Uso de técnicas de navegação em árvores para auxílio na visualização de dados multidimensionais

Marcel Yugo Nakazaki

Uso de técnicas de navegação em árvores para auxílio na visualização de dados multidimensionais

Marcel Yugo Nakazaki

Orientador(a): Profa. Dra. Rosane Minghim

Monografia apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC/USP, como parte dos requisitos para a obtenção do título de Mestre na Área de Ciências de Computação e Matemática Computacional.

USP – São Carlos/SP
Fevereiro/2010

Agradecimentos

Agradeço a toda a minha família, especialmente à minha Mãe, que sempre esteve ao meu lado, me apoiando e me dando forças para continuar sempre seguindo em frente. À minha namorada Lika, que com seu Amor e Carinho, sempre me traz conforto e vontade de continuar batalhando. Sem vocês eu não teria chegado aqui.

Agradeço à Prof. Rosane, que me aceitou como aluno e depositou sua confiança e toda a sua paciência, me ensinando diversas coisas novas. Pelo exemplo de competência, capacidade e seriedade nos momentos em que mais precisei.

Agradeço também ao CNPq pelo auxílio financeiro que foi de grande importância para início dos trabalhos do mestrado.

Resumo

Com base em métodos de extração de características de imagens e extração de vocabulários de textos, podemos empregar técnicas para posicionamento de dados multidimensionais no plano para mapear conjuntos de dados em espaços visuais, auxiliando usuários na interpretação e análise dos dados. Alguns desses métodos constroem árvores de similaridade, impondo uma hierarquia sobre as relações entre as características extraídas dos dados.

Em um ambiente de análise exploratória, é natural que se procurem métodos e técnicas capazes de manipular e interagir com os dados de forma rápida e eficiente.

Nesse contexto, o trabalho visa implementar e aplicar técnicas conhecidas de navegação e interação em árvores no contexto de visualizações baseadas em posicionamento de pontos no plano. Em particular as técnicas *NJ* e *MST*, implementadas e utilizadas com sucesso na ferramenta *PEx-Image*, tornaram-se pontos chave para o auxílio na exploração dos dados através das apresentações radial e hiperbólica, implementadas juntamente com ferramentas de exploração.

Este trabalho implementa e apresenta a capacidade exploratória dessas duas formas de apresentação de árvores sobre as visualizações *NJ* e *MST*.

Abstract

Based on methods of feature extraction for images and vocabulary exploration for text, we can apply point placement techniques to multidimensional data in order to map data sets into visual spaces, assisting users on data analysis and interpretation. Some of these methods build similarity trees, imposing a hierarchy on the relationship between the characteristics extracted from data.

In an exploratory analysis environment, it is natural to use methods and techniques capable of manipulating and interacting data quickly and efficiently.

In this context, this paper aims to implement and apply known techniques of tree navigation and interaction in the context of point placement visualizations. In particular the *NJ* and *MST* techniques, implemented and successfully used in the system *PEx-Image*, are the main focus for helping data exploration through Radial and Hyperbolic Layouts, implemented with exploration tools.

This work implements Radial and Hyperbolic layouts to support exploration of *NJ* and *MST* views.

Sumário

Lista de Figuras

Lista de Siglas

BLAST	Basic Local Alignment Search Tool
CP	Componente Principal
CT	Tomografia Computadorizada por Raios-X
FA	Factor Analysis
FDP	Force-Directed Placement
FMRP	Faculdade de Medicina de Ribeirão Preto
InCor	Instituto do Coração do Hospital das Clínicas
LSP	Least Square Projection
MDS	Multidimensional Scaling
MRI	Ressonância Magnética
MST	Minimum Spanning Tree
NCD	Normalized Compression Distance
NJ	Neighbor-Joining
NNP	Nearest-Neighbor Projection
PCA	Principal Component Analysis
PDF	Portable Document Format
PET	Tomografia por Emissão de Pósitrons
PEx	Projection Explorer
PEx-Image	Projection Explorer for Images
PP	Projection Pursuit
ProjClus	Projection by Clustering
SPECT	Tomografia por Emissão de Fótons
SVD	Singular Value Decomposition
TME	Text Map Explorer
US	Ultra-Sonografia
XML	Extensible Markup Language

Capítulo

1

Introdução

1.1 Contexto e Motivação

Buscando meios de armazenar, recuperar, compreender e manipular grandes conjuntos de dados, identificando relacionamentos ou tendências presentes nas informações disponíveis, tanto para resoluções de problemas quanto para tomada de decisões importantes, surgiu a área de visualização computacional (Wong, 1999).

Informações exibidas visualmente são melhor e mais rapidamente interpretadas pelo ser humano do que em qualquer outra forma. É natural, portanto, que se procurem métodos e técnicas capazes de apresentar os resultados da manipulação de análise de dados em um formato gráfico, exibindo os relacionamentos entre os elementos de informação de maneira clara e objetiva. Há então a necessidade de utilizarmos técnicas que facilitem o usuário a navegar e explorar os dados de forma fácil e rápida.

Nesse contexto, o presente trabalho busca adicionar, às ferramentas de visualização desenvolvidas pelo grupo de visualização do ICMC-USP, diferentes formas de interação interação, auxiliando o usuário na análise, navegação e interpretação de dados multidimensionais.

Este trabalho estende resultados anteriores do grupo em visualização de informação baseada na interação e navegação em visualizações baseadas em árvores de similaridade obtidas através de posicionamento de pontos no plano, aplicadas a conjunto de imagens e textos.

Uma ferramenta de visualização de dados multidimensionais previamente desenvolvida, denominada *PEx-Image (Projection Explorer for Images)* e com foco principal em coleção de imagens, foi adaptada para os requisitos deste trabalho. Técnicas disponíveis na ferramenta

geram visualizações por mapeamento de pontos no plano através de projeções multidimensionais e árvores de similaridade. Em particular, este trabalho implementa o *layout* e a navegação em árvores Radial e Hiperbólica para permitir flexibilidade na exploração de árvores de similaridade. A utilidade dessas abordagens para exploração é evidenciada na forma de estudos de casos.

1.2 Organização da Dissertação

Nesta dissertação são apresentados os conceitos necessários para mostrar o trabalho desenvolvido e seus resultados, sendo estruturada da seguinte forma:

- No Capítulo 2 é apresentada uma introdução às áreas de visualização científica e visualização de informação.
- No Capítulo 3 são apresentadas técnicas de visualização multidimensional, destacando as técnicas desenvolvidas no grupo de pesquisa de visualização do ICMC, seguidas da introdução das duas técnicas de desenho de árvores aplicadas neste trabalho.
- No Capítulo 4 é apresentada a ferramenta base para o estudo das técnicas empregadas para auxílio na navegação das visualizações, seguida da descrição detalhada das técnicas implementadas. Finalmente, são apresentados estudos de casos que utilizam dados de múltiplas instâncias, tais como textos de notícias, imagens médicas e sequências genéticas.

Capítulo

2

Técnicas de Visualização

2.1 Considerações Iniciais

Uma enorme quantidade de informações de diversos tipos estão sendo geradas, armazenadas e disseminadas, levantando questões sobre como tornar tais informações úteis. A necessidade de entender e extrair conhecimento de informações armazenadas está se tornando uma tarefa presente em diversas áreas de pesquisa e é de grande importância e utilidade tanto para os pesquisadores quanto para as próprias pessoas envolvidas com a extração de conhecimento, seja para tomada de decisões ou para o estudo e exploração dos mais diversos fenômenos.

Seres humanos procuram, por natureza, estruturas, padrões, tendências, anomalias e relações entre dados. Seguindo este propósito, surgiu a *Visualização Computacional*, uma área de pesquisa que estuda estratégias e algoritmos para mapear informações em representações gráficas legíveis para o ser humano, possibilitando uma melhor compreensão do conteúdo de grandes conjuntos de dados e dos fenômenos que os geram, envolvendo, portanto, a transformação e mapeamento de dados em objetos gráficos, além da sua exploração por meios computacionais (Minghim e Oliveira, 1997).

Existem várias definições de Visualização. No seu sentido mais amplo, a palavra significa a geração de imagens mentais para organização e entendimento de um conceito, ideia ou informação. O problema fundamental na visualização de dados é encontrar uma representação gráfica para um conjunto de dados que reflita seu conteúdo e significado. Tal tarefa depende da classe de aplicações para a qual uma técnica específica é utilizada, podendo basear-se nos dados a serem tratados, nas dimensões das representações ou nas estruturas intrínsecas aos dados. Nesta linha, podemos citar duas grandes sub-áreas da visualização (Levkowitz e Oliveira, 2003):

1. Visualização Científica: Aplicação das técnicas gráficas para ampliar a capacidade de interpretação de dados medidos e simulados através de experimentos científicos e de engenharia, possuindo sempre um atributo espacial inerentemente associado.

2. Visualização de Informação: Engloba o desenvolvimento de técnicas de visualização de dados multidimensionais que não possuem geometria intrínseca ou natureza espacial bem definida.

Visualização, portanto, envolve aspectos em áreas da computação gráfica, interação humano-computador, processamento de imagens, ciência cognitiva e processamento de sinais. Formalmente estas áreas são independentes; no entanto, uma convergência está sendo formada pelo uso de técnicas análogas em diferentes áreas. Além disso, a visualização intersecta os objetivos de outras áreas de pesquisa que se ocupam da análise de dados complexos, tais como estatística e mineração de dados. Sendo assim, as características da visualização a tornam uma ferramenta viável e muitas vezes essencial para diversas áreas de aplicação.

Nas seções seguintes, são introduzidos conceitos e técnicas básicas das áreas de Visualização Científica e Visualização de Informação, sendo que para o perfeito entendimento destas, admite-se que o leitor tenha domínio de conceitos básicos de computação, computação gráfica e processamento de imagens.

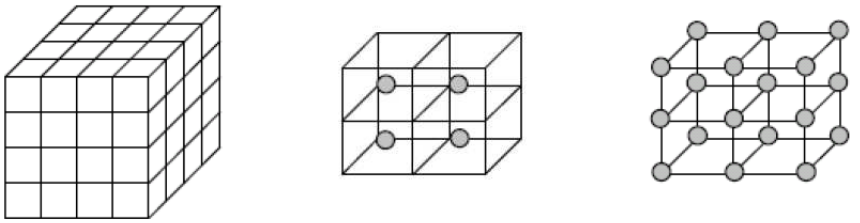
2.2 Visualização Científica

Visualização Científica é uma área preocupada com a exploração de dados e informações de forma gráfica. Normalmente seus dados já possuem uma geometria implícita ou intuitiva associada e representam fenômenos de natureza científica, tais como fenômenos físicos, químicos ou biológicos.

Este tipo de visualização têm sido uma importante área de pesquisa e, nos últimos anos, atingiu um grau de maturidade considerável, tendo como principal sub-área a *Visualização Volumétrica*. Técnicas de visualização volumétrica têm sido desenvolvidas para proporcionar uma melhor compreensão de conjuntos de dados que possuam uma representação tridimensional inerente, oferecendo técnicas para manipulação de malhas multidimensionais e de apresentação em um espaço tridimensional (Elvins, 1992). Tais dados são obtidos por amostragens, simulações e técnicas de modelagem. Originalmente, dados volumétricos eram visualizados através de aproximações poligonais das isosuperfícies, obtidas por interpolação dentro dos elementos de volume, também chamadas de *voxels*[1]. A Figura 2.1 ilustra a organização típica desses dados.

No entanto, o principal problema das técnicas de interpolação é detectar quando uma isosuperfície passa por um *voxel*. Além das naturais ambiguidades de reconstrução, as estruturas geométricas extraídas são grandes e complexas e muita informação é perdida no processo.

[1]Os dados volumétricos são geralmente tratados como uma matriz de elementos de volume, sendo cada elemento denominado *voxel* (análogo 3D de um *pixel*).



(a) Malha regular volumétrica (b) Malha de voxels (c) Malha de células

Figura 2.1: Células através de interpolação (Adaptada de (Paiva et al., 1999)).

Uma outra forma de visualização volumétrica foi então apresentada por Levoy (Levoy, 1990), como uma técnica que permitia a visualização de pequenos detalhes internos ao volume, através do controle de transparência dos *voxels*, removendo trivialmente as partes escondidas atrás de partes definidas como opacas e visualizando o volume a partir de qualquer direção. Sua ideia principal era conseguir uma técnica de visualização eficiente e precisa, que conseguisse "sintetizar" todas as informações contidas em um conjunto de dados volumétricos em uma única imagem, de forma que se tivesse a impressão de estar olhando para dados reais. Neste caso, ao contrário das técnicas existentes anteriormente, não são extraídas estruturas geométricas intermediárias.

Posteriormente, esta técnica recebeu o nome de *Visualização Volumétrica Direta (Direct Volume Rendering)*, tendo como algoritmo principal de *Ray-casting* (Drebin et al., 1988). No entanto, os algoritmos para visualização volumétrica direta possuem um alto custo computacional, uma vez que necessitam percorrer todo o volume de dados, tal como matrizes de dimensões 256x256x160 contendo um valor escalar a cada posição de visualização escolhida de observação.

Essas duas abordagens básicas (extração de superfícies e *rendering* direto de volumes) diferem basicamente pela utilização ou não de representações intermediárias dos dados volumétricos para a geração da visualização adequada à aplicação. Enquanto no *rendering* direto de volumes a projeção é realizada diretamente a partir dos dados volumétricos, na extração de superfícies os dados volumétricos são convertidos para uma representação geométrica (polígonos) a partir da qual são usados os métodos tradicionais de *rendering* de polígonos para geração da visualização. A Figura 2.2 apresenta uma ideia esquemática da conexão entre estas técnicas.

As técnicas envolvidas no processo de visualização volumétrica (Figura 2.2) podem ser resumidas na execução de quatro passos básicos, definidos na Figura 2.3. No entanto, a visualização é realizada através da implementação apenas dos três últimos passos, pois consideramos apenas a visualização de volumes já pré-processados.

A seguir, são apresentadas, a título de ilustração didática, técnicas clássicas de visualização volumétrica.

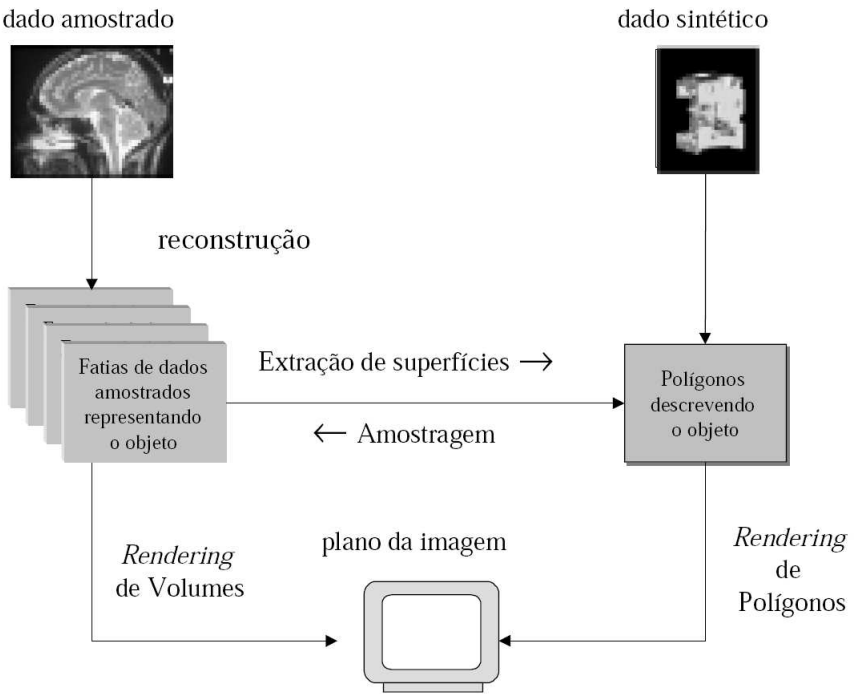


Figura 2.2: Representação do método de visualização volumétrica (Retirada de (Paiva et al., 1999)).

2.2.1 Técnicas de Visualização Volumétrica

Na literatura, vários termos e expressões são utilizados para caracterizar as diferentes classes de técnicas de visualização de volumes. Dentre as nomenclaturas que permanecem em uso, Elvins (Elvins, 1992) considera duas categorias de algoritmos: (1) *Direct Volume Rendering* e (2) *Surface-fitting*, enquanto Brodlie et al. (Brodlie et al., 1992) reconhece essas duas classes como dois grupos de algoritmos: (1) *Direct Volume Rendering* e (2) *Indirect Volume Rendering*. Entretanto, apesar das diferentes classificações e nomenclaturas, as duas categorias de técnicas de visualização de volumes se traduzem nas que trabalham com a extração de uma isosuperfície representada por primitivas gráficas, e nas que trabalham gerando a imagem diretamente a partir do volume, conforme citado na Seção 2.2.

A seguir, usaremos a categorização de Brodlie et al. para mostrar as diferentes classes de técnicas de visualização de volumes.

2.2.1.1 Visualização Volumétrica Indireta

Extração de isosuperfícies é uma importante ferramenta para visualização de campos escalares multidimensionais. Ao expor contornos de valores constantes, isosuperfícies oferecem mecanismos para compreensão da estrutura do campo escalar. Tais contornos isolam superfícies de interesse, centrando a atenção sobre importantes características nos dados, tal como as fronteiras entre os diferentes tipos de materiais. Esta técnica mostra indiretamente os dados

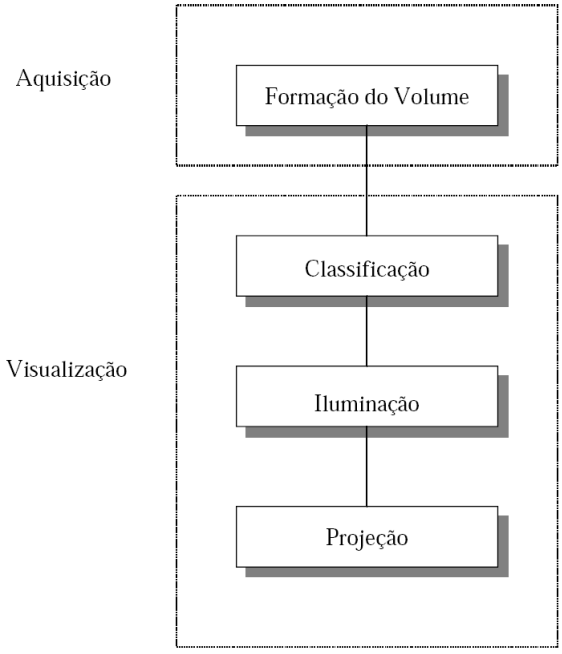


Figura 2.3: *Pipeline* de visualização volumétrica (Retirada de (Paiva et al., 1999)).

originais modelando-os com a ajuda de um algoritmo de *surface-fitting*. A seguir, são descritos alguns algoritmos que implementam técnicas de extração de isosuperfícies.

Marching Cubes O algoritmo de extração de superfícies mais popular e um dos mais estudados é o *Marching Cubes* (Lorensen e Cline, 1987). Este algoritmo é muito empregado, ainda hoje, em diversas aplicações médicas. Seu funcionamento básico é: dado um valor de limiar, as células[2] que contribuem para formação da superfície são pesquisadas. Foram analisadas 256 possíveis configurações das intersecções dos triângulos (no máximo quatro triângulos por célula) com uma célula cúbica a partir dos valores dos seus vértices e por argumentos de simetria (reflexão e rotação). Estes casos foram reduzidos para 15, como mostra a Figura 2.4. Podemos resumir as operações desse algoritmo em quatro passos:

1. Detecção dos vértices cujos valores estão acima do limiar e cálculo de um índice para uma tabela de intersecção de bordas, definindo a configuração dos triângulos dentro da célula.
2. Definição dos vértices dos triângulos por interpolação linear entre os valores dos vértices das células.
3. Cálculo dos gradientes em cada vértice das células para utilização no processo de sombreamento.
4. *Rendering* do modelo tridimensional (3D).

[2]Um cubo formado por oito voxels vizinhos como vértices; quatro da fatia k e quatro da fatia $k+1$.

Uma vez extraída a superfície, a visualização segundo qualquer ponto de observação torna-se rápida.

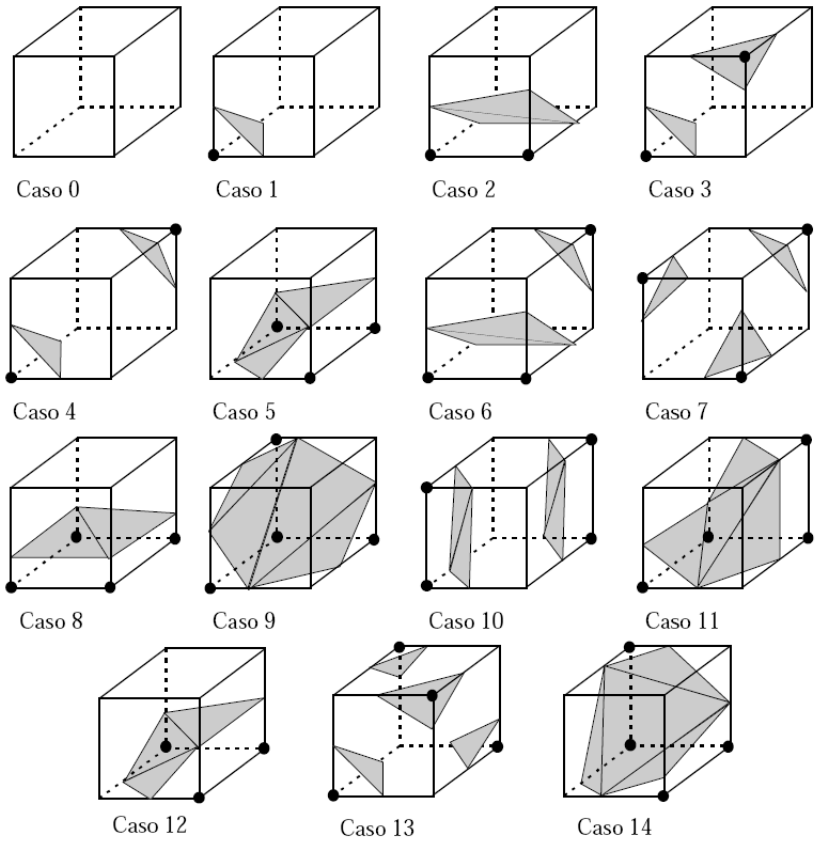


Figura 2.4: Configurações básicas para triangulação de uma única célula (Retirada de (Minghim e Oliveira, 1997)).

Dividing Cubes Lorensen e Cline (Lorensen e Cline, 1987) descobriram que o tamanho de alguns triângulos gerados pelo algoritmo *Marching Cubes* era menor que o tamanho de um *pixel*. Um novo algoritmo, denominado *Dividing Cubes* foi então desenvolvido, tirando vantagem desta observação. O algoritmo efetua a projeção das células na tela, verificando somente aquelas em que a projeção é maior que um *pixel*. Se isso ocorrer, a célula é dividida em sub-células, cada uma das quais determinando um ponto. Caso contrário, a célula toda é visualizada também como um ponto.

2.2.1.2 Visualização Volumétrica Direta

Técnicas de visualização volumétrica direta têm sido desenvolvidas para obter melhor compreensão de conjuntos de dados tridimensionais, obtidos por simulações de elementos físicos ou métodos de engenharia. Originalmente, dados volumétricos eram visualizados através de aproximações poligonais das isosuperfícies, obtidas por interpolação dentro dos elementos de

volume *(voxels)*. Sua principal vantagem é a visualização direta do volume, sem usar representações geométricas intermediárias (Kaufman, 1998), evitando problemas de detecção imprecisa que geraram inconsistências geométricas e topológicas, apresentadas visualmente como falhas ou superfícies espúrias (Elvins, 1992).

Para isso, o volume de dados é varrido, acumulando os valores de opacidades calculados, até chegar ao fim do volume ou atingir um valor máximo de opacidade.

A seguir, são descritos sucintamente o algoritmo de *Splatting* (Westover, 1990) e o algoritmo de *Ray-casting* (Drebin et al., 1988).

Splatting O algoritmo *Splatting* (Westover, 1990) é inspirado na estrutura do *pipeline* de *rendering* de polígonos, em que cada primitiva passa ao longo dos vários estágios, um por vez. Neste algoritmo, cada elemento é mapeado no plano da tela. Em seguida, através de um processo de acumulação, tem sua contribuição adicionada à formação da imagem. O algoritmo termina quando todas as primitivas tiverem sido mapeadas na tela. Podemos definir o algoritmo de *Splatting* em quatro etapas principais: (1) transformação, (2) classificação/iluminação, (3) reconstrução e (4) visibilidade.

Ray-Casting Um algoritmo de visualização volumétrica bastante conhecido é o *Ray-casting* (Drebin et al., 1988; Tide et al., 1990; Ney et al., 1990; Stytz et al., 1991; Levoy et al., 1990). Como a maioria dos algoritmos de visualização volumétrica, o *Ray-casting* constrói todo o volume contínuo do conjunto de dados discretos por meio de alguma função de interpolação de grau maior ou igual a zero. Essa função é reamostrada e projetada na tela 2D produzindo a imagem final. Esse algoritmo usa a técnica *Image-order* (Levoy, 1998, 1990), que dispara raios partindo dos *pixels*. Estes raios atravessam todo o volume somando os valores de cor e opacidade ao longo dos raios e, assim, definindo uma contribuição dos *pixels* na imagem final. Outros algoritmos podem empregar uma técnica oposta, no sentido do volume para a imagem *(Object-order)* (Udupa e Odhner, 1993; Westover, 1990), ou uma combinação das duas técnicas.

2.3 Visualização de Informação

Visualização de Informação pode ser descrita como o uso de representações visuais interativas de dados abstratos apoiadas por computador, com o objetivo de ampliar a *cognição*[3] (Card et al., 1999), combinando mineração de dados, processamento de imagens e interação usuário-computador (Gershon e Eick, 1995; Robertson et al., 1993). Apesar da similaridade com a visualização científica, os dados a serem representados são abstratos, ou seja, não há necessariamente uma representação geométrica inerente a eles. Neste caso, uma imagem deve ser gerada com base nos relacionamentos ou informações que podem ser inferidos acerca dos dados, fornecendo faixas maiores de elementos facilmente distinguíveis pela percepção humana,

[3]Neste contexto, cognição significa aquisição ou uso de conhecimento.

quando comparada aos outros sentidos de percepção, embora a representação de dados abstratos possa ser realizada por meio de sistemas que abordem múltiplas formas de percepção.

2.3.1 Pipeline para Visualização de Informação

Podemos compreender melhor os métodos de visualização de informação através de um modelo formal (Chi e Riedl, 1998), que chamaremos de *pipeline*. Este modelo, descrito pela Figura 2.5, é constituído de 4 fases, das quais 3 são pré-processamento dos dados, que é o processo computacional de converter informação em uma forma visual com a qual o usuário pode interagir (Card et al., 1999). Cada fase deste modelo é executada por um operador que mapeia a representação dos dados de um estágio para uma outra representação do estágio seguinte.

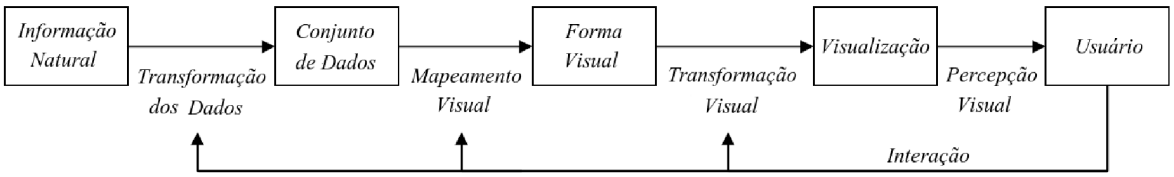


Figura 2.5: *Pipeline* para Visualização de Informação (Adaptada de (Minghim e Oliveira, 1997)).

A primeira fase é transformar os dados brutos em informações organizadas. O formato resultante tipicamente consiste em uma matriz contendo uma série de entidades, possuindo cada uma, um atributo associado. Dados derivados, tais como os promovidos por mineração de dados ou por algoritmos de agrupamento, podem ser úteis no auxílio à transformação dos dados brutos em informações organizadas (Gershon e Eick, 1995). A segunda fase é o mapeamento da matriz resultante em formas visuais. Essa forma contém figuras visuais que correspondem às entidades da matriz. A terceira fase intercala esta forma visual em visualizações, que são responsáveis pela amostragem das formas visuais, oferecendo vários tipos de transformações, tais como navegação e filtragem.

Finalmente, o usuário pode interagir em qualquer um dos passos do *pipeline*, alterando o resultado da visualização e realizando diversas interpretações.

2.3.2 Classificação dos Métodos de Visualização de Informação

Várias tentativas foram feitas para classificar métodos de visualização de informação, a fim de se obter uma visão geral do campo (Levkowitz e Oliveira, 2003). Keim (Keim, 2000, 2002) propõe uma análise dos métodos de visualização a partir de três eixos ortogonais: (1) a técnica de visualização propriamente dita, (2) a técnica de interação e (3) o tipo de dado amostrado. Sua análise é baseada no fato de os métodos de interação (tais como *Zooming* (Bederson e Hollan, 1994), *Linking and Brushing* (Becker e Cleveland, 1987), *Dynamic Distortion* (Leung e Apperley, 1994), etc) poderem ser livremente combinados com técnicas de visualização aplicadas

a diferentes tipos de dados (vetores, árvores, grafos (Herman et al., 2000), texto (Hascoët e Baudouin-Lafon, 2001), etc).

A seguir, são descritos alguns métodos de visualização de informação.

2.3.3 Métodos de Visualização de Informação

As técnicas de visualização de informação já desenvolvidas utilizam representações ou metáforas visuais para exibir graficamente os dados que geralmente não possuem representação direta, óbvia ou natural. Em diversas técnicas, frequentemente os autores buscam inspiração em objetos do mundo real (ou geométricos) para mapear o conjunto de informações. As técnicas de visualização podem utilizar representações visuais unidimensionais (1D), bidimensionais (2D) ou tridimensionais (3D), não necessariamente de acordo com a dimensão do espaço de informação (Luzzardi, 2003).

A escolha de uma técnica de visualização de informação para um determinado conjunto de dados depende de diversos fatores. Assim como no processo da criação de um gráfico em programas de planilhas eletrônicas, no qual são apresentados diversos formatos para criação do gráfico, a construção de ferramentas para visualização de informações segue a mesma lógica. Há ferramentas mais adequadas ou menos adequadas para representar determinados conjuntos de informações.

A seguir, são abordados, para efeito de ilustração, alguns diferentes tipos de representações para visualização dos dados. No Capítulo seguinte, detalharemos técnicas de mapeamento de pontos no plano, e em seguida faremos uma breve introdução à visualização de árvores, que servirão como base para a proposta do trabalho.

Coordenadas Paralelas Inicialmente proposta por Inselberg (Inselberg, 1997) como uma técnica geométrica computacional, e posteriormente contextualizada em visualização de informação (Ferreira e Nascimento, 2005), a técnica de coordenadas paralelas destaca-se justamente pela perspectiva multidimensional conferida à representação visual. Nela, um espaço de dimensão k é mapeado para um espaço visual bidimensional, usando k eixos equidistantes e paralelos a um dos eixos principais (x ou y). Cada eixo representa uma dimensão do conjunto de dados, sobre o qual é mapeado linearmente, do menor ao maior, o intervalo de valores de dados correspondente. Cada item de dado é exibido como uma linha poligonal que intercepta cada eixo no ponto correspondente ao valor do atributo associado ao eixo. Embora simples, esta técnica mostra-se poderosa para identificar diferentes distribuições de dados e dependência funcional entre os atributos.

A Figura 2.6 ilustra um exemplo do uso de coordenadas paralelas. Neste exemplo, é ilustrado um conjunto de indivíduos distribuídos pelo grau de instrução, raça, idade, sexo e valor de renda. Após uma análise rápida, podemos concluir que a quantidade de indivíduos do sexo masculino e pertencentes à raça branca que tiveram um período maior de preparação educacional tendem a ganhar mais em relação aos outros indivíduos.

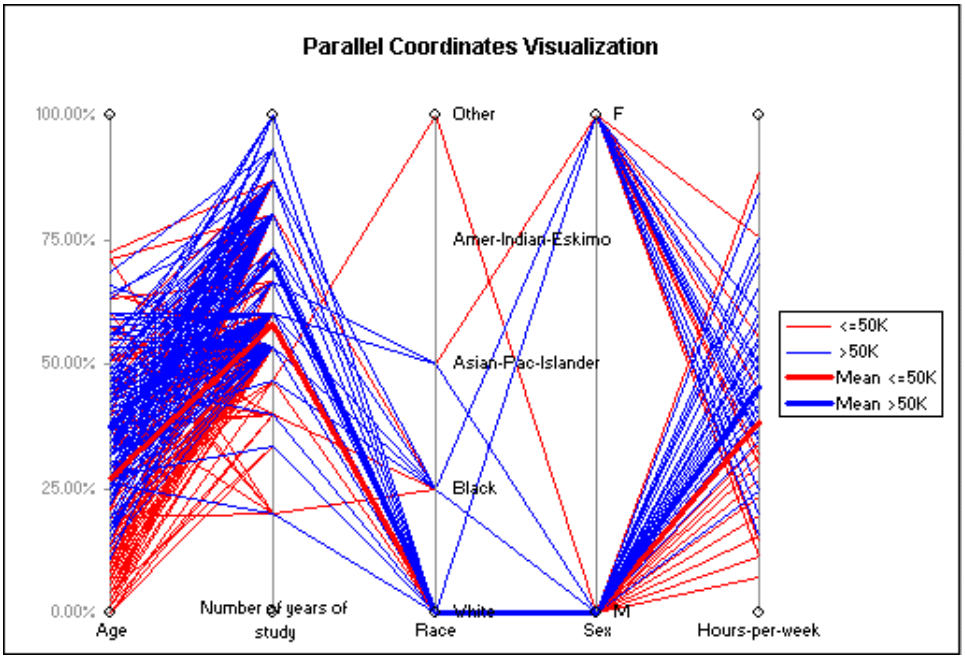


Figura 2.6: Exemplo da utilização de coordenadas paralelas para auxílio na análise de dados (Retirada de http://www.xlstat.com em Fevereiro de 2010).

Técnicas Iconográficas Métodos iconográficos representam cada objeto como um complexo ícone ou figura (Ward, 2002). Famosos exemplos de ícones são as *faces de Chernoff (Chernoff faces)*(Chernoff, 1973): cada objeto é representado por uma pequena face, na qual diferentes dimensões são mapeadas em diferentes características faciais (tamanho dos olhos, tamanho da boca, etc). Outros exemplos incluem *Star-Glyphs* (Siegel et al., 1972) e *Stick-Figures* (Pickett e Grinstein, 1988). No entanto, apesar da facilidade na representação, a comparação entre os objetos se torna difícil, uma vez que o problema de posicionar os objetos não é trivial (Ward, 2002).

As duas técnicas de visualização apresentadas anteriormente são exemplos de visualizações de informação baseadas em características, isto é, as características dos dados são mapeadas para atributos visuais. No próximo capítulo detalharemos uma outra classe de técnicas de visualização de informação que visa representar indivíduos do conjunto no espaço visual, baseando-se em posicionamento de pontos no plano.

2.4 Considerações Finais

Neste capítulo foi apresentado um estudo sobre as áreas de visualização científica e visualização de informação.

A área de visualização é normalmente focada em representar adequadamente dados brutos na forma de imagens, e assim fornecer meios de analisar visualmente conjuntos de dados de elevada dimensionalidade e complexidade, sendo de grande utilidade na descoberta de relacionamentos

e dependências entre os dados. No entanto, a forma como as pessoas percebem e reagem ao resultado da visualização influencia fortemente no entendimento e utilidade dos dados.

Apesar de as técnicas de visualização possuirem um grau de amadurecimento considerável, nota-se que a visualização de informação, ao contrário da científica, ainda não passou a ser uma solução para o usuário final. São necessárias técnicas auxiliares para tratamento dos dados em um domínio sobre uma aplicação específica. Técnicas de mapeamento de pontos no plano agregadas a técnicas de pré-processamento possibilitam o refinamento das visualizações, melhorando os resultados e facilitando o usuário na navegação e exploração dos dados. Uma breve introdução às técnicas e conceitos de mapeamento no plano é apresentada no próximo capítulo.

Capítulo

3

Visualização Multidimensional Baseada em Posicionamento de Pontos no Plano

3.1 Considerações Iniciais

Com o acúmulo na quantidade de informação e a atual dependência de métodos computacionais para armazená-los e consultá-los, áreas de apoio à interpretação das informações desenvolveram-se com o passar dos anos. Em particular, a Mineração de Dados e a Visualização de Informação são áreas que despertam grande interesse da comunidade científica. Por terem objetivos similares e serem complementares, a fusão de ambas em uma área onde a mineração e a visualização co-existem na busca de soluções para interpretação de conjunto de dados complexos é natural. Essa área é denominada *Mineração Visual de Dados* (Wong, 1999).

Um dos maiores problemas encontrados no processo de mineração visual de dados está ligado à dimensionalidade dos dados a serem analisados. Conforme a dimensionalidade aumenta, a identificação de padrões e modelos torna-se cada vez mais complexa e difícil, causando um impacto no processo de interpretação.

Nesse contexto, uma das formas de tratar a alta dimensionalidade é reduzir a dimensão dos dados de forma a ser possível aplicar estratégias e algoritmos que funcionem com dados de menor dimensão. Dentre as possíveis estratégias para a redução de dimensionalidade, as técnicas de *Projeção Multidimensional de Dados* têm despertado grande interesse. Utilizadas para reduzir a

dimensionalidade até dimensão 2 ou 3, elas podem ser empregadas como forma de mapeamento de dados em espaços visuais, com os quais o usuário possa interagir para identificar padrões relevantes nos dados. A seguir, são descritas algumas técnicas de projeção multidimensional, com ênfase em técnicas desenvolvidas pelo grupo de pesquisa ao qual este projeto é vinculado. Em seguida, mapeamentos de dados no plano baseados em árvores de similaridade são discutidos. Ao final, são apresentados conceitos e técnicas de visualização de árvores, que são base para a extensão das técnicas de auxílio na navegação e interação dos dados implementadas neste trabalho.

3.2 Técnicas para Projeção de Dados Multidimensionais

Técnicas de projeção de dados multidimensionais para o apoio à visualização tradicionalmente contornam o problema da alta dimensionalidade reduzindo a dimensão dos dados analisados para uma, duas ou três dimensões, tornando possível uma representação gráfica desses dados, de forma a explorar a capacidade visual humana no reconhecimento de estruturas interessantes, padrões ou anomalias inerentes aos dados.

Em termos gerais, uma técnica para projeção multidimensional visa mapear os dados originais, compostos por m atributos espalhados em um espaço m-dimensional, em um espaço p-dimensional, com $p \ll m$, preservando informações sobre as relações de distância entre as instâncias (objetos) de dados, de forma a revelar o máximo possível das estruturas existentes.

O resultado da projeção é um conjunto de pontos X' em $\mathbb{R}^p$, cada um representando uma instância dos dados, cujas coordenadas podem ser usadas para gerar uma representação gráfica. A similaridade é expressa por meio da vizinhança espacial entre os elementos na projeção, permitindo utilizar a habilidade de interpreção visual para analisar os dados. Essas representações fornecem uma visão geral dos dados e favorecem a identificação de elementos com padrões similares ou dissimilares, provendo um ponto de partida para uma exploração mais detalhada. Formalmente uma técnica de projeção multidimensional pode ser definida como:e

Definição 3.1 (Projeção Multidimensional (Tejada et al., 2003)) *Seja X um conjunto de objetos em $\mathbb{R}^m$ com $\delta : \mathbb{R}^m x \mathbb{R}^m \rightarrow \mathbb{R}$ um critério de proximidade entre objetos em $\mathbb{R}^m$, e Y um conjunto de pontos em $\mathbb{R}^p$ para $p = 1, 2, 3$ e $d : \mathbb{R}^p x \mathbb{R}^m \rightarrow \mathbb{R}$ um critério de proximidade em $\mathbb{R}^p$. Uma técnica de projeção multidimensional pode ser descrita como uma função $f : X \rightarrow Y$ que visa tornar $|\delta(x_i, x_j) - d(f(x_i), f(x_j))|$ o mais próximo possível de zero, $\forall x_i, x_j \in X$.*

Dentre as possíveis funções f, existem dois tipos diferentes de funções que podem ser utilizadas, levando a dois grandes grupos de projeções multidimensionais: (1) técnicas lineares de projeção e (2) técnicas não-lineares de projeção. Podemos definir uma técnica linear de projeção como:

Definição 3.2 (Projeção Multidimensional Linear (Kirby, 2001)) *Uma projeção multidimensional $f : X \rightarrow Y$ é dita ser linear se $f(\alpha x_i + \beta x_j) = \alpha f(x_i) + \beta f(x_j)$ para todo $x_i, x_j \in X$ e $\alpha, \beta \in \mathbb{R}$.*

Uma técnica é dita não-linear se a mesma não obedecer a essa definição.

Como as técnicas lineares criam projeções por meio da combinação linear entre os diferentes eixos que definem o espaço m-dimensional, o resultado do *layout* será satisfatório, isto é, conseguirá expressar as estruturas dos dados originais, somente quando existir uma dependência linear entre tais eixos. Quando esses dados apresentarem estruturas não-lineares, como agrupamentos de formato arbitrário ou *manifolds* curvos, a melhor escolha seria a utilização de uma técnica não-linear.

3.2.1 Técnicas Lineares

Técnicas lineares de projeção criam projeções por meio da combinação linear dos atributos dos dados, definindo-os em uma nova base ortogonal de menor dimensão. Dentre as técnicas lineares, as mais conhecidas são as técnicas de segunda ordem, tais como *Principal Component Analysis (PCA)* (Jolliffe, 1986), *Singular Value Dexomposition (SVD)* (Demmel, 1997) e *Factor Analysis (FA)* (Anderson, 2003). Técnicas de segunda ordem são as que empregam somente a informação contida na matriz de covariância, sendo particularmente apropriadas para dados que apresentem uma distribuição Gaussiana (normal), uma vez que em tais casos, toda a distribuição dos dados pode ser capturada (Mardia et al., 2000). Por outro lado, técnicas de maior ordem, tais como *Projection Pursuit (PP)* (Sun, 1993; Cook et al., 1993; Fyfe e Baddeley, 1995; Friedman e Tukey, 1974; Friedman, 1987; Huber, 1985; Jones e Sibson, 1987), *Redundancy Reduction* (Barlow et al., 1989; Schmidhuber et al., 1996; Deco e Obradovic, 1995) e *Blind Deconvolution* (Haykin, 1994; Shalvi e Weinstein, 1990, 1993; Donoho, 1981) utilizam informações que não estão contidas na matriz de covariância, sendo mais apropriadas para dados não-gaussianos. Descreveremos a seguir duas técnicas lineares de projeção: (1) Principal Component Analysis e (2) Projection Pursuit.

Principal Component Analysis (PCA) *Principal Component Analysis (PCA)* (Jolliffe, 1986) também conhecido com *Expansão de Karhunen-Loève* (Fukanaga, 1990; Duda e Hart, 1973) ou *Empirical Orthogonal Functions* (Lorenz, 1956), é uma técnica que procura transformar linearmente um conjunto original de dados que possuam um alto grau de dependências entre eles (correlações) em um conjunto de menor dependência e dimensionalidade.

O processo utilizado pelo PCA é baseado em determinar combinações lineares ortogonais, os chamados *Componentes Principais*, que melhor capturem a variabilidade dos dados. Nesse processo, o primeiro componente principal será a combinação linear com maior variância, o segundo componente principal será a combinação linear, ortogonal à primeira, com maior variância, e assim por diante. Existem tantos componentes principais quanto o número original de atributos,

mas normalmente os primeiros componentes principais capturam a maior parte da variância dos dados de forma que muitos podem ser descartados com pequena perda de informação.

As principais características do PCA são: permitir identificar as tendências dos atributos mais relevantes, capturar a maior variabilidade dos dados em poucas dimensões e permitir reduzir o ruído, redundância e ambiguidade (Ding, 2000; Paulovich, 2006). PCA porém, conta com a desvantagem na determinação do número correto de dimensões: fornecendo um número pequeno, tende-se a perder características importantes dos dados, caso contrário capturam-se características importantes, mas a representação visual se torna difícil.

Projection Pursuit (PP) A *Projection Pursuit (PP)* (Sun, 1993; Cook et al., 1993; Fyfe e Baddeley, 1995; Friedman e Tukey, 1974; Friedman, 1987; Huber, 1985) é uma técnica desenvolvida no campo da estatística que, diferente da técnica PCA representada anteriormente, pode incorporar mais informação do que informação de segunda-ordem, portanto sendo útil dados não-gaussianos (Fodor, 2002). A PP visa encontrar projeções de dados multidimensionais que podem ser usadas para a visualização da estrutura de agrupamentos dos dados, e para propósitos como estimativa de densidade e regressão.

O ponto central da *Projection Pursuit* é a definição e otimização de um índice de projeção que defina as direções de projeção mais interessantes. Normalmente, esse índice é alguma medida não-normal, sendo a escolha mais natural a *entropia diferencial* (Jones e Sibson, 1987), também conhecida como *entropia negativa de Shannon* (Huber, 1985). A entropia diferencial H de um vetor aleatório y cuja densidade é dada por $f(\cdot)$ é definida como:

$$H(y) = -\int f(y) \log f(y) dy \tag{3.1}$$

Considere variáveis y com média zero e diferentes densidades f, sendo a covariância de y fixa. A entropia diferencial $H(y)$ será maximizada com respeito a f quando f for uma densidade Gaussiana. Para qualquer outra distribuição, a entropia é estritamente menor. Assim, busca-se encontrar as direções da PP por meio da minimização de $H(\mathbf{w}^T\mathbf{x})$ com respeito a $\mathbf{w}$, restringindo a variância de $\mathbf{w}^T\mathbf{x}$ ser constante.

Embora a PP seja uma grande contribuição para a análise de dados de alta dimensionalidade, ainda apresenta diversas limitações (Crawford e Fall, 1990). Um dos problemas mais comuns se refere à dificuldade em determinar o que realmente as soluções encontradas significam para um dado *índice de projeção*. Além disso, a PP não tem a habilidade de fazer inferências, de forma que possa retornar falsas estruturas. Finalmente, é difícil, se não impossível, especificar algoritmicamente o que constitui uma estrutura nos dados.

3.2.2 Técnicas Não-Lineares

Diferente das técnicas lineares apresentadas na seção anterior, que se baseiam em combinações lineares para definir o *layout* final dos dados, as técnicas não-lineares visam minimizar

uma função de perda de informação. Como geralmente essa função baseia-se na dissimilaridade entre os objetos m-dimensionais e nas distâncias entre os pontos p-dimensionais, a aplicação de técnicas não-lineares não necessita que os dados originais tenham uma representação vetorial, só sendo necessário que se defina as dissimilaridades entre objetos e as distâncias entre os pontos no *layout* final.

Para definirmos a similaridade ou a dissimilaridade, podemos usar métricas de distância tais como Euclidiana, Manhattan e Cosseno do ângulo entre os vetores. Esta última determina o cosseno entre dois vetores $\vec{u}$ e $\vec{v}$ aplicando o produto escalar dos vetores entre seus módulos:

$$cos(\vec{u}, \vec{v}) = \frac{\sum_{i=1}^{n}(u_i \times v_i)}{\sqrt{\sum_{i=1}^{n} u_i^2} \times \sqrt{\sum_{i=1}^{n} v_i^2}} \tag{3.2}$$

Outra técnica altamente utilizada no grupo de pesquisa é a *Complexidade de Kolmogorov* (Telles et al., 2007), a qual define uma medida de similaridade a partir de operações simples sobre o tamanho compactado dos dados, fugindo assim, da representação vetorial. A métrica é baseada em uma medida de similaridade denominada *Normalized Compression Distance (NCD)*, que inicialmente foi concebida como uma medida de similaridade entre sequências genéticas. Podemos formular NCD por (Telles et al., 2007):

$$NCD(x, y) = \frac{C(xy) - min\{C(x), C(y)\}}{max\{C(x), C(y)\}} \tag{3.3}$$

onde $C(\cdot)$ é o tamanho de x comprimida por um algoritmo de compressão e xy é a concatenação de x e y.

Dentre as técnicas não-lineares de projeção, uma das mais conhecidas é a *Multidimensional Scaling (MDS)* (Cox e Cox, 2000). Na verdade, MDS é um conjunto de técnicas definidas em dois grandes grupos: (1) técnicas baseadas em gradientes descendentes e (2) técnicas de posicionamento baseado em força (ou *Force-Directed Placement (FDP)*). Além da MDS, há também as técnicas *Nearest-Neighbor Projection (NNP)* (Tejada et al., 2003) e *Fastmap* (Faloutsos e Lin, 1995).

A seguir, descreveremos as técnicas não-lineares desenvolvidas pelo grupo de pesquisa do Instituto de Ciências Matemáticas e de Computação, denominadas *Projection by Clustering (ProjClus)* (Paulovich e Minghim, 2006) e *Least Square Projection (LSP)* (Levkowitz et al., 2007).

Projection by Clustering (ProjClus) Visando reduzir a complexidade do modelo *Force Scheme* (Tejada et al., 2003), a técnica *Projection by Clustering (ProjClus)* (Paulovich e Minghim, 2006) separa os N objetos em $\sqrt{N}$ agrupamentos com uso da técnica *k-means* por bissecção (Tan et al., 2006) para calcular o centróide (ou média aritmética das instâncias de cada grupo). Em seguida, os centróides de cada um dos agrupamentos são projetados com uso da técnica *Fastmap* (Faloutsos e Lin, 1995) e de uma da técnica de posicionamento baseado em forças *Force Scheme* (Tejada et al., 2003).

Em seguida, os pontos de cada agrupamento são projetados em um espaço normalizado contendo apenas os elementos de cada um deles. Por fim, os agrupamentos já projetados são posicionados no espaço comum, de acordo com a posição dos centróides.

Por possuir uma menor complexidade, da ordem de $O(\sqrt{N^3})$, está técnica é indicada para o processamento de conjuntos de dados maiores que as anteriores, pois consegue preservar a proximidade dos pontos em uma vizinhança ao espaço n-dimensional original. Porém, estruturas importantes podem se perder no resultado final, devido principalmente à criação de grupos na fase inicial.

Least Square Projection (LSP) Partindo do estudo feito por Sorkine e Cohen-Or (Sorkine e Cohen-Or, 2004) o qual aplica mínimos quadrados na recuperação e edição de malhas *(least-square meshes)*, *Least Square Projection* (Levkowitz et al., 2007) seleciona um subconjunto S de elementos em uma coleção de documentos, chamados *pontos de controle*, e os projeta com alguma técnica convencional de projeção, tal como *Fastmap* (Faloutsos e Lin, 1995). Em seguida, se constrói um sistema linear esparso baseado nas relações de vizinhança dos pontos em seu espaço original $\mathbb{R}^n$ e nas coordenadas cartesianas dos pontos de controle no espaço reduzido $\mathbb{R}^m$.

A complexidade computacional da *LSP* é determinada pelo número de agrupamentos. Para $\sqrt{n}$ agrupamentos, a complexidade computacional será de $O(n\sqrt{n})$. A Figura 3.1 ilustra uma projeção utilizando o conjunto de dados *news-2* contendo 200 notícias retiradas da internet[1].

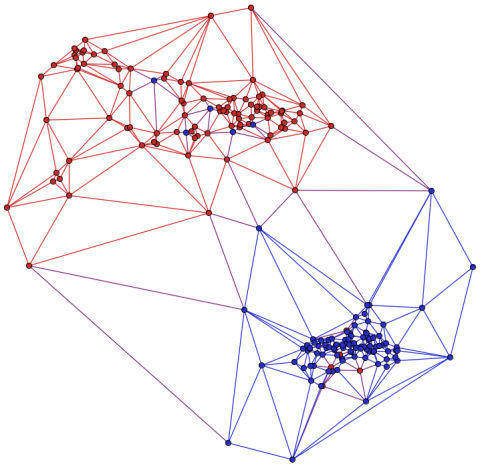


Figura 3.1: Projeção do conjunto de dados *news-2* utilizando a técnica LSP.

3.3 Técnicas de Criação de Árvores de Similaridade

Uma forma de representar e explorar dados consiste em mapear graficamente os elementos em formas visuais a fim de refletir suas relações de similaridade, relevâncias e a possível organização de áreas e sub-áreas relacionadas a um determinado evento. O uso de árvores como forma de

[1]Disponível em http://infoserver.lcad.icmc.usp.br

representação dos mapas visuais a partir de uma matriz de similaridade utiliza conceitos de diversas áreas para a melhoria na qualidade da representação e, por conta disso, é utilizada com sucesso como modelo para posicionamento de pontos na ferramenta *PEx (Projection Explorer)* (Paulovich et al., 2007).

A seguir, descrevemos duas técnicas que estão presentes na ferramenta PEx, e que são utilizadas neste trabalho para a exploração das técnicas de visualização e navegação de dados mapeados no plano.

3.3.1 Neighbor Joining (NJ) Trees

O método *neighbor joining*, inicialmente proposto para árvores filogenéticas[2] (Saitou e Nei, 1987) e posteriormente adaptado para construção de mapas de documentos (Valdivia, 2007), constrói uma árvore não-enraizada a partir de uma matriz de distâncias evolutivas, adaptando o critério de evolução mínima[3]. No caso específico de dados multidimensionais, como textos e imagens, essas distâncias são representadas a partir de uma *matriz de similaridade*. Uma matriz de similaridade M é uma matriz de números reais e dimensões $n \times n$ onde M_{ij} é a distância entre os itens de dado i e j. Para o cálculo dessa matriz, métricas de distâncias são empregadas (ver Seção 3.2.2). Um estudo mais detalhado de árvores (enraizada e não-enraizada) é tratado na Seção 3.4.

A ideia central da técnica *neighbor joining* é identificar pares de objetos mais próximos. Esses pares de objetos, ou vizinhos, são conectados por um nó interno em uma árvore bifurcada. A relação de vizinhança é ilustrada na Figura 3.2, em que os nós A e B são vizinhos, mas os nós A e C não são. Se A e B forem combinados em um único nó, então nessa nova combinação A e C tornam-se vizinhos. Na Figura 3.3 é ilustrado o uso da técnica *neighbor joining* para o conjunto *news-2*.

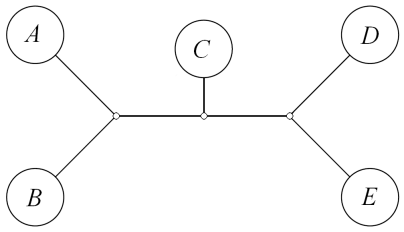


Figura 3.2: Árvore não-enraizada para a técnica *Neighbor Joining* (Adaptada de (Valdivia, 2007)).

[2]Árvore Filogenética ou Cladograma é uma exibição em forma de árvore das relações evolutivas entre várias espécies ou outras entidades que podem ter um antepassado em comum.

[3]O critério da evolução mínima tenta minimizar a soma dos tamanhos de todos os nós da árvore, isto é, busca encontrar sequencialmente vizinhos que minimizem o comprimento total da árvore (Schneider, 2003).

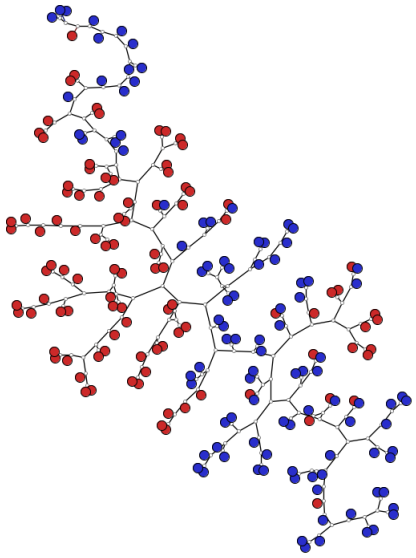


Figura 3.3: Técnica *Neighbor Joining* aplicada para o conjunto *news-2*.

3.3.2 Minimum Spanning Tree (MST)

O problema da Árvore Geradora Mínima ou *Minimum Spanning Tree (MST)* é um dos mais típicos problemas estudados em teoria dos grafos. Podemos defini-lo da seguinte forma: dado um grafo conectado não direcionado $G = (V, E)$, com n nós e m arestas (com pesos), encontre um conjunto de arestas $E' \subseteq E$ de peso mínimo e que conecta todos os nós. O peso do conjunto de arestas é a soma de todos os pesos das arestas (pesos positivos) contidas no conjunto.

Um procedimento padrão para construção de uma árvore geradora mínima é conhecido como algoritmo de Prim (Prim, 1957). A ideia central do algoritmo é, partindo-se de um nó arbitrário, adicionar sucessivamente à árvore geradora mínima o nó remanescente cuja aresta que conecte à árvore tenha menor peso, até que todos os nós do grafo estejam contidos na árvore.

No caso específico de coleção de documentos, podemos utilizar a abordagem MST na matriz de similaridade (a qual representa as distâncias entre todos os nós) para criar a árvore geradora mínima visando preservar as distâncias locais dos pontos no espaço multidimensional (Yang, 2003). Nas Figuras 3.4 e 3.5 podemos ver, respectivamente, a árvore geradora mínima para um grafo com pesos nas arestas e, na Figura 3.5 o uso da Técnica *Minimum Spanning Tree* aplicada para o conjunto *news-2*, citada anteriormente.

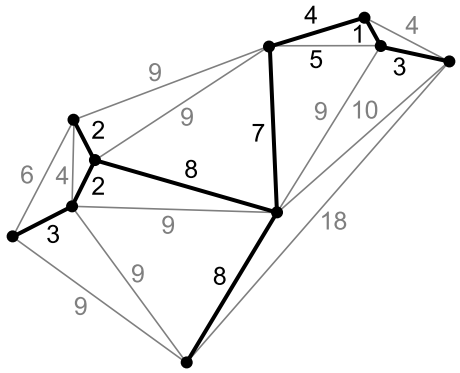


Figura 3.4: Árvore não-enraizada para a técnica *Minimum Spanning Tree* (Retirada do endereço http://en.wikipedia.org), acessada em Janeiro, 2010.

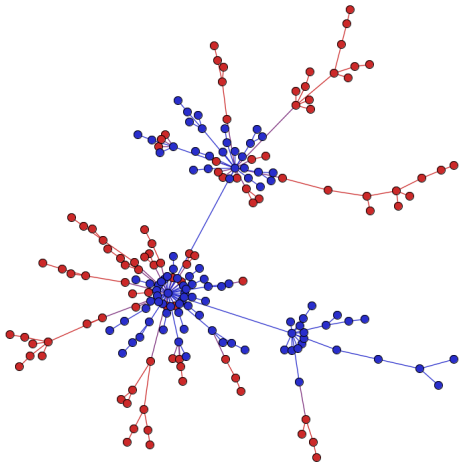


Figura 3.5: Técnica *Minimum Spanning Tree* aplicada para o conjunto *news-2*.

As técnicas *Projection by Clustering*, *Least Square Projection*, *Neighbor Joining* e *Minimum Spanning Tree* foram implementadas na ferramenta *PEx (Projection Explorer)* (Paulovich et al., 2007). Em uma projeção, a exploração se dá por seleção e foco em áreas de interesse e, em árvores a seleção se dá por foco em ramos de interesse. Entretanto, as técnicas baseadas em árvore requerem estratégias adicionais específicas de apresentação e interação, de forma que se tornem úteis em visualização exploratória.

Um breve estudo sobre árvores e técnicas de desenho é apresentado na Seção seguinte.

3.4 Visualização de Árvores

Em função da simplicidade da estrutura e da popularidade no meio científico, as árvores estão entre as primeiras classes estudadas em teoria e desenho de grafos. As técnicas de desenho de árvores consideram dois casos distintos: desenho de árvores com raiz (*rooted trees*) e desenho de árvores livres (*free trees*). *Rooted Trees* podem também englobar outros tipos particulares de

árvores, as árvores *binárias*, nas quais os nós possuem 2 ou menos nós filhos e as árvores gerais, as quais não possuem esta limitação no número de nós filhos.

Nas seções seguintes, descreveremos as árvores com raiz e as árvores livres, uma vez que as utilizaremos em diferentes técnicas, utilizadas pela ferramenta *PEx-Image*. Em seguida abordaremos brevemente as técnicas de Visualização Radial e Visualização Hiperbólica

Um estudo mais profundo de grafos e técnicas de desenho, assim como suas propriedades básicas, podem ser encontrados em diversos livros tais como Baker e Ebert, Bollobás, Bond e Murty e West (Baker e Ebert, 1997; Bollobás, 1990; Bondy e Murty, 1976; West, 1996).

3.4.1 Árvores com Raiz

Árvores com raiz são comumente utilizadas para representar hierarquias, tais como diagramas organizacionais, árvores de busca e registros de chamadas de sub-rotinas. A principal ideia das árvores com raiz é utilizar um nó base, e conectar todos os outros nós a ele ou um a nó já existente. Sendo assim, cada nó possui exatamente um nó-pai (com exceção do nó base, ou nó raiz) e podendo ter vários nós-filhos. Ao final, os nós que não tiverem nós-filhos associados são chamados de nós-folhas. A Figura 3.6 ilustra como uma árvore com raiz pode ser representada, evidenciando a hierarquia associada.

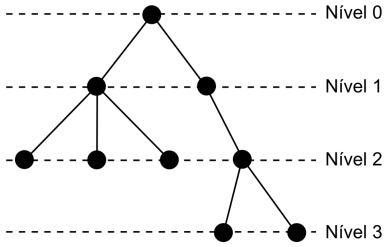


Figura 3.6: Estrutura hierárquica de uma árvore com raiz.

Existem diversas técnicas utilizadas para desenhar esse tipo de árvore, que destacam principalmente as seguintes características:

- Distribuir os nós sobre níveis hierárquicos horizontais, de acordo com a sua profundidade na árvore.
- Para árvores binárias, posicionar os filhos esquerdo e direito de cada nó, respectivamente, à esquerda e à direita do nó pai.
- Centralizar os pais sobre os seus filhos.
- Minimizar a largura do desenho.

Uma das técnicas comumente empregadas no desenho de árvores com raiz será tratada na Seção 3.4.3 e detalhada no próximo Capítulo.

3.4.2 Árvores Livres

Árvores livres são árvores que não possuem raiz e não possuem uma hierarquia intrínseca. Uma vez que o desenho não depende do sistema de coordenadas, por não possuirem hierarquia global associada, o *layout* básico foge, muitas vezes, do modelo adotado por árvores com raiz. A Figura 3.7 ilustra uma árvore livre.

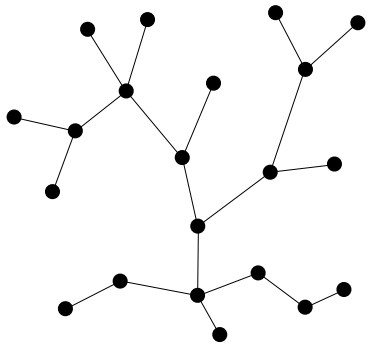


Figura 3.7: Exemplo de uma árvore livre.

Para o desenho de árvores livres, algums critérios estéticos são altamente utilizados nas diferentes abordagens existentes:

- Remover os cruzamentos entre as arestas.
- A distribuição dos nós deve evitar agrupamentos.
- As arestas possuem aproximadamente o mesmo comprimento.

Embora existam diversas técnicas para desenho de árvores livres, uma maneira simples de se desenhar esse tipo de estrutura é utilizar técnicas para desenho de árvores com raiz, bastando escolher um dos nós como raiz. Eades (Eades, 1992) apresenta estudos sobre desenho de árvores livres, destacando não apenas algoritmos para obter desenhos radiais, mas também a utilização do método *Springs* (Eades, 1984; Kamada e Kawai, 1989).

O método *Springs*, também chamado *Force Directed Layout*, proposto por Eades (Eades, 1984) modela um grafo (no nosso caso a árvore) como um sistema dinâmico de molas no plano, no qual cada nó é representado por uma partícula que exerce uma força de repulsão sobre os demais nós. As arestas, por sua vez, são substituídas por uma mola conectando dois nós (partículas). O objetivo do método é encontrar um estado de baixa energia do sistema, que está geralmente associado a um desenho de boa qualidade. Os desenhos gerados pelo método *Springs* costumam ter arestas de comprimento uniforme e apresentam muitas simetrias. O método, contudo, não trata o problema de minimizar o número de cruzamentos entre arestas e exige ajustes para que não caia em um mínimo local (Lin, 1992).

Nas Seções seguintes, citaremos brevemente duas técnicas empregadas para o desenho de árvores. Estas técnicas são utilizadas neste projeto para os mapeamentos no plano, gerados

pela ferramenta *PEx-Image* (Eler et al., 2008, 2009), que utilizam a estrutura de árvore para representação. No Capítulo seguinte, as duas técnicas são detalhadas, evidenciando exemplos da própria ferramenta estendida neste projeto.

3.4.3 Visualização Radial

O Modelo Radial, inicialmente proposto por Eades (Eades, 1992) e por Battista et al. (Battista et al., 1999) como uma variação da abordagem *H-tree* (Shiloach, 1976) pode ser empregado para visualizar árvores genéricas, isto é, independentemente do tipo. Este modelo utiliza a ideia de círculos concêntricos, deslocando os nós sobre os círculos e simultaneamente mantendo a hierarquia da árvore, com o raio de cada círculo representando um nível da árvore e a raiz localizada ao centro de todos os círculos. A distância de um nó para o nó central determina o círculo concêntrico ao qual este pertence. Um exemplo de árvore que utiliza o modelo de visualização radial pode ser visto na Figura 3.8.

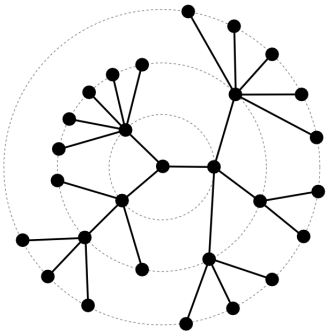


Figura 3.8: *Layout* Radial para representação de uma Árvore (Adaptada de (Wills, 1997)).

3.4.4 Visualização Hiperbólica

A abordagem de árvores Hiperbólicas surgiu primeiramente com Lamping et al. (Lamping et al., 1995; Lamping e Rao, 1996) e posteriormente com Munzner (Munzner e Burchard, 1995; Munzner, 1997, 1998). Esta técnica, que pode ser implementada utilizando espaço bi ou tridimensional, proporciona uma visão distorcida de uma árvore, permitindo a interação com árvores potencialmente grandes, tornando-a adequada para problemas e aplicações reais. Nesta técnica, nós mais distante do centro são apresentados em pequena escala de tamanho e as linhas de relacionamento (arestas) são projetadas como curvas no espaço hiperbólico. A Figura 3.9 mostra esse tipo de desenho.

3.5 Considerações Finais

Conforme visto neste capítulo, a principal ideia das técnicas de mapeamento multidimensional é posicionar um conjunto de dados em um espaço de baixa dimensionalidade, preservando

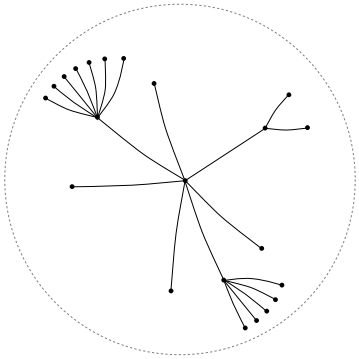


Figura 3.9: *Layout* Hyperbólico para representação de uma Árvore.

ao máximo as relações de similaridade existentes entre eles. Dessa forma, dados multidimensionais representados em espaços de baixa dimensionalidade podem fazer uso das técnicas de visualização e classificação de forma mais eficiente, revelando características interessantes.

A aplicação de técnicas de mapeamento de pontos no plano para representação de coleção de documentos e para coleção de imagens em particular, demonstrou, em trabalhos anteriores (Paulovich et al., 2007), ser capaz de separar e agrupar conjuntos que apresentem alta correlação de conteúdo e também de aproximar em uma vizinhança textos de alta similaridade. Nesse contexto, as técnicas e ferramentas desenvolvidas no grupo de pesquisa do ICMC-USP permitiram não apenas a projeção de qualquer conjunto de dados multidimensionais como também a diminuição da complexidade computacional das técnicas que anteriormente vinham sendo empregadas, sendo que muitas vezes é mais interessante possuir técnicas rápidas e altamente interativas do que processar longamente os dados sem garantia dos resultados.

Este trabalho adaptou e testou algumas técnicas de visualização e navegação em mapeamentos de dados no plano que utilizam a estrutura de árvores para visualização. Em particular as técnicas NJ e MST foram utilizadas para exploração das técnicas Radial e Hiperbólica. No próximo capítulo apresentamos a implementação dessas técnicas no contexto de exploração de árvores e apresentamos estudos de casos para dados como texto e imagens.

Capítulo

4

Implementações e Resultados

4.1 Considerações Iniciais

Técnicas de navegação e interação tornam-se indispensáveis para a interpretação e manipulação dos dados, principalmente daqueles representados em espaços de alta dimensão.

Neste capítulo descrevemos uma ferramenta desenvolvida com o propósito de utilizar técnicas de posicionamento de pontos no plano para dados com imagens e texto, bem como as extensões à ferramenta implementadas neste trabalho. O uso de coordenação entre múltiplas visões e visualização de informação anteriormente empregada com sucesso na ferramenta *PEx-Image* (Eler et al., 2008, 2009) foi adaptado para os novos tipos de desenho de árvores aqui discutidos. Ao final, estudos de caso são apresentados.

4.2 PEx-Image e Coordenação entre Múltiplas Visões

Projection Explorer for Images (PEx-Image)[1] (Eler et al., 2008, 2009) é uma adaptação da ferramenta *Projection Explorer* (Paulovich et al., 2007), a qual foi desenvolvida para auxiliar na análise e exploração de dados multidimensionais em geral e mapas de documentos em particular. A principal funcionalidade da PEx-Image é auxiliar o usuário na análise de um conjunto de imagens, possuindo uma interface para a análise e exploração das imagens mapeadas como pontos no plano.

A entrada dos dados para a PEx-Image pode ser um conjunto de imagens ou documentos, uma matriz de distância contendo a distância ou dissimilaridade entre os objetos sob análise ou

[1]Disponível em: http://infoserver.lcad.icmc.usp.br, acessado em Fevereiro, 2010.

um conjunto de imagens. Para o conjunto de imagens em particular, são extraídas características por meio de algoritmos implementados utilizando a biblioteca *Scilab*[2] em conjunto com *Scilab Image Processing Toolbox*[3]. Essas características podem ser interpretadas como um conjunto de vetores, representando o conjunto de imagens (no qual cada vetor representa uma única imagem) em um espaço m-dimensional, onde m é o número de características extraídas das imagens.

Podemos utilizar também um conjunto de pontos no espaço m-dimensional como entrada, isto é, no caso de imagens, as características podem ser extraídas externamente e os vetores de características alimentados diretamente como entrada do sistema. Com isso, algoritmos podem fazer comparações entre diferentes imagens com base em suas características, tornando o processo mais eficiente, uma vez que o número de características é muito menor do que o tamanho da própria imagem.

Inicialmente, PEx-Image extrai 28 características, representadas por um vetor de 29 posições. As 20 primeiras posições do vetor (posições 1..20) são representações dos 20 *descritores de Fourier* (Costa e Junior, 2000; Azencott et al., 1997) para o histograma da imagem. O histograma é uma representação da distribuição de frequência dos *pixels* de uma imagem. A intensidade de cor dos *pixels* de uma imagem varia de 0 a 255. Para não utilizar todas as 256 posições como característica, é aplicada a transformada de Fourier 1D (Huang e Aviyente, 2006) e desta, apenas os 20 descritores que possuem a maior quantidade de informação, que são os descritores que estão na extremidade inicial e na extremidade final do vetor resultante da transformada, são utilizados.

Em seguida são extraídos os 6 descritores de Fourier (posições 21..26) para uma imagem bidimensional. Uma das formas utilizadas para extrair esses descritores é a criação de círculos, os quais são construídos no centro da matriz; esta que por sua vez é o resultado da aplicação da transformada de Fourier 2D (Huang e Aviyente, 2006). Quando um círculo é aplicado na matriz, é extraído um descritor baseado na energia dos elementos que estão dentro do círculo. Para o cálculo desta energia, PEx-Image utiliza a seguinte equação:

$$E_c = \frac{1}{N} \sum_{k=1}^{N} |C_k| \qquad c = 1, ..., NC \tag{4.1}$$

Onde E_c é a energia do círculo c, N é o número de elementos que estão dentro do círculo c, C_k é o elemento k que está dentro do círculo c e NC é o número de círculos utilizados para extrair os descritores de Fourier.

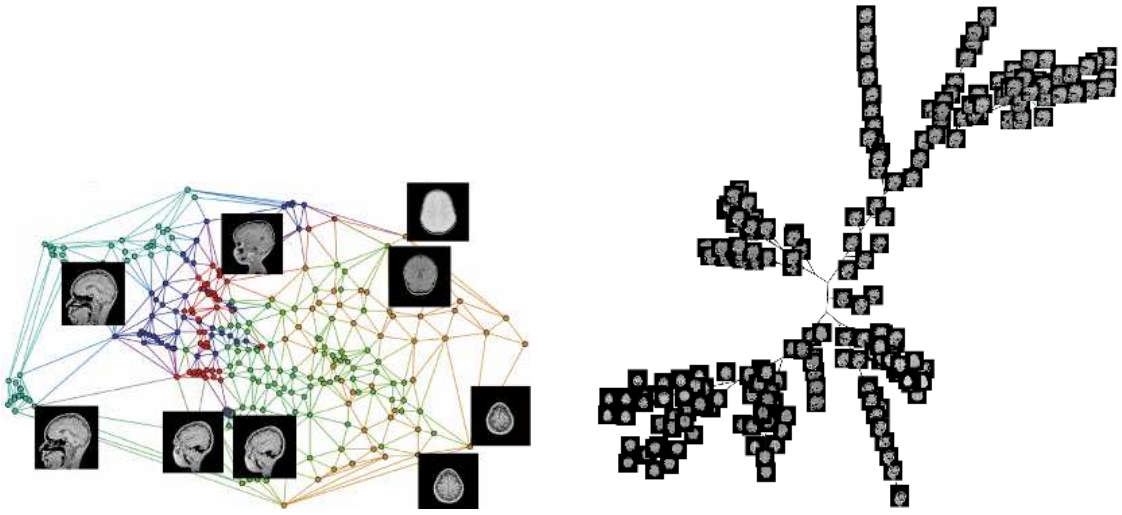
Por fim, nas posições 27, 28 e 29 do vetor temos, respectivamente, a média, desvio padrão e uma informação adicional a critério do usuário (tal como classe da imagem ou um valor escalar associado). Lembramos que a média e o desvio padrão, neste caso, são calculados a partir dos níveis de cinza dos *pixels* de uma imagem, ou seja, das intensidades das imagens, aplicados à imagem inteira.

[2]Disponível em http://www.scilab.org, acessado em Janeiro, 2010.
[3]Disponível em http://siptoolbox.sourceforge.net, acessado em Janeiro, 2010.

Após o processo de alimentação do PEx-Image com os dados, é possível escolher o tipo de técnica que será utilizada para mapear os pontos do espaço m-dimensional em um espaço p-dimensional (sendo, neste caso, $p = 2$ e $m = 29$). Além de selecionar o tipo de projeção e configurar os seus parâmetros específicos, pode-se também escolher o tipo de métrica de distância a ser utilizada na projeção. Na PEx-Image, podemos citar entre as principais métricas disponíveis a distância Euclidiana, cosseno do ângulo entre vetores (ver Seção 3.2.2) e *City Block* (Widmann e Schröger, 1999).

Após as configurações iniciais e o processamento da projeção, os pontos projetados são representados na tela por meio de elementos visuais. Os elementos utilizados, assim como na PEx-Image, são círculos e linhas, os quais representam as imagens e as ligações entre imagens, respectivamente. Quando um conjunto de imagens possui um valor escalar associado, como um rótulo pré-determinado ou algum peso associado, as cores ou tamanhos dos círculos podem mapear tais valores, possibilitando a representação de dois valores escalares ao mesmo tempo, facilitando o processo de análise visual. Linhas de ligação entre os círculos podem indicar a ligação construída por uma *triangulação de Delaunay* (Aurenhammer, 1991), o vizinho mais próximo no espaço $\mathbb{R}^2$ ou no espaço $\mathbb{R}^m$ e também, no caso de uma árvore, a relação hierárquica construída pelo algoritmo.

A pré-visualização através de uma miniatura da imagem pode ser vista quando se posiciona o *cursor* do *mouse* sobre qualquer ponto da projeção (Figura 4.1(a)). Além de círculos, as próprias imagens podem ser apresentadas (Figura 4.1(b)). Essas visualizações auxiliam na verificação de agrupamentos e na validação de classificações. No entanto, dependendo do número de imagens dentro de um agrupamento (ou mesmo na própria projeção em si), pode ocorrer uma sobrecarga visual, ocasionada por sobreposições das imagens.



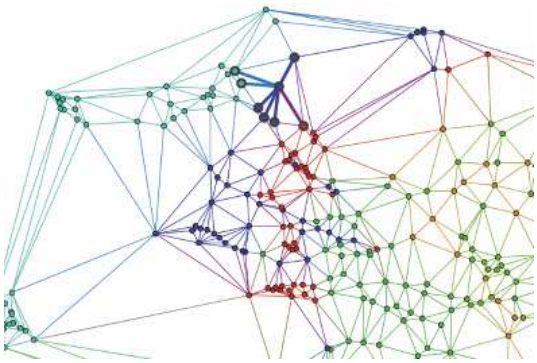
(a) Detalhe dos vértices

(b) Projeção utilizando as próprias imagens como formas visuais

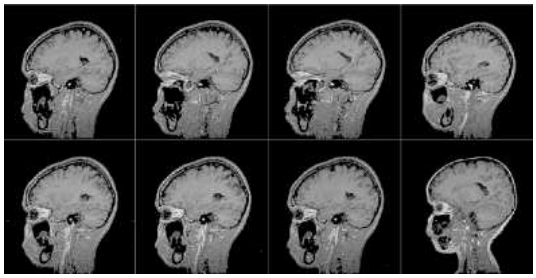
Figura 4.1: Projeção para um conjunto de imagens de Ressonância Magnética.

Para lidar com problemas de sobreposição, estão presentes ferramentas de navegação tais como *zooming* e deslocamento automático ou a possibilidade de poder recortar um conjunto de imagens e analisá-lo isoladamente em uma nova janela. Caso sobreposições ainda estejam aparentes, significa que diversas imagens possuem características muito semelhantes. Resolve-se este problema movendo as imagens para outra posição da tela, através do clique e arraste do *mouse* sobre a imagem ou círculo.

A Figura 4.2 ilustra uma interação que recupera os vizinhos mais próximos de um dado selecionado segundo as arestas no *layout* atual. Ao pressionar uma vez o *mouse* sobre uma imagem, os seus vizinhos mais próximos são destacados (Figura 4.2(a)). Se um duplo clique do *mouse* for executado sobre esta imagem, ela e seus vizinhos mais próximos podem ser visualizados lado a lado (Figura 4.2(b)), facilitando a comparação das imagens que foram consideradas semelhantes no posicionamento.



(a) Seleção de um vértice com destaque dos seus vizinhos



(b) Conteúdo de um vértice e de seus vizinhos

Figura 4.2: Recuperação de imagens semelhantes.

Outra característica importante para análise envolvendo múltiplas visões[4] é a coordenação, que é uma ferramenta essencial na exploração de dados multidimensionais, auxiliando o usuário a determinar a qualidade de agrupamentos formados pela utilização de diferentes características, métricas de distância e técnicas de projeção.

Na PEx-Image foi implementada a coordenação por identidade, coordenando dados que possuem mesmo nome de arquivo e, que pode destacar em diferentes projeções um conjunto de imagens selecionadas, isto é, caso a mesma imagem ocorra em diferentes projeções, pode-se visualizá-las nessas projeções.

Ferramentas de visualização de informações coordenadas são úteis para análise e compreensão de grandes volumes de dados, sendo estes sistemas altamente interativos para auxiliar as pessoas a executarem tarefas de análise. Diversas ferramentas e técnicas de visualização de dados multidimensionais têm sido desenvolvidas nos últimos anos, mas poucos trabalhos abordam questões relacionadas à avaliação de sistemas ou aplicações que implementam estas técnicas

[4]Sistemas de múltiplas visões usam duas ou mais visões distintas para auxiliar o processo de investigação de uma única entidade conceitual (Kuchinsky e Baldonado, 2000).

(Pillat et al., 2005). Uma vez que a ferramenta PEx-Image visualiza tipos de dados diferentes (texto e imagem) em um único ambiente, a necessidade de múltiplas visões é evidente. Uma vez que os dados (ou um subconjunto considerável deles) são referentes aos mesmos objetos, a coordenação, juntamente com técnicas eficazes de navegação e visualização dos dados também se faz necessária.

4.3 Implementação das Interações sobre Árvores

Visando aproveitar as características das técnicas de posicionamento de pontos no plano e seu sucesso na aplicação para coleção de documentos e de imagens, este projeto propôs o desenvolvimento de um processo que auxilie o usuário a interagir com os dados, possibilitando a navegação e a análise mais rápida e eficiente, mantendo a possibilidade de coordenação de conjuntos de dados de tipos diferentes (imagem e texto) empregada por PEx-Image.

Com o auxílio de técnicas de desenho e navegação de árvores, foi possível adaptar a ferramenta PEx-Image e as técnicas de projeção multidimensional *Neighbor-Joining* e *Minimum Spanning Tree* para novos tipos de interação, usando os *layouts* de árvore interativos Radial e Hiperbólico.

A seguir, descreveremos as técnicas implementadas na PEx-Image para auxílio na navegação e interação dos dados para as projeções multidimensionais baseadas em árvores.

4.3.1 Técnica Radial

A técnica Radial, proposta por Eades (Eades, 1992) e por Battista et al. (Battista et al., 1999) utiliza a ideia de árvores com raiz, focando-se em um nó específico, podendo facilmente ser empregada para representar hierarquias, tais como árvores de busca, diagramas organizacionais e árvores filogenéticas.

Uma vez que a ideia principal da técnica é utilizar um nó central, é natural que este nó seja posicionado ao centro da visualização e todos os outros nós em círculos concêntricos (com centro no nó central). Cada círculo concêntrico possuirá um raio definido como sendo $r_n = r_{n-1} + c$, onde c é uma constante positiva. Uma vez definidos os círculos com raios r_0, r_1, r_2, ..., r_n, posicionamos os demais nós nos círculos correspondentes relativos à distância do nó central, mantendo-se a hierarquia original.

Após o posicionamento (em relação ao nó central) de todos os nós em seus respectivos círculos concêntricos, calcula-se o setor correspondente de desenho, levando em consideração o tamanho da sub-árvore que tem raiz em cada nó. A definição deste setor de desenho garante que toda a árvore poderá ser desenhada sem que haja sobreposicionamento dos nós ou cruzamento das arestas de ligação (planaridade).

Wills (Wills, 1997) define uma maneira de calcular o ângulo do setor de desenho utilizando como base a seguinte função:

$$A_n = \frac{A_m * W_n}{W_m} \qquad \forall(m,n),\ \texttt{tal que}\ m\ \texttt{é nó pai de}\ n. \tag{4.2}$$

Onde A_x é o ângulo do nó x e W_x é o número de folhas presentes no nó x. Por exemplo, para calcularmos o valor do ângulo do nó V da Figura 4.3, teremos que calcular o valor do ângulo do nó S (nó pai). Assim, $S = 360 * 10/20 = 180$ graus e $V = 180 * 3/10 = 54$ graus, lembrando que $A_{ncentral} = 360$ graus.

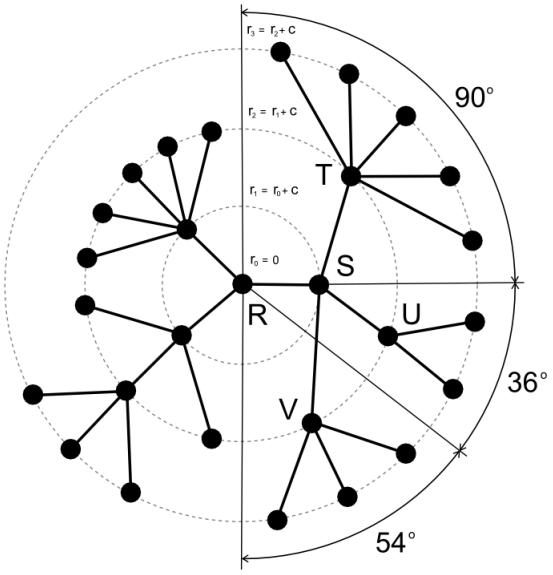


Figura 4.3: Cálculo do Setor de Desenho para uma Árvore com Raiz (Adaptada de (Wills, 1997)).

Finalmente, basta calcularmos a posição (x, y) de um vértice. Como um exemplo, utilizamos as seguintes funções para cálculo das posições dos nós do primeiro nível (diretamente ligados ao nó raiz):

$$P_x(n) = (r_n) * sin(I_n * A_n) \tag{4.3}$$

$$P_y(n) = (r_n) * cos(I_n * A_n) \tag{4.4}$$

Onde $P_x(n)$ e $P_y(n)$ são, respectivamente, as posições x e y do nó n, r_n é o raio do círculo concêntrico do nó n, I_n é o índice do nó n dentro do círculo e A_n o ângulo do nó n. Para o cálculo dos demais nós, podemos utilizar as funções descritas por (Book e Keshary, 2001).

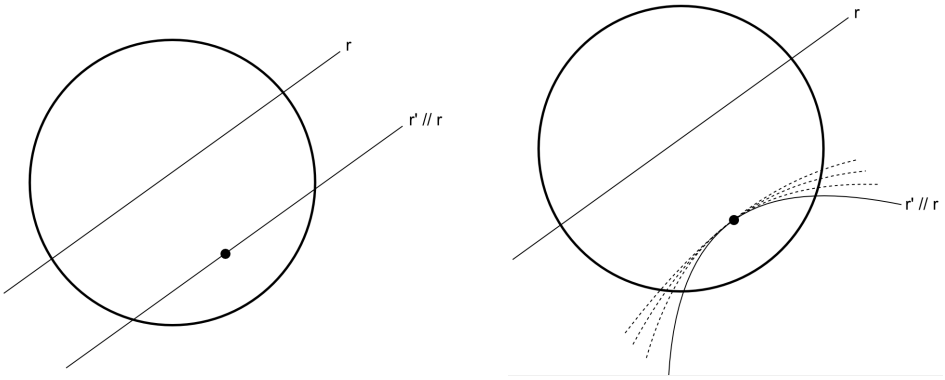
4.3.2 Técnica Hiperbólica

Devido aos efeitos de navegação e uma certa semelhança visual com a técnica *fish-eye* (Furnas, 1981), a Técnica Hiperbólica pode ser facilmente considerada umas das técnicas mais

interessantes para visualização de árvores (Spritzer e Freitas, 2008) e seu uso torna-se ideal para representar grandes estruturas hierárquicas, uma vez que a base principal é a geometria hiperbólica.

A ideia central desta técnica é, baseando-se no plano hiperbólico, construir o *layout* com a técnica radial (4.3.1) e, posteriormente mapear este *layout* para o plano euclidiano, por meio do Modelo Euclidiano do Disco Unitário de Poincaré (Lamping et al., 1995).

Uma geometria hiperbólica é baseada em axiomas idênticos aos tradicionais axiomas Euclidianos, com exceção do *Quinto Postulado*. Em geometria Euclidiana, o quinto postulado afirma que dada uma reta r e um ponto p, e qualquer outra reta r' que passe pelo ponto p, existe apenas uma única reta r' que seja paralela a r. Porém o mesmo não ocorre na geometria hiperbólica, na qual podem existir infinitas retas r' paralelas a r. A Figura 4.4 ilustra a diferença entre o quinto postulado para as geometrias euclidiana e hiperbólica.



(a) Uma única reta r' é paralela a r.

(b) Existem infinitas retas r' paralelas a r.

Figura 4.4: O Quinto Postulado para as Geometrias (a) Euclidiana e (b) Hiperbólica.

Assim, linhas paralelas no plano hiperbólico divergem umas das outras (Figura 4.5), o que implica que a circunferência de um círculo cresce exponencialmente em relação ao raio (Lamping e Rao, 1996). Desta forma, o espaço cresce exponencialmente com o aumento da distância em relação ao ponto central, tornando viável o uso deste tipo de geometria para a visualização de grandes conjuntos de dados hierárquicos.

4.3.3 Inclusão dos Layouts Radial e Hiperbólico na PEx-Image

Para mostrar que os *layouts* Radial e Hiperbólico fornecem meios mais eficazes de navegação e exploração dos dados, foi necessário estender os *layouts* originalmente utilizados pela ferramenta PEx-Image bem como implementar a coordenação entre eles e os novos *layouts*. A extensão para os *layouts* Radial e Hiperbólico partiu dos *layouts Neighbor Joining* e *Minimum Spanning Tree*.

O processo de criação do *layout* através da técnica *Neighbor Joining*, por exemplo, inicia-se com o cálculo da matriz de distâncias entre os objetos em análise. No caso da ferramenta PEx-Image (Seção 4.2) podemos utilizar um conjunto de documentos ou um conjunto de imagens

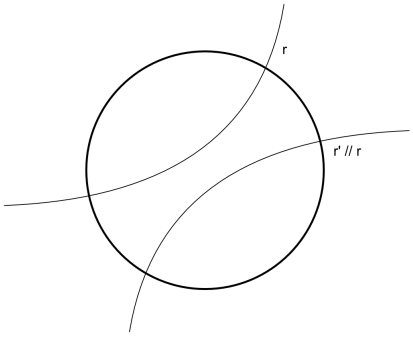


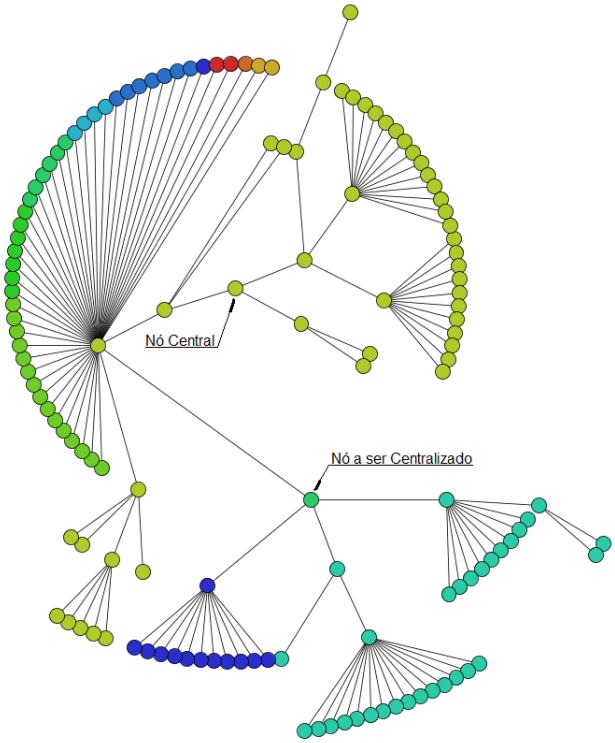
Figura 4.5: Retas Paralelas na Geometria Hiperbólica divergem exponencialmente com o aumento do raio.

e calcular a matriz de distância de duas formas: (1) através de um pré-processamento dos textos ou imagens a partir do espaço vetorial e (2) através por métricas diretas de cálculo de similaridades. A partir da construção da matriz de distâncias, é efetuada a construção de uma árvore sem raiz através da técnica de geração de árvores filogenéticas (Ver Seção 3.3.1) ou de uma *MST* (Ver Seção 3.3.2).

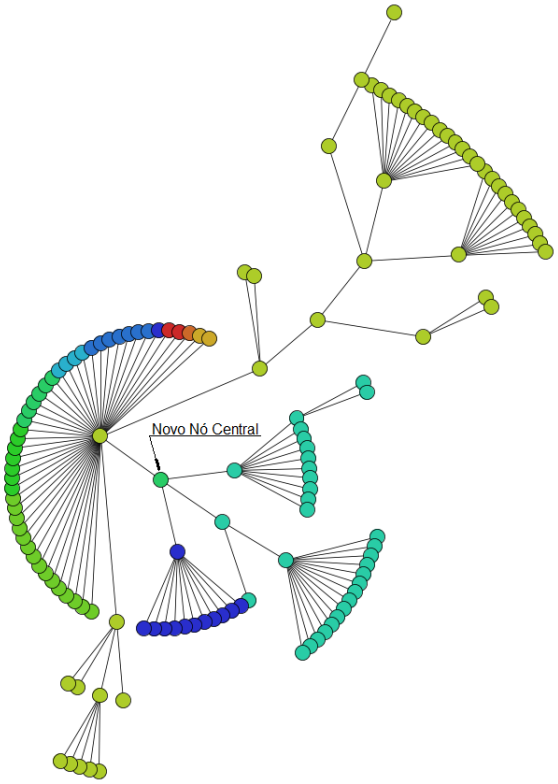
A partir do mapa gerado, representado por uma árvore, pode-se criar os novos *layouts* Radial e Hiperbólico. A criação dos layouts Radial e Hiperbólico utiliza os mesmos conjuntos de vértices e arestas da ferramenta PEx-Image, representado, respectivamente, pelas classes *vertex* e *edge*, para construir as novas posições dos pontos no plano. Uma vez criado o novo mapa de pontos, os recursos de interação e navegação da visualização também foram estendidos.

Uma vez que as funções de interação da ferramenta PEx-Image são implementadas considerando a seleção de vértices (ou elementos de dados) no plano de visualização, as mesmas funções foram utilizadas nos *layouts* Radial e Hiperbólico e outras adaptadas a fim de promover formas mais eficientes de navegação e interação em árvore. Dentre as principais funcionalidades, podemos destacar:

Foco de um Vértice Como os *layouts* Radial e Hiperbólico possuem como base a centralização de um nó específico, a funcionalidade de Foco de um vértice foi utilizada a fim de se poder utilizar qualquer vértice como centro da visualização. Uma vez que o usuário seleciona um vértice como novo centro, todos os outros elementos são reposicionados. Podemos destacar essa funcionalidade na Figura 4.6 para o *layout* Radial e na Figura 4.7 para o *layout* Hiperbólico.

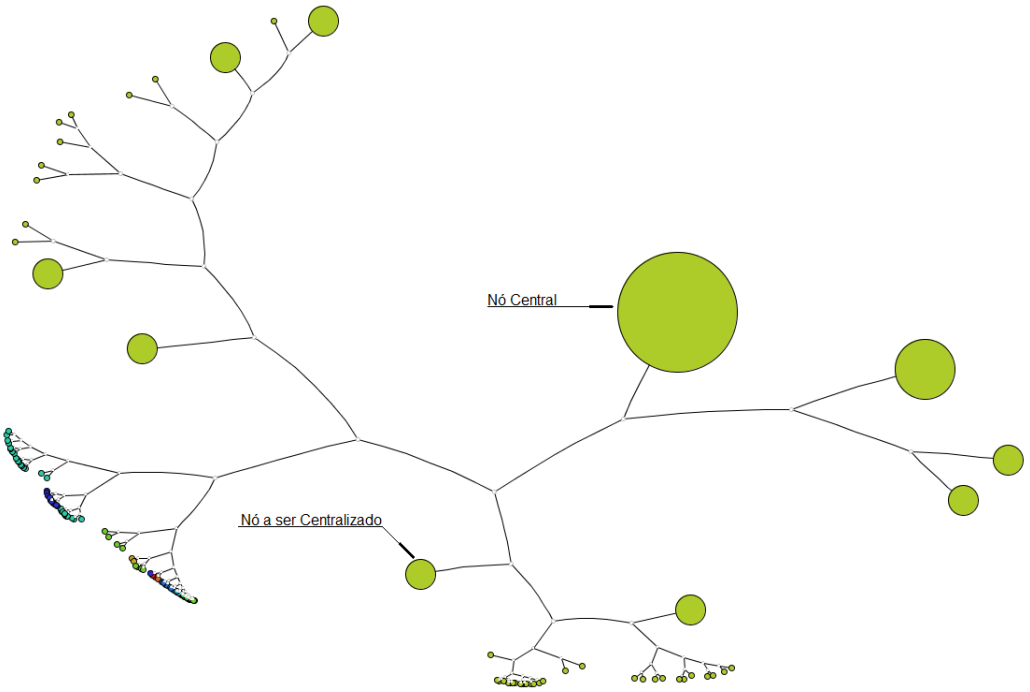


(a) Visualização do *Layout* Radial para um conjunto de dados.

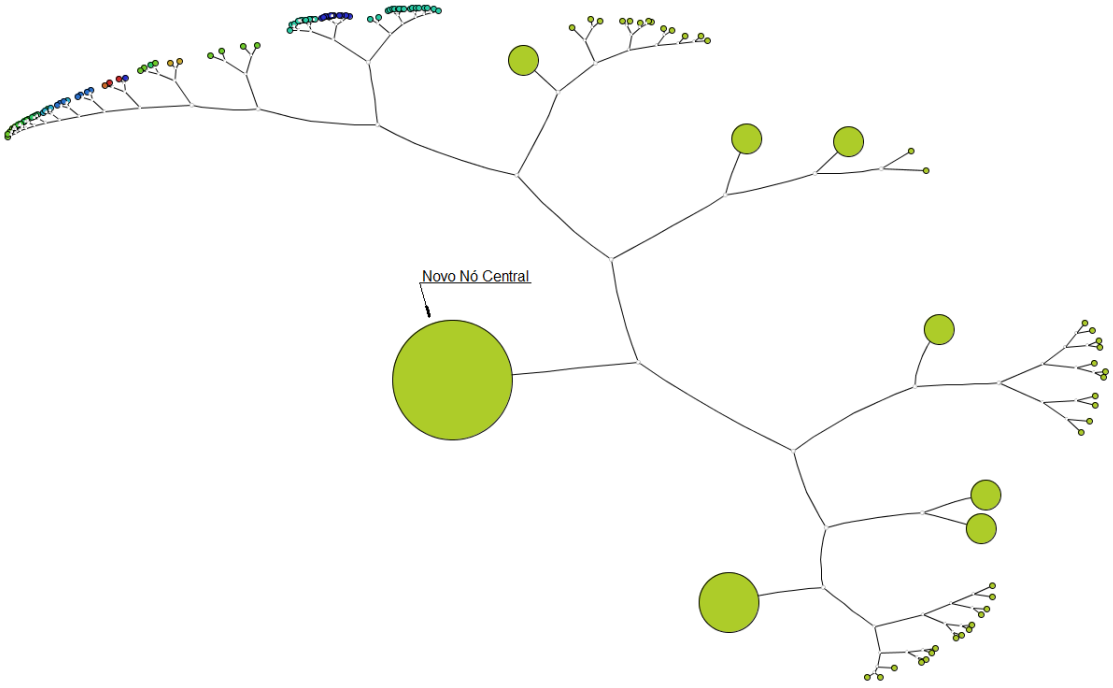


(b) Nova Visualização do *Layout* Radial para o mesmo conjunto de dados, porém centrado em outro Elemento.

Figura 4.6: Seleção de um vértice no *Layout* Radial.



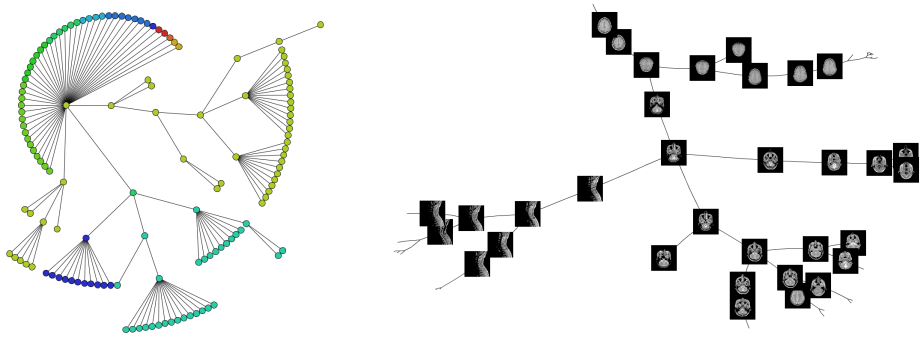
(a) Visualização do *Layout* Hiperbólico para um donjunto de dados.



(b) Nova Visualização do *Layout* Hiperbólico para o mesmo conjunto de dados, porém centrado em outro Elemento.

Figura 4.7: Seleção de um Vértice no *Layout* Hiperbólico.

Representação dos Elementos de Dados As mesmas formas do PEx-Image de representação dos elementos de dados no plano foram utilizadas. Pode-se adotar a representação de círculos para textos e de círculos ou imagens para imagens. A Figura 4.8 representa os dois tipos de visualização que podemos adotar nos *Layouts* Radial e Hiperbólico.



(a) Representação de um conjunto de textos através de círculos no plano no *layout* Radial.

(b) Representação de um conjunto de imagens através das próprias imagens no *layout* Hiperbólico.

Figura 4.8: Representação dos elementos de dados.

Apesar da representação ser da mesma maneira originalmente adotada por PEx-Image, para o *layout* Hiperbólico foram adicionados fatores de escalas; isto é, como a ideia central da técnica Hiperbólica é aumentar a distância exponencialmente a partir do centro, quanto mais distante os elementos estiverem posicionados do centro da visualização, menor será a escala de sua representação. Podemos notar as escalas utilizadas na Figura 4.9.

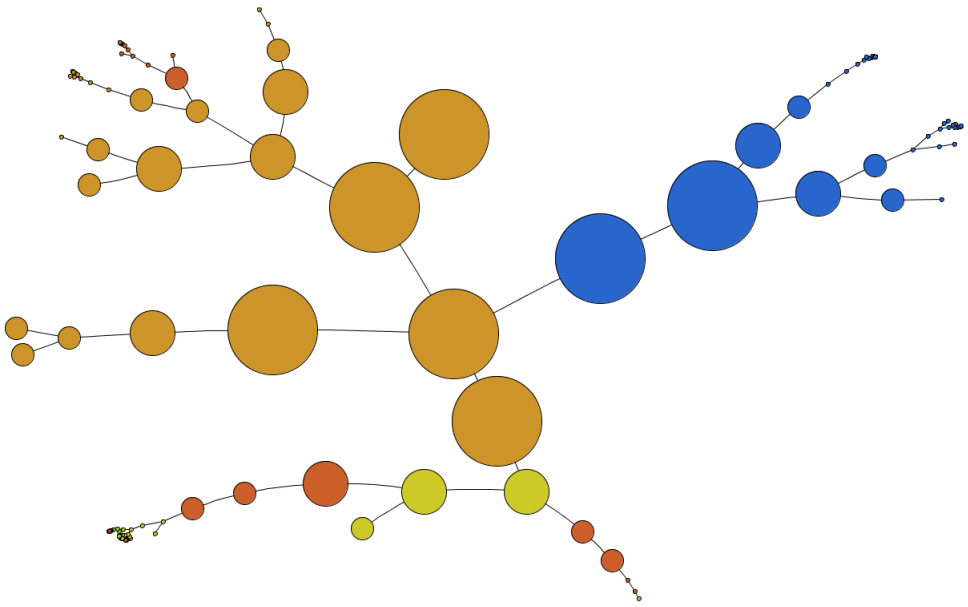


Figura 4.9: Escala utilizada para representar elementos no *Layout* Hiperbólico.

Coordenação dos Elementos Uma vez que a representação dos elementos de dados utiliza a mesma abordagem da ferramenta PEx-Image, a coordenação também utiliza a mesma estrutura de representação, destacando os vértices selecionados ou coordenados em diferentes tipos de *layouts*. A Figura 4.10 ilustra o uso da coordenação para o mesmo conjunto de elementos, porém exibida com diferentes técnicas de posicionamento de pontos no plano.

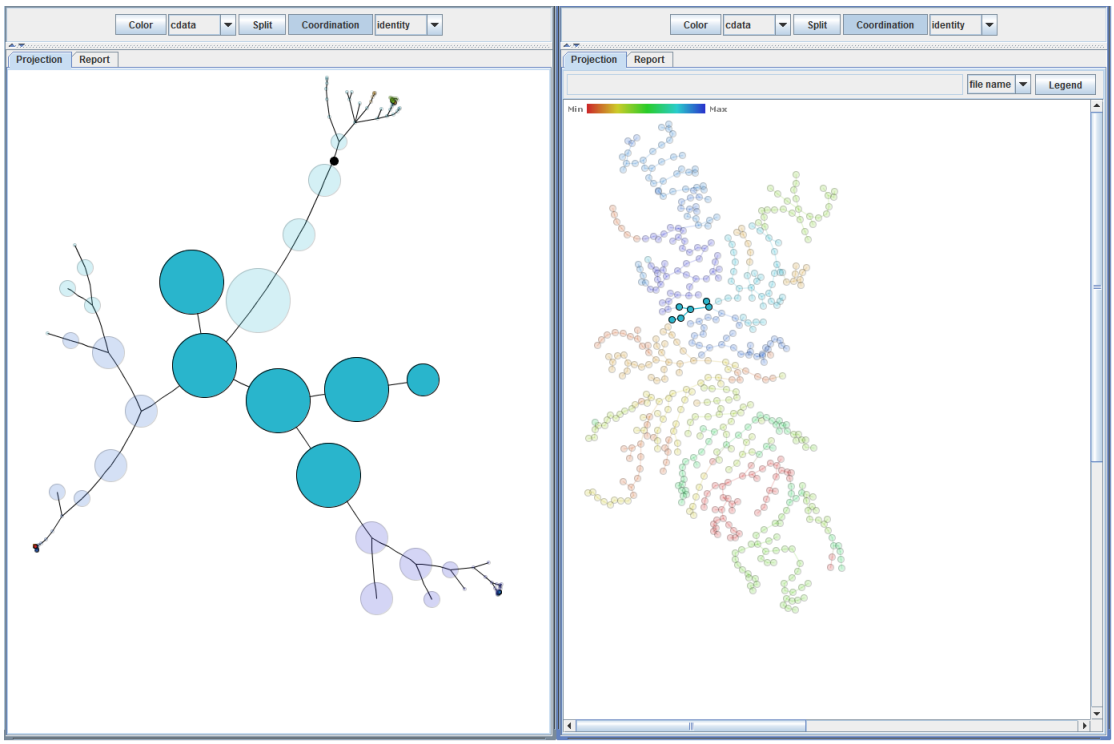
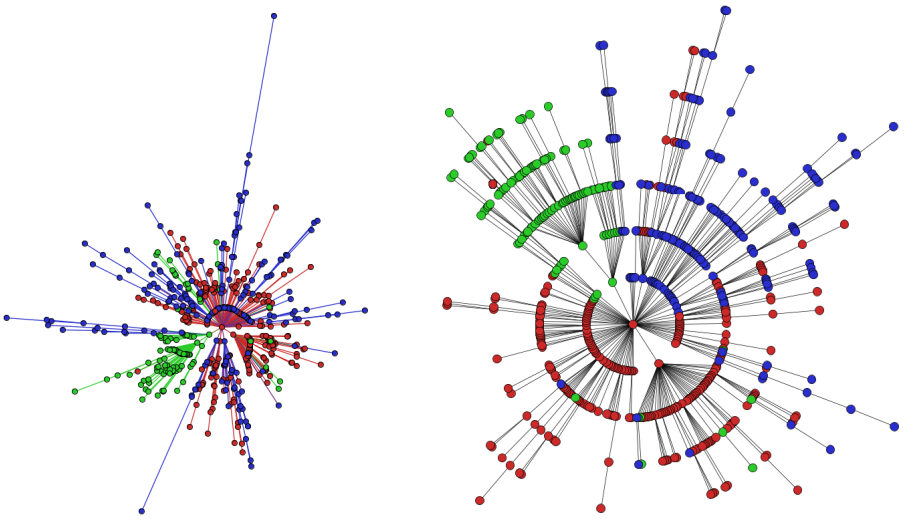


Figura 4.10: Coordenação entre o *layout* Hiperbólico e o *layout*, ambos da Técnica NJ.

Com a utilização das técnicas, vemos que o uso dos *Layouts* Radial e Hiperbólico aumenta o grau de interação e promovem formas mais eficientes de navegação dos dados posicionados no plano. As Figuras 4.11 e Figura 4.12 ilustra, a título de exemplo, os *layouts* das técnicas em seu estado original e utilizando os *layouts* Radial e Hiperbólico para o conjunto de dados cbr-ilp-ir[5].

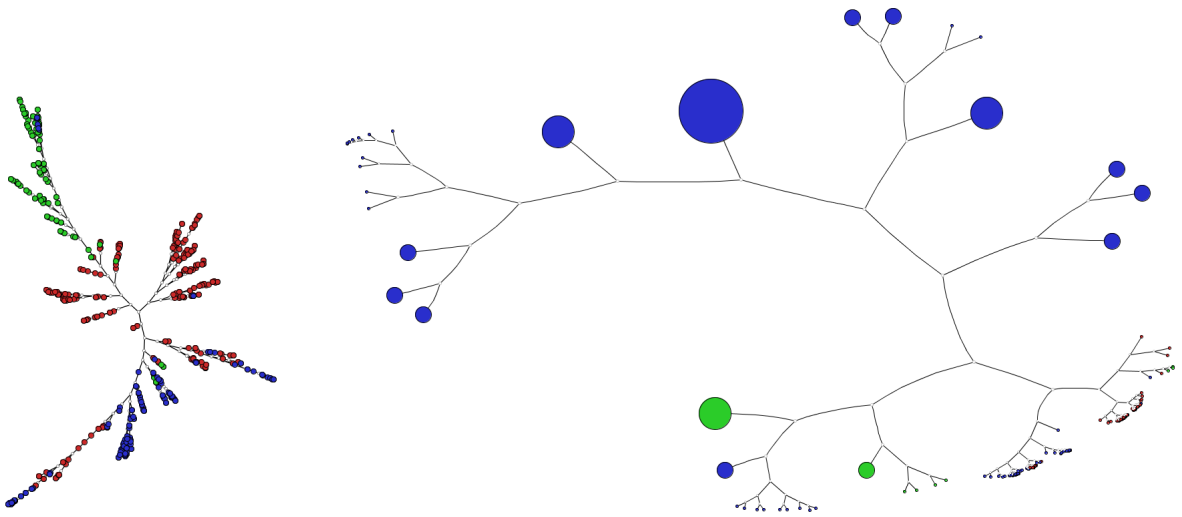
[5]Artigos coletados das áreas de *Case Based Reasoning*, *Inductive Logic Programming* e *Information Retrieval*, disponíveis em http://infoserver.lcad.icmc.usp.br



(a) Representação do conjunto cbr-ilp-ir utilizando a técnica *Minimum Spanning Tree*.

(b) Representação do conjunto cbr-ilp-ir utilizando o *Layout* Radial e a técnica *Minimum Spanning Tree*.

Figura 4.11: Representação dos dados para as técnicas MST e Radial.



(a) Representação do Conjunto cbr-ilp-ir utilizando a técnica *Neighbor Joining*.

(b) Representação do Conjunto cbr-ilp-ir utilizando o *Layout* Hiperbólico e a técnica *Neighbor Joining*.

Figura 4.12: Representação dos dados para as técnicas NJ e Hiperbólica.

4.4 Estudos de Caso

A seguir, descreveremos alguns estudos de caso que apresentam o uso das técnicas radial (4.3.1) e hiperbólica (4.3.2) em comparação com as técnicas anteriormente abordadas por PEx-Image.

4.4.1 Visualização de Sequências Genéticas

Aproveitando conceitos e técnicas de Bioinformática[6], nos quais estruturas de proteínas podem ser representadas como uma matriz bidimensional (Philips, 1970) e utilizando a ferramenta PROTMAP2D (Pietal et al., 2007), com a qual essas informações podem ser utilizadas para gerar mapas bidimensionais *(contact maps)* (Hu et al., 2002), podemos extrair imagens que carregam informações de uma sequência genética e mostrar o uso da coordenação para dados multi-modais (texto e imagem), isto é, sequências genéticas e suas imagens correspondentes.

Foram utilizadas 138 sequências com diferentes classificações, escolhidas aleatoriamente e disponíveis publicamente[7]. Após a obtenção das sequências, estas foram comparadas utilizando-se o BLAST[8] (Altschul et al., 1990) contra um banco de dados contendo 63.762 sequências e então selecionadas aquelas que possuíam *e-value*[9] menor ou igual a e^{-100}, resultando em um total de 138 sequências de 11 classes diferentes.

De posse dos arquivos contendo as informações estruturais das 138 sequências, foi possível gerar os mapas de contatos *(contact maps)* para cada proteína e, aproveitando-se das próprias cadeias de aminoácidos, representar separadamente as sequências (texto) e os *contact maps* (imagens) coordenando suas visualizações, a fim de comparar as informações de cada projeção.

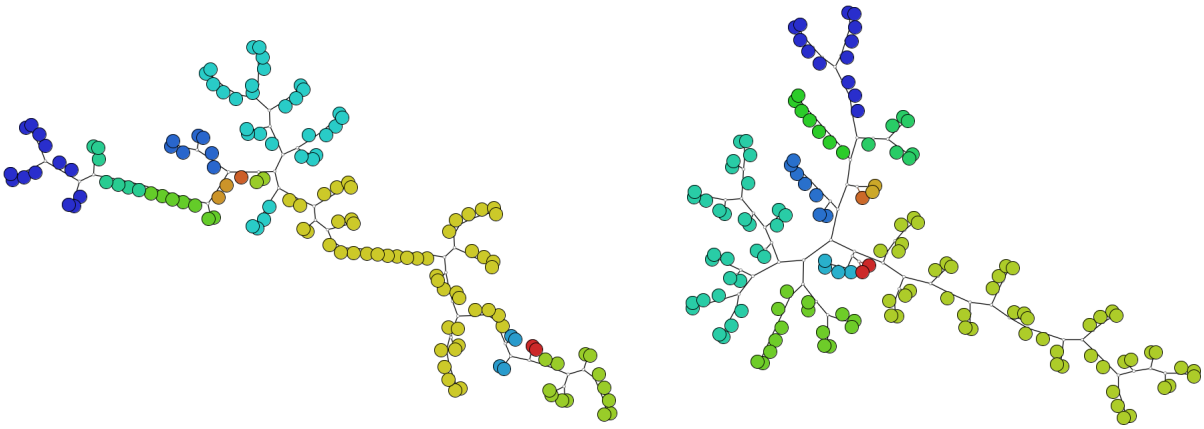
As Figuras 4.13(a) e 4.13(b) ilustram o mapeamento das imagens pela técnica NJ (Ver Seção 3.3.1) e o mapeamento das sequências utilizando também, a técnica NJ e adicionalmente a métrica NCD (Ver Seção 3.2.2), respectivamente, utilizando as técnicas já empregadas na PEx-Image. Podemos notar que ambas as projeções conseguiram separar os dados analisados em suas respectivas classes, previamente rotuladas e destacadas por cores distintas entre as classes.

[6]Bioinformática é a área que estuda algoritmos e técnicas para manipular dados gerados pelas técnicas de Biologia Molecular, a qual por sua vez estuda a estrutura e a atividade de macromoléculas essenciais à vida.

[7]*RCSB Protein Data Bank*, disponível em http://www.rcsb.org.

[8]BLAST é uma ferramenta de Bioinformática altamente utilizada para alinhamento de sequências genéticas.

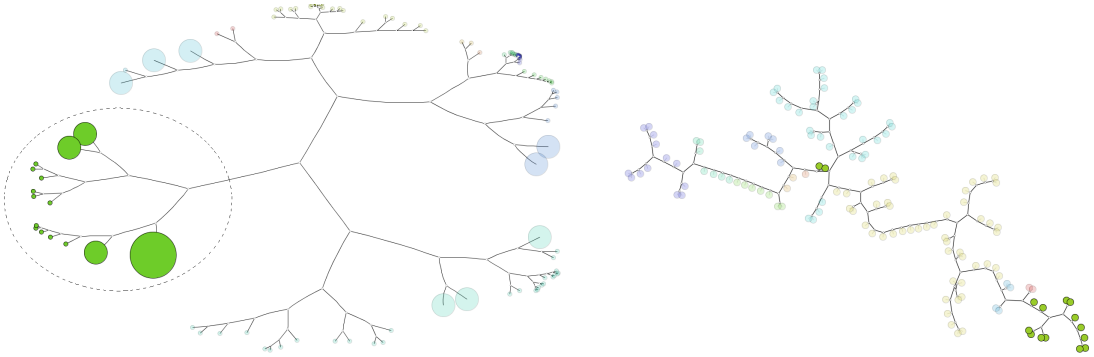
[9]*Expectation value* ou *e-value* é o resultado de cálculos estatísticos que indicam o grau de probabilidade de um alinhamento entre sequências genéticas ter ocorrido ao acaso.



(a) Mapemanto dos mapas de contato (b) Mapeamento das sequências

Figura 4.13: Visualização dos mapeamentos utilizando a técnica NJ.

Apesar da separação visual entre classes, através do *Layout* Hiperbólico aplicado para as sequências genéticas, é possível selecionar um ramo específico (Figura 4.14(a)) com facilidade e visualizar através da coordenação os pontos correspondentes na projeção das imagens (Figura 4.14(b)). Podemos notar na Figura 4.14(b) que dois objetos foram separados de sua classe de origem na projeção das imagens. Embora neste caso todos os objetos das duas projeções sejam pareados, isto é, dado um objeto x na projeção das sequências e um objeto x' na projeção das imagens, temos sempre uma associação entre x e x', dada por uma função ou caracterização a um mesmo fato, variações como estas podem ocorrer, indicando mais detalhes para o especialista analisar.



(a) Seleção de um ramo específico no *Layout* Hiperbólico para a Projeção das Sequências.

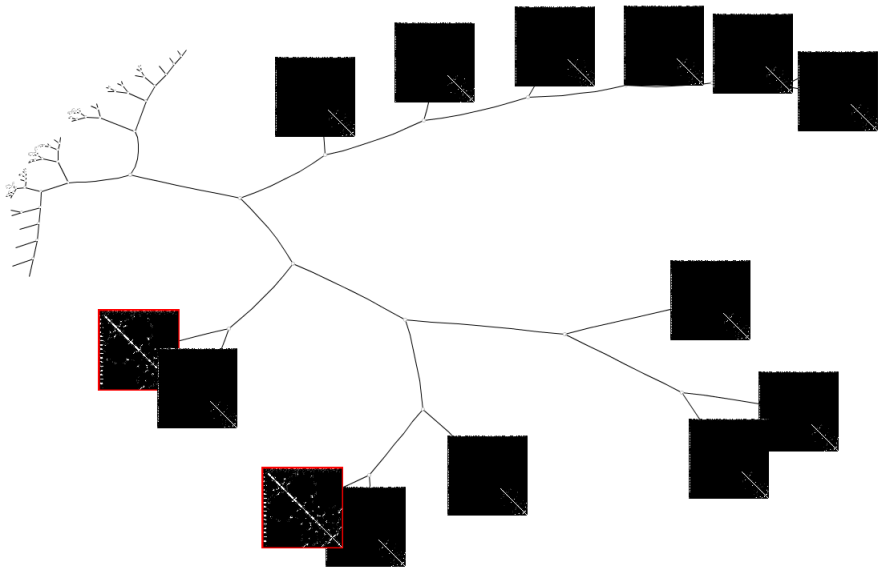
(b) Coordenação da Seleção na Árvore NJ das Imagens.

Figura 4.14: Coordenação das projeções de sequências (com *Layout* Hiperbólico) e da projeção de imagens.

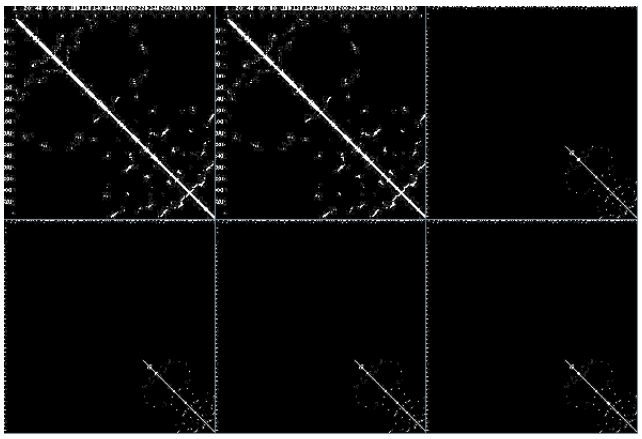
Para uma rápida análise desta situação, utilizamos a técnica Hiperbólica e a opção *"Force Reload Data"* [10], carregando os dados de imagens na projeção de sequências, sendo possível

[10]Esta opção implementada na PEx-Image permite que os outros tipos de dados, tais como imagens e textos sejam carregados em qualquer projeção, desde que estes sejam associados pelo nome.

analisar com exatidão as diferenças entre as imagens (Figura 4.15(a)) e que, por mais que os dois objetos destacados pertençam a uma determinada classe, suas imagens possuem padrões totalmente diferentes do padrão da classe, conforme ilustrado na Figura 4.15(b). A Técnica Hiperbólica nesse caso foi de grande ajuda para a navegação e fácil interpretação dos dados analisados.



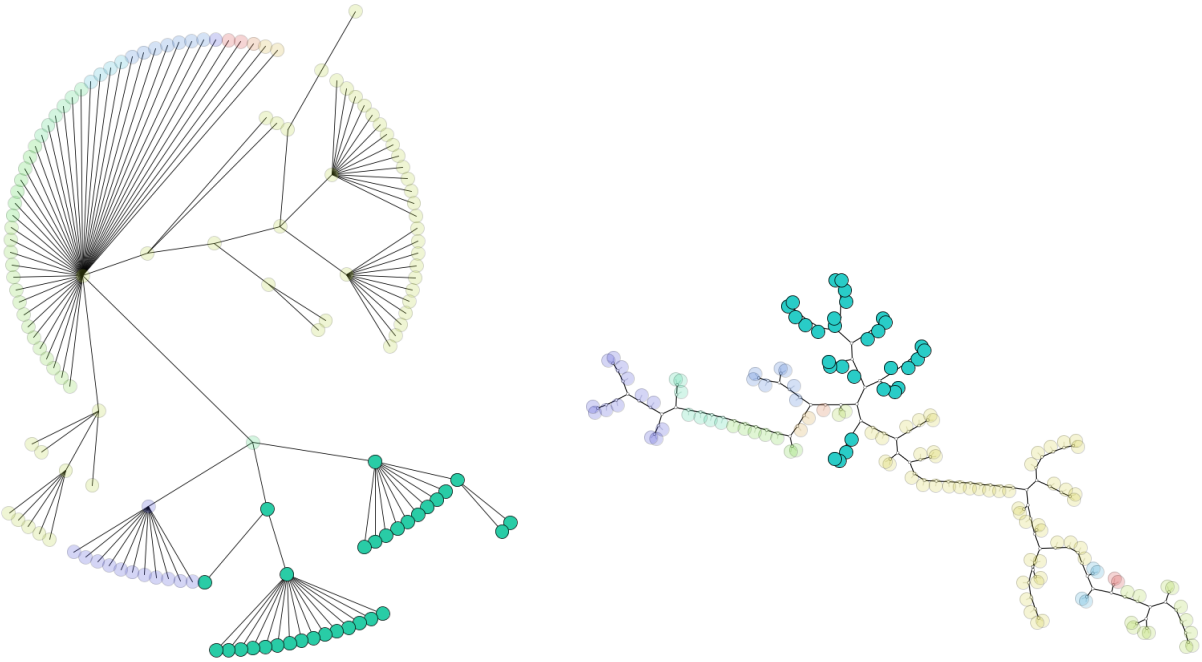
(a) Detalhe para 2 objetos na Árvore NJ das Imagens



(b) Detalhe para os mesmos 2 objetos (Linha 1, Colunas 1 e 2) em relação a outros objetos da mesma classe.

Figura 4.15: Análise por coordenação da discrepância entre as visualizações de imagem e sequência.

De forma análoga, podemos coordenar utilizando o *Layout* Radial. A Figura 4.16 ilustra a mesma coordenação para o conjunto de sequências e imagens, utilizando a Técnica Radial.



(a) Seleção de um ramo específico no *Layout* Radial para a Projeção das Sequências da técnica NJ.

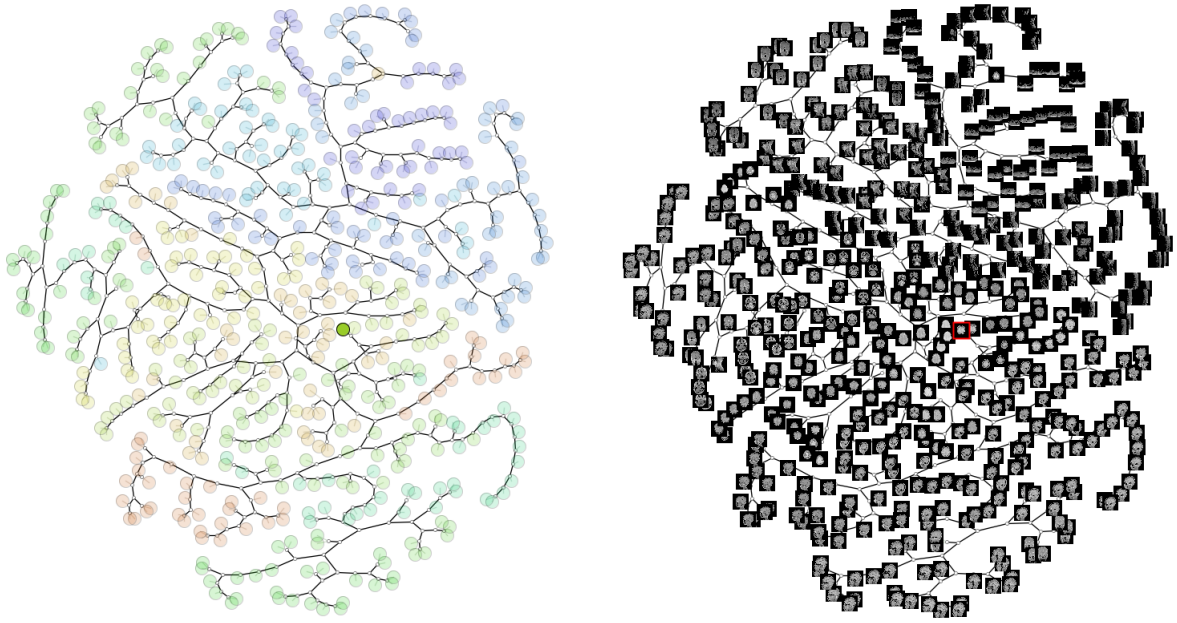
(b) Coordenação da seleção na projeção das Imagens utilizando a técnica NJ.

Figura 4.16: Coordenação das projeções de sequências (com *Layout* Radial) e imagens.

4.4.2 Visualização de Conjunto de Imagens Médicas

Um outro estudo de caso feito, aplicando as técnicas a um conjunto de 512 imagens de Ressonância Magnética, distribuídas em 12 classes e indicadas por diferentes cores. Como esse conjunto de dados possui somente as imagens como entrada para o mapeamento, é importante que sejam utilizados meios que abordem a fácil manipulação e navegação dos dados projetados, visando identificar quais técnicas são mais eficientes para extração das características.

A Figura 4.17 ilustra o *layout* por *Neighbor-Joining* das imagens utilizando, círculos para representação dos elementos e as próprias imagens.



(a) Árvore NJ utilizando círculos para representar os elementos analisados e cores representando as classes.

(b) Árvore NJ utilizando as próprias imagens para representar os elementos analisados.

Figura 4.17: *Layout* por *Neighbor-Joining* utilizando diferentes formas visuais de representação dos elementos.

No *Layout* gerado por NJ, a navegação entre os dados torna-se uma tarefa difícil para o usuário. Porém, através da Técnica Hiperbólica, podemos navegar entre os elementos, identificando rapidamente quais são os elementos próximos e as similaridades entre eles. Na Figura 4.18, podemos verificar que o mesmo nó, selecionado na Figura 4.17, é visualizado sem grandes problemas ou sobreposição com outros elementos. Também é possível ver mais claramente quais são os elementos próximos.

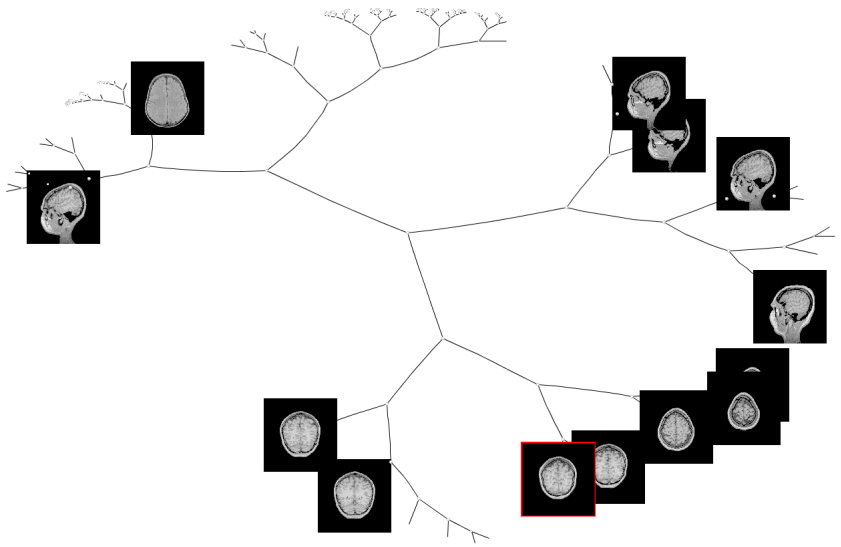
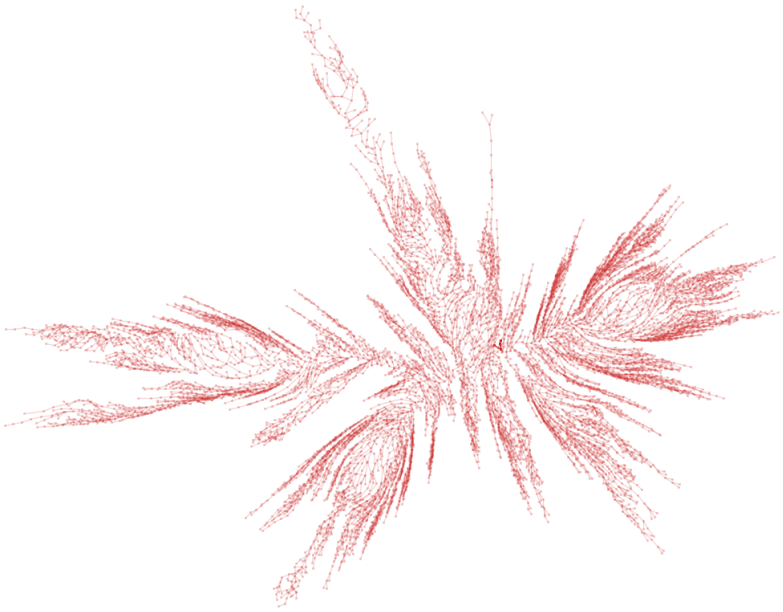


Figura 4.18: *Layout* utilizando a técnica Hiperbólica e as imagens como representação visual dos elementos analisados.

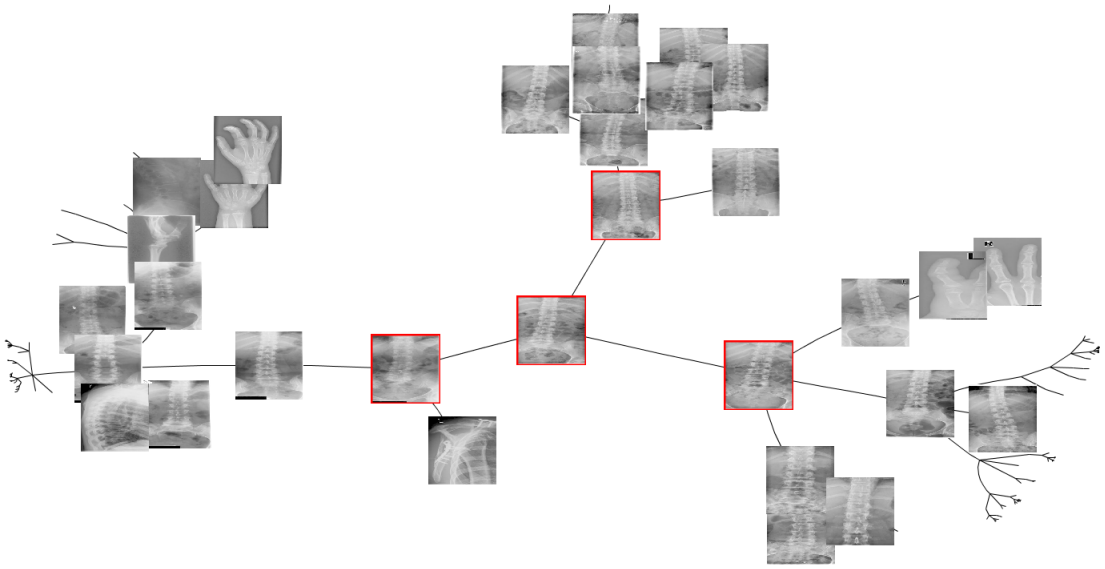
Para expandirmos a análise para outros elementos, basta clicarmos em qualquer ponto do mapeamento ou arrastar a árvore, fazendo com que o nó que esteja mais ao centro da visualização seja melhor visualizado e os elementos que estejam mais distantes sejam ocultados.

Seguindo a mesma ideia, foi utilizado um outro conjunto contendo 9000 imagens de Raios-X[11] de diversas partes do corpo. A Figura 4.19(a) ilustra a projeção para o conjunto de imagens utilizando a técnica MST. A Figura 4.19(b) utiliza a técnica Hiperbólica para visualizar determinados elementos, aumentando o grau de interação com o usuário.

[11]ImageCLEF 2006 Data Set. Departamento de Informática Médica, Alemanha.



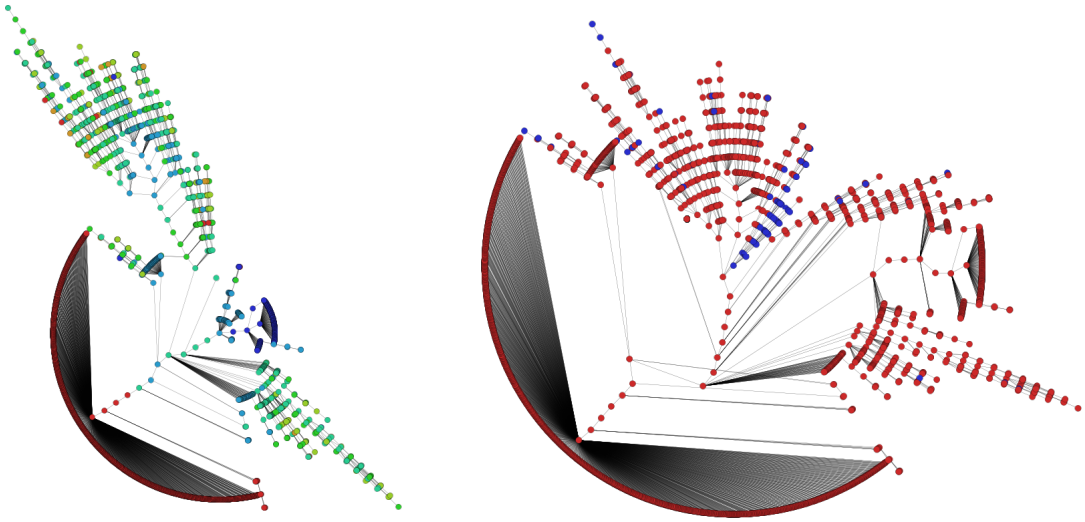
(a) *Layout* para o conjunto de Imagens de Raios-X utilizando MST.



(b) *Layout* para o conjunto de Imagens de Raios-X utilizando o *Layout* Hiperbólico.

Figura 4.19: Projeção para um conjunto de 9000 imagens de Raios-X com seleção de um elemento e seus vizinhos mais próximos.

Podemos notar que, com a utilização do *Layout* Hiperbólico, podemos facilmente analisar um dado específico e seus elementos mais próximos. Através da seleção de um elemento específico, todos os seus vizinhos diretamente ligados são também selecionados, conforme ilustra a Figura 4.19(b). O mesmo ocorre utilizando a Técnica MST (Figura 4.19(a)), porém nesse caso, a navegação se torna mais difícil e menos intuitiva para o usuário.



(a) Busca pelo padrão *normal* (b) Busca pelo padrão *osteofitose*

Figura 4.20: Projeção de um conjunto de laudos médicos utilizando o *Layout* Radial.

Além destes estudos de caso, foram feitas projeções de um pequeno conjunto de laudos médicos enviados pelo InCor. Esses laudos foram projetados utilizando-se a técnica NJ (Ver Seção 3.3.1). A Figura 4.20(a) ilustra uma tendência formada quando realizamos uma busca pela palavra *normal* em uma projeção contendo 2.954 laudos médicos. Pela escala e cores, podemos notar que os grupos formados na parte superior da imagem (definidos pela cor azul), que possuem maior frequência da palavra buscada, são posicionados de forma oposta aos grupos que não possuem nenhuma frequência da palavra (definidos pela cor vermelha). Na Figura 4.20(b) podemos verificar os agrupamentos formados quando é feita a busca por *osteofitose*[12], definidos pela cor azul.

4.4.3 Visualização de Conjunto de Paisagens Diversas

Por fim, foi utilizado um conjunto de 658 imagens[13] que possuíam características diversas, isto é, que descreviam vários cenários, tais como imagens de ônibus, imagens de dinossauros, imagens de flores, entre outras, caracterizando 9 tipos diferentes de cenários.

[12]Osteofitose é uma formação óssea anormal, produzida na proximidade das articulações das vértebras para absorver melhor a sobrecarga desta região.

[13]Disponível em http://wang.ist.psu.edu/docs/related/, acessado em Fevereiro de 2010.

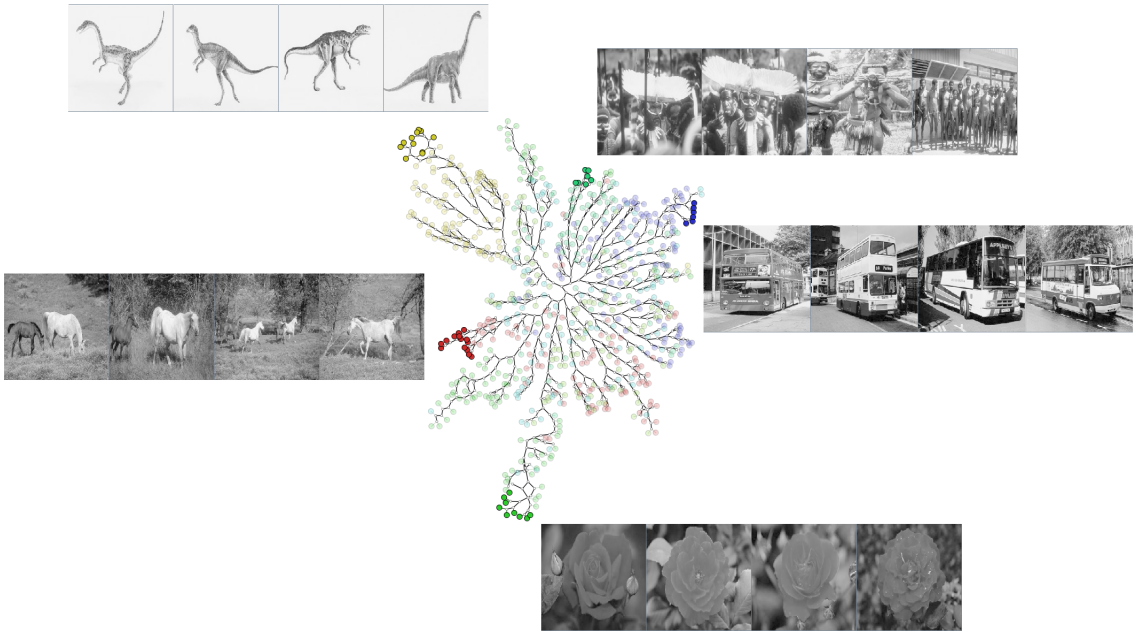
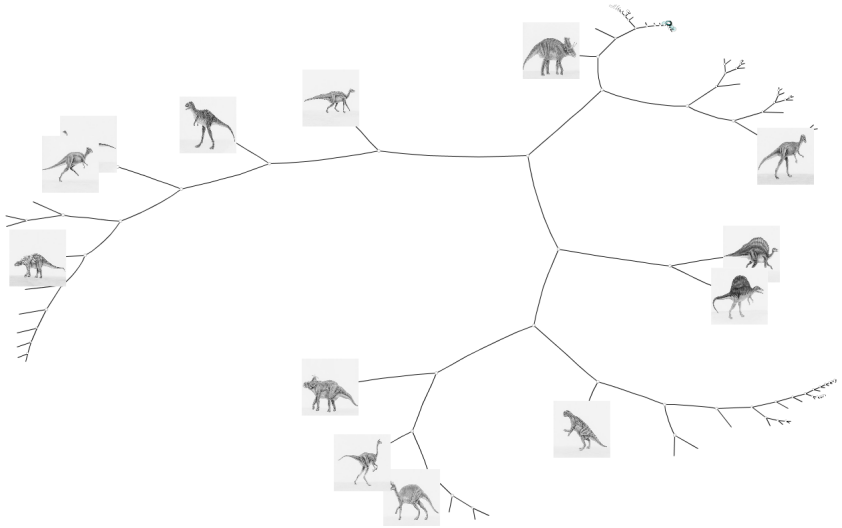
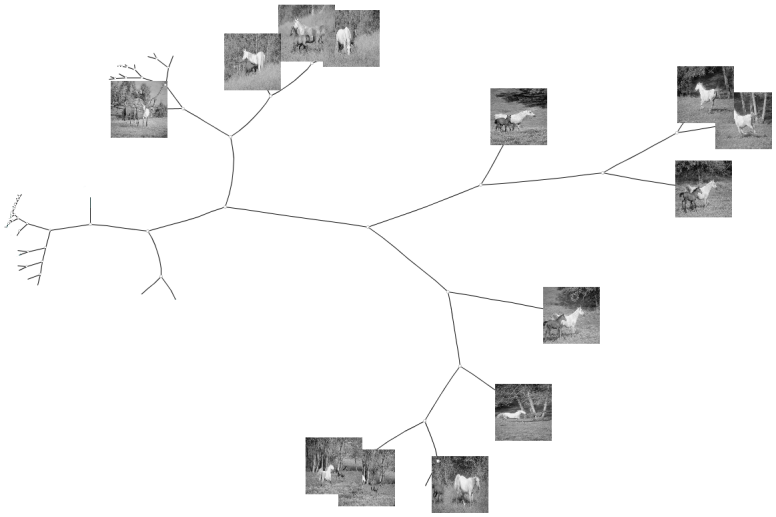


Figura 4.21: Agrupamentos formados pelo mapeamento de 658 imagens utilizando a técnica MST.

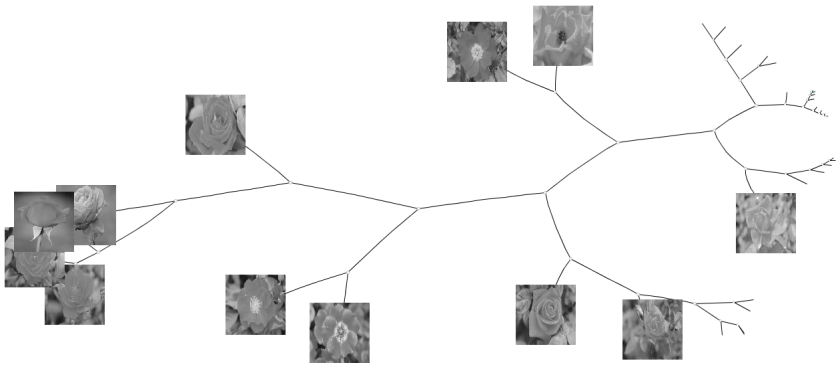
A Figura 4.21 ilustra o mapeamento do conjunto destacando alguns agrupamentos interessantes. A partir de um determinado agrupamento é fácil navegar entre os dados através do *layout* Hiperbólico, bastando selecionar um determinado vértice que seu correspondente é automaticamente centralizado pela técnica hiperbólica. A Figura 4.22 apresenta alguns desses agrupamentos.



(a) Visão detalhada para o agrupamento contendo imagens de dinossauros.



(b) Visão detalhada para o agrupamento contendo imagens de cavalos.



(c) Visão detalhada para o agrupamento contendo imagens de flores.

Figura 4.22: Navegação entre os agrupamentos utilizando o *Layout* Hiperbólico.

4.5 Conclusões

Ferramentas e técnicas de visualização são importantes para o gerenciamento e extração de qualquer tipo de informação. Seus princípios fundamentais são utilizados em diversas áreas para auxílio aos profissionais a obterem resultados mais precisos e concretos. Apesar de existirem diversas técnicas capazes de analisar diferentes tipos de dados, podemos destacar o uso de técnicas de projeção de dados multidimensionais. A ideia central das técnicas de Projeção Multidimensional baseia-se no mapeamento das instâncias de dados em um espaço de 1, 2 ou 3 dimensões, preservando informações de distância entre os dados. Dessa forma, uma representação gráfica pode ser criada de forma a utilizar a habilidade visual humana para reconhecer estruturas, padrões ou anomalias baseadas em similaridade, tais como grupos de elementos similares ou relações entre elementos diferentes.

A aplicação de técnicas de projeção multidimensional no contexto de imagens e textos demonstrou ser capaz de separar e agrupar conjuntos que apresentem alta correlação de conteúdo e também de aproximar em uma vizinhança instâncias de alta similaridade. Técnicas de projeção multidimensional agregadas a técnicas de pré-processamento, tal como a escolha de métricas de distâncias, possibilitaram o refinamento das visualizações, melhorando os resultados e facilitando a interpretação dos dados.

O uso de técnicas de desenho de árvores tornou-se indispensável para a manipulação, navegação e interação com dados projetados, uma vez que facilmente analisamos centenas de dados simultaneamente. Em particular, a Técnica Hiperbólica une a possibilidade de interação e contextualização acerca dos dados, provendo uma forma mais rápida e eficaz para a interpretação dos dados e, adicionalmente, contribui para a análise de um conjunto de dados maior, uma vez que sua geometria hiperbólica o permite. A ferramenta estendida por este trabalho foi disponibilizada juntamente com as demais ferramentas de visualização produzidas pelo grupo no endereço: http://infoserver.lcad.icmc.usp.br.

De forma geral, além das técnicas utilizadas para auxílio na visualização dos dados, este trabalho contribui para um melhor entendimento das características desejáveis para exploração de dados multidimensionais que, apesar de serem altamente exploradas pelo meio científico e acadêmico, dificilmente podem ser empregadas para análise de quaisquer tipos de dados, sem que haja alguma intervenção humana para pré-processamento ou adequação dos meios visuais para o contexto em análise.

Referências Bibliográficas

Altschul, S. F.; W. Gish, W. M.; Myers, E. W.; Lipman, D. J. A basic local alignment search tool. *Journal of Molecular Biology*, v. 215, n. 3, p. 403–410, 1990.

Anderson, T. W. *An introduction to multivariate statistical analysis.* Third ed. Wiley, 2003.

Aurenhammer, F. Voronoi diagrams - a survey of a fundamental geometric data structure. *ACM Computing Surveys*, v. 23, n. 3, p. 345–405, 1991.

Azencott, R.; Wang, J.; Younes, L. Texture classification using windowed fourier filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 19, n. 2, p. 148–153, 1997.

Baker, R. D.; Ebert, G. L. *Discrete mathematics.* St. Louis: Kendall/Hunt Publishing Company, 1997.

Barlow, H. B.; Kaushal, T. P.; Mitchison, G. J. Finding minimum entropy codes. *Neural Computation*, v. 1, n. 3, p. 412–423, 1989.

Battista, G. D.; Eades, P.; Tamassia, R.; Tollis, I. G. *Graph drawing: Algorithms for the visualization of graphs.* New Jersey: Prentice Hall, 1999.

Becker, R. A.; Cleveland, W. S. Brushing scatterplots. *Technometrics*, v. 29, n. 2, p. 127–142, 1987.

Bederson, B. B.; Hollan, J. D. Pad++: a zooming graphical interface for exploring alternate interface physics. In: *Proceedings of the 7th annual ACM symposium on User interface software and technology*, New York - NY, USA: ACM Press, 1994, p. 17–26.

Bollobás, B. *Graph theory.* Springer-Verlag, 1990.

Bondy, G. A.; Murty, U. S. R. *Graph theory with applications.* New York: North Holland, 1976.

Book, G.; Keshary, N. *Radial tree graph drawing algorithm for representing large hierarchies.* Relatório Técnico, 2001.

Brodlie, K. W.; Carpenter, L. A.; Earnshaw, R. A.; Gallop, J. R.; Hubbold, R. J.; Mumford, A. M.; Osland, C. D.; Quarendon, P. *Scientific visualization: techniques and applications.* New York - NY, USA: Springer-Verlag New York, Inc., 1992.

Card, S. K.; MacKinlay, J. D.; Shneiderman, B. *Readings in information visualization: Using vision to think.* San Francisco - CA, USA: Morgan Kaufmann Publishers Inc., 1999.

Chernoff, H. The use of faces to represent points in k-dimensional space graphically. *Journal of the American Statistical Association*, v. 68, n. 342, p. 361–368, 1973.

Chi, E. H.; Riedl, J. T. An operator interaction framework for visualization systems. In: *Proceedings of the 1988 IEEE Symposium on Information Visualization*, Washington - DC, USA: IEEE Computer Society Press, 1998, p. 63–70.

Cook, D.; Buja, A.; Cabrera, J. Projection pursuit indexes based on orthonormal function expansions. *Journal of Computational and Graphical Statistics*, v. 2, n. 3, p. 225–250, 1993.

Costa, L. F.; Junior, R. M. C. *Shape analysis and classification: Theory and practice.* Boca Raton - FL, USA: CRC Press Inc., 2000.

Cox, T. F.; Cox, M. A. A. *Multidimensional scaling.* Second ed. Chapman & Hall/CRC, 2000.

Crawford, S. L.; Fall, T. C. Projection pursuit techniques for visualizing high-dimensional data sets. In: Nielson, G. M.; Shriver, B., eds. *Visualization in Scientific Computing*, IEEE Computer Society Press, 1990, p. 94–108.

Deco, G.; Obradovic, D. Linear redundancy reduction learning. *Neural Networks*, v. 8, n. 5, p. 751–755, 1995.

Demmel, J. W. *Applied numerical linear algebra.* Society for Industrial and Applied Mathematics, 1997.

Ding, C. A probabilistic model for dimensionality reduction in information retrieval and filtering. In: *Proceedings of 1st SIAM Computational Information Retrieval Workshop*, Raleigh - NC, USA, 2000.

Donoho, D. On minimum entropy deconvolution. In: *Applied Time Series Analysis II*, New York - NY, USA: Academic Press, 1981, p. 565–608.

Drebin, R. A.; Carpenter, L.; Hanrahan, P. Volume rendering. In: *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, New York - NY, USA: ACM Press, 1988, p. 65–74.

Duda, R. O.; Hart, P. E. *Pattern classification and scene analysis.* New York - NY, USA: Wiley-Interscience Publication, 1973.

Eades, P. A heuristic for graph drawing. *Congressus Numerantium*, v. 42, p. 149–160, 1984.

Eades, P. Drawing free trees. *Bulletin of the Institute for Combinatorics and its Applications*, p. 10–36, 1992.

Eler, D. M.; Nakazaki, M. Y.; Paulovich, F. V.; Santos, D. P.; Andery, G. F.; Oliveira, M. C. F.; Neto, J. B.; Minghim, R. Visual analysis of image collections. *The Visual Computer*, v. 25, p. 923–937, 2009.

Eler, D. M.; Nakazaki, M. Y.; Paulovich, F. V.; Santos, D. P.; Oliveira, M. C. F.; Neto, J. B. E. S.; Minghim, R. Multidimensional visualization to support analysis of image collections. *In: XXI Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2008)*, p. 289–296, 2008.

Elvins, T. A survey of algorithms for volume visualization. *SIGGRAPH Computer Graphics*, v. 26, n. 3, p. 34–44, 1992.

Faloutsos, C.; Lin, K. Fastmap: A fast algorithm for indexing, datamining and visualization of traditional and multimedia databases. In: *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, New York - NY, USA: ACM Press, 1995, p. 163–174.

Ferreira, C. B. R.; Nascimento, H. A. D. Visualização de informações - uma abordagem prática. *Jornadas de Atualização em Informática*, p. 1262–1312, 2005.

Fodor, I. K. *A survey of dimension reduction techniques.* Relatório Técnico, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 2002.

Friedman, J. H. Exploratory projection pursuit. *Journal of the American Statistical Association*, v. 82, n. 397, p. 249–266, 1987.

Friedman, J. H.; Tukey, J. W. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions Computers*, v. C-23, n. 9, p. 881–890, 1974.

Fukanaga, K. *Introduction to statistical pattern recognition.* Second ed. San Diego - CA, USA: Academic Press Professional Inc., 1990.

Furnas, G. W. The fisheye view: a new look at structured files. *Bell Laboratories*, 1981.

FYFE, C.; BADDELEY, R. Non-linear data structure extraction using simple hebbian networks. *Biological Cybernetics*, v. 72, n. 6, p. 533–541, 1995.

GERSHON, N.; EICK, S. G. Visualization's new tack: Making sense of information. *IEEE Spectrum*, v. 32, n. 11, p. 38–56, 1995.

HASCOËT, M.; BAUDOUIN-LAFON, M. Visualization interactive d'infomation. *Revue 13*, v. 1, n. 1, p. 77–108, 2001.

HAYKIN, S. *Blind deconvolution.* Prentice Hall, 1994.

HERMAN, I.; MELANÇON, G.; MARSHALL, M. S. Graph visualization and navigation in information visualization: a survey. *IEEE Transactions on Visualization and Computer Graphics*, v. 6, n. 1, p. 24–43, 2000.

HU, J.; SHEN, X.; SHAO, Y.; BYSTROFF, C.; ZAKI, M. J. Mining protein contact maps. In: *Proceedings of 2nd BIOKDD Workshop on Data Mining in Bioinformatics*, 2002, p. 3–10.

HUANG, K.; AVIYENTE, S. Rotation invariant texture classiŕcation with ridgelet transform and fourier transform. In: *Proceedings of the ICIP*, 2006, p. 2141–2144.

HUBER, P. J. Projection pursuit. In: *Annals of Statistics*, 1985, p. 435–475.

INSELBERG, A. Multidimensional detective. In: *Proceedings of the 1997 IEEE Symposium on Information Visualization*, Washington - DC, USA: IEE Computer Society Press, 1997.

JOLLIFFE, I. T. Principal component analysis. *Springer-Verlag New York, Inc.*, 1986.

JONES, M. C.; SIBSON, R. What is projection pursuit? *Journal of the Royal Statistical Society*, v. A, n. 150, p. 1–36, 1987.

KAMADA, T.; KAWAI, S. An algorithm for drawing general undirected graphs. *Information Processing Letters*, v. 31, p. 7–15, 1989.

KAUFMAN, A. Advances in volume visualization. *SIGGRAPH'98: Course Notes*, 1998.

KEIM, D. A. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics*, v. 6, n. 1, p. 59–79, 2000.

KEIM, D. A. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, v. 1, n. 7, p. 100–107, 2002.

KIRBY, M. *Geometric data analysis: An empirical approach to dimension reduction and the study of patterns.* John Wiley and Sons, 2001.

KUCHINSKY, A.; BALDONADO, M. Guidelines for using multiple views in information visualization. In: *Proceedings of the working conference on Advanced visual interfaces*, New York - NY, USA: ACM Press, 2000, p. 110–119.

LAMPING, J.; RAO, R. The hyperbolic browser: A focus+context technique for visualizing large hierarchies. *Journal of Visual Languages and Computing*, v. 7, n. 1, p. 33–55, 1996.

LAMPING, J.; RAO, R.; PIROLLI, P. A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In: *CHI'95 Conference Proceedings*, ACM Press, 1995.

LEUNG, Y. K.; APPERLEY, M. D. A review and taxonomy of distortion-oriented presentation techniques. *ACM Transactions on Computer-Human Interaction*, v. 1, n. 2, p. 126–160, 1994.

LEVKOWITZ, H.; MINGHIM, R.; NONATO, L. G.; PAULOVICH, F. V. Least square projection: a fast high precision multidimensional projection technique and its application to document mapping. *IEEE Transactions on Visualization and Computer Graphics*, v. 14, n. 3, p. 564–575, 2007.

LEVKOWITZ, H.; OLIVEIRA, M. C. F. From visual data exploration to visual data mining: a survey. *IEEE Transactions on Visualization and Computer Graphics*, v. 9, n. 3, p. 378–394, 2003.

LEVOY, M. Efficient ray tracing of volume data. *ACM Transaction on Graphics*, v. 9, n. 3, p. 245–261, 1990.

LEVOY, M. Display of surface from volume data. *IEEE Computer Graphics and Applications*, v. 8, n. 3, p. 29–37, 1998.

LEVOY, M.; HANRAHAN, P.; HOEHNE, K. H.; KAUFMAN, A.; LORENSEN, W. Volume visualization algorithms and architectures. *SIGGRAPH'90: Course Notes*, 1990.

LIN, X. *Analysis of algorithms for drawing graphs.* Tese de Doutorado, Department of Computer Science, University of Queensland, Brisbane, Australia, 1992.

LORENSEN, W. E.; CLINE, H. E. Marching cubes: a high resolution 3d surface construction algorithm. *Computer Graphics*, v. 21, n. 4, p. 163–169, 1987.

LORENZ, E. *Empirical orthogonal eigenfunctions and statistical weather prediction.* Relatório Técnico, M.I.T., Cambridge, MA, 1956.

LUZZARDI, P. R. G. *Critérios de avaliação de técnicas de visualização de informações hierárquicas.* Tese de Doutorado, Universidade Federal do Rio Grande do Sul, 2003.

MARDIA, K. V.; KENT, J. T.; BIBBY, J. M. *Multivariate analysis.* Seventh ed. Academic Press, 2000.

MINGHIM, R.; OLIVEIRA, M. C. F. Uma introdução à visualização computacional. *Jornadas de Atualização em Informática*, p. 85–131, 1997.

MUNZNER, T. H3: Laying out large directed graphs in 3d hyperbolic space. In: *Proceedings of the 1997 IEEE Symposium onlic Space*, IEEE CS Press, 1997, p. 2–10.

MUNZNER, T. Drawing large graphs with h3viewer and site manager. In: *Proceedings of the Symposium on Graph Drawing GD'98*, Springer-Verlag, 1998, p. 384–393.

MUNZNER, T.; BURCHARD, P. Visualizing the structure of the world wide web in 3d hyperbolic space. In: *Proceedings of the VRML'95 Symposium*, ACM Press, 1995.

NEY, D. R.; FISHMAN, E. K.; MAGID, D. Volume rendering of computed tomography data: principles and techniques. *IEEE Computer Graphics and Applications*, v. 10, n. 2, p. 24–32, 1990.

PAIVA, A. C.; SEIXAS, R. B.; GATTASS, M. *Introdução à visualização volumétrica.* Relatório Técnico, Departamento de Ciência da Computação, Pontifícia Universidade Católica do Rio de Janeiro, 1999.

PAULOVICH, F. V. *Técnicas geométricas para análise visual de dados - integrando mineração e visualizações.* Exame de qualificação, Instituto de Ciências Matemáticas e de Computação - USP, 2006.

PAULOVICH, F. V.; MINGHIM, R. Text map explorer: a tool to create and explore document map. In: *Proceedings of the conference on Information Visualization: IV'06*, Washington - DC, USA: IEEE Computer Society Press, 2006, p. 245–251.

PAULOVICH, F. V.; OLIVEIRA, M. C. F.; MINGHIM, R. The projection explorer: A flexible tool for projection-based multidimensional visualization. In: *Proceedings of XX Brazilian Sysmposium on Computer Graphics and Image Processing - SIBGRAPI 2007*, Belo Horizonte, Brazil: IEEE Computer Society Press, 2007, p. 27–34.

PHILIPS, D. C. The development of crystallographic enzymology. In: *Proceedings of the 1970 Biochemical Society Annual Symposium*, 1970, p. 11–28.

PICKETT, R. M.; GRINSTEIN, G. G. Iconographic displays for visualizing multidimensional data. In: *Proceedings of the 1988 IEEE International Conference on Systems, Man and Cybernetics*, 1988, p. 514–519.

PIETAL, M. J.; TUSZYNSKA, I.; BUJNICKI, J. M. Protmap2d: visualization, comparison and analysis of 2d maps of protein structure. *Bioinformatics*, v. 23, n. 11, p. 1429–1430, 2007.

PILLAT, R. M.; VALIATI, E. R.; FREITAS, C. M. D. Experimental study on evaluation of multidimensional information visualization techniques. In: *CLIHC'05*, Cuernavaca, Mexico, 2005, p. 20–30.

PRIM, R. C. Shortest connection network and some generalizations. *Bell Syst. Tech.*, v. 36, p. 1389–1401, 1957.

ROBERTSON, G. G.; CARD, S. K.; MACKINLAY, J. D. Information visualization using 3d interactive animation. *Communications of the ACM*, v. 4, n. 36, p. 57–71, 1993.

Saitou, N.; Nei, M. The neighbor joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, v. 4, n. 4, p. 406–425, 1987.

Schmidhuber, J.; Eldracher, M.; Foltin, B. Semilinear predictability minimization produces well-known feature detectors. *Neural Computation*, v. 8, n. 4, p. 773–786, 1996.

Schneider, H. *Métodos de análise filogenética.* Ribeirão Preto: Holos Editora, 2003.

Shalvi, O.; Weinstein, E. New criteria for blind deconvolution of nonminimum phase systems (channels). *IEEE Transactions on Information Theory*, v. 36, n. 2, p. 312–321, 1990.

Shalvi, O.; Weinstein, E. Super-exponential methods for blind deconvolution. *IEEE Transactions on Information Theory*, v. 39, n. 2, p. 204–519, 1993.

Shiloach, Y. *Arrangements of planar graphs on the planar lattices.* Tese de Doutorado, Weizmann Institute of Science, Rehovot, Israel, 1976.

Siegel, J. H.; Farrell, E. J.; Goldwyn, R.; Friedman, H. The surgical implication of physiologic patterns in myocardial infarction shock. *Surgery*, v. 72, n. 1, p. 126–141, 1972.

Sorkine, O.; Cohen-Or, D. Least-square meshes. In: *Proceedings of Shape Modeling International*, IEEE Computer Society Press, 2004, p. 191–199.

Spritzer, A. S.; Freitas, C. M. D. S. Navigation and interaction in graph visualizations. *Revista de Informática Teórica e Aplicada*, v. 15, n. 1, p. 111–136, 2008.

Stytz, M. R.; Frieder, G.; Frieder, O. Three-dimensional medical imaging: algorithms and computer systems. *ACM Computing Surveys*, v. 23, n. 4, p. 423–499, 1991.

Sun, J. Some practical aspects of exploratory projection pursuit. *SIAM Journal of Science Computer*, v. 14, n. 1, p. 68–80, 1993.

Tan, P.; Steinbach, M.; Kumar, V. *Introduction to data mining.* Addison-Wesley Longman Publishing Co., Inc., 2006.

Tejada, E.; Minghim, R.; Nonato, L. G. On improved projection techniques to support visual exploration of multi-dimensional data sets. *Information Visualization*, v. 2, n. 4, p. 218–231, 2003.

Telles, G. P.; Minghim, R.; Paulovich, F. V. Normalized compression distance for visual analysis of document collections. *Computer & Graphics, Special Issue on Visual Analytics*, v. 31, n. 3, p. 327–337, 2007.

Tide, U.; Hoehne, K.; Bomans, M.; Pommert, A.; Riemer, M.; Wiebecke, G. Investigation of medical 3d-rendering algorithms. *IEEE Computer Graphics and Applications*, v. 10, n. 2, p. 41–53, 1990.

UDUPA, J. K.; ODHNER, D. Shell rendering. *IEEE Computer Graphics and Applications*, v. 13, n. 6, p. 58–67, 1993.

VALDIVIA, A. M. C. *Mapeamento de dados multidimensionais usando árvores filogenéticas: foco em mapeamento de textos.* Dissertação de Mestrado, USP, 2007.

WARD, M. O. A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization*, v. 1, n. 3–4, p. 194–210, 2002.

WEST, D. B. *Introduction to graph theory.* Prentice Hall, 1996.

WESTOVER, L. Footprint evaluation for volume rendering. *SIGGRAPH'90: Proceedings of the 17th annual conference on Computer Graphics and Interactive Techniques*, v. 4, n. 4, p. 367–376, 1990.

WIDMANN, A.; SCHRÖGER, E. Bootstrapping the distribution of the city-block distance between two repeated measures. Online, available http://www.uni-leipzig.de/~biocog/widmann/minkowski.html. Last Access: February, 2008, 1999.

WILLS, G. J. Nicheworks - interactive visualization of very large graphs. In: *Proceedings of Graph Drawing*, 1997, p. 403–414.

WONG, P. C. Visual data mining. *IEEE Computer Graphics and Applications*, v. 19, n. 5, p. 20–21, 1999.

YANG, L. Distance-preserving projection of high dimensional data. *Pattern Recognition Letters*, v. 25, p. 259–266, 2003.