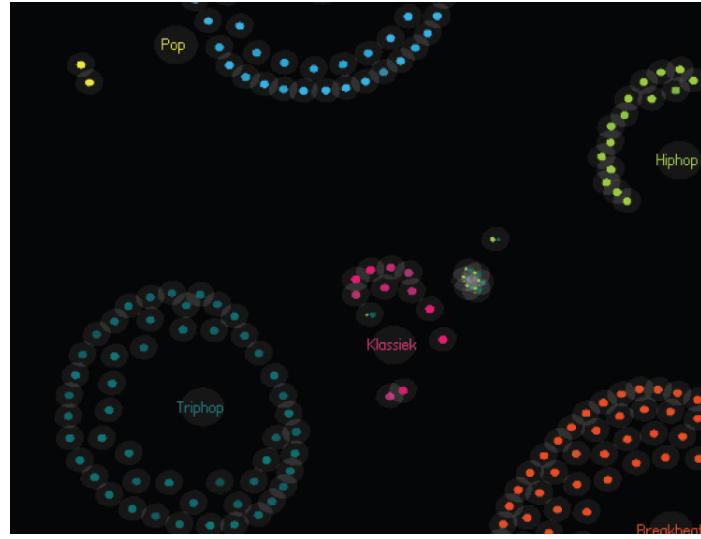


Figura 3.7: Base de músicas representada com *MusicNodes*. Figura adaptada de (Dalhuijsen et al., 2010).



Figura 3.8: Representação de um conjunto de músicas com *Islands of Music*. Figura adaptada de (Pampalk, 2001).

MusicBox (Lillie, 2008) também realiza o mapeamento de músicas mp3 no espaço bidimensional baseando-se em características extraídas do sinal. Primeiramente, descritores são extraídos do sinal de áudio (duração da música, tempo, timbre e histograma de ritmo) e de meta-dados de fontes online (popularidade, humor e gênero). Essas características são então mapeadas para o espaço 2D por meio da técnica *Principal Component Analysis* (PCA). A ferramenta possibilita a customização do *layout* gerado e a criação de *playlists* de forma visual. A Figura 3.9 apresenta um exemplo de execução do sistema.



Figura 3.9: Exemplo de execução da ferramenta *MusicBox*. Figura adaptada de (Lillie, 2008).

Muelder et al. (2010) desenvolveram uma técnica para visualizar bibliotecas de música com grafos. O trabalho calcula a média e o desvio padrão dos MFCCs (*Mel Frequency Cepstrum Coefficients*) de cada música e usa essas características para medir a similaridade entre os sinais de áudio com a distância euclidiana. No grafo proposto, cada música é representada por um nó e as relações entre músicas, por arestas ponderadas. Os nós são posicionados no espaço bidimensional por meio de uma técnica de mapeamento descrita em (Muelder et al., 2008). O algoritmo, baseado em curvas de preenchimento de espaço, evita sobreposições e possibilita que as capas dos álbuns sejam utilizadas como *glyphs*. É estabelecida uma relação entre a transparência das arestas e o grau de similaridade entre músicas. A Figura 3.10 apresenta um exemplo da execução desse algoritmo com uma coleção de músicas da banda The Beatles.

Por fim, o trabalho de Paulovich et al. (2011) propõe uma técnica de projeção multidimensional local denominada *Piecewise Laplacian-based Projection* (PLP) e a aplica no contexto de visualização de bases de música. Um vetor de 78 características é extraído para cada uma das 3.857 músicas da base de dados, por meio da ferramenta *JAudio* (McKay et al., 2005). Entre as características utilizadas, estão medidas estatísticas, histograma de batidas e análises espectrais. Por fim, a dimensionalidade dos dados é reduzida de 78 para 2, por meio da PLP. A Figura 3.11 ilustra um exemplo de projeção de bases de música com PLP. Seleções podem ser feitas (na figura, em verde e rosa) para organização e criação de *playlists*.

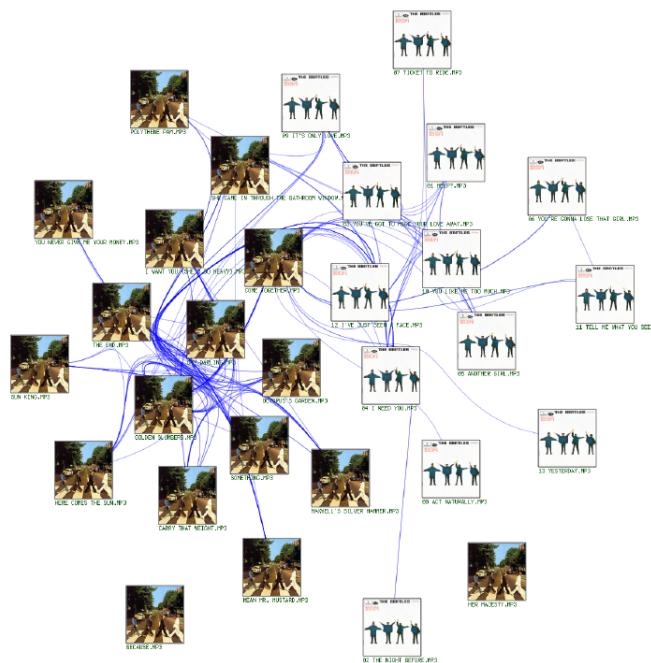


Figura 3.10: Organização de uma base de dados da banda The Beatles com curvas de preenchimento de espaço. Figura adaptada de (Muelder et al., 2010).

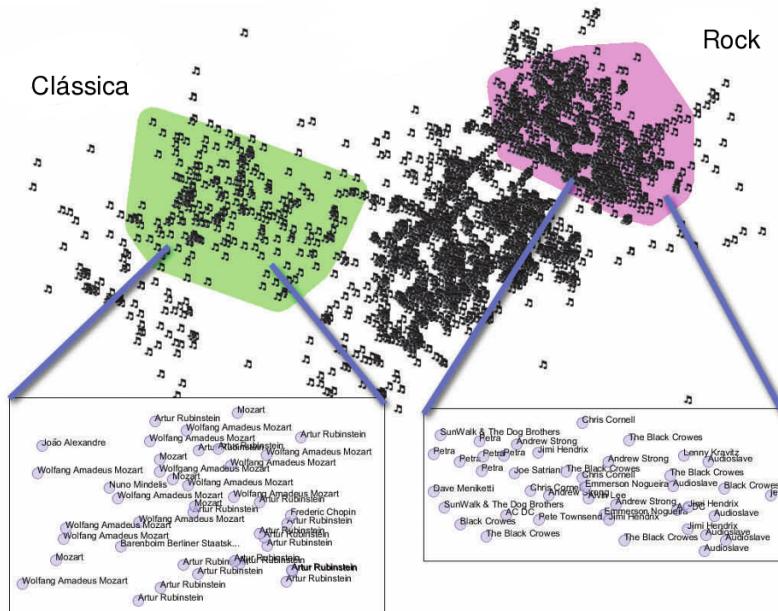


Figura 3.11: Sistema de criação de Playlists com a projeção multidimensional PLP. Figura adaptada de (Paulovich et al., 2011).

A tabela 3.2 apresenta uma comparação de técnicas de visualização de coleções de músicas. As técnicas são classificadas quanto ao tipo de dado analisado, metáfora visual

empregada e informação apresentada: visualização de gênero ou informação de similaridade.

Tabela 3.2: Comparação de técnicas de visualização de coleções de música

Técnica	Tipo de dado	Metáfora visual	Gênero	Similaridade
Torreens et al. (2004)	<i>Tags</i>	Subdivisão do Espaço	✓	
<i>Music Nodes</i>	<i>Tags + Usuário</i>	Modelo de forças	✓	
<i>Islands of Music</i>	Características	Mapa de Densidade		✓
<i>MusicBox</i>	Características	Projeção PCA		✓
Muelder et. al (2010)	Características	Grafos + <i>Edge Bundle</i>		✓
Paulovich et. al (2011)	Características	Projeção PLP		✓

3.3 Considerações finais

Neste capítulo, foram descritos os principais trabalhos sobre visualização de composições musicais e de coleções de músicas.

A maioria das técnicas de visualização de coleções de músicas utilizam uma métrica de dissimilaridade entre interpretações e mapeiam os dados na tela, por meio uma técnica de redução de dimensionalidade. Entretanto, a análise é feita de modo superficial, não apresentando ao usuário informações mais detalhadas, por exemplo, onde as similaridades ocorreram. No Capítulo 4, a visualização Grafo de Similaridades é proposta. Nela, são apresentadas informações de similaridade entre pares de músicas de forma hierárquica, isto é, o usuário consegue enxergar não apenas os grupos de músicas similares na base, como também a posição em que as similaridades aconteceram.

Uma segunda deficiência das técnicas de visualização de bases de músicas apresentadas é que elas fazem, em geral, o mapeamento das composições para o espaço visual levando em consideração apenas dois aspectos: similaridades par-a-par e o gênero musical. A técnica RadViz Concêntrico, proposta no Capítulo 5, tenta suprir essa deficiência. Com ela, o usuário pode explorar a base de músicas por meio de diferentes pontos de vista, entre eles, gênero musical, humor e “danceabilidade”, possibilitando uma análise visual mais completa e abrangente do conteúdo.

Grafo de Similaridades

No Capítulo 3, foram apresentadas duas categorias de visualizações musicais: visualização de música individual, em que propriedades de uma única peça são estudadas, e visualização de coleções de músicas, cujo objetivo é apresentar as relações de similaridade entre composições graficamente.

A visualização de músicas proposta neste capítulo pertence às duas categorias simultaneamente: com o Grafo de Similaridades (GS), uma base de músicas é representada de forma visual por meio de *glyphs*. Entretanto, ao contrário de outras visualizações de coleções musicais presentes na literatura, detalhes de cada música também são representados individualmente. Mais especificamente, a visualização proposta apresenta ao usuário não apenas quão similares as músicas são, por meio relações de proximidade, mas também quais trechos de cada música podem ser encontrados em outras composições. A construção da visualização é feita por meio de uma analogia com grafos, em que músicas são representadas por nós e similaridades entre trechos, por arestas.

As funcionalidades presentes na visualização proposta podem ser úteis para diversas tarefas, entre elas, identificação de músicas *cover*, identificação de plágio (mesmo em pequenos trechos) e *mixagem* (combinação) de músicas por DJs. O GS é, portanto, uma ferramenta importante não apenas para usuários sem conhecimento musical, que desejam realizar tarefas simples, como organizar suas bases de dados e criar *playlists*, mas também para especialistas, cujo objetivo é entender e analisar a base de dados em diferentes níveis de detalhe.

A técnica proposta foi aplicada a duas bases de dados: *Covers80* (Ellis, 2007) e *CoversYoutube*. *Covers80* é uma coleção de músicas *cover* que contém 80 músicas, cada uma interpretada por dois artistas. *CoversYoutube* foi construída no contexto do presente trabalho e consiste em um conjunto de músicas *cover* disponíveis no site de compartilhamento de vídeos *youtube*¹. Contém trinta composições, cada uma com cinco interpretações diferentes (versões *cover*), totalizando 150 músicas. Os nomes e intérpretes das músicas estão disponíveis no Apêndice A. As trinta composições escolhidas pertencem a seis gêneros musicais, *Country*, Eletrônica, *Folk*, Pop, R&B e Rock, sendo que cada gênero é representado por cinco músicas na base.

Este capítulo está organizado da seguinte maneira: primeiramente, é apresentada uma breve revisão sobre duas interpolações polinomiais, empregadas na visualização proposta. Em seguida, a metodologia para a criação do Grafo de Similaridades é descrita. Por fim, a técnica é utilizada para visualizar duas coleções de músicas.

4.1 Conceitos básicos sobre interpolações polinomiais para o desenho de curvas

Nesta seção, será realizada uma breve revisão sobre duas interpolações polinomiais para o desenho de curvas, a interpolação *spline* Bézier e a interpolação *spline* Hermite. Ambas serão utilizadas para o desenho do Grafo de Similaridades proposto neste capítulo.

A interpolação Bézier está amplamente disponível em pacotes gráficos, graças a sua simplicidade para especificar curvas e facilidade de implementação. Dados $n + 1$ pontos de controle x_k , com $k = 0 \dots n$, a interpolação Bézier descreve uma curva com início em x_0 e fim em x_n , guiada pelos pontos de controle $x_1 \dots x_{n-1}$ (Hearn et al., 1997). A figura 4.1 ilustra uma interpolação Bézier quadrática com três pontos de controle: *A*, *B* e *C*.

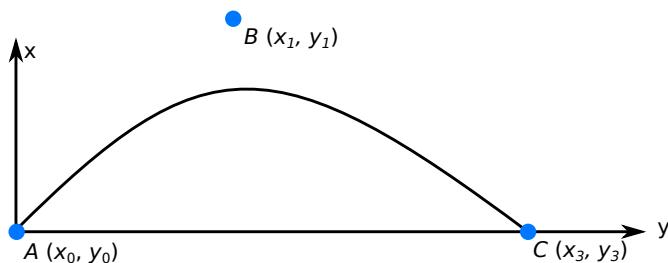


Figura 4.1: Curva gerada por uma interpolação Bézier quadrática. Figura adaptada de (Hearn et al., 1997).

¹<http://youtube.com>

No caso bidimensional, $x \in \mathbb{R}^2$, $x = (x, y)$, a curva de Bézier é descrita por:

$$\begin{aligned} x(t) &= \sum_{k=0}^n x_k \frac{n!}{k!(n-k)!} t^k (1-t)^{n-k} \\ y(t) &= \sum_{k=0}^n y_k \frac{n!}{k!(n-k)!} t^k (1-t)^{n-k} \end{aligned} \quad (4.1)$$

, em que t é o parâmetro para a geração da curva ($0 \leq t \leq 1$).

Hermite é uma interpolação polinomial que define um polinômio $p(x)$, de grau $2n+1$ ou menor, com valores pré-definidos em $p(x_k)$ e derivadas $p'(x_k)$, sendo x_k , $k = 0 \dots n$, um conjunto de $n+1$ nós (pontos de controle). Uma formulação mais geral para essa interpolação permite que sejam definidos valores para $p(x)$, $p'(x)$, $p''(x)$, etc. (Kreyszig, 2011).

Para o caso de bidimensional, $x = (x, y)$ e $n = 1$, a interpolação Hermite cúbica especifica uma curva no plano cujos pontos inicial e final têm posições e tangentes pré-definidas. A Figura 4.2 ilustra esse processo: dados os pontos inicial ($A = (x_0, y_0)$), final ($B = (x_1, y_1)$) e vetores tangente ($A' = (x'_0, y'_0)$ e $B' = (x'_1, y'_1)$, respectivamente), define-se uma curva que respeita as restrições impostas.

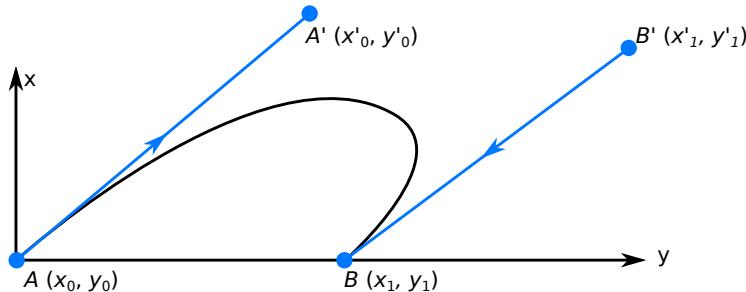


Figura 4.2: Curva gerada por uma interpolação Hermite cúbica. Figura adaptada de (Kreyszig, 2011).

As coordenadas da curva no plano são dadas por:

$$\begin{aligned} x(t) &= x_0 + x'_0 t + (3(x_1 - x_0) - (2x'_0 + x'_1))t^2 + (2(x_0 - x_1) + x'_0 + x'_1)t^3 \\ y(t) &= y_0 + y'_0 t + (3(y_1 - y_0) - (2y'_0 + y'_1))t^2 + (2(y_0 - y_1) + y'_0 + y'_1)t^3 \end{aligned} \quad (4.2)$$

, em que t é o parâmetro da interpolação ($0 \leq t \leq 1$). Assim como na interpolação de Bézier, a curva pode ser obtida incrementando-se t em intervalos regulares e conectando-se os pontos resultantes por meio de uma poli-linha (Kreyszig, 2011).

4.2 Método

A metodologia para a criação do Grafo de Similaridades é apresentada graficamente na Figura 4.3. Em termos gerais, pode-se dividir o método em duas partes: pré-processamento e visualização da base. As duas etapas serão descritas a seguir.

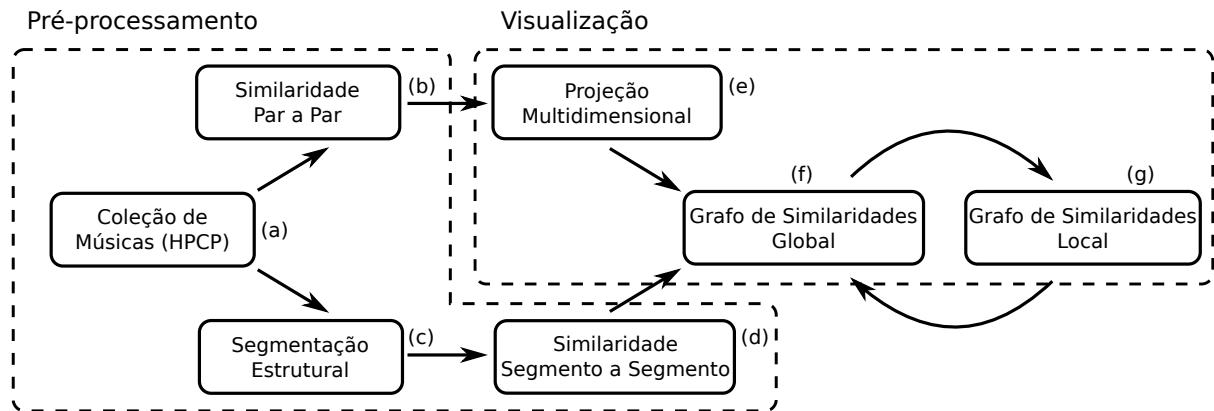


Figura 4.3: Metodologia para a criação do Grafo de Similaridades.

4.2.1 Pré-processamento da base de dados

Dada uma coleção de músicas, (a) são extraídas características tonais dos sinais. Mais especificamente, para cada música, é extraído um vetor HPCP, como descrito na Subseção 2.3.2. Em seguida, (b) a similaridade de *chroma* entre pares de músicas Q_{max} é calculada, por meio do uso de *Cross Recurrence Plots* (Serrà et al., 2009), como visto na Subseção 2.4.2. Paralelamente, (c) as músicas devem ser segmentadas em trechos com significado musical, de modo que esses trechos possam ser comparados entre si. O algoritmo descrito por McFee et al. (2014), visto na Subseção 2.5.4, é utilizado, possibilitando a segmentação musical de forma hierárquica. Finalmente, (d) as similaridades entre todos os pares de trechos musicais são calculadas, também por meio do uso de *Cross Recurrence Plots*.

Note que a maioria dos métodos de redução de dimensionalidade, utilizados na etapa de visualização, têm como entrada uma matriz de dissimilaridade. Neste trabalho, a dissimilaridade D entre duas músicas, ou trechos de músicas, foi calculada por meio de:

$$D = \frac{1}{Q_{max} + 1} \quad (4.3)$$

Como $Q_{max} \geq 0$, $0 < D \leq 1$. Quanto maior o D , mais dissimilares são as duas músicas.

4.2.2 Visualização da base de dados

A fim de visualizar a base de dados como um todo, representando as relações de similaridade entre músicas por meio de distâncias no espaço visual, as músicas devem ser posicionadas no plano. Portanto, uma técnica de redução de dimensionalidade (no caso do mapeamento em duas ou três dimensões, também conhecida como projeção multidimensional) deve ser utilizada. Uma característica desejável para a técnica escolhida é que o mapeamento mantenha as relações de vizinhança na base de dados, isto é, se uma instância p é vizinha de q no espaço original, o ponto mapeado p' também deve ser vizinho de q' no espaço de dimensão reduzida (\mathbb{R}^2).

De modo geral, há um consenso na literatura de que a técnica *t-Distributed Stochastic Neighbor Embedding* (t-SNE) produz um dos melhores mapeamentos em termos de preservação de vizinhanças e segregação de grupos no espaço visual (Ingram et al., 2015; Fadel et al., 2015). Foram realizados testes com seis técnicas de projeção multidimensional, MDS (Torgerson, 1952), Isomap (Tenenbaum et al., 2000), Force Scheme (Tejada et al., 2003), LSP (Paulovich et al., 2008), LAMP (Joia et al., 2011) e t-SNE (Maaten et al., 2008) e, como esperado, os melhores resultados de projeção foram obtidos com a t-SNE. Adotou-se o parâmetro “número de vizinhos”=15 para o cálculo das projeções Isomap, LSP e t-SNE.

A Figura 4.4 ilustra a aplicação das técnicas na base *CoversYoutube* e a métrica de dissimilaridade baseada em Q_{max} . Observe que t-SNE é a única técnica que consegue mapear as músicas de forma que *covers* fiquem próximas. A medida Preservação de Vizinhança (PV), definida como a porcentagem dos k vizinhos mais próximos no espaço original que permanecem vizinhos no espaço projetado (Joia et al., 2011), também é apresentada na figura. Por seus resultados superiores, a t-SNE foi escolhida para realizar a projeção na visualização proposta (Figura 4.3 (e)).

t-SNE é uma técnica de redução de dimensionalidade baseada em probabilidades que tem como objetivo posicionar dados multidimensionais no espaço bidimensional, preservando tanto estruturas locais quanto globais e evitando o *clutter* visual de pontos, isto é, evitando que pontos sejam mapeados muito próximos uns dos outros. Nessa técnica, as similaridades entre pares de instâncias no espaço original são modeladas como uma distribuição de probabilidades *t-Student*. Mais especificamente, quanto mais similares dois elementos forem, maior será a probabilidade associada a eles. Do mesmo modo, as distâncias entre pares de pontos projetados também são modeladas como uma distribuição de probabilidades. O mapeamento dos pontos em duas dimensões é feito por meio de uma otimização por gradiente descendente, que minimiza a divergência Kullback-Leibler entre as duas distribuições de probabilidade calculadas (Maaten et al., 2008).

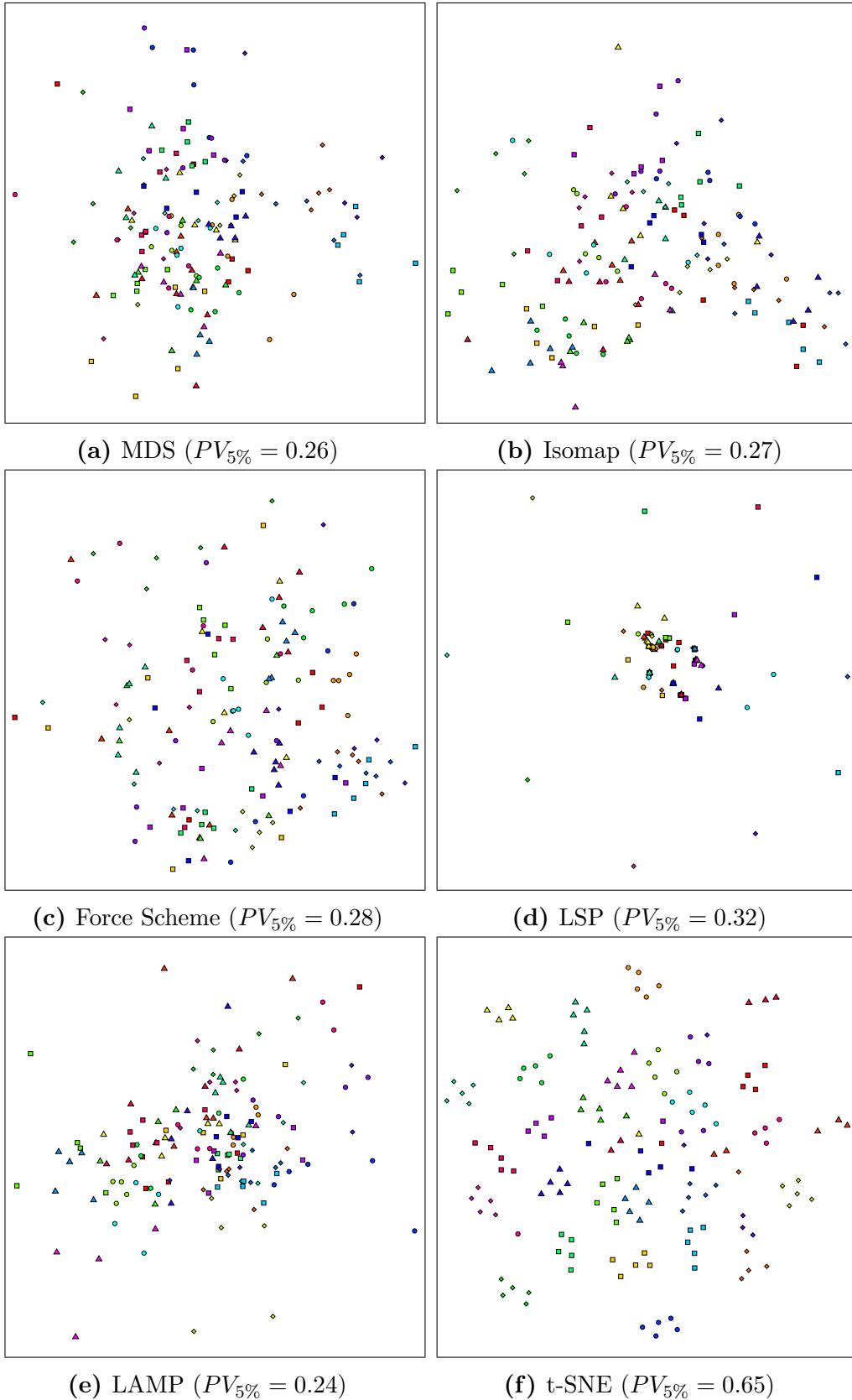


Figura 4.4: Visualização da coleção de músicas *Covers Youtube* com seis técnicas de redução de dimensionalidade. O par [forma,cor] representa uma classe de música (conjunto de cinco músicas *cover*). Considera-se 5% dos vizinhos mais próximos para o cálculo da PV.

De posse da projeção das músicas no espaço visual e das similaridades entre pares de segmentos musicais, o Grafo de Similaridades Global (GSG) é construído (Figura 4.3 (f)). Primeiramente, cada música é representada por um retângulo, de altura maior que largura. A posição do retângulo na tela é dado pela projeção t-SNE. O topo do retângulo representa o começo da música, e a base, o final da mesma.

Em seguida, os segmentos similares entre todas as músicas e seus k vizinhos mais próximos são conectados por meio de uma interpolação Hermite cúbica. Mais especificamente, a posição de início do segmento S_i é conectado à posição de início do segmento S_j e a posição de término do segmento S_i é conectado à posição de término do segmento S_j , se S_i e S_j forem similares. São conectados apenas os k vizinhos mais próximos para evitar poluição visual de arestas na visualização. Neste trabalho, adotou-se $k = 10$.

Considera-se que um segmento é similar a outro se a medida Q_{max} normalizada (Q'_{max}) for maior que uma constante h . Dados l_i o tamanho do segmento S_i e l_j o tamanho do segmento S_j , Q'_{max} é calculada por meio da equação:

$$Q'_{max} = \frac{Q_{max}}{\min(l_i, l_j) - 2} \quad (4.4)$$

Com a normalização, $0 \leq Q'_{max} \leq 1$. Neste trabalho, foi adotado $h = 0.8$. Como o tamanho dos segmentos podem variar muito (relembre que é feita uma segmentação hierárquica), impõe-se mais uma restrição para que dois segmentos sejam semelhantes: $l_i \leq 2l_j$ e $l_j \leq 2l_i$.

As tangentes associadas às posições de início e término dos segmentos (Hermite cúbica) são vetores de direção horizontal, isto é, sua componente $y = 0$. A Figura 4.5 mostra um esquema de como o sentido e o módulo dos vetores são calculados. O sentido dos vetores é dado pelo sentido da conexão, isto é, se a linha conectará o retângulo pela esquerda ou pela direita. O módulo dos vetores é dado pelo tempo, em segundos, em que ocorre o início da similaridade na caixa mais alta. Quanto mais próximo de 0 segundos, maior será o módulo do vetor. Note que reflexões verticais ou horizontais das duas situações mostradas na Figura 4.5 podem ocorrer.

GS é uma visualização interativa, sendo que o usuário pode escolher um conjunto de músicas para explorar com mais atenção. A interação acontece da seguinte maneira: seleciona-se uma pequena região do GSG para ser visualizada com mais detalhes. As músicas selecionadas serão então representadas por meio de arcos em um grafo em formato circular, chamado de Grafo de Similaridade Local (GSL), etapa (g) da metodologia apresentada na Figura 4.3. Essa visualização circular conecta os trechos parecidos entre músicas de maneira similar ao grafo global. Entretanto, não utiliza curvas Hermite para

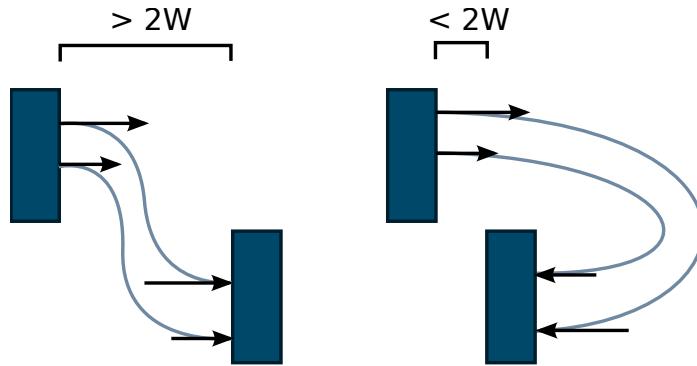


Figura 4.5: Esquema de curvas Hermite conectando duas músicas no Grafo de Similaridade Global. Duas situações podem ocorrer: Se a distância entre os retângulos for maior que duas vezes a largura da caixa (W), a conexão é feita como na figura da esquerda. Caso contrário, conecta-se as caixas como na figura da direita.

representar similaridades, mas sim curvas Bézier quadráticas. Mais especificamente, são utilizadas duas curvas, a primeira para conectar o início do segmento S_i ao início do segmento S_j e a segunda para conectar o término dos dois segmentos. São utilizados três pontos de controle para definir a curva: o primeiro e o terceiro são os pontos de início e término da similaridade. O segundo ponto de controle é o centro do círculo, o que distorce as curvas na direção da origem da visualização. A Figura 4.6 ilustra como as conexões acontecem.

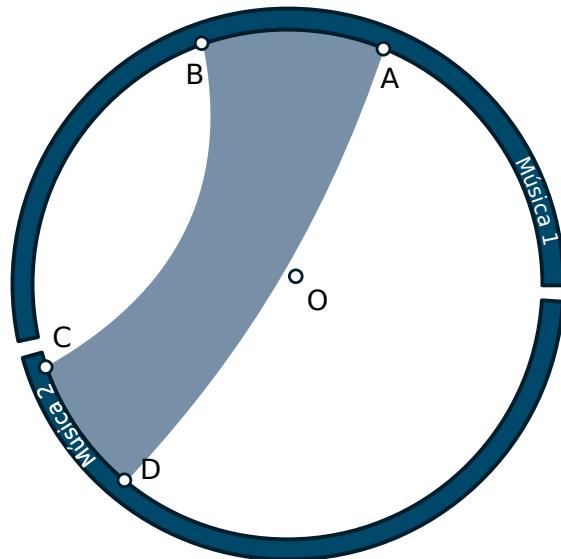


Figura 4.6: Esquema de curvas Bézier conectando duas músicas no Grafo de Similaridade Local. O trecho da Música 1 começa em A e termina em B. O trecho da Música 2 começa em C e termina em D.

Nesta visualização, o usuário pode reproduzir trechos das músicas em destaque, de modo a verificar e analisar regiões similares e dissimilares. Note que o cálculo da medida de similaridade Q_{max} é invariante a transposições, portanto os trechos não necessariamente estão na mesma tonalidade. Relembre que transposições correspondem a rotações no vetor HPCP.

Um segundo mecanismo de interação facilita a identificação de segmentos semelhantes: enquanto o usuário escuta um trecho de música, todas as regiões similares a esse trecho serão destacados com uma cor mais intensa. Assim, mesmo que diversas similaridades ocorram, resultando em muitos cruzamentos de arestas no grafo, o usuário conseguirá enxergar as similaridades do segmento que está sendo reproduzido.

A exploração da base de dados é, portanto, realizada de forma cíclica: o usuário pode visualizar a base de dados como um todo na visualização global (GSG) e explorar regiões de interesse na visualização local (GSL). Esse ciclo de interações é representado na Figura 4.3, ítems (f) e (g).

4.3 Aplicação

Um protótipo da técnica de visualização foi desenvolvido na linguagem C++ com a biblioteca QT². A etapa de pré-processamento dos dados foi implementada com a linguagem R, sendo que o cálculo do HPCP foi realizado com o código da biblioteca Essentia (Bogdanov et al., 2013) e a projeção dos pontos no espaço visual, com a implementação da t-SNE do pacote “tsne”³.

A Figura 4.7 apresenta o Grafo de Similaridades Global da *CoversYoutube*. As cores dos retângulos representam as classes de música, isto é, grupos de músicas *cover* são representados por retângulos de mesma cor. Observe que as músicas *cover* são posicionadas muito próximas umas das outras. Isso é uma forte indicação de que a métrica de comparação de músicas funciona corretamente e que a técnica de projeção multidimensional mapeia os dados respeitando as relações de vizinhança na base.

É possível notar também que podem existir várias conexões entre pares de músicas não *covers*. Isto ocorre porque músicas populares de mesmo gênero compartilham de progressões de acordes similares. Esse fato é evidenciado no trabalho de Anglade et al. (2009), em que uma árvore de decisão foi utilizada para classificar o gênero de músicas com base em partituras digitais, isto é, progressões de notas e acordes, obtendo acurácia de 66%.

²<http://www.qt.io/>

³<http://cran.r-project.org/package=tsne>

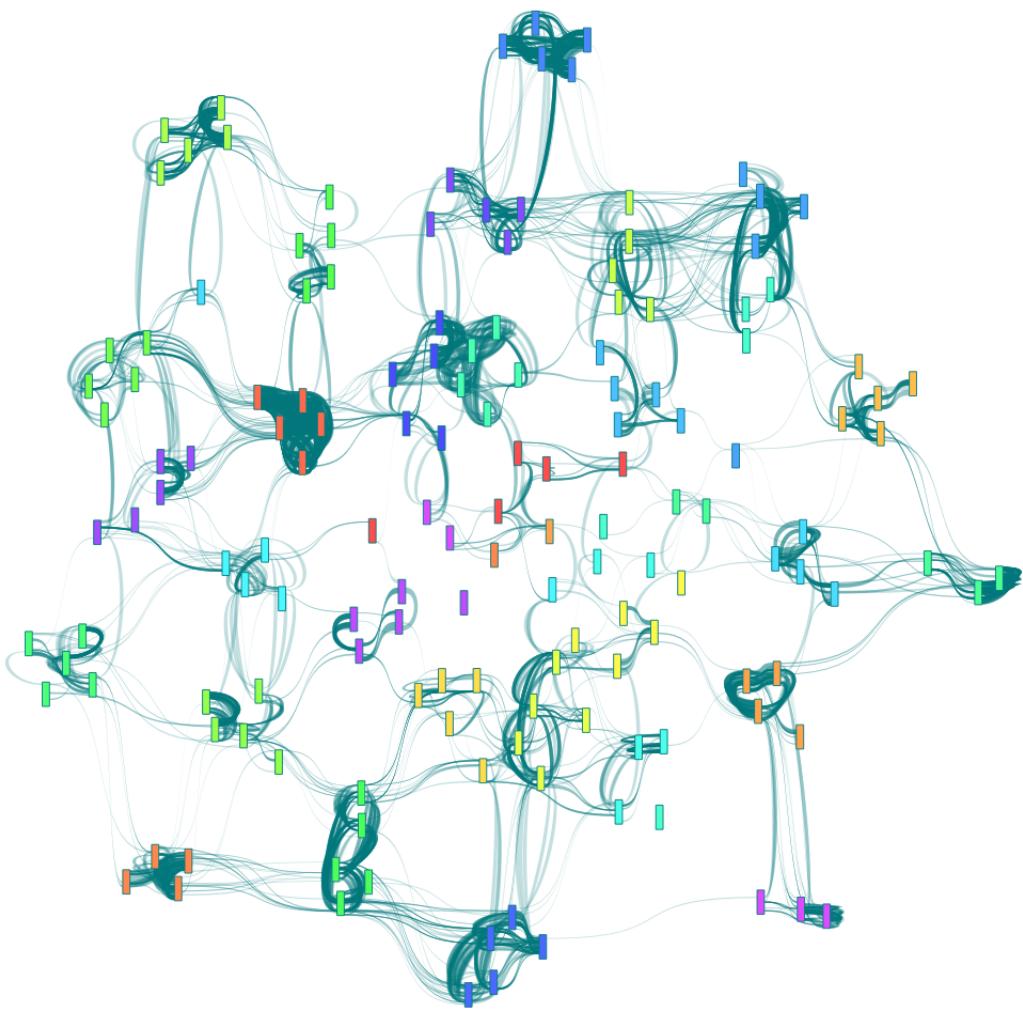


Figura 4.7: Visualização da base *CoversYoutube* com o Grafo de Similaridades Global.

No protótipo desenvolvido, o usuário pode selecionar um conjunto de músicas para ser explorado por meio do Grafo de Similaridades Local. A Figura 4.8 ilustra essa operação.

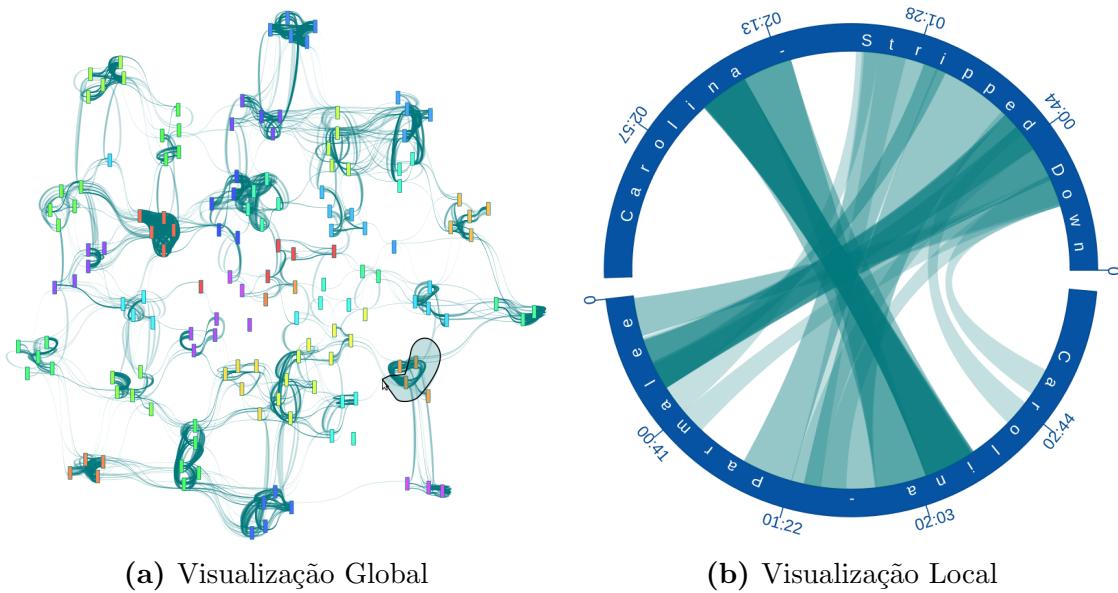


Figura 4.8: Exploração da base de dados com o Grafo de Similaridades Local e Global. O usuário selecionou duas músicas na visualização (a) para explorar com mais detalhes em (b). Base de dados *CoversYoutube*.

Como descrito na Sessão 4.2, o usuário pode reproduzir trechos de músicas na visualização local. Conforme a música progride, as regiões de repetição são destacadas, como é mostrado na Figura 4.9. Mais de duas músicas podem ser exploradas simultaneamente com o GSL. Entretanto, conforme arestas de mais músicas são adicionadas, torna-se difícil identificar onde começam e terminam os trechos similares. A Figura 4.10 ilustra esse cenário.

A Figura 4.11 apresenta o GSG da base de dados *Covers80*. Mais uma vez, músicas *cover* ficaram próximas na projeção bidimensional. Entretanto, como existem apenas duas versões de cada música, não foram criados *clusters* de músicas *cover*, como na visualização da base *CoversYoutube* (Figura 4.7).

Por fim, a Figura 4.12 apresenta duas versões da música “Let it be”, de The Beatles, com o GSL. Nota-se, novamente, que o mecanismo de interatividade proposto facilita a interpretação do grafo.

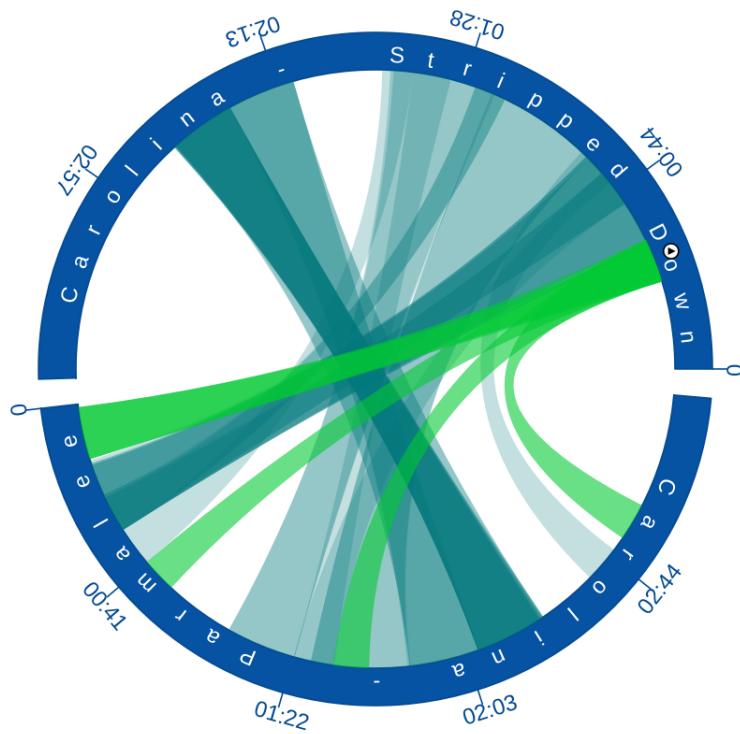


Figura 4.9: Reprodução de um segmento na visualização local. Os trechos similares à região reproduzida estão destacados em verde claro. Base de dados *CoversYoutube*.

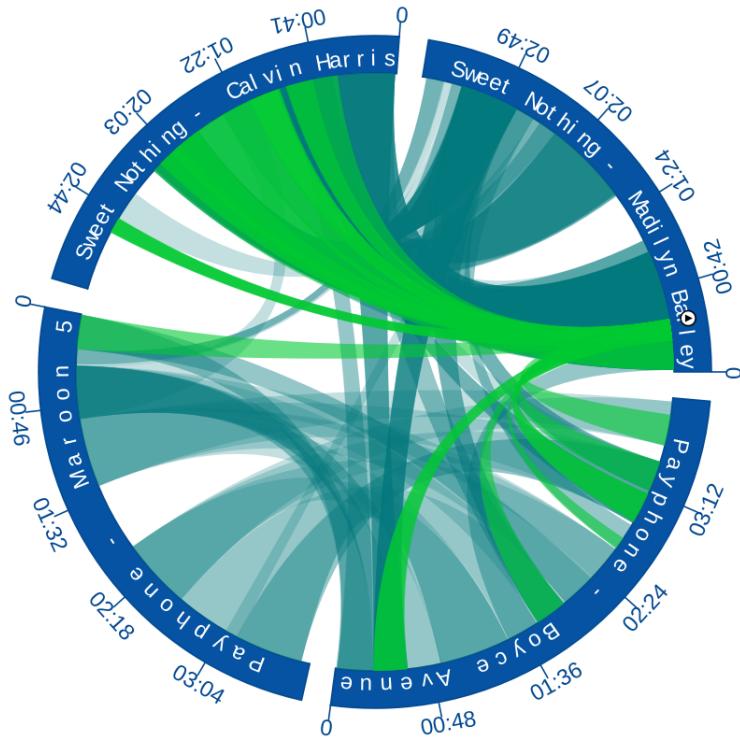


Figura 4.10: Visualização de quatro músicas com o Grafo de Similaridades Local. Base de dados *CoversYoutube*.

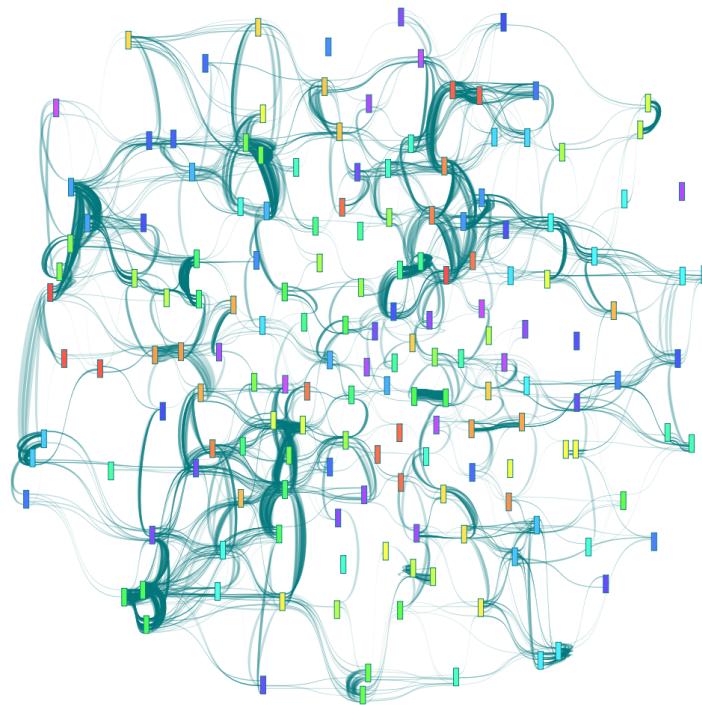


Figura 4.11: Visualização da base de dados *Covers80* com o Grafo de Similaridades Global.

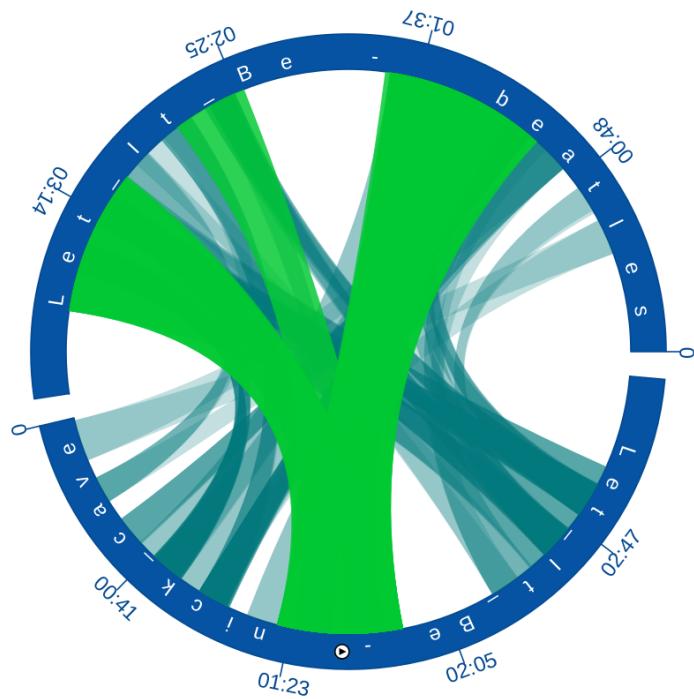


Figura 4.12: Visualização de duas versões de “Let it be” com o Grafo de Similaridades Local. Base de dados *Covers80*.

4.4 Considerações finais

Neste capítulo, uma nova visualização musical foi proposta. O Grafo de Similaridades é capaz de apresentar informações sobre trechos de uma música individual e informações de similaridade sobre a base de dados como um todo. Combinadas, essas duas funcionalidades podem ser muito úteis ao entendimento e exploração da coleção musical, facilitando a interação com o conjunto de músicas e permitindo a identificação de trechos parecidos em diferentes níveis de detalhes, tanto a partir das relações de proximidade entre as músicas projetadas quanto por meio das conexões do grafo.

A validação desta técnica de visualização é uma tarefa complicada, pois não existe na literatura bases de dados com anotações de similaridade entre trechos de pares de músicas. Construir tal base foge do contexto deste mestrado, pois, a fim de identificar as semelhanças de trechos de n músicas, seria necessário anotar $n(n - 1)/2$ pares de músicas. Além disso, essa tarefa requereria um conhecimento musical muito profundo, dado que o especialista teria que identificar progressões de acordes similares em tonalidades diferentes.

Portanto, defende-se a validade da técnica proposta da seguinte forma: em termos gerais, o Grafo de Similaridades segue uma metodologia que compara pares de músicas e seus segmentos para gerar uma visualização de grafo que conecta trechos musicais semelhantes. Os estados da arte em segmentação estrutural e comparação de músicas foram utilizados. O mapeamento dos dados para o espaço visual é realizado com uma das melhores técnicas de projeção multidimensional em termos de preservação de vizinhança e a representação de segmentos semelhantes é um mapeamento direto dos dados para a tela. Com base no fato de que as partes constituintes da visualização funcionam corretamente, a técnica proposta é capaz de representar graficamente as similaridades presentes na base de músicas.

RadViz Concêntrico

Layouts baseados em projeção são fundamentais para a área de pesquisa de visualização de dados multidimensionais. A principal razão para o crescimento do interesse nestes mecanismos é que usuários podem identificar grupos, padrões e tendências nos dados baseados somente na disposição dos pontos na tela (Kovalerchuk et al., 2014). Sistemas de visualização atuais combinam métodos de projeção com outros recursos matemáticos e computacionais para melhorar a capacidade de ferramentas de análise visual e revelar informações contidas nos dados. Nesse contexto, houve muito esforço para combinar técnicas de projeção multidimensional com ferramentas de aprendizado de máquina, principalmente em trabalhos relacionados ao aprendizado ativo (McCarthy et al., 2004).

Por exemplo, Teoh et al. (2003) desenvolveram *PaintingClass*, um sistema que projeta dados multidimensionais no espaço visual, disponibilizando recursos interativos para os usuários direcionarem a construção de uma árvore de decisão, de onde é possível descobrir informações importantes sobre os dados. Geng et al. (2005) propuseram Isomap Supervisionado, uma técnica que leva em consideração as informações de classe para definir a métrica de distância usada no processo de projeção, resultando em *layouts* com boa separabilidade de classes. Em um trabalho mais recente, Seifert et al. (2010) adotaram RadViz (Hoffman et al., 1997) para visualizar incertezas no resultado de classificadores, o que facilitou o processo de aprendizado ativo: no sistema desenvolvido, os usuários selecionam pontos para serem etiquetados de acordo com sua disposição.

Uma característica comum de todos os métodos citados é que classificadores são utilizados para auxiliar no processo de análise visual de dados. Mais especificamente, o classificador atribui um rótulo para cada instância. Entretanto, em muitas aplicações, as instâncias de dados estar associadas a mais de uma classe simultaneamente, o que é conhecido na literatura como problema *multi-label*. Por exemplo, um filme pode ser do gênero “comédia” e “romance” (Read et al., 2011), assim como documentos textuais podem ser classificados como “evidência histórica” e “artigo científico” (Schapire et al., 2000). Um problema muito parecido é a classificação multi-tarefa, onde diversas tarefas de classificação são executadas simultaneamente (Evgeniou et al., 2004), por exemplo, características musicais, entre elas, gênero musical, humor e sexo do intérprete.

Alguns classificadores *multi-label* e multi-tarefa realizam uma classificação “fuzzy”, isto é, calculam as probabilidades de instâncias pertencerem às classes. Mais especificamente, dada uma instância e um conjunto de classes, o algoritmo estima a probabilidade da instância pertencer a cada classe do conjunto. Técnicas de visualização que utilizam propriedades de projeção multidimensional lidando com classificadores *multi-label* ou multi-tarefa não são abundantes, e existe uma clara necessidade de se operar nesse cenário.

Neste capítulo, o método RadViz Concêntrico (RC) é proposto. RC é uma técnica baseada em projeção desenvolvida para operar no contexto de classificadores *multi-label* e multi-tarefa durante o processo de visualização. Uma aplicação direta desta visualização é a exploração de bases de dados de música, onde um classificador multi-tarefa pode ser utilizado para classificar coleções musicais em termos de descrições semânticas da música, por exemplo, gênero, sexo do intérprete (masculino / feminino) e humor (feliz / triste). Com base nessas informações, RC apresenta as informações relevantes ao usuário e permite a interação direta com a projeção dos dados.

O método utiliza RadViz como um mecanismo de projeção. Entretanto, diferentemente de outras técnicas baseadas em RadViz, a metodologia utilizada é capaz de apresentar dados multi-tarefa enquanto produz resultados melhores de posicionamento de pontos em termos de desordem visual e ambiguidade, comumente presentes em visualizações baseadas em projeção. Este capítulo está organizado da seguinte maneira: primeiramente, uma revisão sobre RadViz será feita. O método de visualização é descrito e validado. Por fim, o RadViz Concêntrico é utilizado em bases de dados reais, ilustrando duas possíveis aplicações.

5.1 Conceitos básicos sobre RadViz

5.1.1 Algoritmo e propriedades

RadViz (Hoffman et al., 1997) é uma técnica de visualização desenvolvida para mapear dados multidimensionais no espaço visual. O algoritmo, inspirado por um sistema massa-mola, opera da seguinte maneira: cada atributo (coordenada) é inicialmente normalizado no intervalo $[0, 1]$. Pontos de referência, chamados Âncoras Dimensionais (AD), são posicionados sobre a circunferência de um círculo, onde cada AD representa uma dimensão. Portanto, n ADs são necessários para representar dados n -dimensionais. Cada instância é amarrada aos ADs por meio de molas, cuja constante de atração conectando a instância i ao AD j é proporcional ao valor do j^{esimo} atributo da instância i . A posição da instância i na tela (\vec{x}_i) é dada por:

$$\vec{x}_i = \frac{\sum_{j=0}^n \vec{S}_j v_{ij}}{\sum_{j=0}^d v_{ij}} \quad (5.1)$$

, onde v_{ij} é o valor do j^{esimo} atributo da instância i e \vec{S}_j é a posição da j^{esima} AD. A Figura 5.1 ilustra uma execução do RadViz tradicional na base de dados Iris.

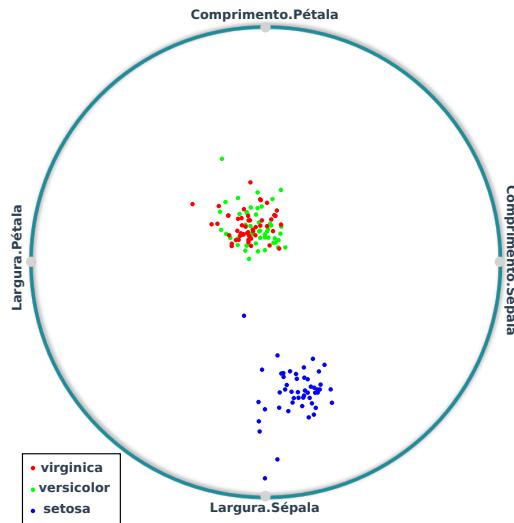


Figura 5.1: Visualização RadViz da base de dados Iris.

Daniels et al. (2012) provaram diversas propriedades geométricas de visualizações radiais, como o RadViz. Entre as principais propriedades, estão: linhas d -dimensionais são mapeadas para linhas ou pontos, hiperesferas são mapeadas para elipses, e hiperplanos,

para polígonos limitados. Outras duas propriedades fazem do RadViz uma importante técnica de visualização. RadViz pode mapear dados com milhares de dimensões no espaço visual de maneira robusta, pois sua complexidade é linear no número de amostras (McCarthy et al., 2004). Além disso, é inherentemente interativo: as ADs podem ser movidas livremente sobre o círculo e o mapeamento pode ser atualizado em tempo real, de acordo com a interação do usuário (Sharko et al., 2008).

A ordenação das ADs no círculo influencia diretamente na qualidade do *layout* final da visualização. Ankerst et al. (1998) formularam o problema da ordenação de dimensões como uma adaptação do problema do caixeiro viajante e provaram a sua NP-Completeness. Os autores propuseram diferentes métricas de dissimilaridade entre as dimensões do conjunto de dados e usaram uma heurística de colônia de formigas para resolver o caixeiro viajante. McCarthy et al. (2004) utilizaram a distribuição t-student para calcular quão efetivamente cada dimensão discrimina as classes e organizaram os ADs de modo a otimizar essa métrica. Recentemente, Caro et al. (2010) apresentaram duas diferentes formulações para o problema da ordenação dos ADs e mostraram que as formulações têm maior probabilidade de encontrar o ótimo global por meio de heurísticas.

5.1.2 Extensões

Diversas extensões foram propostas para o RadViz original, de modo a adaptá-lo para problemas específicos ou suprir as deficiências da técnica. Os principais trabalhos serão descritos nesta subseção.

Modificações já foram feitas para trabalhar com dados resultantes de classificação e agrupamento. Seifert et al. (2010) modificaram RadViz para visualizar o resultado de classificação de imagens em termos de probabilidades de classe. Primeiramente, cada classe é posicionada no círculo como ADs. As instâncias são então projetadas em termos de suas probabilidades de classe e representadas por *Glyphs* coloridos, que indicam a classe de maior probabilidade. Imagens próximas ao centro do círculo tem um grau de incerteza maior na classificação. Em um processo iterativo, o usuário pode rotular essas instâncias e retreinar o classificador, formando o ciclo de aprendizado ativo. A Figura 5.2 ilustra a execução do algoritmo.

Sharko et al. (2008) desenvolveram *Vectorized RadViz* (VRV), uma técnica que possibilita a avaliação visual de algoritmos de agrupamento. Com VRV, é possível identificar padrões nos dados, por exemplo: grupos similares obtidos em diferentes algoritmos; e grupos instáveis, que mudam de grupos constantemente. O VRV funciona da seguinte maneira: particiona-se as dimensões categóricas, no caso, a identificação de *cluster*, transformando atributos com n valores possíveis em n atributos binários. Em seguida,

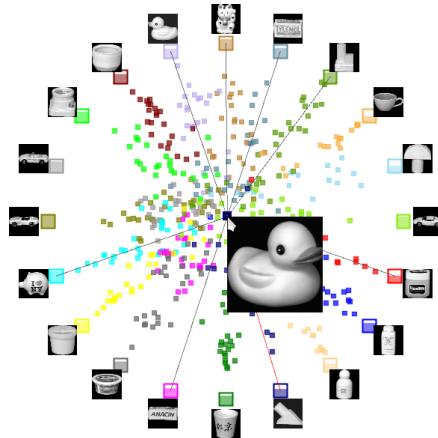


Figura 5.2: RadViz aplicado no contexto de classificação de bases de imagens. As probabilidades de classe da instância selecionada são apresentadas como barras próximas às âncoras dimensionais. Figura adaptada de (Seifert et al., 2010)

posiciona-se cada nova dimensão independentemente como um AD. Isso resulta em uma visualização onde grupos estáveis ficam nas bordas e instáveis, no centro do círculo. A Figura 5.3 ilustra a execução do VRV com dados do agrupamento de micro-vetores de DNA e quatro técnicas. É possível identificar grupos estáveis e instáveis na visualização. A escala de cor indica o resultado de um agrupamento com o algoritmo *K-Means*.

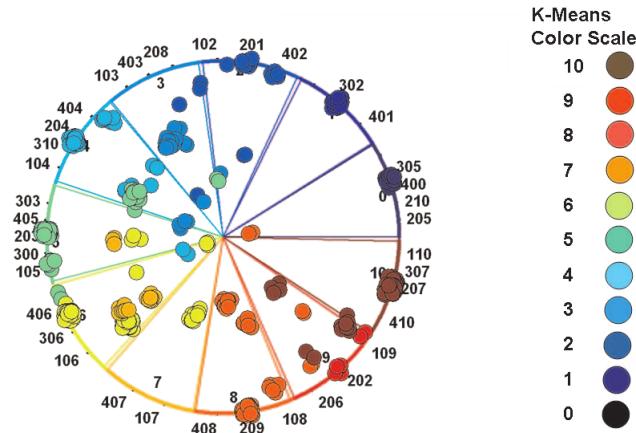


Figura 5.3: VRV aplicado ao agrupamento de micro-vetores de DNA. Pontos próximos ao centro do círculo representam grupos instáveis. Figura adaptada de (Sharko et al., 2008).

Embora seja uma visualização com boas propriedades e complexidade linear, RadViz tem uma limitação: muitas instâncias de dados podem ser mapeados para o mesmo ponto. De fato, instâncias em uma linha contendo a origem do espaço original serão mapeados para a mesma posição no espaço visual, dado que a proporção entre as dimensões originais é mantida sobre essa linha. Isso prejudica a visualização de grupos cujos

centroides estão próximos de uma linha passando pela origem, já que os grupos não serão separáveis no espaço visual (Daniels et al., 2012). Novakova et al. (2009) desenvolveram um pré-processamento interativo para lidar com esse problema: a ferramenta permite que usuários rotacionem e espelhem algumas dimensões dos dados no espaço original antes de realizar a projeção com RadViz, o que pode gerar projeções com grupos mais separáveis. Daniels et al. (2012) automatizaram esse processo com um algoritmo genético, que realiza um grande número de rotações e escolhe o conjunto com a melhor separação de grupos. Apesar de essas abordagens melhorarem a separação de grupos visualmente, as ADs não correspondem às dimensões originais, prejudicando a interpretação direta do *layout*.

Extensões tridimensionais para RadViz também foram propostas. Novakova et al. (2009) incluíram um terceiro eixo na visualização para representar a distância de cada instância até a origem no espaço original dos dados. A visualização proposta, denominada RadVizS, reduz o problema dos grupos alinhados com a origem, que podem ser separados no novo eixo, como ilustra a Figura 5.4. No contexto de aplicações de realidade virtual, Doulis et al. (2007) desenvolveram *SphereViz*, um sistema que mapeia dados de alta dimensão para o interior de uma esfera. O mapeamento é realizado de modo similar ao RadViz: ADs são posicionados sobre a superfície de uma esfera e os dados são interpolados no espaço 3D por meio de uma analogia com o sistema de molas. A figura 5.5 apresenta um esquema de como ocorre a projeção com *SphereViz*.

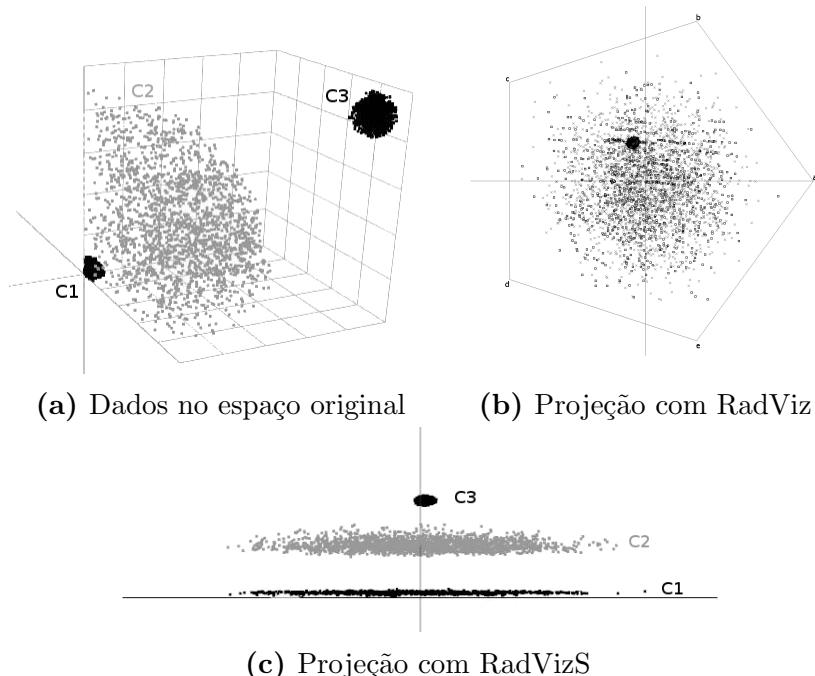


Figura 5.4: RadVizS, extensão do RadViz tradicional em três dimensões. Imagem adaptada de (Novakova et al., 2009).

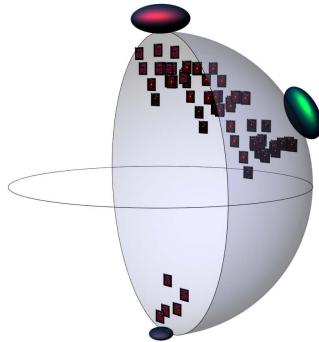


Figura 5.5: *Sphereviz*, extensão tridimensional do RadViz em que âncoras são posicionadas sobre uma esfera. Imagem adaptada de (Doulis et al., 2007).

5.2 Método

Nesta seção, serão propostas duas modificações para o algoritmo RadViz tradicional, desenvolvidas para habilitar a exploração visual de dados de classificação multi-classe, *multi-label* e multi-tarefa. Assim como (Seifert et al., 2010), RadViz é utilizado para visualizar estimativas de probabilidade de classe. Entretanto, neste trabalho, deseja-se explorar os resultados de classificação, ao invés de usar um procedimento relacionado ao aprendizado ativo para atualizar os modelos de classificação.

5.2.1 RadViz Concêntrico

O RadViz tradicional pode ser utilizado para visualizar o resultado de uma classificação, considerando-se cada possível classe como uma dimensão a ser mapeada por meio de ADs. Entretanto, não pode representar múltiplas tarefas de classificação ao mesmo tempo, mesmo se as ADs forem ordenadas de acordo com uma medida de similaridade entre dimensões, por exemplo, Caro et al. (2010). Considere, por exemplo, dados resultantes de classificação de música multi-tarefa, consistindo de gênero musical (rock, pop, etc.), sexo (masculino, feminino) e humor (festivo, agressivo ou relaxado). As três tarefas de classificação seriam representadas por meio de uma única visualização circular, o que levaria a diversos problemas:

1. Tarefas de classificação binárias exerceriam mais influência na posição de cada ponto, em comparação com tarefas multi-classe (probabilidades de classificação são comumente mais altas no problema de classificação binário);

2. Classes de diferentes tarefas de classificação seriam apresentadas em um único círculo como ADs, o que dificultaria a interpretação dos dados e a interação com a visualização;
3. ADs com significados opostos poderiam estar próximos na circunferência do círculo, o que acarretaria na apresentação de dados incorretos ao usuário.

Considere, por exemplo, um conjunto de dados artificial de formas geométricas, em que três tarefas de classificação são realizadas: “geometria” (“triângulo”, “quadrado”, “círculo”, “pentágono”), “preenchimento” (“preenchido”, “vazio”) e “cor” (“verde”, “vermelho”, “azul”). Na Figura 5.6a, o *dataset* é visualizado com o algoritmo RadViz tradicional, e os três problemas descritos estão presentes: (1) a tarefa “preenchimento” (binária) exerce grande influência na projeção; (2) classes das três tarefas estão misturadas em um único círculo e (3) ADs com significados opostos (“verde” e “vermelho”, por exemplo) são posicionados lado a lado na visualização.

A fim de resolver esses problemas e visualizar mais de uma tarefa de classificação ao mesmo tempo, foi desenvolvido RadViz Concêntrico (RC). Com RC, cada tarefa de classificação é representada por um círculo particular, chamado Grupo de Dimensões (GD). GDs são representados por círculos concêntricos no espaço visual, onde as ADs são posicionadas. A Figura 5.6b ilustra a base de dados de formas geométricas com a técnica proposta.

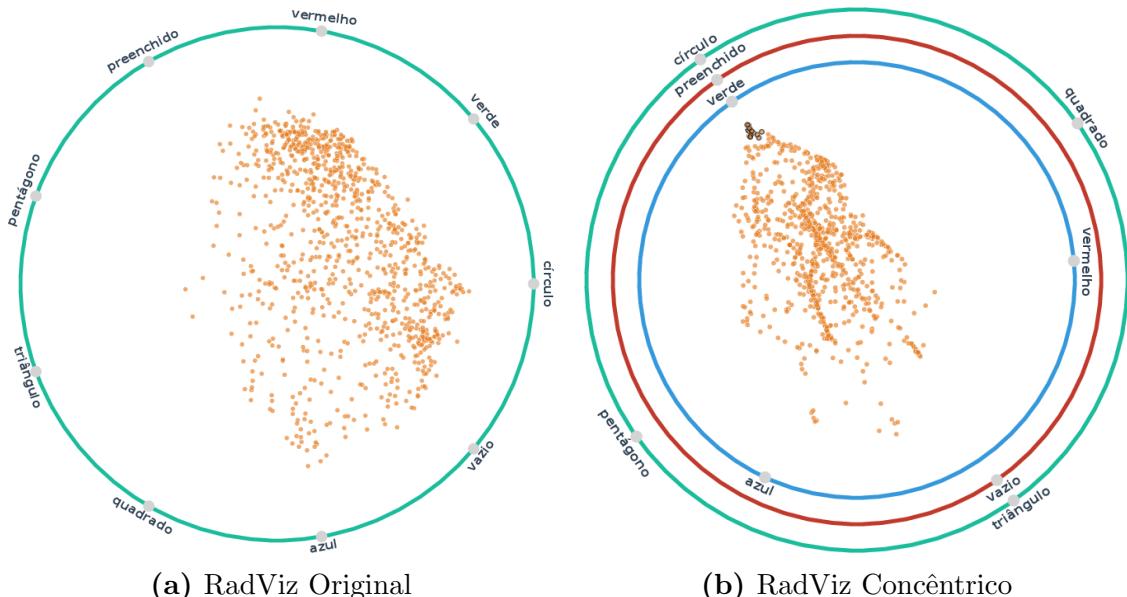


Figura 5.6: Base de dados de formas geométricas visualizada com RadViz e RadViz Concêntrico

O algoritmo para o desenho do RC é dado a seguir: cada instância é associada a um vetor m -dimensional, onde cada coordenada corresponde a uma classe. Portanto, m é o número total de classes em todos os GDs. O valor de cada coordenada é igual a probabilidade da instância pertencer à classe correspondente a essa coordenada. A instância será mapeada para o espaço visual por meio do RadViz tradicional, levando em consideração o ângulo das ADs nos círculos concêntricos e as probabilidades do vetor m -dimensional. ADs são posicionadas em cada círculo por meio da resolução de um problema do caixeiro viajante, como proposto por Ankerst et al. (1998).

Cada dimensão deve ser normalizada como é feito no RadViz tradicional. Além disso, a fim de trabalhar com múltiplos GDs, o vetor m -dimensional também é normalizado por partes, de acordo com o GD a qual cada dimensão pertence. Mais especificamente, cada grupo de coordenadas é normalizado, fazendo com que a dimensão de maior probabilidade em cada GD seja igual a 1. Isso garante que as classes de maior probabilidade de cada GD influenciem igualmente na projeção.

Com RC, o usuário pode combinar classes de diferentes grupos para explorar os dados e extrair informações relevantes da visualização. Por exemplo, na Figura 5.6b, os grupos de dimensões foram rotacionados de modo a alinhar as âncoras “círculo”, “preenchido” e “verde”. A projeção resultante agrupou os itens nessas categorias de forma que eles ficasse próximos às âncoras alinhadas.

5.2.2 Ponderação sigmoidal

Com o RadViz tradicional, todas as ADs contribuem igualmente na posição dos pontos mapeados. No contexto de visualização de probabilidades de classe, isso pode não ser interessante. Considere o caso de um problema multi-classe onde a instância i pertence a classe A com probabilidade 0.5, portanto, a probabilidade de i pertencer a todas as outras classes totalizam 0.5. O classificador pode estar com muita certeza de que a instância pertence à classe A , mas no RadViz, o ponto é atraído para o centro da circunferência, como ilustrado na Figura 5.7.

A fim de amenizar esse problema, um mecanismo de filtragem de forças que exclui classes com baixa probabilidade é proposto neste trabalho. A filtragem ocorre ao multiplicar cada dimensão v_{ij} por uma função sigmoide normalizada no intervalo $[0, 1]$, apresentada abaixo:

$$\hat{\sigma}(x, s, t) = \begin{cases} \frac{\sigma(x, s, t) - \sigma(0, s, t)}{\sigma(1, s, t) - \sigma(0, s, t)} & , \text{ se } \sigma(1, s, t) \neq \sigma(0, s, t) \\ 1 & , \text{ caso contrário} \end{cases} \quad (5.2)$$

$$\text{com } \sigma(x, s, t) = \frac{1}{1 + \exp(-s(x + t))}.$$

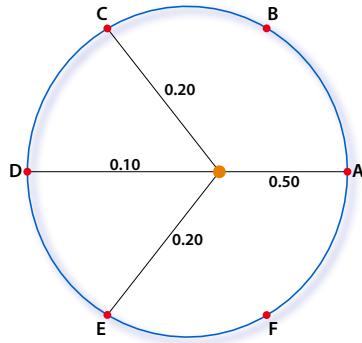


Figura 5.7: Visualização de uma tarefa de classificação com RadViz. Note como as forças se cancelam e o ponto é atraído para a origem.

O parâmetro de escala, s , $s \geq 0$, e translação, t , $-1 \leq t \leq 1$, controlam o limiar a partir do qual dimensões (classes) com probabilidades baixas são canceladas. Por fim, o mapeamento ponderado com a função $\hat{\sigma}$ é dado por:

$$\vec{x}_i = \frac{\sum_{j=0}^d \vec{S}_j v_{ij} \hat{\sigma}(v_{ij}, s, t)}{\sum_{j=0}^d v_{ij} \hat{\sigma}(v_{ij}, s, t)}, \quad (5.3)$$

, onde \vec{S}_j é o vetor posição da AD de índice j e v_{ij} é o valor da coordenada j da instância i .

Um esquema de ponderação similar foi proposto para a técnica Gravi++ (Hinum et al., 2005): nesta técnica, o usuário pode atribuir pesos às dimensões de modo a dar mais, ou menos, importância a elas durante a etapa de interpolação. No Radviz Concêntrico, por outro lado, os pesos são atribuídos automaticamente às dimensões de cada instância.

O usuário pode controlar interativamente os parâmetros de escala e translação da sigmoide. Conforme a área da sigmoide diminui, os pontos ficam mais agrupados. Os grupos se formam em posições bem definidas, próximos às âncoras, o que reduz ambiguidades sobre a classificação, isto é, as instâncias tendem a ficar mais próximas da classe de maior probabilidade. Para ilustrar esta situação, considere a Figura 5.7: os pesos originais são utilizados para posicionar o ponto, portanto é uma configuração equivalente a projetá-lo com a ponderação sigmoidal e parâmetros $s = 0$ e $t = 0$. Se cancelarmos algumas molas, utilizando parâmetros $s = 15$ e $t = -0.5$, os o ponto será atraído para a AD A. A Figura 5.8 apresenta o efeito da ponderação sigmoidal: A Figura 5.8a apresenta o peso original

(x) , a função sigmoid normalizada ($\hat{\sigma}(x)$) e a influência de $\hat{\sigma}(x)$ em x ($x\hat{\sigma}(x)$). A Figura 5.8b apresenta a nova posição do ponto após a ponderação.

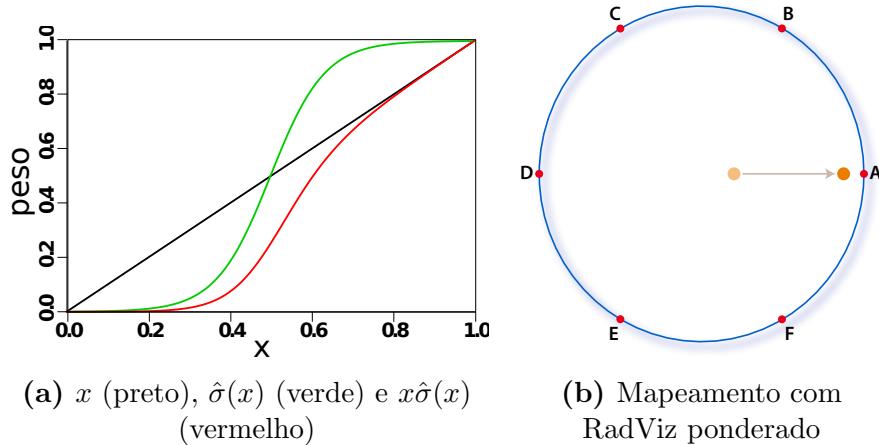


Figura 5.8: RadViz com ponderação sigmoidal. Parâmetros: $s = 15$ e $t = -0.5$.

5.3 Validação

Um protótipo para o sistema foi implementado utilizando a linguagem *JavaScript* com a biblioteca *Data Driven Documents* (D3) (Bostock et al., 2011). O problema do caixeiro viajante foi resolvido com a biblioteca Concorde (Applegate et al., 1998).

Nesta seção, são apresentados alguns resultados com o intuito de validar a técnica RadViz Concêntrico. A fim de direcionar a análise apenas na técnica de visualização proposta, assumimos que os dados para análise foram classificados por um método que satisfaz a seguinte afirmação: “O classificador é altamente preciso em termos de estimativa de probabilidades de classe”.

O RC é validado por meio de duas técnicas bem conhecidas de recuperação de informação: *Mean Average Precision* (MAP) e *R-Precision* (Manning et al., 2008). A MAP é utilizada para verificar a qualidade da vizinhança presente no *layout* da projeção, ou seja, mede-se quão similar pontos vizinhos são em termos de suas múltiplas classificações (derivadas de tarefas de classificação distintas). A *R-Precision*, por sua vez, é utilizada para medir o alinhamento dos pontos projetados com relação às âncoras dimensionais.

Foram realizados experimentos utilizando duas bases de classificação multi-tarefas distintas: a base de músicas *Dortmund* (Homburg et al., 2005) e a base de imagens faciais *Multi-task Facial Landmark* (MTFL) (Zhang et al., 2010).

A base de dados *Dortmund* contém 1.886 músicas divididas em nove gêneros: *Alternative* (145), *Blues* (120), *Electronic* (113), *Folk-Country* (222), *Funk-Soul-R&B* (47),

Jazz (319), *Pop* (116), *Rap-Hiphop* (300) e *Rock* (504). São utilizados modelos de classificação pré-treinados da biblioteca Essentia (Bogdanov et al., 2013) para classificar e obter as probabilidades de cada instância à partir dos arquivos de áudio, com respeito a gênero musical, sexo (masculino / feminino), humor (feliz / triste), instrumento (voz / instrumental) e tipo (festivo / agressivo / relaxado). A acurácia dos classificadores pré-treinados da biblioteca estão disponíveis em (Bogdanov et al., 2013; Bogdanov, 2013).

A base MTFL contém 12.995 imagens de faces humanas e um conjunto de características que descrevem informações geométricas de pontos de referência faciais, por exemplo, olhos, nariz e boca. Cada imagem é anotada com informações de quatro atributos: sexo (masculino / feminino), pose (perfil esquerdo / esquerdo / frontal / direito / perfil direito), óculos (usando / não usando) e expressão facial (sorrindo / não sorrindo). A fim de predizer probabilidades de classe neste *dataset*, foi utilizado o classificador Floresta Randômica (Breiman, 2001), implementado na biblioteca *Scikit-Learn* (Pedregosa et al., 2011). Foram utilizados como parâmetros os valores padrão da biblioteca.

O MAP é utilizado para verificar se pontos no espaço visual tendem a pertencer às mesmas classes de seus vizinhos. Diferentes MAPs podem ser obtidos com o RC, modificando-se as rotações nos grupos de dimensão. A fim de fazer uma comparação com o Radviz tradicional, um algoritmo genético foi utilizado para calcular os ângulos de rotação que resultem no melhor MAP. Relembre que as âncoras dimensionais são arranjadas em cada grupo por meio da resolução de um problema do caixeiro viajante. Uma vez que os círculos concêntricos estejam rotacionados, computamos o MAP ranqueando os vizinhos de cada ponto p de acordo com a distância Euclidiana. A média das precisões (MP) de cada ponto p é calculada e, ao fim do processo, o MAP é obtido como a média de todos os MPs. Um vizinho de p é considerado relevante se o seu conjunto de rótulos (classe de maior probabilidade em cada GD) é idêntico ao de p .

A Tabela 5.1 apresenta o MAP obtido para os dois conjuntos de dados. Foram realizados oito experimentos com diferentes parâmetros para a função sigmoide. Os experimentos foram executados variando-se o número de tarefas de classificação, da seguinte maneira:

1. Base de músicas *Dortmund*:
 - (a) Gênero Musical;
 - (b) Gênero Musical & Humor;
 - (c) Gênero Musical & Humor & Sexo;
 - (d) Gênero Musical & Humor & Sexo & Instrumento;
2. Base de faces MTFL:

- (a) Pose;
- (b) Pose & Sexo;
- (c) Pose & Sexo & Óculos;
- (d) Pose & Sexo & Óculos & Expressão Facial.

Tabela 5.1: Mean Average Precision de RadViz e RadViz Concêntrico.

	Radviz	RadViz Concêntrico		
	$\hat{\sigma}(x, 0, 1)$ 	$\hat{\sigma}(x, 10, -0.8)$ 	$\hat{\sigma}(x, 20, -1)$ 	
1-a)	0.7141	0.7141	0.8846	0.9062
1-b)	0.5420	0.7043	0.8201	0.8985
1-c)	0.3842	0.6869	0.7560	0.8470
1-d)	0.2783	0.6931	0.7328	0.7418
2-a)	0.9150	0.9150	0.9787	0.9852
2-b)	0.6350	0.7010	0.8840	0.9110
2-c)	0.5709	0.5937	0.8113	0.8368
2-d)	0.3574	0.3921	0.6975	0.7500

A fim de realizar a comparação, executa-se o RadViz original com ADs, de todas as tarefas de classificação, organizadas em um único círculo com a otimização do caixeiro viajante. Os resultados apresentados na Tabela 5.1 mostram que o RadViz original não é capaz de apresentar múltiplas tarefas de classificação tão bem quanto o RadViz Concêntrico. Por exemplo, quando quatro GDs são visualizados (experimentos 1-d e 2-d), o MAP do RadViz original cai consideravelmente, enquanto o MAP do RadViz Concêntrico é aproximadamente duas vezes melhor com a configuração $\hat{\sigma}(x, 20, -1)$.

O segundo conjunto de experimentos utiliza a medida *R-Precision* para avaliar o quão próximos os pontos projetados se alinham com as âncoras dimensionais. Neste caso, os dados são projetados utilizando o RadViz Concêntrico e consultas são feitas rotacionando-se os grupos de dimensões de modo a alinhar classes de interesse. Por exemplo, no caso da base de dados de música, uma consulta pode ser efetuada alinhando-se as classes “pop”, “mulher” e “feliz” para identificar músicas que pertencem às três classes simultaneamente. A Tabela 5.2 apresenta o resultado de *R-Precision* para oito consultas com o RadViz Concêntrico (três parâmetros de sigmoid) e com o RadViz original. A fim de realizar o experimento com o RadViz original, as classes foram inicialmente posicionadas por meio de uma execução do caixeiro viajante e, em seguida, classes de interesse foram sobrepostas no círculo. Os seguintes alinhamentos foram realizados:

1. Base de músicas *Dortmund*:
 - (a) Gênero Musical: Pop;
 - (b) Gênero Musical: Alternative; Instrumento: Instrumental;
 - (c) Gênero Musical: Rock; Sexo: Masculino; Humor: Triste;
 - (d) Gênero Musical: Rap; Tipo: Agressivo; Sexo: Masculino; Humor: Feliz;
2. Base de faces MTFL:
 - (a) Pose: Perfil Direito;
 - (b) Pose: Perfil Esquerdo; Sexo: Masculino;
 - (c) Pose: Esquerdo; Sexo: Feminino; Óculos: Não usando;
 - (d) Pose: Frontal; Sexo: Masculino; Expressão Facial: Sorrindo; Óculos: Não usando.

Tabela 5.2: *R-Precision* para consultas utilizando RadViz e RadViz Concêntrico.

	Radviz	Radviz Concêntrico		
	$\hat{\sigma}(x, 0, 1)$ 	$\hat{\sigma}(x, 10, -0.8)$ 	$\hat{\sigma}(x, 20, -1)$ 	
1-a)	0.1000	0.1000	0.9250	0.9650
1-b)	0.2307	0.4146	0.8780	0.9024
1-c)	0.4062	0.5833	0.6875	0.8541
1-d)	0.6388	0.6111	0.7777	0.9166
2-a)	0.8260	0.8260	1.0000	1.0000
2-b)	0.3750	0.5416	0.83333	0.8333
2-c)	0.6173	0.7198	0.90660	0.9339
2-d)	0.5894	0.5621	0.77052	0.8063

Assim como nos experimentos anteriores, conforme a sigmoide é transladada para a direita, os pontos são posicionados de acordo com as maiores probabilidades de classe, o que resulta em um valor mais alto de *R-Precision*. A Tabela 5.2 mostra como o RadViz Concêntrico supera o RadViz tradicional nesta métrica de avaliação.

5.4 Aplicações

Nesta seção, RC é utilizado no contexto de duas bases de dados: análise de uma coleção de músicas e de uma base de descrição textual de filmes. A efetividade do Radviz

Concêntrico é evidenciada em ambas as aplicações, que possibilitam ao usuário explorar a base de dados e facilmente recuperar a informação de interesse.

5.4.1 Base de músicas Dortmund

Uma maneira tradicional de organizar coleções de música é identificar e agrupar músicas que compartilham as mesmas características. No contexto de recuperação de informação musical, músicas podem ser caracterizadas por timbre, melodia, ritmo, harmonia, instrumentação, entre outras estruturas. Uma grande variedade de características de áudio foram propostas na literatura para visualização e classificação de músicas (Fu et al., 2011; Dalhuijsen et al., 2010; Torrens et al., 2004).

De modo geral, as características podem ser divididas em três categorias: baixo, médio e alto nível. Características de baixo nível são consideradas a base de sistemas de classificação de áudio, pois são facilmente extraídas e possuem bons resultados na maioria dos sistemas de classificação. Entre elas, podemos citar o espectrograma, MFCC e o centróide espectral (Fu et al., 2011).

Características de médio nível ganharam importância recentemente na literatura de MIR, pois são capazes de descrever semanticamente algumas informações musicais, por exemplo, ritmo (*beat tracking*), tom (PCP) e harmonia (progressão de acordes). Por fim, as características de alto nível são obtidas a partir de descritores mais simples (baixo e médio nível) e constituem um vasto conjunto de propriedades semânticas de música, por exemplo, gênero, estilo, humor, sexo do cantor, entre outras.

Na aplicação de visualização de coleções de músicas com o Radviz Concêntrico, características de baixo e médio nível são utilizadas como entrada para os classificadores, que geram características musicais de alto nível em termos de probabilidade de classe. As probabilidades de classe são obtidas por meio da execução de classificadores SVM pré-treinados, implementados na biblioteca Essentia (Bogdanov et al., 2013). Entre as características utilizadas, estão medidas estatísticas (média, mediana, variância, obliquidade e curtose) do espectrograma, MFCC e HPCP, obtidos a partir dos arquivos de áudio.

As seguintes características de alto nível (tarefas de classificação) foram utilizadas na visualização:

- **Gênero Musical:** Alternative, Blues, Electronic, Folk-Country, Funk-Soul-R&B, Jazz, Pop, Rap-Hiphop, Rock;
- **Humor 1:** Feliz, Não Feliz;
- **Humor 2:** Triste, Não Triste;

- **Tipo 1:** Festivo, Não Festivo;
- **Tipo 2:** Relaxado, Não Relaxado;
- **Tipo 3:** Agressivo, Não Agressivo;
- **Gênero do cantor:** Masculino, Feminino;
- **Instrumento:** Instrumental, Voz.

Assim como na seção de validação, a técnica RadViz Concêntrico foi utilizada para explorar a base de dados de música *Dortmund* (Homburg et al., 2005), descrita previamente na Seção 5.3. A Figura 5.9 apresenta a projeção resultante do Radviz Concêntrico aplicado à coleção de músicas. Cada GD corresponde à descriptores semânticos disponibilizados pela biblioteca Essentia. No sistema, o usuário pode escolher quais tarefas de classificação utilizar na análise de dados. Por exemplo, é possível explorar a base com relação à gênero musical, humor, sexo do cantor e tipo de música, como apresentado na Figura 5.9.

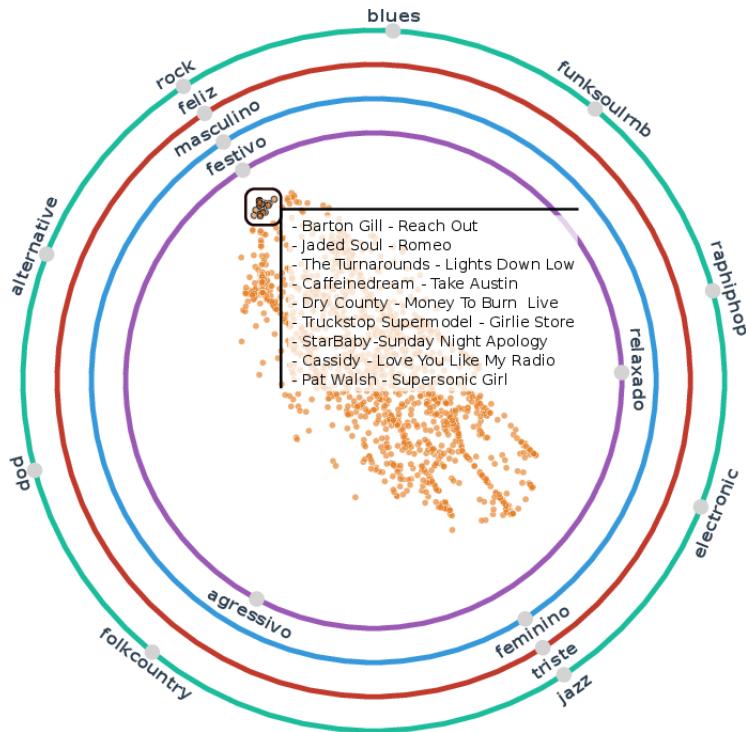


Figura 5.9: Visualização da base de dados *Dortmund* com o Radviz Concêntrico. As músicas selecionadas pertencem às classes gênero “rock”, humor “feliz”, sexo “masculino” e tipo: “festivo”. Em destaque, os títulos das músicas selecionadas.

A visualização apresentada na Figura 5.9 filtra probabilidades baixas usando a ponderação sigmoidal com parâmetros $s = 10$ e $t = -0.8$. Perceba que quatro dimensões estão

alinhas: “rock”, “feliz”, “masculino” e “festivo”. O Radviz Concêntrico está, portanto, agrupando músicas que compartilham desses quatro rótulos, de modo que elas fiquem próximas às âncoras. O usuário pode, então, interagir com a visualização, selecionando e inspecionando uma pequena porção das músicas próximas das âncoras direcionais, como apresentado nesta figura. A interação pode ser utilizada para criar *playlists* ou explorar e entender melhor a base de dados, identificando as suas principais classes de músicas.

É importante ressaltar que os interesses do usuário direcionam a exploração dos dados com o RadViz Concêntrico. Por exemplo, se o objetivo da exploração for buscar por músicas eletrônicas cantadas por mulheres, pode-se filtrar os dados como mostrado na Figura 5.10.

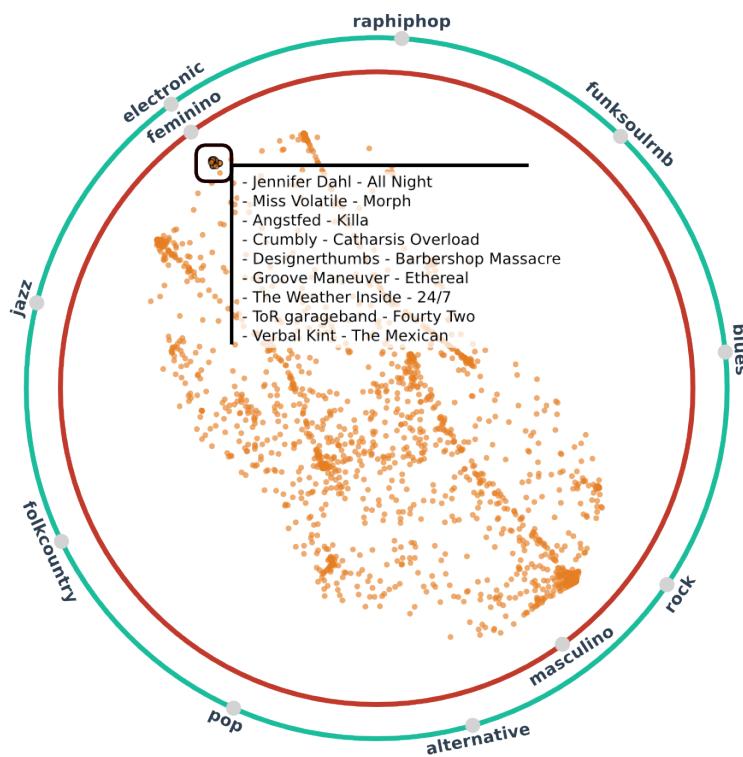


Figura 5.10: Exemplo de interação: busca por músicas eletrônicas cantadas por mulheres. Base de dados *Dortmund*.

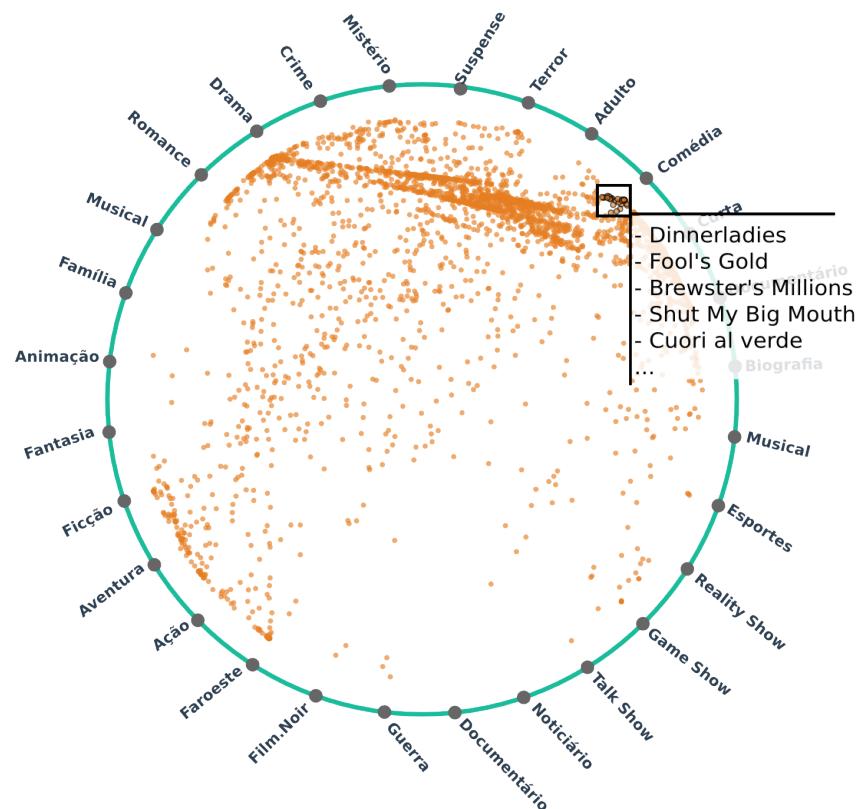
As tarefas de classificação originais indicam se uma música é “feliz”, ou “não feliz”, assim como “triste”, e “não triste”. Entretanto, o usuário pode interpretar as classes e observar que elas são complementares, mesmo não pertencendo à mesma tarefa de classificação. Assim, é possível posicionar apenas os rótulos “feliz” e “triste” em um grupo de dimensões para simplificar a visualização. Graças à normalização das amostras, apesar de “feliz” e “triste” não pertencerem à mesma tarefa de classificação, as instâncias projetadas serão levadas para o AD de maior probabilidade.

5.4.2 Base de filmes IMDB-F

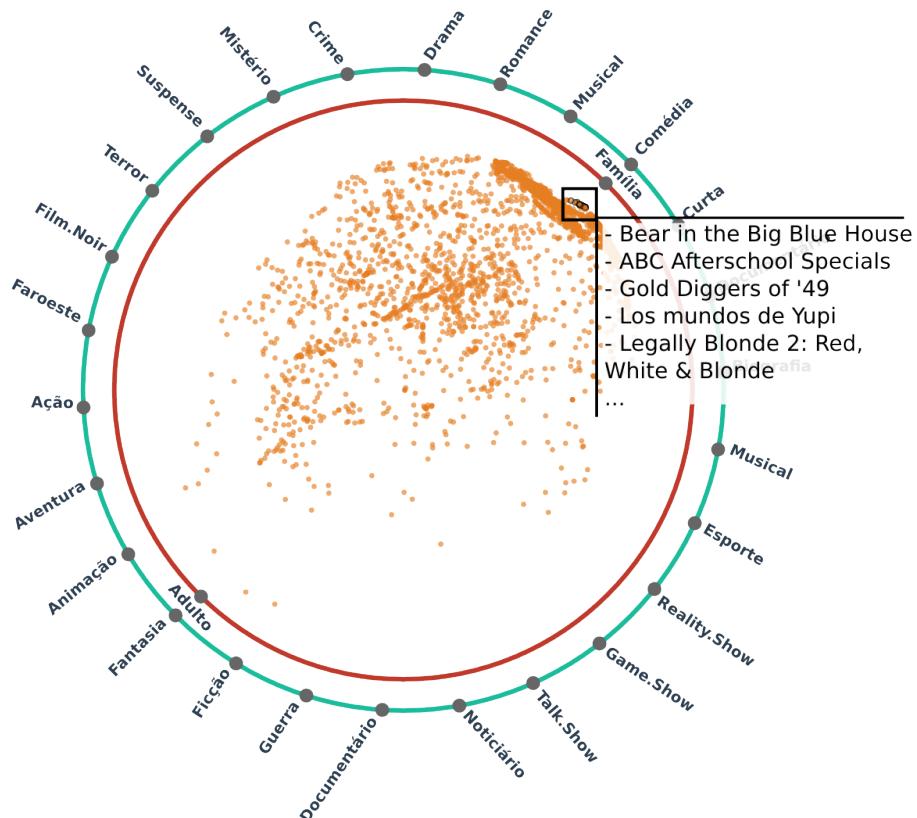
A técnica proposta também foi utilizada para visualizar a base de dados *multi-label* IMDB-F (Read et al., 2011), composta por metadados textuais associados a filmes. A base consiste em uma matriz de termos por documento, contendo informações de sinopses de 120.919 filmes com relação à 1.000 termos (palavras). Mais especificamente, a entrada (i, j) da matriz informa o número de vezes que o termo de índice i aparece nos meta-dados do filme de índice j . Os filmes são anotados com relação ao gênero a que pertencem. Ao todo, existem 28 gêneros na base de dados, entre eles, “Romance”, “Comédia”, “Aventura”, “Drama” e “Fantasia”. Para obter as probabilidades de classe, é utilizado o algoritmo Naive-Bayes Multinomial (Rennie et al., 2003), comumente utilizado em tarefas de classificação textual (Nigam et al., 2000). Foi adotada a implementação do algoritmo da biblioteca *Scikit-learn* (Pedregosa et al., 2011), com os parâmetros padrão.

A Figura 5.11 apresenta a projeção da base de dados de filmes com o Radviz Concêntrico e parâmetros $s = 30$ and $t = -0.6$. Como o *dataset* não é multi-tarefas, apenas uma dimensão foi anotada, ou seja, o gênero do filme. Na Figura 5.11a, um conjunto de filmes do gênero “comédia” é selecionado. Note que, devido à ponderação com função sigmoide, os pontos projetados podem formar estruturas lineares, por exemplo, a linha Drama-Comédia. Essas estruturas são fortes indicadores de que filmes projetados sobre as linhas pertencem às duas categorias simultaneamente, ou seja, possuem duas *labels*. A linha Drama-Comédia, por exemplo, representa filmes do gênero “Comédia dramática”.

Usuários podem interagir e modificar a visualização, impondo interesses específicos à base de dados. Note que as categorias “comédia” e “adulto” estão posicionadas lado a lado na visualização. Se um usuário deseja encontrar filmes da categoria “comédia” e “família” que não pertençam à classe “adulto”, ele pode criar um novo GD com as dimensões “família” e “adulto” e então alinhar “comédia” com “família”, como mostrado na Figura 5.11b. Os pontos projetados próximos das ADs alinhadas respeitarão os interesses do usuário.



(a) Radviz Concêntrico considerando apenas um grupo de dimensões: gênero do filme.



(b) Inclusão de uma dimensão adicional, permitindo ao usuário visualizar interesses específicos.

Figura 5.11: Radviz Concêntrico da base de filmes IMDB-F.

Percebe-se que o RC é uma visualização flexível, pois permite que a interpretação do usuário seja utilizada para modificar a análise dos dados. Portanto, é possível criar novas possibilidades de descoberta de informação por meio de interações com os grupos de dimensão, mesmo se o *dataset* não for multi-tarefa.

5.5 Considerações finais

Neste capítulo, uma nova técnica de visualização de resultados de classificação foi proposta. Resumidamente, o Radviz Concêntrico introduz dois conceitos novos à visualização RadViz tradicional:

1. Um novo mecanismo de ponderação sigmoidal que melhora a projeção com RadViz consideravelmente, reduzindo a desordem visual e ambiguidades na projeção;
2. Um esquema interativo de círculos concêntricos que permite ao usuário visualizar, explorar e organizar dados de classificação multi-tarefa no espaço visual;

A técnica foi validada com duas métricas de recuperação de informação, indicando que RC tem um desempenho melhor que RadViz em termos de vizinhança (MAP) e consultas ranqueadas (*R-Precision*). Além disso, foram realizados estudos de caso com duas bases de dados que mostraram a efetividade da metodologia como uma ferramenta de análise visual.

A aplicação do RC na base de dados *Dortmund* mostrou que essa visualização pode ser muito útil à organização de coleções de músicas, possibilitando a exploração de diferentes aspectos musicais simultaneamente.

Conclusões

Neste trabalho, foram propostas duas técnicas de visualização de informações musicais inéditas na literatura da área: Grafo de Similaridades e RadViz Concêntrico. Cada técnica apresenta a coleção de músicas de um ponto de vista distinto.

O Grafo de Similaridades ressalta relações de semelhança tonal na base de dados de forma hierárquica, provendo informações mais detalhadas sobre a coleção de músicas, em comparação com as visualizações presentes na literatura. Mais especificamente, músicas são projetadas no espaço visual por meio de suas relações de similaridades globais e um paradigma de grafo é utilizado para apresentar detalhes de similaridades locais, isto é, similaridades entre trechos de segmentos. A fim de facilitar o processo de análise de dados, foram propostas duas versões para o GS: o Grafo de Similaridades Global, que apresenta a base de dados como um todo, e o Grafo de Similaridades Local, que possibilita a análise detalhada de uma pequena porção da coleção de músicas.

O RadViz Concêntrico, por sua vez, permite a visualização de coleções musicais de diversos pontos de vista. Mais especificamente, a base de músicas é pré-processada em termos de tarefas de classificação (gênero, sexo do cantor, humor, etc.) e a visualização apresenta similaridades das músicas em termos dessas tarefas. A visualização possibilita a interação do usuário por meio da criação de grupos de dimensões e da rotação dos círculos concêntricos, que resultam em uma organização diferente da base e, possivelmente, em consultas visuais, provenientes do alinhamento das âncoras dimensionais.

6.1 Discussões, limitações e trabalhos futuros

As aplicações das duas técnicas em bases de dados de música evidenciaram o potencial das representações visuais propostas no contexto de exploração e descoberta de informação musical. A discussão deste trabalho será dividida em duas subseções, referentes às duas metodologias desenvolvidas.

6.1.1 Grafo de Similaridades

A escalabilidade é uma limitação da primeira visualização, Grafo de Similaridades, uma vez que o pré-processamento dos dados é uma tarefa demorada. A métrica de comparação utilizada, Q_{max} , tem complexidade quadrática. Além disso, é necessário comparar os segmentos de kn pares de músicas, sendo k a quantidade de vizinhos ligados no grafo e n a quantidade de músicas na base. Desta forma, um trabalho futuro é avaliar o uso de métricas de comparação mais simples entre trechos de músicas, de modo a tornar a etapa de pré-processamento mais rápida.

A segunda limitação da técnica é a utilização do espaço visual. Claramente, o Grafo de Similaridades é uma visualização que ocupa muito espaço visual, o que limita sua aplicabilidade a um conjunto de poucas centenas de músicas. Metáforas visuais para aliviar este problema serão buscadas em trabalhos futuros. Em especial, serão tomadas como inspiração técnicas que utilizam o princípio de foco + contexto e a metáfora visual *fish eye*.

Por fim, é necessário definir um critério de avaliação para o método proposto. Um trabalho futuro nesta linha é a anotação manual de um conjunto pequeno de pares de músicas, em termos de similaridade entre trechos musicais. A anotação, realizada por um especialista da área, possibilitaria que métricas de recuperação de informação fossem utilizadas para avaliar quantitativamente os resultados obtidos.

6.1.2 RadViz Concêntrico

Radviz Concêntrico é uma técnica de visualização de propósito geral que possibilita que usuários explorem conjuntos de dados em termos de similaridade entre descritores semânticos (saída de classificadores previamente treinados nos dados). Devido ao tamanho limitado da tela, o sistema de visualização restringiu o número de grupos de dimensão a seis círculos concêntricos. Um trabalho futuro nesta linha é o estudo de novas metáforas visuais que possam ser combinadas com o RadViz Concêntrico para habilitar o uso de um número maior de grupos de dimensão.

A técnica foi avaliada em termos de duas métricas comumente utilizadas no contexto de recuperação de informação. Entretanto, não foram realizados estudos de caso com usuários. Como trabalho futuro, a qualidade da visualização será investigada em termos da perspectiva do usuário, medindo assim a efetividade da metodologia proposta de forma qualitativa.

Apêndices

Listagem de Músicas da Base *CoversYoutube*

1. Carolina - Taylor Moreau
2. Carolina - Stripped Down
3. Carolina - Parmalee Live
4. Carolina - Marcus Brown
5. Carolina - Parmalee
6. Our Song - JustinGodsey
7. Our Song - Kristen Roth
8. Our Song - Renee Dominique
9. Our Song - Taylor Swift
10. Our Song - The Better Fight
11. Shotgun Rider - Seth Cook
12. Shotgun Rider - Arabella Jones
13. Shotgun Rider - Coal Mountain Band
14. Shotgun Rider - Tim McGraw
15. Shotgun Rider - Shane Lee
16. Something in the Water - Carrie Underwood
17. Something in the Water - Maddie Wilson

-
- 18. Something in the Water - Anthem Lights
 - 19. Something in the Water - Marina Morgan
 - 20. Something in the Water - Arabella Jones
 - 21. This is How We Roll - Florida Georgia Line
 - 22. This is How We Roll - Buddy Brown
 - 23. This Is How We Roll - Florida Georgia Line Live
 - 24. This is How We Roll - Jerrad Hayes
 - 25. This is How We Roll - Arabella Jones
 - 26. Break Free - Ariana Grande
 - 27. Break Free - Craig Yopp
 - 28. Break Free - Hannah Emerson
 - 29. Break Free - Leroy Sanchez
 - 30. Break Free - Sam Tsui
 - 31. Clarity - Alex Goot
 - 32. Clarity - GLEE
 - 33. Clarity - Our Last Night
 - 34. Clarity - Tanner Patrick
 - 35. Clarity - Zedd
 - 36. Get Lucky - Ivan Radenov
 - 37. Get Lucky - Daft Punk
 - 38. Get Lucky - Miracles of Modern Science
 - 39. Get Lucky - Nicole Cross
 - 40. Get Lucky - Macy Kate
 - 41. Sweet Nothing - Madilyn Bailey
 - 42. Sweet Nothing - Calvin Harris
 - 43. Sweet Nothing - The Score
 - 44. Sweet Nothing - Robert Caroll
 - 45. Sweet Nothing - Patrick Lentz
 - 46. Under Control - Calvin Harris
 - 47. Under Control - Gabriele Giudici
 - 48. Under Control - Foster The People
 - 49. Under Control - Pete Lunn

-
50. Under Control - Beth
51. Ho Hey - All Instruments riptard
52. Ho Hey - AJR
53. Ho Hey - Alyssa Bernal
54. Ho Hey - Boyce Avenue
55. Ho Hey - The Lumineers
56. I Will Wait - Daniel Ma
57. I Will Wait - Mumford Sons
58. I Will Wait - The Goodnight
59. I Will Wait - Patrick Lentz
60. I Will Wait - Hannah Trigwell
61. King And Lionheart - Megan Collins
62. King And Lionheart - Brett & Casey
63. King And Lionheart - Brandon Li
64. King And Lionheart - Of Monsters and Men (Live on KEXP)
65. King And Lionheart - Of Monsters and Men
66. Like a Rolling Stone - Green Day
67. Like a Rolling Stone - John Mayer
68. Like a Rolling Stone - Bob Dylan
69. Like a Rolling Stone - Mumbrew
70. Like a Rolling Stone - Vulgo
71. Little Talks - Alex and Sierra -XFactor
72. Little Talks - Alexi Blue
73. Little Talks - Julia Sheer
74. Little Talks - Daniela Andrade
75. Little Talks - Of Monsters and Men
76. Counting Stars - Alex Goot
77. Counting Stars - Gardiner Sisters
78. Counting Stars - Jake Coco Corey Gray Alexi Blue
79. Counting Stars - One Republic
80. Counting Stars - Youngbok Gomez
81. Halo - Ane Brun
82. Halo - Beyonce

-
- | | |
|--|----------------------------------|
| 83. Halo - Catie Lee | 100. Skyfall - Our Last Night |
| 84. Halo - Lisa Lavie | 101. Adorn - Fourtunate |
| 85. Halo - Sam Tsui | 102. Adorn - Miguel |
| 86. Payphone - Maroon 5 | 103. Adorn - Rudy Currence |
| 87. Payphone - Alex Goot feat Eppic | 104. Adorn - GBLIVE in San Mateo |
| 88. Payphone - Boyce Avenue | 105. Adorn - Travis Garland |
| 89. Payphone - Tanner Patrick | 106. All of Me - Boyce Avenue |
| 90. Payphone - Tyler Ward feat Katy McAllister | 107. All of Me - Luciana Zogbi |
| 91. Roar - Katy Perry | 108. All of Me - Justin Rhodes |
| 92. Roar - Alex G | 109. All of Me - Before You Exit |
| 93. Roar - Alex Goot and Sam Tsui | 110. All of Me - John Legend |
| 94. Roar - Boyce Avenue | 111. Price Tag - Jessie J |
| 95. Roar - Tanner Patrick | 112. Price Tag - Jessica Hammond |
| 96. Skyfall - Adele | 113. Price Tag - Mariana Nolasco |
| 97. Skyfall - Mariangeli | 114. Price Tag - Chiara Grispo |
| 98. Skyfall - Amy Guess | 115. Price Tag - Stela Petrova |
| 99. Skyfall - Helena Maria | 116. Problem - Carson Lueders |

-
117. Problem - Ariana Grande
118. Problem - Pentatonix
119. Problem - Jessica Sanchez
120. Problem - Kurt Schneider
121. Pusher Love Girl - ReggieWillMusic
122. Pusher Love Girl - Justin Timberlake(Ellen -Live 2013)
123. Pusher Love Girl - Todrick Hall
124. Pusher Love Girl - Pentatonix
125. Pusher Love Girl - The Shadowboxers
126. Clocks - Alicia Keys
127. Clocks - Andrew and Kitch
128. Clocks - Kevin Laurence
129. Clocks - Sofia Nicole
130. Clocks - Coldplay
131. Iris - The Goo Goo Dolls
132. Iris - Alex Goot
133. Iris - Austin and Diana Hein
134. Iris - Boyce Avenue
135. Iris - The Wanted
136. Raise Your Glass - GLEE
137. Raise Your Glass - Michelle Chamuel
138. Raise Your Glass - Pink
139. Raise Your Glass - Gavin Mikhail
140. Raise Your Glass - Megan Nicole and Jason Chen
141. Times Like These - boldtben
142. Times Like These - Fair Strikes
143. Times Like These - Foo Fighters
144. Times Like These - Shinedown
145. Times Like These - Daisy & Nino
146. Use Somebody - Boyce Avenue
147. Use Somebody - Kings Of Leon
148. Use Somebody - Lascel Wood
149. Use Somebody - Paramore
150. Use Somebody - Matt Beilis

Bibliografia

- Alligood, K. T., T. D. Sauer e J. A. Yorke (2000). *Chaos: An Introduction to Dynamical Systems*. Corrected edition. New York: Springer. 604 pp.
- Anglade, A., R. Ramirez e S. Dixon (2009). “First-order logic classification models of musical genres based on harmony”. Em: *6th Sound and Music Computing Conference*.
- Ankerst, M., S. Berchtold e D. Keim (1998). “Similarity clustering of dimensions for an enhanced visualization of multidimensional data”. Em: *IEEE Symposium on Information Visualization, 1998. Proceedings*.
- Applegate, D. et al. (1998). *On the solution of traveling salesman problems*. Vol. Extra Volume. Rheinische Friedrich-Wilhelms-Universität Bonn.
- Bai, M. R. e G.-Y. Shih (2007). “Upmixing and Downmixing Two-channel Stereo Audio for Consumer Electronics”. Em: *IEEE Transactions on Consumer Electronics* 53.3.
- Bogdanov (2013). “Semantic audio content-based music recommendation and visualization based on user preference examples”. Em: *Information Processing & Management* 49.1.
- Bogdanov, D. et al. (2013). “Essentia: An audio analysis library for music information retrieval”. Em: *Proceedings of ISMIR*.
- Bonada, J. (2000). “Automatic technique in frequency domain for near-lossless time-scale modification of audio”. Em: *Proceedings of International Computer Music Conference*.
- Bostock, M., V. Ogievetsky e J. Heer (2011). “D3 Data-Driven Documents”. Em: *Visualization and Computer Graphics, IEEE Transactions on* 17.12.
- Breiman, L. (2001). “Random Forests”. Em: *Machine Learning* 45.1.
- Caro, L. D., V. Frias-Martinez e E. Frias-Martinez (2010). “Analyzing the Role of Dimension Arrangement for Data Visualization in Radviz”. Em: *Advances in Knowledge Discovery and Data Mining*. Ed. por M. J. Zaki et al. Lecture Notes in Computer Science 6119. Springer Berlin Heidelberg.
- Chediak, A. (1986). *Harmonia & Improvisação-Vol. 1*. Vol. 1. Irmãos Vitale.

- Chen, Q. et al. (2010). "Analysis of Music Representations of Vocal Performance Based on Spectrogram". Em: *2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*.
- Cooper, M. e J. Foote (2003). "Summarizing popular music via structural similarity analysis". Em: *Applications of Signal Processing to Audio and Acoustics, IEEE*.
- Cui, W. et al. (2011). "TextFlow: Towards Better Understanding of Evolving Topics in Text". Em: *IEEE Transactions on Visualization and Computer Graphics* 17.12.
- Dalhuijsen, L. e L. van Velthoven (2010). "MusicalNodes: The Visual Music Library". Em: *Proceedings of the 2010 International Conference on Electronic Visualisation and the Arts*. EVA'10. Swinton, UK, UK: British Computer Society.
- Daniels, K. et al. (2012). "Properties of normalized radial visualizations". Em: *Information Visualization*.
- Deezer (2015). *Recursos Deezer*. Deezer. URL: <http://www.deezer.com/features> (acesso em 02/07/2015).
- Doulis, M., M. Soldati e A. Csillaghy (2007). "SphereViz - Data Exploration in a Virtual Reality Environment". Em: *Information Visualization, 2007. IV '07. 11th International Conference*.
- Echonest, T. (2013). *The Echonest API*. URL: <http://the.echonest.com/> (acesso em 10/11/2013).
- Eckmann, J.-P., S. O. Kamphorst e D. Ruelle (1995). "Recurrence Plots of Dynamical Systems". Em: Ruelle, D. *Turbulence, Strange Attractors And Chaos*. Vol. 16. WORLD SCIENTIFIC.
- Ellis, D. P. W. (2007). *The "covers80" cover song data set*. URL: <http://labrosa.ee.columbia.edu/projects/coversongs/covers80/> (acesso em 06/11/2013).
- Evgeniou, T. e M. Pontil (2004). "Regularized MultiTask Learning". Em: *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '04. New York, NY, USA: ACM.
- Fadel, S. G. et al. (2015). "LoCH: A neighborhood-based multidimensional projection technique for high-dimensional sparse spaces". Em: *Neurocomputing* 150, Part B.
- Foote, J. (2000). "Automatic audio segmentation using a measure of audio novelty". Em: *Multimedia and Expo, IEEE International Conference on*. Vol. 1. IEEE.
- Fu, Z. et al. (2011). "A Survey of Audio-Based Music Classification and Annotation." Em: *IEEE Transactions on Multimedia* 13.2.
- Fujishima, T. (1999). "Realtime Chord Recognition of Musical Sound : a System Using Common Lisp Music". Em: *Proc. ICMC, 1999*.

- Furnas, G. W. (1986). "Generalized fisheye views". Em: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '86. New York, NY, USA: ACM.
- Geng, X., D.-C. Zhan e Z.-H. Zhou (2005). "Supervised nonlinear dimensionality reduction for visualization and classification". Em: *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 35.6.
- Gomez, E. G. (2006). "Tonal description of music audio signals". Tese de doutorado. Barcelona: Universitat Pompeu Fabra.
- Goto, M. (2003). "A chorus-section detecting method for musical audio signals". Em: *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)*. Vol. 5.
- Hadlak, S., H. Schulz e H. Schumann (2011). "In Situ Exploration of Large Dynamic Networks". Em: *IEEE Transactions on Visualization and Computer Graphics* 17.12.
- Han, W. et al. (2006). "An efficient MFCC extraction method in speech recognition". Em: *2006 IEEE International Symposium on Circuits and Systems, 2006. ISCAS 2006. Proceedings*.
- Hearn, D. e M. P. Baker (1997). *Computer Graphics, C Version*. Prentice Hall. 680 pp.
- Hinum, K. et al. (2005). "Gravi++: Interactive Information Visualization to Explore Highly Structured Temporal Data". Em: *Journal of Universal Computer Science* 11.11. [http://www.jucs.org/jucs_11_11/gravi_interactive_information_visualization].
- Hiraga, R., F. Watanabe e I. Fujishiro (2002). "Music learning through visualization". Em: *Second International Conference on Web Delivering of Music, 2002. WEDELMUSIC 2002. Proceedings*.
- Hoffman, P. et al. (1997). "DNA visual and analytic data mining". Em: *Visualization '97., Proceedings*.
- Homburg, H. et al. (2005). "A Benchmark Dataset for Audio Classification and Clustering." Em: *ISMIR*. Vol. 2005.
- Hunt, M., M. Lennig e P. Mermelstein (1980). "Experiments in syllable-based recognition of continuous speech". Em: *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '80*. Vol. 5.
- Ingram, S. e T. Munzner (2015). "Dimensionality reduction for documents with nearest neighbor queries". Em: *Neurocomputing. Selected papers from the Workshop on Visual Analytics using Multidimensional Projections, held at EuroVis 2013* 150, Part B.
- Joia, P. et al. (2011). "Local Affine Multidimensional Projection". Em: *IEEE Transactions on Visualization and Computer Graphics* 17.12.
- Keogh, E. e C. A. Ratanamahatana (2005). "Exact indexing of dynamic time warping". Em: *Knowledge and Information Systems* 7.3.

- Kostka, S. et al. (1995). *Tonal Harmony with an Introduction to Twentieth-Century Music*. McGraw-Hill.
- Kovalerchuk, B. e V. Grishin (2014). “Collaborative Lossless Visualization of n-D Data by Collocated Paired Coordinates”. Em: *Cooperative Design, Visualization, and Engineering*. Ed. por Y. Luo. Lecture Notes in Computer Science 8683. Springer International Publishing.
- Kreyszig, E. (2011). *Advanced Engineering Mathematics*. 10 edition. Hoboken, NJ: Wiley. 1280 pp.
- Lamere, P. (2012). *Infinite Jukebox*. Versão 1.0. URL: <http://labs.echonest.com/Uploader/index.html> (acesso em 08/11/2013).
- Lerch, A. (2012). *An Introduction to Audio Content Analysis: Applications in Signal Processing and Music Informatics*. John Wiley & Sons. 248 pp.
- Lillie, A. S. (2008). “MusicBox: Navigating the space of your music”. Tese de doutorado. Massachusetts Institute of Technology.
- Maaten, L. Van der e G. Hinton (2008). “Visualizing data using t-SNE”. Em: *Journal of Machine Learning Research* 9.2579.
- Malinowski, S. (2013). *The Music Animation Machine*. URL: <http://www.musanim.com/> (acesso em 07/11/2013).
- Manning, C. D., P. Raghavan e H. Schütze (2008). *Introduction to information retrieval*. Vol. 1. Cambridge University Press Cambridge.
- McCarthy, J. F. et al. (2004). “Applications of Machine Learning and High-Dimensional Visualization in Cancer Detection, Diagnosis, and Management”. Em: *Annals of the New York Academy of Sciences* 1020.1.
- McFee, B. e D. P. Ellis (2014). “Analyzing song structure with spectral clustering”. Em: *ISMIR - International Society for Music Information Retrieval Conference*.
- McKay, C., I. Fujinaga e P. Depalle (2005). “jAudio: A feature extraction library”. Em: *Proceedings of the International Conference on Music Information Retrieval*.
- Muelder, C. e K.-L. Ma (2008). “Rapid Graph Layout Using Space Filling Curves”. Em: *IEEE Transactions on Visualization and Computer Graphics* 14.6.
- Muelder, C., T. Provan e K.-L. Ma (2010). “Content Based Graph Visualization of Audio Data for Music Library Navigation”. Em: *2010 IEEE International Symposium on Multimedia (ISM)*.
- Müller, M. (2007). *Information retrieval for music and motion*. Springer. 313 pp.
- Nigam, K. et al. (2000). “Text classification from labeled and unlabeled documents using EM”. Em: *Machine learning* 39.2.
- Novakova, L. e O. Stepankova (2009). “RadViz and Identification of Clusters in Multidimensional Data”. Em: *Information Visualisation, 2009 13th International Conference*.

- Ono, J. H. P. et al. (2015a). “Concentric RadViz: Visual Exploration of Multi-Task Classification”. Em: *Sibgrapi 2015 (28th Conference on Graphics, Patterns and Images)*.
- Ono, J. H. P. et al. (2015b). “Similarity Graph: Visual Exploration of Song Collections”. Em: *Electronic Proceedings of Sibgrapi 2015 (28th Conference on Graphics, Patterns and Images). 6th Workshop on Visual Analytics, Information Visualization and Scientific Visualization*.
- Oppenheim, A. V., R. W. Schafer, J. R. Buck et al. (1999). *Discrete-time signal processing*. Vol. 5. Prentice Hall Upper Saddle River.
- Orio, N. (2006). “Music Retrieval: A Tutorial and Review”. Em: *Foundations and Trends in Information Retrieval* 1.1.
- O’Shaughnessy, D. (1987). *Speech communication: human and machine*. Addison-Wesley series in electrical engineering. Reading, Mass: Addison-Wesley Pub. Co. 568 pp.
- Pampalk, E. (2001). “Islands of music: Analysis, organization, and visualization of music archives”. Tese de doutorado.
- Pampalk, E., A. Rauber e D. Merkl (2002). “Content-based organization and visualization of music archives”. Em: *Proceedings of the tenth ACM international conference on Multimedia*. MULTIMEDIA ’02. New York, NY, USA: ACM.
- Paulovich, F. et al. (2008). “Least Square Projection: A Fast High-Precision Multidimensional Projection Technique and Its Application to Document Mapping”. Em: *IEEE Transactions on Visualization and Computer Graphics* 14.3.
- Paulovich, F. et al. (2011). “Piece wise Laplacian-based Projection for Interactive Data Exploration and Organization”. Em: *Computer Graphics Forum* 30.3.
- Paulus, J., M. Müller e A. Klapuri (2010). “Audio-Based Music Structure Analysis.” Em: *ISMIR*.
- Pedregosa, F. et al. (2011). “Scikit-learn: Machine Learning in Python”. Em: *Journal of Machine Learning Research* 12.
- Read, J. et al. (2011). “Classifier chains for multi-label classification”. Em: *Machine Learning* 85.3.
- Rennie, J. D. et al. (2003). “Tackling the poor assumptions of naive bayes text classifiers”. Em: *ICML*. Vol. 3. Washington DC).
- Schapire, R. E. e Y. Singer (2000). “BoosTexter: A Boosting-based System for Text Categorization”. Em: *Machine Learning* 39.2.
- Seifert, C. e M. Granitzer (2010). “User-Based Active Learning”. Em: *2010 IEEE International Conference on Data Mining Workshops (ICDMW)*.
- Serra, J. et al. (2008). “Chroma Binary Similarity and Local Alignment Applied to Cover Song Identification”. Em: *IEEE Transactions on Audio, Speech, and Language Processing* 16.6.

- Serrà, J., X. Serra e R. G. Andrzejak (2009). “Cross recurrence quantification for cover song identification”. Em: *New Journal of Physics* 11.9.
- Sharko, J., G. Grinstein e K. A. Marx (2008). “Vectorized radviz and its application to multiple cluster datasets”. Em: *Visualization and Computer Graphics, IEEE Transactions on* 14.6.
- Spotify (2015). *Spotify Information*. Spotify Press. URL: <https://press.spotify.com/au/information/> (acesso em 02/07/2015).
- Stevens, S. S., J. Volkmann e E. B. Newman (1937). “A Scale for the Measurement of the Psychological Magnitude Pitch”. Em: *The Journal of the Acoustical Society of America* 8.3.
- Tejada, E., R. Minghim e L. G. Nonato (2003). “On Improved Projection Techniques to Support Visual Exploration of Multi-Dimensional Data Sets”. Em: *Information Visualization* 2.4.
- Tenenbaum, J. B., V. d. Silva e J. C. Langford (2000). “A Global Geometric Framework for Nonlinear Dimensionality Reduction”. Em: *Science* 290.5500.
- Teoh, S. T. e K.-L. Ma (2003). “PaintingClass: Interactive Construction, Visualization and Exploration of Decision Trees”. Em: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '03. New York, NY, USA: ACM.
- Torgerson, W. S. (1952). “Multidimensional scaling: I. Theory and method”. Em: *Psychometrika* 17.4.
- Torrens, M., P. Hertzog e J. L. Arcos (2004). “Visualizing and Exploring Personal Music Libraries.” Em: *ISMIR*.
- Tzanetakis, G. e P. Cook (2002). “Musical genre classification of audio signals”. Em: *IEEE Transactions on Speech and Audio Processing* 10.5.
- Ware, C. (2013). *Information visualization: perception for design*. Third edition. Interactive technologies. Waltham, MA: Morgan Kaufmann. 512 pp.
- Wattenberg, M. (2002). “Arc diagrams: Visualizing structure in strings”. Em: *Information Visualization, 2002. INFOVIS 2002. IEEE Symposium on*. IEEE.
- Withall, M., I. Phillips e D. Parish (2007). “Network visualisation: a review”. Em: *IET Communications* 1.3.
- Yknk (2012). *MIDITrail*. URL: <http://en.sourceforge.jp/projects/miditrail/> (acesso em 05/11/2013).
- Zhang, C.-T. et al. (2010). “Phase space reconstruction and prediction of multivariate chaotic time series”. Em: *2010 International Conference on Machine Learning and Cybernetics (ICMLC)*. Vol. 5.