

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Análise e Implementação de Técnicas de Visualização de Bases de Dados para Extratores de Características de Imagens

Helena do Amparo Teixeira Aravechia

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-C²MC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Helena do Amparo Teixeira Aravechia

Análise e Implementação de Técnicas de Visualização de Bases de Dados para Extratores de Características de Imagens

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestra em Ciências – Ciências de Computação e Matemática Computacional. *EXEMPLAR DE DEFESA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Agma Juci Machado Traina

USP – São Carlos
Dezembro de 2022

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

A658a Aravechia, Helena do Amparo Teixeira
Análise e Implementação de Técnicas de
Visualização de Bases de Dados para Extratores de
Características de Imagens / Helena do Amparo
Teixeira Aravechia; orientadora Profa. Dra. Agma
Juci Machado Traina. -- São Carlos, 2022.
97 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2022.

1. visualização de informação. 2. recuperação de
imagens baseada em conteúdo. 3. busca por
similaridade de imagens. I. Traina, Profa. Dra.
Agma Juci Machado, orient. II. Título.

Helena do Amparo Teixeira Aravechia

**Analysis and Implementation of Database Information
Visualization Techniques for Image Feature Spaces**

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

Concentration Area: Computer Science and Computational Mathematics

Advisor: Profa. Dra. Agma Juci Machado Traina

**USP – São Carlos
December 2022**

RESUMO

ARAVECHIA, H. A. T. **Análise e Implementação de Técnicas de Visualização de Bases de Dados para Extratores de Características de Imagens**. 2022. 94 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

O crescimento das tecnologias de captura de imagens digitais é também refletido em diversas áreas da ciência, como por exemplo redes sociais, e também no volume crescente de exames médicos por imagens, possibilitando a criação de conjuntos de imagens referentes a diversas enfermidades, indicando grandes possibilidades para implementação de sistemas de apoio a decisão médica. Neste cenário, a proposta desta pesquisa foi alinhar técnicas de visualização de informação à análise de extratores de características no contexto de consultas por similaridade, por meio da visualização e interpretação dos vetores de características gerados a partir de imagens. É esperado que esta aplicação seja capaz de oferecer uma prévia da eficácia de uma consulta por similaridade a partir da análise visual de seu espaço de características e reduzir a carga de processamento a posteriori que é usualmente realizado em análises baseadas em médias de consultas realizadas e curvas de precisão versus revocação.

Palavras-chave: visualização de informação, consultas por similaridade, imagens médicas, recuperação de imagens baseada em conteúdo.

ABSTRACT

ARAVECHIA, H. A. T. **Analysis and Implementation of Database Information Visualization Techniques for Image Feature Spaces**. 2022. 94 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

The recent growth in image capturing technologies can be seen in several areas, such as social networks, and medical applications, which make it possible the creation of numerous image datasets and indicate great possibilities for clinical decision support systems. In this context, this project aims to apply information visualization technics into the analysis of feature extractors, focused on similarity search systems, using visualization to analyze and comprehend the behavior of images feature spaces. It is expected that this work will be able of providing a forecast for the effectiveness of a similarity query from the visual analysis of its feature space and reduce the prior processing load that is usually made in analyses based on similarity queries, by the means of precision versus recall curves.

Keywords: information visualization, similarity queries, medical images, content based image retrieval.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de consulta do tipo <i>range</i> , com raio r e centro de consulta q . Elaborado pela autora.	26
Figura 2 – Exemplo de consulta do tipo <i>k-nn</i> , para $k=4$ e centro de consulta q . Elaborado pela autora.	27
Figura 3 – Fotos e respectivos histogramas RGB	28
Figura 4 – Pipeline da visualização de informação utilizado neste trabalho. Baseado em proposta apresentada em (WARD; GRINSTEIN; KEIM, 2015)	30
Figura 5 – Exemplo de visualização em gráfico de dispersão de conjunto de pontos multidimensionais sintéticos processados com a técnica t-SNE (MAATEN; HINTON, 2008). Imagem retirada de (WATTENBERG; VIÉGAS; JOHNSON, 2016)	34
Figura 6 – Achados em gráficos de dispersão. Imagem adaptada de (BEHRISCH <i>et al.</i> , 2018)	35
Figura 7 – Matriz de dispersão gerada a partir do dataset Iris. Autoria de Mike Bostock, disponível em < https://beta.observablehq.com/@mbostock/d3-scatterplot-matrix >	36
Figura 8 – Achados em matrizes de dispersão. Imagem adaptada de (BEHRISCH <i>et al.</i> , 2018)	37
Figura 9 – Coordenadas Paralelas geradas a partir do dataset Iris. Autoria de Mike Bostock, disponível em < http://mbostock.github.io/d3/talk/20111116/iris-parallel.html >	38
Figura 10 – Achados em coordenadas paralelas. Imagem adaptada de (BEHRISCH <i>et al.</i> , 2018)	38
Figura 11 – Achados em visualizações baseadas em pixels. Adaptada de (BEHRISCH <i>et al.</i> , 2018)	39
Figura 12 – Achados em visualizações radiais. Imagem adaptada (BEHRISCH <i>et al.</i> , 2018)	39
Figura 13 – Otimização e achados em glyphs. Imagem adaptada de (BEHRISCH <i>et al.</i> , 2018)	40
Figura 14 – Conjuntos de dados escolhidos	52
Figura 15 – Ferramentas da visualização	54
Figura 16 – Exemplos de oclusão	57
Figura 17 – Etapas da implementação do projeto	58
Figura 18 – Estado da visualização com variação no número de iterações	59
Figura 19 – Tela inicial da aplicação	60

Figura 20 – Diferentes imagens e classes representados na visualização do conjunto SegMRF	61
Figura 21 – Formação de Agrupamentos nas visualizações do conjunto SegMRF	62
Figura 22 – Percepção de cor na visualização do dataset val2017	63
Figura 23 – Percepção de cor na visualização do dataset val2017 - Ampliação e exemplos	64
Figura 24 – Distribuição dos gradientes de cores aleatórias em espalhamento	65
Figura 25 – Representação de tipos de folhas distribuídas	66
Figura 26 – Ampliação de tipos de folhas distribuídas	67

LISTA DE CÓDIGOS-FONTE

Código-fonte 1 – Extração de características do conjunto val2017	79
Código-fonte 2 – Extração de características do conjunto Swedish Leaves	81
Código-fonte 3 – Extração de características do conjunto SegMRF	82
Código-fonte 4 – Definição de casos teste na redução de dimensionalidade	83
Código-fonte 5 – Definição do dataset sintético de imagens coloridas	87
Código-fonte 6 – Cálculo de silhueta	90

LISTA DE TABELAS

Tabela 1 – Tecnologias testadas e conformidade com requisitos propostos	47
Tabela 2 – Conjuntos de dados selecionados	51
Tabela 3 – Parâmetros recomendados na documentação da técnica t-SNE	59
Tabela 4 – Casos teste técnica UMAP	59
Tabela 5 – Casos Teste t-SNE	60
Tabela 6 – Taxas de oclusão classificadas	65
Tabela 7 – Valores brutos de taxas de oclusão	93
Tabela 8 – Taxas de oclusão por caso de teste em cada conjunto de dados em ordem de casos	94

LISTA DE ABREVIATURAS E SIGLAS

CBIR	Content-Based Image Retrieval
PCA	Principal Component Analysis
t-SNE	t-distributed Stochastic Neighbor Embedding
UMAP	Uniform Manifold Approximation and Projection

SUMÁRIO

1	INTRODUÇÃO	19
1.1	Motivação e Contextualização	19
1.2	Objetivos	20
1.2.1	<i>Hipótese</i>	21
1.3	Organização da monografia	21
2	RECUPERAÇÃO DE IMAGENS BASEADA EM CONTEÚDO E CONSULTAS POR SIMILARIDADE	23
2.1	Considerações Iniciais	23
2.2	Visão geral	23
2.3	Extração de características e representação dos dados	24
2.3.1	<i>Dados complexos</i>	24
2.3.2	<i>Extratores de características</i>	24
2.3.3	<i>Descritores de Imagens e Funções de Distância</i>	25
2.4	Consultas por similaridade	26
2.4.1	<i>Técnicas de Avaliação</i>	26
2.4.2	<i>Lacuna Semântica</i>	27
2.4.3	<i>Realimentação por relevância</i>	28
2.5	Considerações Finais	28
3	VISUALIZAÇÃO DE INFORMAÇÃO	29
3.1	Considerações Iniciais	29
3.2	Visão Geral	29
3.3	Visualização de dados de alta dimensionalidade	32
3.3.1	<i>Técnicas de redução de dimensionalidade</i>	33
3.3.2	<i>Gráficos de dispersão</i>	35
3.3.3	<i>Matrizes de dispersão</i>	36
3.3.4	<i>Coordenadas paralelas</i>	37
3.3.5	<i>Visualizações baseadas em pixels</i>	37
3.3.6	<i>Visualização radial</i>	39
3.3.7	<i>Glyphs</i>	39
3.4	Qualidade em visualizações de dados	39
3.4.1	<i>Abordagens Quantitativas</i>	40

3.4.2	<i>Abordagens Qualitativas</i>	41
3.5	Aplicações relacionadas	41
3.6	Tecnologias em Visualização	42
3.7	Considerações Finais	42
4	DESENVOLVIMENTO	45
4.1	Considerações iniciais	45
4.2	Proposta	45
4.3	Metodologia	46
4.3.1	<i>Seleção de técnicas de visualização e redução de dimensionalidade</i>	47
4.4	Bases de dados	49
4.4.1	<i>Crítérios de seleção e escolha de bases</i>	49
4.4.2	<i>Bases Escolhidas</i>	50
4.5	Visualização Interativa	53
4.6	Análise de silhueta e oclusão	53
4.7	Testes e análise	56
4.7.1	<i>Parâmetros na redução de dimensionalidade</i>	58
4.7.2	<i>Aplicação</i>	59
4.7.3	<i>Análise</i>	60
4.8	Considerações finais	67
5	CONCLUSÕES	69
5.1	Conclusões	69
5.2	Limitações	69
5.3	Propostas futuras	70
	REFERÊNCIAS	71
	APÊNDICE A CÓDIGOS-FONTE	79
	APÊNDICE B DADOS BRUTOS	93

INTRODUÇÃO

1.1 Motivação e Contextualização

Um modo direto e eficiente de comunicar informações entre pessoas é utilizando imagens e recursos gráficos. Além disso, com o barateamento das tecnologias de aquisição de imagens têm aumentado exponencialmente o volume de imagens coletadas na prática clínica, em exames de imagens. Desse modo, desenvolver recursos para tirar proveito desse grande volume de dados, de modo a apoiar a elaboração de diagnósticos mais confiáveis e de modo mais rápido, tem sido um desafio importante colocado para os pesquisadores da área de processamento de imagens, recuperação de informação e bases de dados. Nesse contexto, os Sistemas de Recuperação de Imagens Médicas por Similaridade, utilizando recuperação por conteúdo (CBIR - *Content-Based Image Retrieval*) (TORRES; FALCAO, 2006) em ambientes reais de estudo, análise e diagnóstico radiológico têm sido foco de pesquisas em vários centros ao redor do mundo (KALPATHY-CRAMER *et al.*, 2015; BEDO *et al.*, 2016; PEDRONETTE; TORRES, 2017; KITANOVSKI *et al.*, 2017; PIRAS; GIACINTO, 2017; MA *et al.*, 2017; BEDO *et al.*, 2022).

Ao longo dos anos, essa área de pesquisa tem encontrado muitos desafios e vem desvendando novas fronteiras. Contudo, ainda persistem obstáculos tais como a divergência entre os resultados obtidos automaticamente pelo sistema e aqueles esperados pelo radiologista (especialista na área), conhecido na literatura como “gap semântico” (DHARANI; AROQUIARAJ, 2013), bem como dificuldades tais como a falta de estudos sobre a viabilidade clínica dessas ferramentas como forma de auxílio ao diagnóstico, sendo eles os principais responsáveis pela não efetivação do uso dessa tecnologia no ambiente médico-hospitalar (GHOSH *et al.*, 2011; KALPATHY-CRAMER *et al.*, 2015).

Um dos grandes problemas com a utilização de sistemas CBIR, como já colocado, é que as características que são extraídas das imagens, e utilizadas para fazer o processo de indexação, comparação e recuperação das imagens nem sempre atendem ao que o usuário entende como

similaridade. O processo de análise e refinamento das características que são extraídas das imagens é trabalhoso e demorado, o que faz que muitos trabalhos percam sua aplicabilidade em ambientes reais. Desse modo, uma técnica que permita escolher e aprimorar algoritmos e técnicas de extração de características de imagens seria muito benéfica para a área de recuperação de imagens por conteúdo.

Considerando o grande volume de dados produzidos, neste e outros contextos, técnicas de visualização vêm sendo aplicadas de forma a permitir maneiras simples, rápidas e efetivas na análise e interação humana com tais dados, evidenciando a presença de padrões de interesse (LIU *et al.*, 2018). É também crescente a sua utilização como parte interativa em sistemas que aplicam aprendizado de máquina, com foco na participação humana no ciclo de repetição de processos automatizado (CHEN; GOLAN, 2016; SACHA *et al.*, 2017).

Desta forma o presente projeto de Mestrado visou aplicar este conceito em sua contribuição: utilizar técnicas de visualização de dados para permitir analisar de modo rápido e eficaz, quão bem um tipo de características de imagens separa e classifica imagens que deverão ser indexadas e recuperadas por conteúdo. A abordagem desenvolvida aspira atuar como alternativa a métodos que exijam maiores graus de treinamento, como interpretação de conjuntos numéricos e métricas, ou táticas que requeiram alto poder de processamento, visto que a renderização das visualizações neste trabalho pôde ser realizada em máquinas pessoais.

1.2 Objetivos

O trabalho teve como objetivo geral a análise de técnicas de visualização aplicadas ao contexto de espaços de características extraídas de imagens médicas.

Buscando explorar esse objetivo foi formulada a seguinte questão de pesquisa para nortear o trabalho:

A análise visual de vetores de características gerados por extratores distintos em imagens médicas possibilita antecipar quão apropriados são tais extratores considerando o contexto em análise para apoiar o processamento de consultas por similaridade.

A partir do embasamento da questão proposta, foram explorados os seguintes tópicos no desenvolvimento do trabalho:

- Estudar, analisar e implementar técnicas de visualização de dados, buscando opções adequadas para o projeto de um modelo de visualização de vetores de características.
- Explorar os resultados da análise visual de vetores de características em conjuntos de imagens do mundo real, de contexto médico ou diverso, visando identificar melhores mecanismos visuais para seleção de extratores de características.

- Realizar estudos e validação da aplicabilidade da ferramenta como auxílio a seleção de extratores de característica mais adequados.

1.2.1 Hipótese

A partir dos objetivos gerais e questão de pesquisa delineados anteriormente, a seguinte hipótese foi definida:

A análise visual de vetores de características gerados por extratores distintos em imagens possibilita antecipar quão apropriados são tais extratores considerando o contexto em análise para apoiar o processamento de consultas por similaridade.

Ou seja, seria possível, antes de empregar tempo e recursos de processamento no cálculo de medidas como de precisão e revocação para análise da eficácia de uma consulta por similaridade, analisar visualmente características que influenciam na eficiência das buscas, como a escolha de extratores e distâncias, oferecendo uma previsão do comportamento do extrator de características, antes do processamento efetivo das consultas. Desse modo, o esforço no desenvolvimento de extratores de características de imagens poderá ser realizado com mais respaldo e de modo mais eficaz.

1.3 Organização da monografia

Esta monografia está organizada em cinco capítulos: no Capítulo 1, são expostos o tema e motivação, juntamente dos objetivos e hipótese da pesquisa realizada no Mestrado. Os dois capítulos seguintes apresentam os conceitos específicos abordados no trabalho e as principais referências relacionadas, sendo os tópicos de bases de dados, recuperação de imagens por conteúdo e busca por similaridade tratados no Capítulo 2; e o conteúdo relacionado aos métodos de visualização de informações, incluindo técnicas e métricas de qualidade, é abordado no Capítulo 3. Na sequência, o Capítulo 4 apresenta todo o conteúdo referente a definição da metodologia, desenvolvimento das ferramentas e análise dos resultados obtidos. Por fim, no Capítulo 5 constam a consolidação das análises, conclusões gerais e sugestões de trabalhos futuros.

RECUPERAÇÃO DE IMAGENS BASEADA EM CONTEÚDO E CONSULTAS POR SIMILARIDADE

2.1 Considerações Iniciais

Neste capítulo são apresentadas as principais técnicas e conceitos de processamento de imagens e bases de dados, que compõem o ferramental necessário para a extração de características de imagens, sua representação através de vetores de características, cálculo de distâncias entre tais vetores e consultas por similaridade.

2.2 Visão geral

A partir da grande quantidade de dados produzidos e armazenados, principalmente imagens, e considerando o avanço na qualidade e acesso às tecnologias de aquisição como câmeras e exames computadorizados, a criação de grandes *datasets* foi possível, trazendo consigo tanto as vantagens da disponibilidade de dados quanto as dificuldades de processamento, indexação e consultas. Tal cenário foi favorável ao desenvolvimento da área de Recuperação de Imagens Baseada em Conteúdo (*Content-Based Image Retrieval*, ou CBIR), que aplica uma abordagem de busca independente de marcações textuais (rótulos, metadados, legendas e palavras-chave), frequentemente trabalhosas para fazer e não representativas da totalidade dos elementos da imagem, podendo apresentar vieses do usuário. Em sistemas CBIR são utilizadas as informações inerentes às imagens, tais como: cor, forma dos elementos e textura para realizar buscas por similaridade, sendo estes parâmetros mais sistemáticos do que anotações (TORRES; FALCAO, 2006).

Para detalhar os processos envolvidos na arquitetura de um sistema CBIR é possível

dividir as etapas em duas partes (TORRES; FALCAO, 2006), primeiramente inserção de dados, que envolve processamento das imagens, extração de características e armazenamento na base; e processamento de consultas, com o processamento da imagem de consulta e descrição da consulta em si, aplicação da métrica de similaridade e organização do resultado e exibição para o usuário.

Entretanto, vale ressaltar que ainda há um conjunto de questões abertas neste contexto, como a seleção de descritores de imagens (extratores e funções de distância) adequados para cada contexto, tratamento da lacuna semântica e otimização de performance.

2.3 Extração de características e representação dos dados

A formalização de um sistema de banco de dados que realiza consultas por similaridade em dados complexos (características extraídas de imagem) requer a definição de descritores, estabelecidos através do par formado entre um algoritmo extrator de características de imagens e uma medida de similaridade, usualmente uma função capaz de determinar a distância entre duas imagens através do seu conjunto de características extraídas (TORRES; FALCAO, 2006; CAZZOLATO *et al.*, 2022). De forma a detalhar estes conceitos, são apresentados abaixo alguns algoritmos de extração de características e funções de distância.

2.3.1 Dados complexos

Dados complexos são definidos no contexto de sistemas CBIR como aqueles que precisam de uma função de avaliação de similaridade para serem comparados, então neste contexto são compostos por atributos que foram extraídos das imagens através de etapas de processamento, em contraparte a outros atributos que não são derivados, como nome ou data. Ou seja, um vetor de características extraído por uma das técnicas descritas a seguir é considerado um dado complexo. Ressaltando que esta taxonomia vale apenas para dados compostos por informações derivadas diretamente da imagem, excluindo informações adicionais como legendas e marcações (TRAINA *et al.*, 2017).

2.3.2 Extratores de características

Visando a representação de aspectos relevantes das imagens, que as descrevam numericamente para o contexto e objetivo das consultas que o usuário deseja, e a minimização das interferências de características irrelevantes, são usualmente aplicados extratores baseados em cores, forma e textura, podendo ser de abrangência geral (imagem inteira) ou local (seleção de trecho segmentado da imagem original, denominado de *região de interesse*). A seleção de um ou mais extratores é realizada a partir dos elementos característicos de cada conjunto de

imagens. Por exemplo, em certos domínios, como alguns exames médicos, a cor representada nas imagens é gerada de forma sintética, podendo não representar uma medida de similaridade entre as imagens, o que levaria a escolha de outro extrator, como um de forma local (SMEULDERS *et al.*, 2000; DATTA; LI; WANG, 2005).

Dentre os extratores de cor, podem ser utilizadas técnicas que atuam a partir de vários modelos de espaço de cor, como RGB (*Red, Green, Blue*) e HSV (*Hue, Saturation e Value*), descrevendo a imagem a partir da quantização de tonalidade e luminância em cada pixel, do posicionamento espacial das cores, e cores dominantes. Sua aplicação costuma ser geral (imagem toda), e uma das representações mais difundidas nesta categoria é o histograma de cor (ver Figura 3 para exemplo de histograma RGB), calculado pela contagem de pixels para cada valor de cor. O histograma é um extrator de baixo custo computacional que pode ser utilizado como um filtro inicial para separar as imagens.

A extração de características que consideram a forma e textura dos objetos da imagens é usualmente feita de forma local (apenas em uma região segmentada da imagem), sendo no caso da forma, extraídas representações geométricas do elemento, como informações de contorno, bordas e curvatura, aplicando-se filtros de Gabor ou derivação direcional (DATTA; LI; WANG, 2005; SMEULDERS *et al.*, 2000) por exemplo; e no caso da textura (HUMEAU-HEURTIER, 2019), recorrência de padrões, como as relações entre pixels calculadas através de matrizes de co-ocorrência pelo algoritmo de Haralick (HARALICK; SHANMUGAM *et al.*, 1973), para exemplificar alguns métodos tradicionais. Um *survey* sintético sobre extratores de imagens pode ser encontrado em (SALAU; JAIN, 2019)

2.3.3 Descritores de Imagens e Funções de Distância

Definidos como um par <Extrator de características (fx), Função de avaliação de similaridade (fa)>, os descritores (d) indicam as propriedades do domínio, compondo a base de um sistema de recuperação de imagens baseado em conteúdo. A função de avaliação de similaridade (fa) indica como comparar imagens cujas características foram extraídas por um extrator (fx). Dentro desta premissa, uma função de avaliação de similaridade pode ser definida através da inversa de uma função de distância calculada em um espaço métrico, ou seja, duas imagens apresentam maior similaridade se a distância entre seus vetores de características é menor.

Diversos tipos e famílias de funções podem ser aplicadas como funções de distância, sendo especialmente desejada a aplicação de métricas (definidas a partir de um espaço métrico e sujeitas às suas regras), visando aproveitamento das propriedades matemáticas na definição das consultas por similaridade e subsequente otimização (ZEZULA *et al.*, 2006). Por exemplo, as distâncias Euclidiana e Manhattan, comumente aplicadas, são parte da família Minkowski de métricas L_p definidas em vetores n -dimensionais, seguindo as propriedades de espaços métricos (não-negatividade, simetria, desigualdade triangular, identidade dos indiscerníveis e reflexividade).

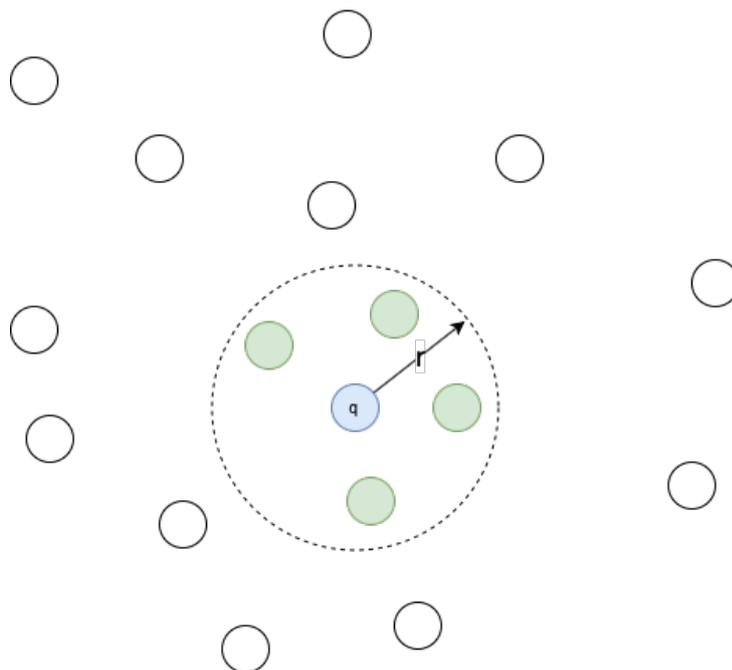


Figura 1 – Exemplo de consulta do tipo *range*, com raio r e centro de consulta q . Elaborado pela autora.

Vale ressaltar também o recente crescimento na aplicação de descritores mistos, que consideram anotações e exploram valores textuais, bem como a utilização de descritores baseados em técnicas de *deep learning* (MÜLLER; UNAY, 2017).

2.4 Consultas por similaridade

Geralmente dois tipos de consulta por similaridade são suportados em sistemas CBIR, as consultas do tipo *textitk-nn* (*K-Nearest Neighbors*, ou em português, *k* vizinhos mais próximos) e as consultas do tipo *range* (alcance, em português) (TORRES; FALCAO, 2006; SMEULDERS *et al.*, 2000; ZEZULA *et al.*, 2006; HAUPTMANN *et al.*, 2007; POBLETE *et al.*, 2010; ZHANG *et al.*, 2012; KITANOVSKI *et al.*, 2017). A figura 1 exemplifica uma consulta do tipo *range* (ou abrangência), que retorna a partir de uma imagem q e um raio r , quais imagens estão contidas no espaço de raio r tendo q como centro; e a figura 2 exemplifica uma consulta do tipo *k-nn*, que a partir de um número k e uma imagem q , retorna as k imagens mais próximas de q .

2.4.1 Técnicas de Avaliação

A avaliação de eficácia de consultas é calculada tradicionalmente por meio do par precisão e revocação, respectivamente quantas das imagens retornadas corresponderam ao esperado e quantas imagens correspondentes ao esperado foram retornadas a partir do possível total. Mais detalhadamente, precisão indica a proporção de resultados corretos retornados por consulta, ou sua acurácia; enquanto revocação indica a proporção de resultados corretos (verdadeiros

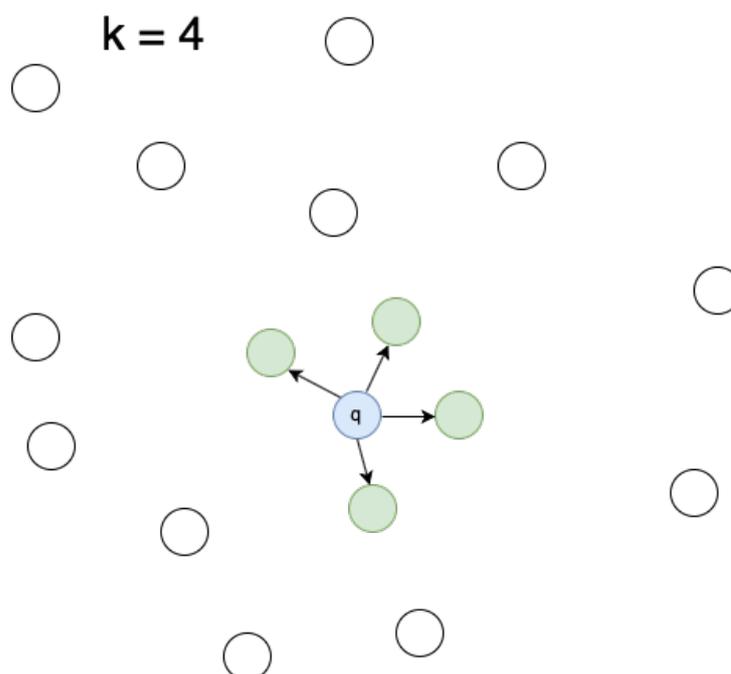


Figura 2 – Exemplo de consulta do tipo k -nn, para $k=4$ e centro de consulta q . Elaborado pela autora.

positivos) retornados pelas consultas a partir do total da base, ou a abrangência dos resultados corretos das consultas (TRAINA; MARQUES; TRAINA, 2006; POWERS, 2020).

2.4.2 Lacuna Semântica

O "gap semântico" ou lacuna semântica, é definido como o desencontro entre as informações extraídas de uma imagem, como representações numéricas de características como cor, formato e textura, e a interpretação que um usuário daria para o mesmo dado original, no caso uma imagem. Ou seja, certas representações para imagens não possibilitam uma definição de similaridade que corresponda às expectativas de um humano (SMEULDERS *et al.*, 2000). Como já colocado anteriormente, existe uma relação forte entre o extrator de características de imagens e a função de distância que irá avaliar as imagens, o que levou ao desenvolvimento de descritores de imagens. Esse relacionamento é muito relevante no processamento de consultas por similaridade e recuperação de imagens baseada em seu conteúdo (LI; YANG; MA, 2021).

Por exemplo, se compararmos as características de cor nos histogramas das duas fotos na figura 3, é simples perceber que não seriam classificadas como similares, porém a interpretação de uma pessoa poderia indicar que são imagens do mesmo assunto (gato na janela). Ou seja, a similaridade depende da motivação do que o usuário está querendo avaliar. A partir disso também é possível realçar a importância da escolha de descritores adequados e a consequente importância de táticas para avaliar a adequação das características extraídas para buscas por similaridade.

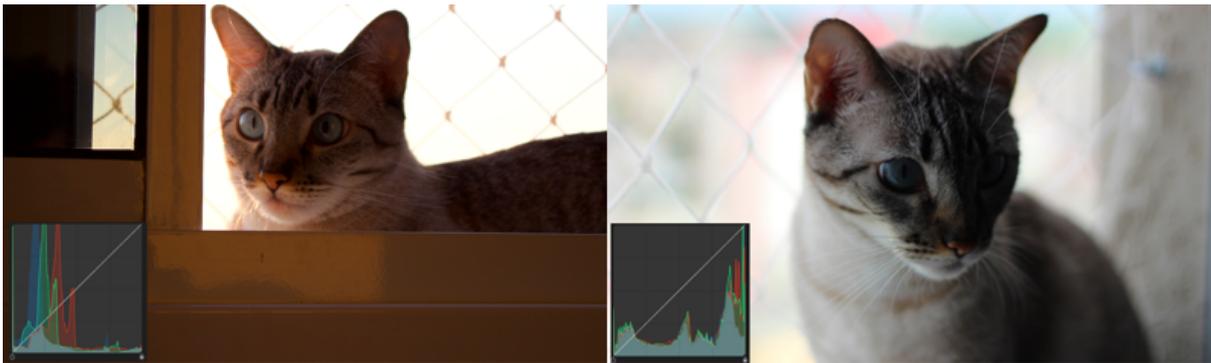


Figura 3 – Fotos e respectivos histogramas RGB

2.4.3 Realimentação por relevância

Visando alinhar as expectativas do especialista e os resultados obtidos pelas consultas, foram propostas técnicas de realimentação por relevância (RR) que permitem que o especialista insira suas considerações sobre os resultados, possibilitando adaptações na consulta de similaridade inicial, e gradual redução da lacuna semântica (AHMED, 2020).

A realimentação por relevância costuma introduzir o *feedback* do usuário especialista segundo sua classificação de relevância para cada um dos elementos do resultado de uma consulta. Desse modo, a partir da classificação apresentada pelo usuário, cada elemento recebe um peso, possibilitando o reajuste na relevância de cada imagem dentro do contexto estabelecido. Este processo é realizado em diversas iterações, sendo aplicado reposicionamento do centro da consulta de acordo com os pesos (positivos ou negativos) atribuídos (TRAINA; MARQUES; TRAINA, 2006; AZEVEDO-MARQUES; RANGAYAN, 2013; AZEVEDO-MARQUES *et al.*, 2008; BUGATTI *et al.*, 2011; LI; ALLINSON, 2013).

2.5 Considerações Finais

Neste capítulo foram abordados conceitos relativos a aplicações de recuperação de imagens baseada em conteúdo e as consultas por similaridade realizadas para tal. Dentro destes temas foram abordados também métodos de avaliação, desvantagens trazidas pela lacuna semântica e otimizações trazidas pela realimentação por relevância. É importante considerar que este trabalho de mestrado não desenvolveu extratores de característica, porém aplica distintos extratores para analisar a formação de bases de imagens que são utilizadas em processos de apoio à tomada de decisão por especialistas na área. Assim, os conceitos apresentados neste capítulo visam trazer essa visão geral ao leitor.

VISUALIZAÇÃO DE INFORMAÇÃO

3.1 Considerações Iniciais

Neste capítulo são apresentadas definições e processos de visualização e técnicas de visualização de informação que podem ser empregadas considerando o tipo de dado apresentado no capítulo anterior. Também é apresentada uma revisão das métricas de qualidade que podem ser aplicadas na avaliação de tais visualizações e algumas das tecnologias atuais que podem ser utilizadas no desenvolvimento do trabalho.

Finalizando, são listadas as técnicas que foram selecionadas no escopo deste trabalho, os critérios utilizados na seleção e uma análise de trabalhos que apresentam abordagens relacionadas ao uso de visualização de dados no refinamento de processos computacionais através da interação humana.

Deve-se ressaltar que alguns exemplos das técnicas de visualização apresentadas neste capítulo foram gerados a partir do dataset Iris ([DUA; GRAFF, 2017](#)), apenas com propósito ilustrativo.

3.2 Visão Geral

Presente em diversos aspectos da vida cotidiana, a visualização pode ser encontrada em gráficos e tabelas contidos em diversos meios, mapas rodoviários, representações tridimensionais do organismo geradas em exames de imagem, entre outros, de forma que suas utilizações permeiam diversas áreas do conhecimento, desde análises financeiras voltadas ao marketing de produtos e mercado acionário, estudos sociológicos e comparativos de populações, design e projeto de estruturas arquitetônicas, até aplicações médicas usadas no diagnóstico e tratamento de enfermidades, sendo eficaz na representação de informações e consequentemente como auxiliar em processos de tomada de decisões ([WARD; GRINSTEIN; KEIM, 2015](#)).



Figura 4 – Pipeline da visualização de informação utilizado neste trabalho. Baseado em proposta apresentada em (WARD; GRINSTEIN; KEIM, 2015)

Dada a abrangência do contexto, uma visualização pode ser definida como a expressão de uma informação (ou conjunto de informações) através de uma representação gráfica, que pode ser chamada simplesmente de imagem ou de projeção (dependendo de suas características). A utilização de visualizações permite a sumarização de informações através destas imagens, sendo interpretadas pelos mecanismos de percepção visuais que são parte dos sentidos básicos humanos, sendo um processo cognitivamente mais rápido do que a leitura (WARD; GRINSTEIN; KEIM, 2015; CARD, 1999).

A partir das últimas décadas a área de visualização passou a distinguir entre visualização científica, que engloba a representação de dados espaciais ou espaço-temporais, como por exemplo mapas de relevo, representações do organismo em exames médicos computadorizados e representações visuais de moléculas; e visualização de informação, que considera dados abstratos (que não são mapeáveis de forma física), como conjuntos de dados do mercado financeiro, textos e características extraídas de imagens (WARD; GRINSTEIN; KEIM, 2015). Especificamente a categoria de visualização de informação é foco deste trabalho, sendo detalhada a seguir.

Até que seja gerada uma imagem ou visualização a partir do conjunto de dados original, os dados passam por uma série de processos, que compõem o modelo chamado de *pipeline* da visualização de informação. Este *pipeline* (ver Figura 4) é composto por transformações nos dados, que passam por diferentes formatos, saindo dos dados originais (*raw data*, em vermelho) até o formato final apresentado numa visualização (em verde) (CARD, 1999; BERTINI; TATU; KEIM, 2011).

Como pode-se observar na Figura 4, os dados passam primeiro por etapas de processamento, que envolvem análise e pré-processamento (tratamento de valores ausentes, identificação de dados errados, entre outros), seguida de filtragem (ou amostragem e agregação) para reduzir a quantidade de dados, gerando dados em formato adequado para as etapas de processamento seguintes. Desta forma o processo continua com o mapeamento da informação para componentes geométricos, como pontos, linha e outras formas, além da posição e características de cor e textura (que irão compor a imagem visualizada) e renderização da visualização, ou seja exibição no dispositivo do usuário. Idealmente o usuário poderá interagir com as transformações nas diversas etapas do processo, tornando o resultado colaborativo (KEIM, 2002).

No que concerne ao processo de visualização, a escolha ou o desenvolvimento do método que será utilizado apresenta diversos fatores a ser considerados e usualmente inclui análises do

tipo de dado que será representado (formato, dimensionalidade e comportamento), da aplicação (exploração/mineração visual, comprovação de hipótese ou apresentação de resultados), e do tipo específico de achado que se busca extrair do conjunto. Algumas características que podem ser realçadas são a presença de (WARD; GRINSTEIN; KEIM, 2015; BEHRISCH *et al.*, 2018):

- **agrupamentos** (*clusters*), indicativos de comportamento similar dos pontos/elementos;
- **correlações**, ou relação entre valores médios das variáveis;
- **tendências**, ou características de comportamento que possam ser aplicadas em previsões;
- **outliers**, pontos do conjunto que apresentam comportamento muito diferente e não parecem fazer parte do conjunto, ou pontos que apresentem erro na coleta;
- **anomalias**, pontos que apresentam comportamento inconsistente com as expectativas.

Mesmo partindo destas características, a seleção de uma visualização efetiva não é trivial (WARD; GRINSTEIN; KEIM, 2015), sendo possível que vários fatores influenciem em sua efetividade, como a inadequação ao público alvo, falta de interatividade, falta de apelo estético e perda de informação no processo de mapeamento, com distorções e oclusões.

No que concerne a interatividade, é recomendado que a visualização ofereça um conjunto básico de características interativas, estabelecendo um padrão mínimo de tarefas possíveis para o usuário. O mantra da busca de informação visual "*Overview first, zoom and filter, then details-on-demand*" ("primeiro visão geral, zoom e filtros, então detalhes sob demanda", em tradução livre) apresentado em (SHNEIDERMAN, 2003) aparece neste contexto. Assim, o conjunto básico de operações de usuário que devem ser suportadas pela visualização inclui:

- visão geral: visualizar todo o conjunto;
- zoom: aproximar regiões de interesse;
- filtro: retirar/destacar elementos não interessantes ou de interesse;
- detalhes sob demanda: selecionar itens ou grupos e exibir detalhes quando necessário;
- relacionar: observar relações entre itens;
- histórico: manter histórico das ações, suportando opções de refazer/desfazer ação e refino;
- extração: permite extração de subgrupos e buscas com parâmetros.

No mesmo trabalho (SHNEIDERMAN, 2003) também foi apresentada uma taxonomia de tipos de dados em visualizações, que divide os tipos de dados em sete categorias: unidimensionais, bidimensionais, tridimensionais, multidimensionais, temporais, árvores e redes. Tal taxonomia

não é universal, havendo outras versões como a proposta por Keim (KEIM, 2002), que divide os tipos em seis categorias: unidimensionais, bidimensionais, multidimensionais, texto e hipertexto, hierarquia e grafos, e algoritmos e software. Em ambos os casos o tipo multidimensional é análogo, indicando consistência na classificação de visualizações para este caso. Na seção a seguir são abordadas técnicas de visualização de dados multidimensionais, sendo importante ressaltar que os termos multidimensional e alta dimensionalidade são empregados como sinônimos.

3.3 Visualização de dados de alta dimensionalidade

Considerando o contexto abordado neste trabalho, de vetores de características extraídos de imagens médicas, é possível afirmar que os dados de alta dimensionalidade tratados dizem respeito a ordem de dezenas ou centenas, visto o grande número de características extraídas de imagens por conjuntos de extratores de cor, forma e textura (indicados no capítulo anterior). Portanto serão abordadas técnicas capazes de oferecer visualizações diretas ou métodos de projeção.

Partindo inicialmente dos formatos primitivos básicos utilizados na renderização das visualizações, segue uma classificação de técnicas para dados multidimensionais, divididas nas seguintes categorias (WARD; GRINSTEIN; KEIM, 2015):

- Técnicas baseadas em pontos, que expressam cada dado n-dimensional em um espaço através de uma pequena marcação, incluindo gráficos de dispersão e matrizes de dispersão;
- Técnicas baseadas em linhas, que traçam conexões entre os pontos de cada dimensão do dado n-dimensional, como gráficos de linha, coordenadas paralelas, visualização radial em linhas;
- Técnicas baseadas em área, que representam os dados através de polígonos, preenchidos ou não, cujo tamanho, formato e posicionamento indicam seus valores, como gráficos de barras, histogramas, *heatmaps* e *Table Lens*;
- Técnicas híbridas, que combinam características das categorias anteriores, como *glyphs* e visualizações baseadas em pixels.

Cada uma das técnicas indicadas pode ser aplicada para tarefas de visualização de padrões indicativos de agrupamentos, *outliers*, correlações e tendências, sendo os padrões indicativos de cada achado tratados individualmente nas seções dedicadas a cada uma das técnicas abaixo, bem como indicadas as possibilidades de otimização mais comuns para cada método (BEHRISCH *et al.*, 2018).

Porém é necessário considerar os efeitos da *maldição da dimensionalidade*, uma expressão utilizada no contexto para descrever os efeitos de rápido crescimento (ordem exponencial)

da complexidade computacional e tempo de processamento, nos casos em que o conjunto de dados apresenta muitos pontos ou muitas dimensões para processamento, tornando a execução muito custosa ou mesmo proibitiva. Tal efeito leva à necessidade de adaptações e estratégias alternativas que permitam a execução de algoritmos sobre estes espaços minimizando seus efeitos. Por exemplo, estratégias que reduzem o número de pontos considerados (KUO; SLOAN, 2005; DECHEVA.; LINSEN., 2018). Tais limitações e adaptações serão individualmente elaboradas para cada técnica a seguir.

3.3.1 Técnicas de redução de dimensionalidade

Uma das primeiras etapas para a análise de conjuntos de dados multidimensionais é a redução de dimensionalidade, cujo objetivo é transformar o conjunto de dados em uma representação significativa em menor dimensionalidade. Tal transformação deve ter como objetivo que o conjunto transformado corresponda às dimensões intrínsecas dos dados, ou seja, o número mínimo de dimensões necessário para que sejam mantidas as propriedades dos dados. Aplicar técnicas de redução também é importante para evitar as consequências da maldição da dimensionalidade, e dentre estas técnicas existem diversas abordagens possíveis, variando entre determinísticas/não-determinísticas, lineares/não-lineares, convexas/não-convexas, com vantagens e desvantagens para cada contexto (MAATEN; POSTMA; HERIK, 2009; NONATO; AUPETIT, 2018; TANG *et al.*, 2016; REES; LARAMEE, 2019).

A figura 5 exemplifica o poder de representação de uma técnica de redução de dimensionalidade, no caso a t-SNE (uma das variantes da técnica SNE, *Stochastic Neighbor Embedding* que será sumarizada a seguir, através do mapeamento do conjunto sintético de pontos de 100 dimensões para uma representação bidimensional, que convergiu para a visualização dos três agrupamentos presentes nos dados (indicados pelas diferentes cores, para possibilitar a comparação) de forma visualmente explícita.

Algumas das técnicas de redução de dimensionalidade aplicadas em contextos relacionados a visualização e mineração visual (MAATEN; HINTON, 2008; MAATEN; POSTMA; HERIK, 2009; SUPIANO *et al.*, 2019; WANG *et al.*, 2021) são:

- Análise de componentes principais ou PCA (*Principal Component Analysis*), definida matematicamente como uma transformação linear ortogonal que realiza a redução de dimensionalidade incorporando os dados a um sistema de coordenadas de menor dimensão, é uma das mais consolidadas e utilizadas técnicas lineares (HOTELLING, 1933; MAATEN; POSTMA; HERIK, 2009);
- *Sammon Mapping*, que objetiva preservar a estrutura de distância entre pontos do espaço de maior dimensão ao escalar o conjunto em menor dimensão, é não-linear e pelas suas características adequa-se às tarefas de análise exploratória de dados; (SAMMON, 1969);

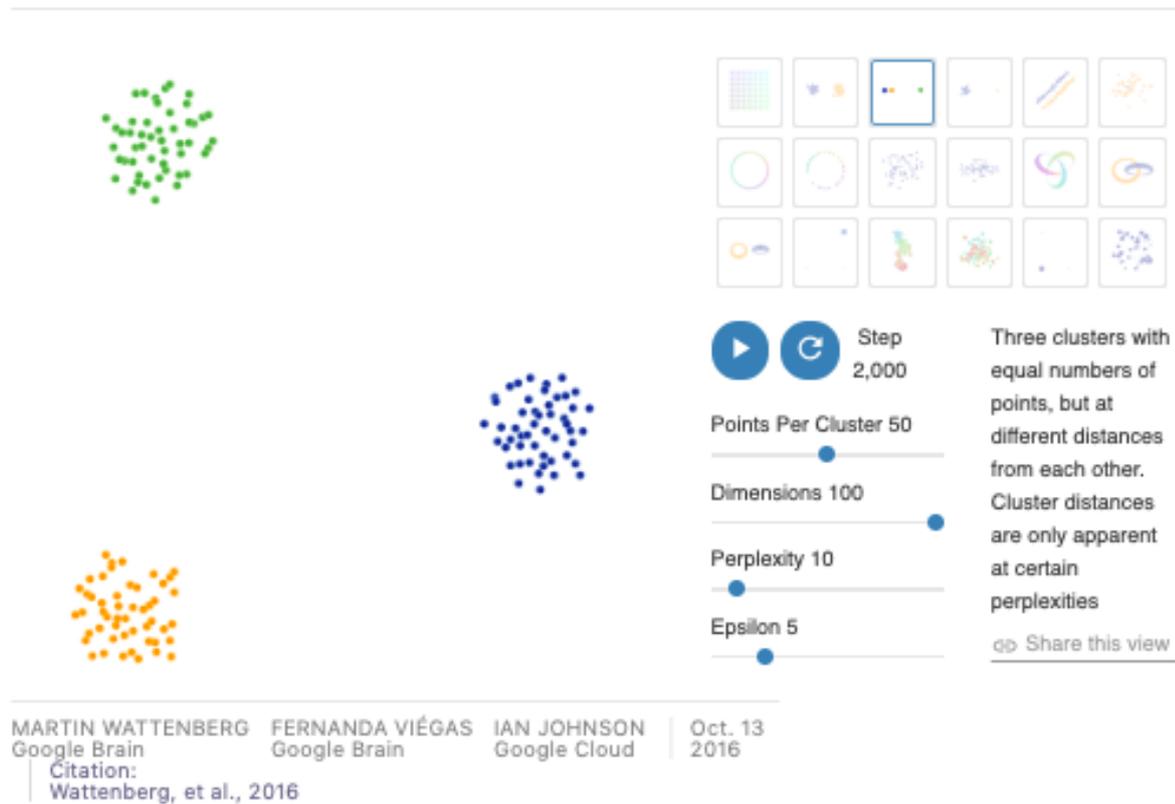


Figura 5 – Exemplo de visualização em gráfico de dispersão de conjunto de pontos multidimensionais sintéticos processados com a técnica t-SNE (MAATEN; HINTON, 2008). Imagem retirada de (WATTENBERG; VIÉGAS; JOHNSON, 2016)

- *t-SNE (t-distributed Stochastic Neighbor Embedding)* (figura 5, uma variante do SNE (apresentada em 2008 por Maaten (MAATEN; HINTON, 2008)) aplica a distribuição t-student calculando a posição dos pontos no espaço de dimensionalidade reduzida através da probabilidade de um ponto pertencer a mesma categoria/classe (estar próximo ao outro), iterando até uma convergência.
- *UMAP (Uniform Manifold Approximation and Projection)*, uma técnica de redução de dimensionalidade não-linear, matematicamente embasada na teoria de espaços topológicos, que modela e aproxima os pontos em menor dimensionalidade mantendo a estrutura que melhor se assimilaria a estrutura topológica original. Apresenta aplicações sugeridas e resultados similares a t-SNE (MCINNES; HEALY; MELVILLE, 2018).

Assim, uma abordagem seria aplicar reduções de dimensionalidade, como mencionado na seção anterior para os gráficos de dispersão, sendo que limitações similares se aplicam, como possível perda de representação de estruturas globais dos dados, sendo esta possibilidade potencialmente compensada na representação adequada de estruturas locais dos dados nas visualizações renderizadas.

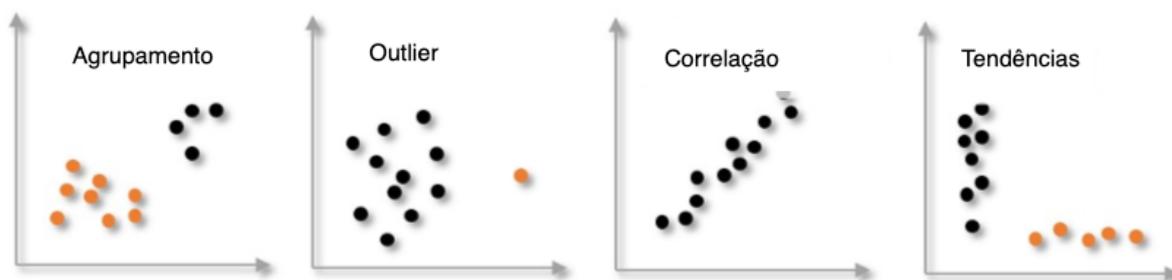


Figura 6 – Achados em gráficos de dispersão. Imagem adaptada de (BEHRISCH *et al.*, 2018)

Porém, é importante ressaltar que técnicas de redução de dimensionalidade também sofrem os efeitos da maldição da dimensionalidade quando o volume de dados tratados é muito grande (TATU *et al.*, 2012), e sua aplicação pode ocasionar distorções no cálculo de similaridade (dependente de métrica de distância), ordenações, distribuição geométrica e probabilidades, à depender do método aplicado para as reduções (NONATO; AUPETIT, 2018).

3.3.2 Gráficos de dispersão

Um dos métodos mais aplicados na visualização de informação, o gráfico de dispersão, em suas versões bidimensionais e tridimensionais representa os dados pelo posicionamento dos pontos considerando as coordenadas de um eixo, aproveitando a compreensão humana inata de julgar posições relativas no espaço. A otimização da representação gerada pode ser feita através de mudanças na escala dos eixos, através da seleção de intervalos, sendo também simples de aplicar. Porém, considerando os limites na representação de espaços com dimensionalidade superior a 3, são aplicadas estratégias de adaptação, como a seleção de dimensões, a redução de dimensionalidade e a incorporação de outros atributos gráficos como cores e formas (WARD; GRINSTEIN; KEIM, 2015).

Isto o torna útil em representações de espaços reduzidos pelas técnicas listadas acima, considerando que o número de dimensões seja inferior a 3. Ainda é possível observar que de forma intuitiva, se as propriedades dos dados forem mantidas, é uma ferramenta valiosa na visão geral do conjunto, pela facilidade de compreensão e capacidade de representação simplificada.

Na figura 6 são indicados os padrões formados na visualização para cada tipo de achado. Estes podem ser interpretados intuitivamente no caso desta visualização, visto que agrupamentos são diretamente mapeados para grupos de pontos, correlações são representadas de forma linear, *outliers* podem ser observados à margem do conjunto e tendências aparecem quando os pontos acompanham determinado sentido.

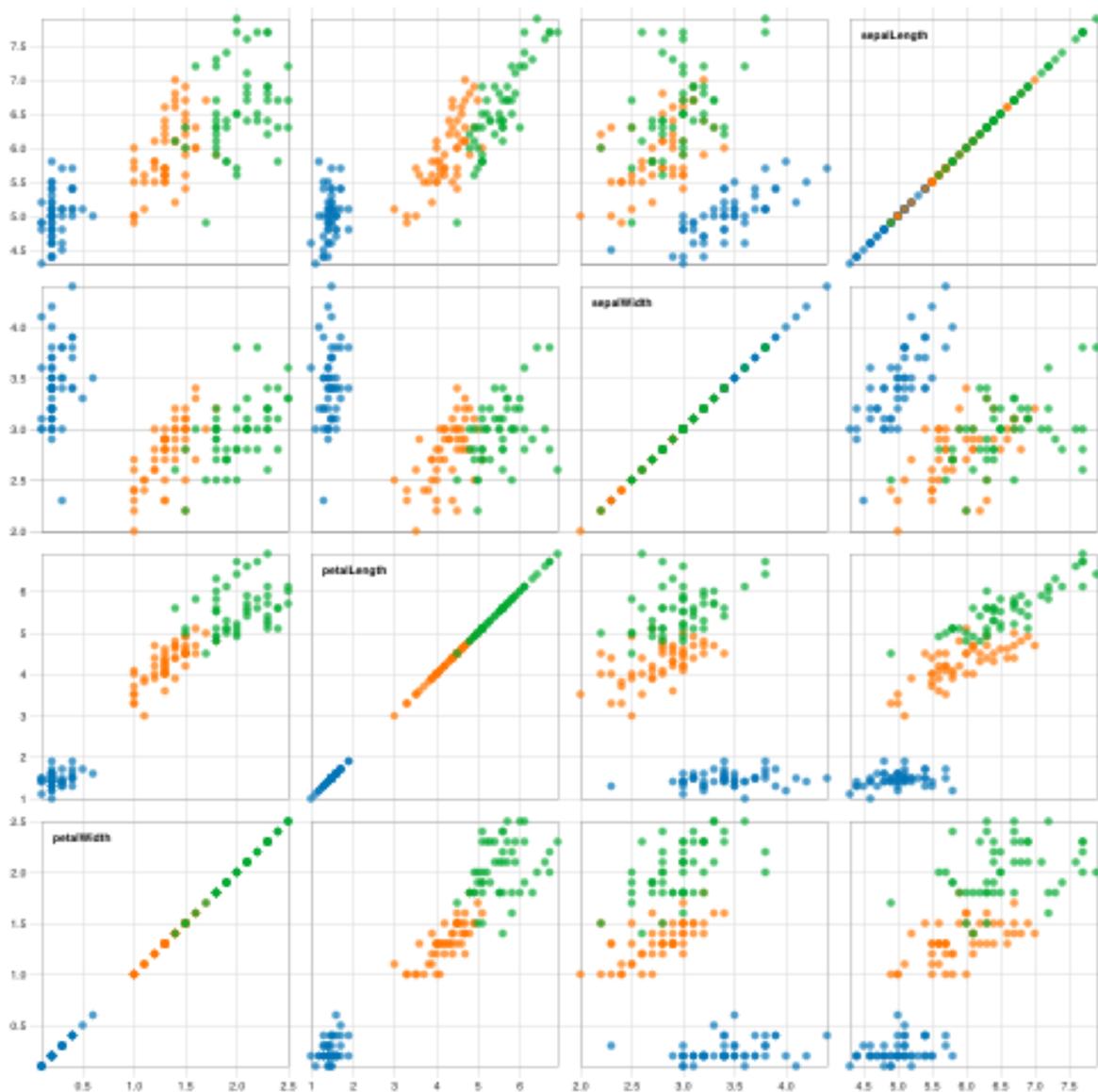


Figura 7 – Matriz de dispersão gerada a partir do dataset Iris. Autoria de Mike Bostock, disponível em <https://beta.observablehq.com/@mbostock/d3-scatterplot-matrix>

3.3.3 Matrizes de dispersão

Como extensão à representação limitada dos gráficos de dispersão simples, para que seja possível representar todas as dimensões de dados com mais de 3 dimensões de forma bidimensional, matrizes de dispersão são compostas por uma matriz quadrada de N dimensões, correspondentes às N dimensões dos dados, combinando cada dimensão par a par através de uma permutação, como é possível observar na figura 7, que apresenta uma matriz 4 por 4 (dimensões do dataset Iris) onde é possível observar as relações entre cada uma das dimensões. As matrizes de dispersão apresentam como tarefa principal de análise, a identificação de correlações entre cada duas variáveis de um espaço de alta dimensionalidade (WARD; GRINSTEIN; KEIM, 2015), identificadas através da estrutura indicada na figura 8.

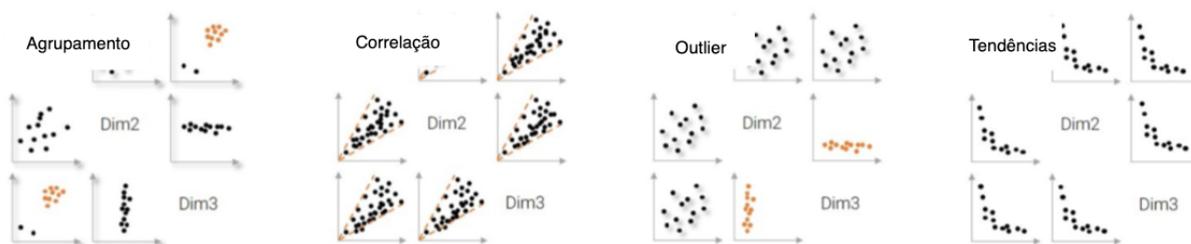


Figura 8 – Achados em matrizes de dispersão. Imagem adaptada de (BEHRISCH *et al.*, 2018)

Porém, a visualização apresenta acúmulos de informação de complexa interpretação se o número N de dimensões for muito alto, de forma que estratégias foram desenvolvidas para otimizar a aplicação desta técnica, sendo a mais comum o reordenamento das dimensões representadas, que podem destacar ou ocultar um padrão se forem apresentadas em diferentes ordens (ALBUQUERQUE *et al.*, 2010; BEHRISCH *et al.*, 2018).

3.3.4 Coordenadas paralelas

Apresentado na década de 80 por Inselberg como um método para visualização de geometria em R_n , suas aplicações foram estendidas para análise de outros tipos de dados no decorrer dos anos. A projeção consiste em n eixos paralelos equidistantes, sendo um para cada variável e em sua determinada escala, a partir dos quais uma linha é traçada por entrada, atravessando cada eixo paralelo no valor correspondente (WARD; GRINSTEIN; KEIM, 2015; INSELBERG; DIMSDALE, 1987).

No caso das coordenadas paralelas, a análise visual em busca de estruturas funciona para os mesmos achados mencionados anteriormente, indicados pela figura 10. Agrupamentos são percebidos através da concentração de linhas em determinados valores de cada eixo, e espaçamento com relação a outros grupos; correlação pode ser constatada em valores de determinado eixo que apresentam linhas que se cruzam (alterando-se os sentidos); enquanto *outliers* e tendências apresentam comportamentos análogos às representações anteriores. Na figura 9 cada uma das quatro dimensões do dataset Iris são representadas nos eixos verticais

As técnicas aplicadas para melhorar as projeções geradas com esta técnica, principalmente para representar grandes conjuntos de dados com muitas dimensões, envolvem a reordenação de eixos e o agrupamento de linhas, visto que sua ordem é crucial na identificação de padrões e o agrupamento auxilia na remoção de *cluttering* e oclusões (JOHANSSON; FORSELL, 2016).

3.3.5 Visualizações baseadas em pixels

Diretamente relacionadas à representação de grandes bases de dados, as visualizações baseadas em pixels representam cada ponto de dados como um pixel colorido na tela, possibilitando a visualização simultânea de grandes conjuntos. Porém indicam apenas um atributo

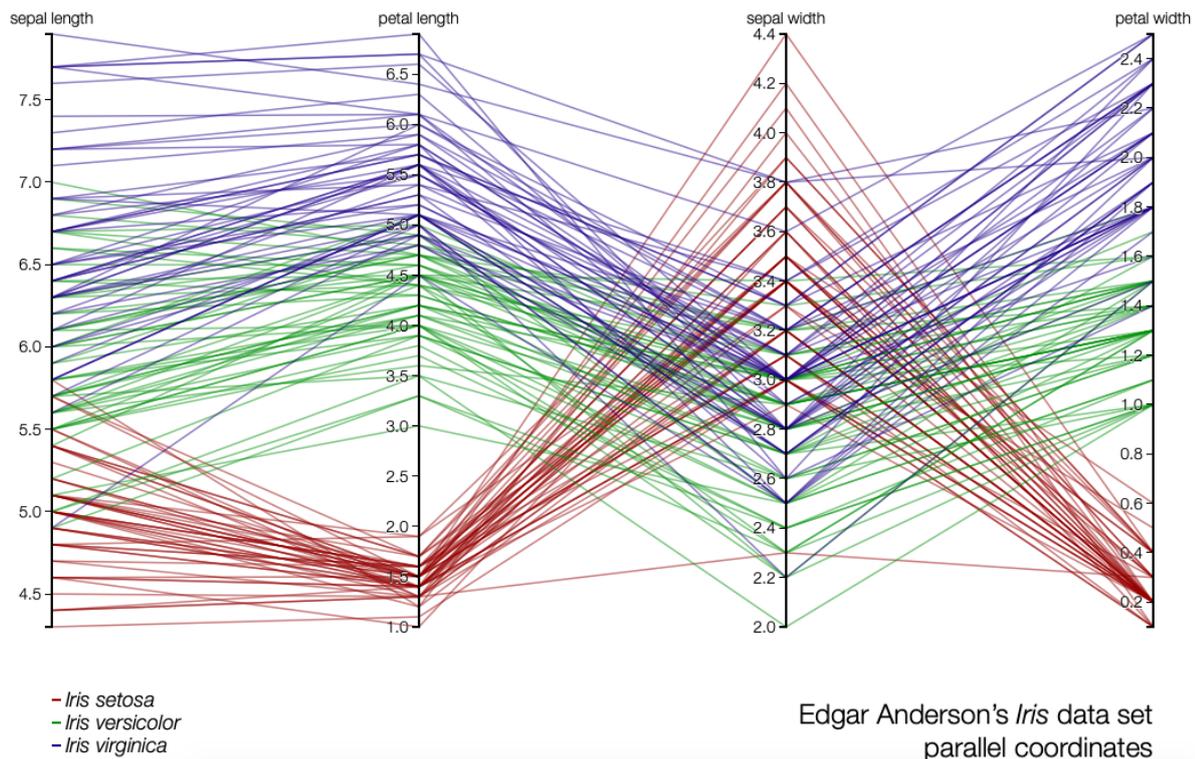


Figura 9 – Coordenadas Paralelas geradas a partir do dataset Iris. Autoria de Mike Bostock, disponível em <<http://mbostock.github.io/d3/talk/20111116/iris-parallel.html>>

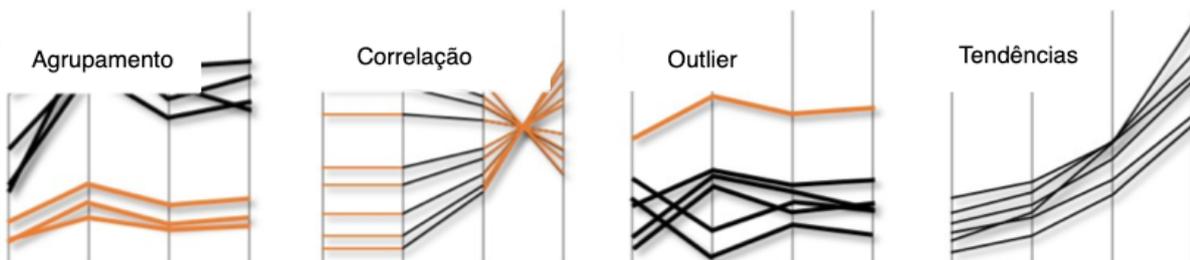


Figura 10 – Achados em coordenadas paralelas. Imagem adaptada de (BEHRISCH *et al.*, 2018)

por pixel (uma dimensão), de forma que múltiplas janelas nas quais os dados seguem a mesma ordem/disposição, são necessárias para representar conjuntos de alta dimensionalidade (KEIM, 1996; KEIM, 2000; ANKERST, 2001).

Na figura 11 é possível observar o comportamento deste tipo de visualização com relação aos padrões buscados por tarefas de análise. Como cada pixel representado corresponde a um ponto do conjunto de dados correspondente àquela variável (cada janela individual) a ordem dos pontos é de extrema importância, podendo ser aplicado um padrão de inserção como curvas de Hilbert ou espirais, bem como realizar a inserção de forma linear. Outros aspectos que podem ser analisados para otimização destas visualizações são mapeamento de cores e formato das janelas.

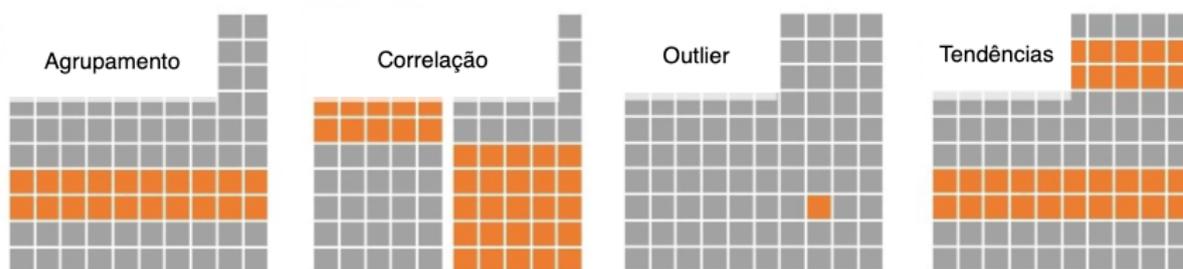


Figura 11 – Achados em visualizações baseadas em pixels. Adaptada de (BEHRISCH *et al.*, 2018)

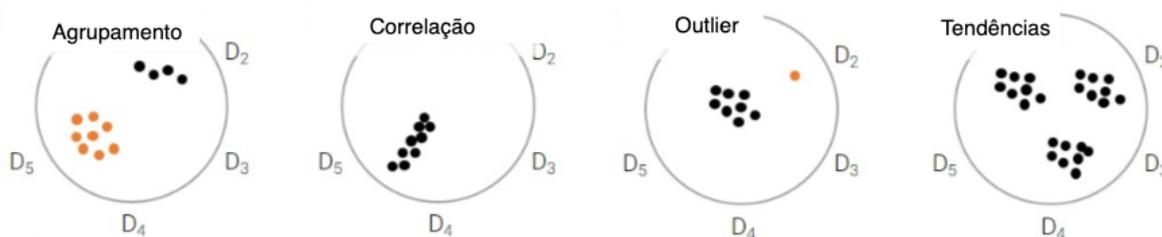


Figura 12 – Achados em visualizações radiais. Imagem adaptada (BEHRISCH *et al.*, 2018)

3.3.6 Visualização radial

A visualização radial ou Radviz representa cada dimensão como uma posição em uma circunferência, assim o ponto que representa uma entrada do conjunto de dados é posicionado dentro do círculo, e sua posição em relação à circunferência é determinada pela aplicação do valor de cada dimensão (posições na circunferência) como se fosse a força exercida por uma mola, sendo a posição final a localização de soma zero das forças. Esta técnica se destaca por apresentar todas as dimensões simultaneamente para cada ponto (ALBUQUERQUE *et al.*, 2010).

3.3.7 Glyphs

Um *glyph* no contexto de visualização é uma representação visual de um ponto da informação que pode ser posicionada de forma independente por já conter todas as informações relativas a seus atributos, ou seja, como é possível observar na figura 13, que exemplifica o modelo de *glyph* comumente utilizado para dados multidimensionais, que mapeia cada dimensão do ponto como uma distância a partir do centro. Cada forma individual é um *glyph* e estes podem ser distribuídos no espaço independentemente. (BORGO *et al.*, 2013; CHUNG *et al.*, 2015)

3.4 Qualidade em visualizações de dados

Para selecionar e adaptar um método de visualização, de forma que ele se adeque aos dados e objetivos de análise desejados, é imprescindível determinar o conjunto de tarefas e qual será o usuário. A partir desta, a aplicação de métricas de qualidade na seleção de técnicas

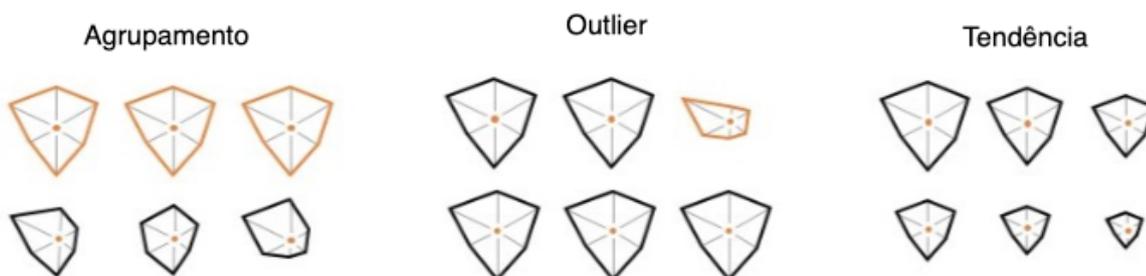


Figura 13 – Otimização e achados em glyphs. Imagem adaptada de (BEHRISCH *et al.*, 2018)

de visualização possibilita escolhas mais eficientes e adequadas de método para os dados em questão, bem como a adaptação da técnica escolhida para as tarefas-objetivo. Por exemplo, ao analisar a relação entre duas variáveis, um gráfico de dispersão é suficiente para indicar correlação (adequação à tarefa-objetivo) e melhoras na análise podem ser obtidas através de mudanças na escala de representação de cada variável, indicando a importância da aplicação de métricas para avaliar se as adaptações ofereceram melhorias ao resultado final (BEHRISCH *et al.*, 2018).

As métricas de qualidade podem atuar em diferentes níveis do processo cognitivo, desde a percepção do usuário e experiência de uso (alto nível) até percepção de baixo nível, que envolve leis da Gestalt e processos pré-atentivos. O enfoque dos estudos realizados neste trabalho está no nível intermediário do processo cognitivo, onde as considerações de qualidade são aplicadas segundo características de percepção relacionadas às tarefas realizadas sobre a visualização. Este nível visa medir efetivamente a presença de oclusão (anti-padrão) e quais padrões visuais são aparentes (ELLIS; DIX, 2007; ALBUQUERQUE *et al.*, 2010; BEHRISCH *et al.*, 2014; BEHRISCH *et al.*, 2018).

3.4.1 Abordagens Quantitativas

Aspectos numéricos podem ser analisados, oferecendo métricas quantitativas na avaliação de uma visualização. Por exemplo, medir a taxa de compressão que a técnica de visualização aplicada representa no alfabeto inicial de dados, ou seja, se o mapeamento dos pontos originais de dados por características visuais em uma projeção representa uma diminuição no espaço de análise. Considerando a definição de um pipeline de transformações dos dados, partindo do formato original, incluindo fases de pré-processamento, visualização gerada e por fim as características encontradas, é possível associar processos qualitativos como a análise do usuário, à função de custo-benefício (comparando, por exemplo um parâmetro de tempo de processamento dos dados pra renderização da visualização e o tempo requerido pelo usuário para realizar a análise do mesmo conjunto sem a visualização) (CHEN; GOLAN, 2016; MICALLEF *et al.*, 2017). Também é possível aplicar técnicas diretamente relacionadas à análise da capacidade do córtex de visão humana de perceber elementos visuais (MATZEN *et al.*, 2018).

Partindo de uma abordagem intermediária na análise de qualidade de visualizações, com enfoque em técnicas associadas à análise de padrões e redução de ruído e oclusão, as métricas que podem ser aplicadas na comparação de visualizações costumam ser embasadas em conceitos similares ao *data-to-ink ratio*, usado para medir a quantia de informação expressa com relação a área da visualização gerada (densidade de dados), ou nas porcentagens de oclusão e pontos identificáveis (BRATH, 1997; BRATH; BANISSI, 2015; PANDEY *et al.*, 2016; BEHRISCH *et al.*, 2018).

3.4.2 Abordagens Qualitativas

Também podem ser aplicadas diversas técnicas qualitativas de avaliação, que são dependentes da presença e análise de um usuário sobre a visualização gerada. Características relacionadas à interação com o usuário podem englobar aspectos do design da visualização, como a aplicação ou não de *guidelines* de cor e posicionamento (mais diretamente, se a imagem apresentada é representativa), e adequação à tarefa, sendo analisado se o método foi eficiente conforme as necessidades de análise do usuário (REDA *et al.*, 2015; BORKIN *et al.*, 2016).

3.5 Aplicações relacionadas

Na revisão realizada na literatura, foram encontrados trabalhos que aplicam técnicas de visualização de informação como auxiliares na compreensão de similaridade em buscas ((RODRIGUES; NUNES, 2014)) e na análise de espaços compostos por vetores de características, inspirando a proposta de uma aplicação que adote visualizações como meio de explorar a eficácia de um determinado algoritmo de extração de características, ou escolha de subconjunto destas características na representação de imagens, em um contexto de buscas por similaridade.

Por exemplo, é possível aplicar técnicas de redução de dimensionalidade e gráficos de dispersão na mineração visual de informações não classificadas, sendo um tipo de aplicação crescente para a exploração de conjuntos de dados em processos de aprendizado de máquina que visam feedback humano em suas performances (DECHEVA.; LINSEN., 2018).

Em trabalho de 2007 de Blaas *et al.* e colegas (BLAAS; BOTHA; POST, 2007), são aplicadas técnicas de visualização interativas em bases de dados de características extraídas de imagens médicas. Os autores realçam as necessidades de estudos em visualização exploratória de dados neste contexto, atribuindo tom de urgência. Contudo este trabalho não realiza comparações entre técnicas distintas de visualização, tendo com objetivo maior elencar vantagens para a abordagem visual desenvolvida. É possível assim inferir uma abertura para a análise de outras técnicas viáveis para tais conjuntos de dados.

Molchanov (MOLCHANOV; CHITIBOI; LINSEN, 2015) aplica visualização de espaço de características como parte do processo de refinamento de uma técnica supervisionada de classificação, indicando viabilidade na aplicação de análise visual para seleção de características

relevantes, no contexto de aprendizado. Projeções aparecem também como instrumentos de *insight* no espaço de características, sendo aplicadas na predição de eficácia em sistemas de classificação (RAUBER; FALCÃO; TELEA, 2017).

Desta forma, foi observado que visualizações de dados vêm também sendo aplicadas para espaços de características em outros contextos próximos; como parte do processo de refinamento de técnicas supervisionadas e seleção de características relevantes para segmentação de imagens (MOLCHANOV; CHITIBOI; LINSEN, 2015); atrelando métodos automatizados de análise de dados como àqueles de aprendizado de máquina associados ao *feedback* humano (SACHA *et al.*, 2017), bem como abordagens que atrelam etapas automatizadas e interação de usuário (TATU *et al.*, 2012).

3.6 Tecnologias em Visualização

As tecnologias disponíveis atualmente para desenvolver e exibir visualizações podem ser divididas nas categorias: sem linguagem de programação e baseadas em linguagens de programação (LIU *et al.*, 2018). No caso de aplicações fechadas que não envolvem programação diretamente, fácil utilização e modelos pré-definidos se apresentam como vantagens, enquanto aquelas que são baseadas em linguagens de programação podem requerer maior flexibilidade para trabalhar diretamente com os dados, por meio da manipulação do código, tornando o desenvolvimento de visualizações novas e modificações mais acessível, que são características valiosas no contexto acadêmico. Dentre estas, do último tipo, são mais difundidas as tecnologias D3.js ¹, Chart.js ², Fusion Charts ³, Flot Charts ⁴, Zing Chart ⁵, Gephi ⁶, Nodebox ⁷, GGPlot2 ⁸, Processing ⁹ e JGraph ¹⁰, tendo como base diversas linguagens como JavaScript ¹¹, Python ¹² e Java ¹³, com destaque às bibliotecas de código aberto e uso gratuito.

3.7 Considerações Finais

Neste capítulo foi realizada uma revisão das principais fundamentações da literatura para área de visualização de informação, com foco em dados multidimensionais, sendo observado que

¹ <<https://d3js.org>>

² <<https://www.chartjs.org>>

³ <<https://www.fusioncharts.com>>

⁴ <<https://www.flotcharts.org>>

⁵ <<https://www.zingchart.com>>

⁶ <<https://gephi.org>>

⁷ <<https://www.nodebox.net>>

⁸ <<https://ggplot2.tidyverse.org>>

⁹ <<https://processing.org>>

¹⁰ <<https://jgraph.net>>

¹¹ <<https://www.javascript.com>>

¹² <<https://www.python.org>>

¹³ <<https://go.java/index.html>>

múltiplas técnicas são vigentes no contexto, bem como são existentes técnicas de otimização e avaliação de qualidade para cada uma delas. Ao considerar a proposta de representar visualmente espaços de características, a revisão realizada realçou a vigência de técnicas de redução de dimensionalidade como métodos para criação de visualizações capazes de transcrever estruturas dos dados que possibilitem o tipo de análise desejado neste trabalho. As escolhas relacionadas à técnicas específicas, tecnologias e bibliotecas são detalhadas no Capítulo 4 a seguir.

DESENVOLVIMENTO

4.1 Considerações iniciais

Neste capítulo são apresentados o desenvolvimento e as decisões de implementação para a solução idealizada para visualização de conjuntos de características extraídas de imagens, bem como a medida desenvolvida para análise de oclusão em tais visualizações interativas. O desenvolvimento e diretrizes de projeto escolhidos são apresentados de forma detalhada, sendo seguidos dos testes e análises realizados.

4.2 Proposta

A proposta deste projeto de mestrado foi analisar a viabilidade da aplicação de técnicas de visualização como parte do processo de avaliação e aprimoramento de extratores de características para o processamento de consultas por similaridade de imagens médicas. As técnicas de visualização foram aplicadas ao espaço de características, verificando sua viabilidade para antecipar a eficácia dos extratores escolhidos para a consulta designada.

Tal aplicação pode ser integrada ao processo de realização de consultas por similaridade, com o objetivo de apresentar um panorama geral dos extratores escolhidos, sendo definida como auxiliar os testes realizados em técnicas de recuperação de imagens baseada em conteúdo; desta forma possibilitaria a otimização de resultados sem requerer a atuação de um especialista médico, considerando testes realizados em uma base previamente anotada. Ou seja, uma primeira análise do comportamento dos extratores com referência aos padrões presentes nas imagens, que devem ser mapeados pelos extratores, pode ser utilizada como um ponto de partida na calibragem dos extratores. Tal iniciativa visou diminuir a carga de processamento a posteriori que é usualmente realizado em análises baseadas em médias de consultas realizadas e curvas de precisão versus revocação.

Assim, a abordagem planejada foi voltada às tarefas de mineração visual de imagens. Estão incluídas a busca de padrões e tendências que indiquem relações entre as características extraídas do conjunto de imagens, além das medidas de similaridade propostas, ou seja, determinar se tais características e medidas geram resultados semanticamente desejáveis.

Neste contexto, o objetivo foi comparar o desempenho de diversas técnicas de visualização, otimizando-as para redução de oclusão de objetos, legibilidade e execução das tarefas de mineração visual indicadas a seguir:

1. Identificar agrupamentos a partir de projeções, preferencialmente contendo todas as dimensões ou atributos dos dados (panorama),
2. Identificar agrupamentos a partir da análise de conjuntos de dimensões, e
3. Identificar correlações entre diferentes dimensões.
4. Identificar relações entre a distribuição de pontos e a semântica visual das imagens correspondentes.

Derivada da necessidade de avaliar o desempenho e qualidade das visualizações propostas, também foi apresentada uma medida visual automatizada para análise das mesmas, uma medida de taxa de oclusão.

4.3 Metodologia

Partindo da exploração da hipótese proposta

A análise visual de vetores de características gerados por extratores distintos em imagens médicas possibilita antecipar quão apropriados são tais extratores considerando o contexto em análise para apoiar o processamento de consultas por similaridade.,

e da revisão da literatura, bem como o contexto de aplicação e as tecnologias disponíveis para implementação da proposta, as seguintes diretrizes de projeto foram adotadas com o objetivo de elucidar a hipótese:

1. **Familiaridade:** reconhecimento e interpretação que não requeressem especialização no tópico, ou seja, seleção de uma técnica de visualização de informação consolidada que não necessitasse de aprendizado prévio aprofundado para interpretação e reconhecimento de padrões em sua análise exploratória.
2. **Interatividade:** Possibilidade de interação, a partir da seleção de uma biblioteca para implementação da técnica que possibilitasse a interação direta do usuário no resultado visual gerado, oferecendo opções de seleção visual e ajustes.

	Interatividade	Documentação	Simplicidade	Comparabilidade	Integração	Distribuição	Livre
D3.js							
Plotly							
Bokeh							

Tabela 1 – Tecnologias testadas e conformidade com requisitos propostos

3. **Documentação:** Utilização de tecnologias consolidadas na implementação, visando minimizar erros e aproveitar literatura disponível.
4. **Simplicidade:** Implementação clara, apoiada na seleção de uma biblioteca que permitisse a implementação simplificada com eficaz custo temporal das funcionalidades desejadas mantendo a flexibilidade de alteração necessária para requisitos específicos do projeto.
5. **Comparabilidade:** Possibilidade de comparação de técnicas, com uma implementação que permitisse a comparação interativa direta e visual entre técnicas distintas aplicadas.
6. **Integração:** Integração de processos e funcionalidades, com a seleção de bibliotecas que permitissem a criação de uma sequência encadeada de processamento dos dados que incluísse a extração de características, redução de dimensionalidade e renderização das visualizações interativas.
7. **Distribuição:** Praticidade na distribuição do resultado final, auxiliando trabalhos futuros.
8. **Livre:** Utilização de software livre e gratuito ou ao menos aberto para comunidade científica.

Desta forma, o requisito 1 é específico da escolha da técnica de visualização, enquanto os demais (de 2 a 8) são mais voltados à decisões tomadas na implementação.

4.3.1 *Seleção de técnicas de visualização e redução de dimensionalidade*

Conforme as diretrizes escolhidas, foram conduzidos testes em tecnologias visando selecionar a alternativa que melhor se adequasse aos requisitos propostos (de 2 a 8), desta forma a partir das especificidades da tecnologia escolhida, a melhor opção de visualização foi selecionada conforme o requisito 1. Desta forma foram testados 3 conjuntos de tecnologias:

- biblioteca D3.js em conjunto com a plataforma Observable
- biblioteca Plotly (Python) em conjunto com Jupyter Notebook
- biblioteca Bokeh (Python)

Nesta etapa de seleção de tecnologias, a primeira tecnologia considerada foi a biblioteca D3.js, disponível desde 2011 (BOSTOCK; OGIEVETSKY; HEER, 2011), consolidada na área em conformidade com o requisito 3, a biblioteca foi testada em conjunto com a plataforma Observable¹ (plataforma de edição colaborativa para código em JavaScript voltada para visualização de dados), especialmente pelo requisito 7, buscado na utilização de uma plataforma colaborativa e fácil de distribuir como Observable. A partir de buscas na D3 Gallery² por visualizações que pudessem ser utilizadas como ponto de partida no desenvolvimento proposto, ou seja, que já correspondessem de alguma forma aos requisitos 2, 5 e 6, foi constatado que as alternativas encontradas iriam requerer mais tempo do que o disponível no projeto para uma implementação inteiramente nova (inviabilizando o requisito 4), visto que não foram encontradas implementações que pudessem ser objetivamente adaptadas. Foi observado que esta alternativa também estaria de acordo com o requisito 8.

O conjunto de tecnologias testado na sequência foi o combo Plotly (em sua versão express) e Jupyter Notebook, escolhido inicialmente por solucionar o requisito 6, já que outras bibliotecas também em Python poderiam ser aplicadas na manipulação dos dados de teste, mantendo a integração. Porém nos testes realizados se constatou que a biblioteca Plotly não apresentava as características desejadas de interatividade (requisitos 2 e 5) nas visualizações multidimensionais disponíveis (matriz de dispersão e coordenadas paralelas), instigando a busca por uma biblioteca alternativa em Python, que fosse capaz de solucionar estes requisitos restantes.

A solução encontrada foi a biblioteca Bokeh³, uma biblioteca de visualização interativa para navegadores web modernos com inspiração conceitual na ideia de abstrações rígidas, bem definidas e combináveis que podem ser vinculadas diretamente aos dados (WILKINSON, 2013) proposta por Wilkinson. A funcionalidade de vínculo de dados pode resolver o requisito de **Comparabilidade**, permitindo que variações do mesmo conjunto de dados fossem exibidas simultaneamente na mesma visualização.

As tecnologias foram inicialmente testadas com conjuntos simples de dados, especificamente iris e mnist, por serem conjuntos disponíveis como exemplo facilmente usado em outras bibliotecas da linguagem Python. Rapidamente foi possível perceber que mesmo com estes conjuntos pequenos de dados as visualizações multidimensionais testadas (matriz de dispersão e coordenadas paralelas) aparentaram estar sobrecarregadas com o volume de dimensões exibidas.

A partir destas observações a decisão seguinte foi adotar métodos de redução de dimensionalidade, que possibilitassem a aplicação de gráficos de dispersão, considerando que dado o número de características médias apresentadas na tela, os padrões tornavam-se difíceis de identificar devido à alta taxa de oclusão. Esta decisão também alinou-se ao escopo proposto do projeto, visto que o desenvolvimento de uma técnica inovadora do zero consumiria mais

¹ <<https://observablehq.com>>

² <<https://observablehq.com/@d3/gallery>>

³ <<https://bokeh.org/>>

tempo do que o proposto. Além disso foram observadas tendências em pesquisas relacionadas que indicariam reduções de dimensionalidade como caminhos promissores de solução.

A partir da decisão de aplicar técnicas de redução de dimensionalidade, a primeira técnica considerada foi o t-SNE, visto a proposta de preservação de estruturas locais após a redução, proposta esta que se alinha com a análise desejada, ou seja, tornaria possível determinar se os conjuntos de características extraídas de imagens mantêm alguma relação com a distribuição das imagens e também se a extração realizada seria adequada. Para proporcionar um parâmetro de comparação, também foi aplicada a técnica UMAP nos conjuntos de dados analisados, visto que é uma técnica de redução de proposta de aplicação similar ao t-SNE que também possui uma implementação na biblioteca escolhida em python, a scikit-learn.

4.4 Bases de dados

Após a seleção das técnicas de redução e métodos de visualização, os desenvolvimentos seguintes foram relativos a escolha de bases de dados de teste que correspondessem ou ao menos se assemelhassem com o contexto proposto de bases médicas. Desta forma as buscas foram focadas em bases de oriundas de contexto médico que apresentassem características extraídas previamente, visto que o desenvolvimento e aprimoramento (de extratores) não é foco deste trabalho. Na falta de bases obtidas conforme estes critérios, foram selecionadas aquelas que melhor representassem ideias gerais de grande variabilidade de imagens, que pudessem ter características extraídas sem grande aprimoramento nas especificações das técnicas. As escolhas visaram apresentar bases distintas, que pudessem exemplificar melhor a aplicação das técnicas de visualização como método de comparação entre técnicas de extração de características. Também foi feito o desenvolvimento de uma base sintética de imagens parametrizável, que proporciona um conjunto conhecido de dados, para minimizar vieses nas análises iniciais.

4.4.1 Critérios de seleção e escolha de bases

No processo inicial de escolha das bases de teste os seguintes critérios foram aplicados:

- Tipo: base de dados de imagens variadas.
- Temática: preferencialmente médica, de exames de imagem.
- Marcação e anotação: presença de classes definidas e marcadas, visando possibilitar comparações de clusterização.
- Características extraídas: disponibilidade de características previamente extraídas conforme métodos desejados de cor, forma e textura.
- Tamanho adequado: que não exigisse capacidade excessiva de processamento, ou seja, pudesse ser processada localmente em uma máquina pessoal.

Na busca realizada em repositórios de bases disponíveis online, foi constatado que o foco primário das bases de imagens disponíveis é possibilitar tarefas nos âmbitos de classificação, segmentação, reconhecimento e visão computacional sendo não usual a disponibilização de conjuntos de imagens com características de cor, forma ou textura previamente extraídas. Desta forma foi decidido que uma etapa de extração de características deveria ser incluída no projeto.

Repositórios online utilizados na busca no ano de 2020:

- UCI Machine Learning Repository ⁴
- Kaggle⁵

Assim, duas bases de imagens foram selecionadas de repositórios virtuais, sendo estas fora do contexto médico com o propósito de incluir diversidade na análise. Foi incluso também um conjunto de dados simples, com a proposta de realizar testes iniciais na visualização, que não incluíssem extração de características.

A partir destas observações, uma busca foi realizada também no repositório de bases do grupo de pesquisas, sendo escolhida uma destas para teste, que se enquadra na categoria médica.

Considerando o contexto de pandemia mundial, também foi analisada a possibilidade de incluir alguma base de exames de imagens da COVID19, porém no momento que foram realizadas as buscas, apesar de grande volume de tópicos relacionados, não havia informação consolidada disponível e as bases encontradas não apresentavam marcação consistente de classes e atributos de diagnóstico, bem como escopo de propostas definido, fator que traria uma incerteza indesejada no desenvolvimento e aplicação das técnicas propostas neste projeto.

Visto que seria necessária uma etapa de extração de característica de todas as bases selecionadas, também foi desenvolvida uma base sintética parametrizável de imagens, com opções de seleção de número de imagens e estilo de imagem, a selecionar entre cor única ou gradiente de cor. A partir da entrada direta das imagens para cálculo de histograma de cor, a proposta desta base foi oferecer um teste sem os vieses de dados do mundo real para os testes iniciais com imagens.

4.4.2 Bases Escolhidas

Foram selecionadas cinco bases de dados, detalhadas a seguir:

1. *Iris Data Set* (DUA; GRAFF, 2017): foi selecionado como primeiro conjunto de dados para simplificar a realização dos testes iniciais de interface e da implementação, incluindo integração do tratamento dos dados e geração da visualização, visto o rápido processamento

⁴ <<https://archive.ics.uci.edu/ml/index.php>>

⁵ <<https://www.kaggle.com/>>

	Tipo	Tamanho	Instâncias	Classes	Descrição
iris	multivariado	5Kb	150	3	atributos numéricos de flores
COCO(val2017)	imagens	776Mb	5000	N/A	fotos diversas
SegMRF	imagens	21.9Mb	714	9	cortes de ressonância magnética
Synthetic	imagens	variável	variável	N/A	gradientes de cor
Swedish Leaves	imagens	6.38Gb	1125	15	folhas

Tabela 2 – Conjuntos de dados selecionados

de suas 150 instâncias. O conjunto de dados Iris não possui imagens e sim atributos referentes ao comprimento e largura de pétalas e sépalas (4 atributos) de três tipos de flores (3 classes), Iris Setosa, Iris Versicolour e Iris Virginica. Visto que é uma base que traz medidas (características) dos tipos de flores, as informações necessárias para a análise já estava presente na base.

2. COCO *Common Objects in Context* (LIN *et al.*, 2014), 2017 validation set (val 2017): foi selecionado por ser um conjunto de imagens reais de contextos variados, sendo a escolha de utilização apenas do conjunto de validação feita por conta do tamanho deste, com 5000 imagens, num total de 777Mb, possibilitando o processamento local em sua utilização.
3. *Swedish Leaves* (SÖDERKVIST, 2001): foi selecionada com o objetivo de oferecer um estilo distinto de imagens reais, mais especificamente de objetos (neste caso folhas) isolados de contexto e fundo com classificação indicada (15 classes).
4. SegMRF (TRAINA *et al.*, 2004): selecionada dentre as bases de estudo do laboratório GBdi-ICMC-USP, o conjunto de imagens de cortes de ressonância magnética oferece a possibilidade de testar as técnicas em contexto médico. É importante ressaltar que as indicações de classes de imagens feitas são referentes ao tipo de corte médico das imagens da ressonância (sagital, horizontal, axial, ...) apresentando alguns subconjuntos de similaridade visual dentro de mesma classe.
5. Base sintética de imagens gradientes cor aleatória: desenvolvida para este projeto, a base parametrizável possibilita testes em condições completamente controláveis.

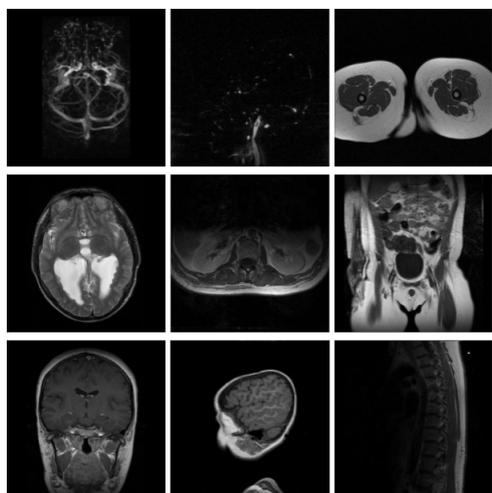
A etapa seguinte foi de processamento dos conjuntos de dados escolhidos, envolvendo extração de características das imagens e redução de dimensionalidade do conjunto de características extraído. A etapa de extração de características não fazia parte da proposta original, supondo-se que conjuntos de imagens com estes atributos adicionais estariam largamente disponíveis. Ao constatar que a especificidade da proposta não seria prontamente obtida, foi tomada a decisão de também extrair as características dos conjuntos de imagens, de forma que métodos simples e consolidados foram aplicados. Não foi garantida a otimização destas técnicas, exatamente por fugir do escopo proposto.



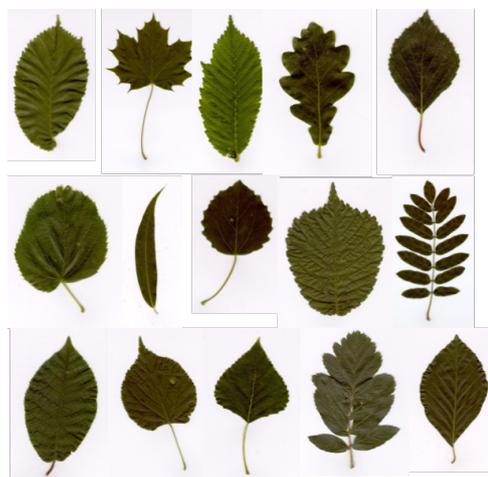
(a) iris



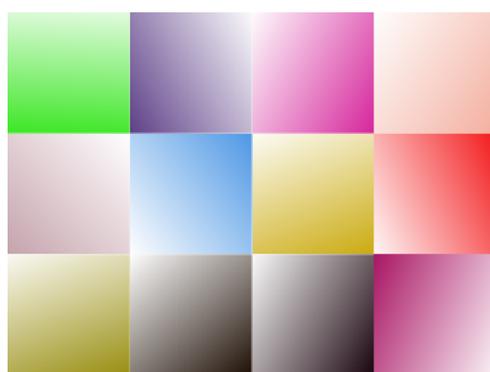
(b) val2017



(c) SegMRF



(d) Swedish Leaves



(e) Synthetic

Figura 14 – Conjuntos de dados escolhidos

Com o intuito de fornecer aos menos uma técnica de extração de cada tipo, sendo estes, cor, forma e textura, foram selecionadas técnicas que pudessem ser aplicadas nos conjuntos de imagem selecionados a partir de bibliotecas Python de acesso livre. Como representante das técnicas de extração foi aplicado um histograma de cor no conjunto de imagens *val2017*, considerando que seria possível obter algum tipo de separação neste caso. O conjunto de imagens sintéticas coloridas também teve suas características extraídas por um histograma de cor. Nas imagens de ressonância magnética foi aplicado de Momentos de Zernike, um técnica de extração de forma, visando posteriormente analisar se o agrupamento corresponderia ao tipo de corte de cada exame. Por fim, no caso das fotos de folha foi aplicada a extração de texturas de Haralick, como meio de considerar as ranhuras intrínsecas de cada tipo de folha.

4.5 Visualização Interativa

A visualização desenvolvida teve como objetivo principal apresentar uma interface interativa, que proporcionasse simultaneamente para dois métodos distintos as possibilidades de mudar parâmetros, efetuar comparações diretas, ampliação/redução sob demanda, exibição da imagem correspondente ao ponto sob o cursor, realizar seleções de pontos por área que exibam seus correspondentes e reestabelecimento das condições iniciais. Além disso também foi incluída uma métrica numérica como parâmetro de comparação entre as duas visualizações ligadas.

A fim de trazer uma avaliação quantitativa das visualizações geradas, foi proposta uma análise de silhueta e oclusão, aplicada automaticamente para cada visualização gerada na interface. Esta taxa proporciona a comparação direta e numérica entre as duas visões e está descrita detalhadamente a seguir.

4.6 Análise de silhueta e oclusão

Visando trazer uma abordagem numérica para a proposta, foi definido um coeficiente de oclusão, calculado automaticamente na visualização a partir da ideia de *data to ink ratio* mencionada na subseção 4.1 do capítulo 3. Para isso, foram geradas as seguintes definições:

Definição 1 (Área suposta). Uma visualização de n pontos, onde cada elemento é renderizado utilizando uma área de a_p pixels, a área suposta A_s é definida por:

$$A_s = n * a_p \quad (4.1)$$

Definição 2 (Área ocupada). Em uma imagem $I(x, y) \in [0, 1]$ renderizada, a área ocupada A_o é definida por:

$$A_o = \sum_x \sum_y f(x, y) \quad (4.2)$$

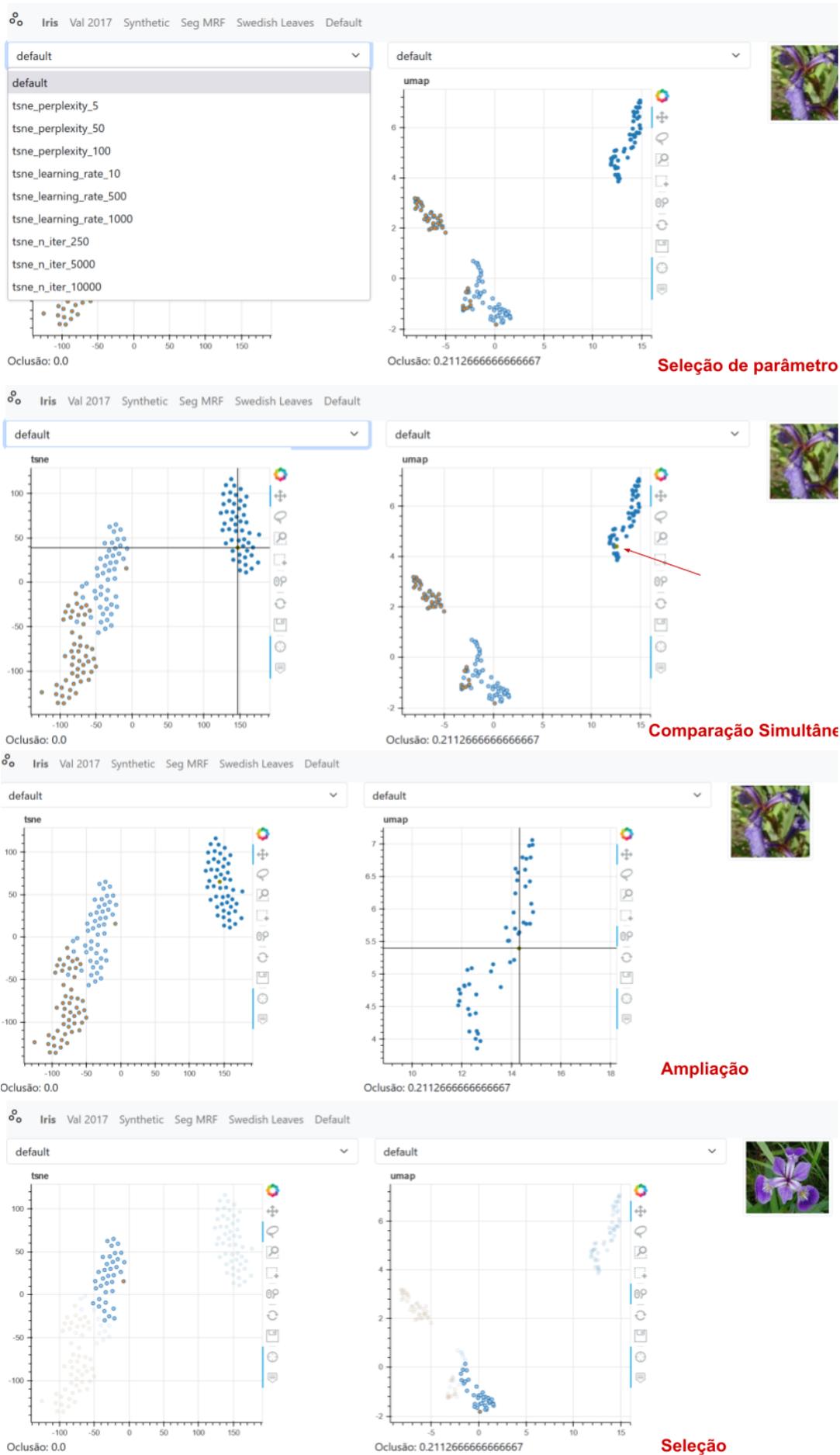


Figura 15 – Ferramentas da visualização

Onde:

$$f(x,y) = \begin{cases} 1 & I(x,y) < 1 \\ 0 & I(x,y) = 1 \end{cases} \quad (4.3)$$

Em uma visualização onde os pontos estão dispersos o suficiente para que não haja sobreposição dos pontos, pode-se afirmar que $A_o = A_s$, pois todos pixels referentes aos pontos de dados são representados na imagem final. Da mesma forma, em uma visualização onde os pontos estão próximos o suficiente para que sejam renderizados com sobreposição, pode-se afirmar que $A_o \ll A_s$. Assim, podemos definir a taxa de oclusão da seguinte forma:

Definição 3 (Taxa de oclusão). Seja A_o a quantidade de pixels preenchidos após a renderização da visualização e A_s a área suposta para tal visualização, a taxa de oclusão é definida por:

$$TxOc = 1 - \frac{A_o}{A_s} \quad (4.4)$$

De acordo com a definição 3, a taxa de oclusão de pontos se dá pela diferença direta da área preenchida na imagem, ou seja, o coeficiente gerado na razão entre o suposto número de pixels que a visualização produziria, e quantidade exata aparente na imagem, sendo esta medida realizada automaticamente utilizando um processamento rápido da imagem.

O funcionamento direto engloba os seguintes passos:

1. Leitura da visualização gerada como imagem em .png
 - a) Geração da visualização com uma área total (tamanho da imagem) definida em dpi
 - b) Pontos da visualização são gerados como quadrados para evitar aproximação de cálculo de área gerada causada pelo serrilhamento das bordas de círculos
 - c) Pontos da visualização são gerados com tamanho que considera as proporções de pixel (constante da biblioteca multiplicada pela proporção em dpi imagem gerada)
 - d) Visualização é salva em imagem temporária recebendo como parâmetro o tamanho em dpi
2. Transformação da imagem colorida original para uma imagem em preto e branco
 - a) Leitura da imagem temporária como preto e branco
3. Cálculo da área ocupada A_o por cada elemento (ponto) da visualização, considerando as proporções de pixels
 - a) Cálculo da área do quadrado a , aplicando o mesmo valor de lateral proporcional ao pixel descrito no item 1c

4. Cálculo da área suposta a ser ocupada pelos pontos caso não haja nenhuma sobreposição de pontos, considerando o cálculo de área por elemento realizado anteriormente
5. Cálculo "visual" da área efetivamente ocupada pelos elementos da visualização, aplicando uma comparação entre os espaços em branco e aqueles preenchidos na imagem salva
 - a) Tratamento da imagem como array e contagem do número de pontos não brancos (como a imagem foi lida em preto e branco, cada posição que contenha valor será parte de um ponto)
 - b) Divisão do resultado da contagem pelo número total de pontos
6. Cálculo da taxa de oclusão entre as áreas supostas e reais

Por exemplo, tomadas as visualizações demonstrativas A, B e C com três elementos cada na figura 16. Primeiramente todos os pontos são tratados como preto e branco, para que sejam feitas as comparações de área. Considerando que cada ponto tenha valor de área $a_p = 1$, temos os valores indicados de área suposta (quanto de área no espaço seria preenchida em preto se nenhum ponto estiver encoberto) e área ocupada (área efetivamente preenchida em preto no espaço) para cada caso. Assim, o caso A representa uma situação sem oclusão de pontos, o caso B a oclusão parcial de um ponto, e o caso C a total sobreposição dos três elementos.

Sua aplicação não apenas possibilita uma avaliação da técnica aplicada na extração de características e subsequente redução de dimensionalidade dos pontos, como também permite avaliar o tamanho ideal dos elementos gráficos na visualização de forma a permitir menos oclusão, visto que o parâmetro do tamanho de elementos também é configurável.

A implementação completa para este método e as outras etapas descritas anteriormente pode ser consultada no Apêndice A, estando disponíveis os passos descritos na figura 17.

4.7 Testes e análise

A aplicação consistindo nas visualizações comparadas entre as duas técnicas de redução de dimensionalidade (t-SNE e UMAP) funciona em servidor local com a opção de seleção entre as cinco bases de dados introduzidas anteriormente, sendo possível também selecionar os parâmetros desejados nas técnicas de redução conforme opções previamente processadas para ambos t-SNE e UMAP. Nas comparações é possível observar os indícios da tradução das características visuais das imagens na representação gráfica dos pontos compostos pela redução de dimensionalidade dos vetores de características, por conseguinte entre os exemplos serão apresentados indícios de percepção de brilho e contraste para grupos de imagens reais não classificados, percepção de classes e similaridade em imagens de exames médicos, distinção entre colorações geradas aleatoriamente, em que não se observa formação de agrupamentos espúrios e

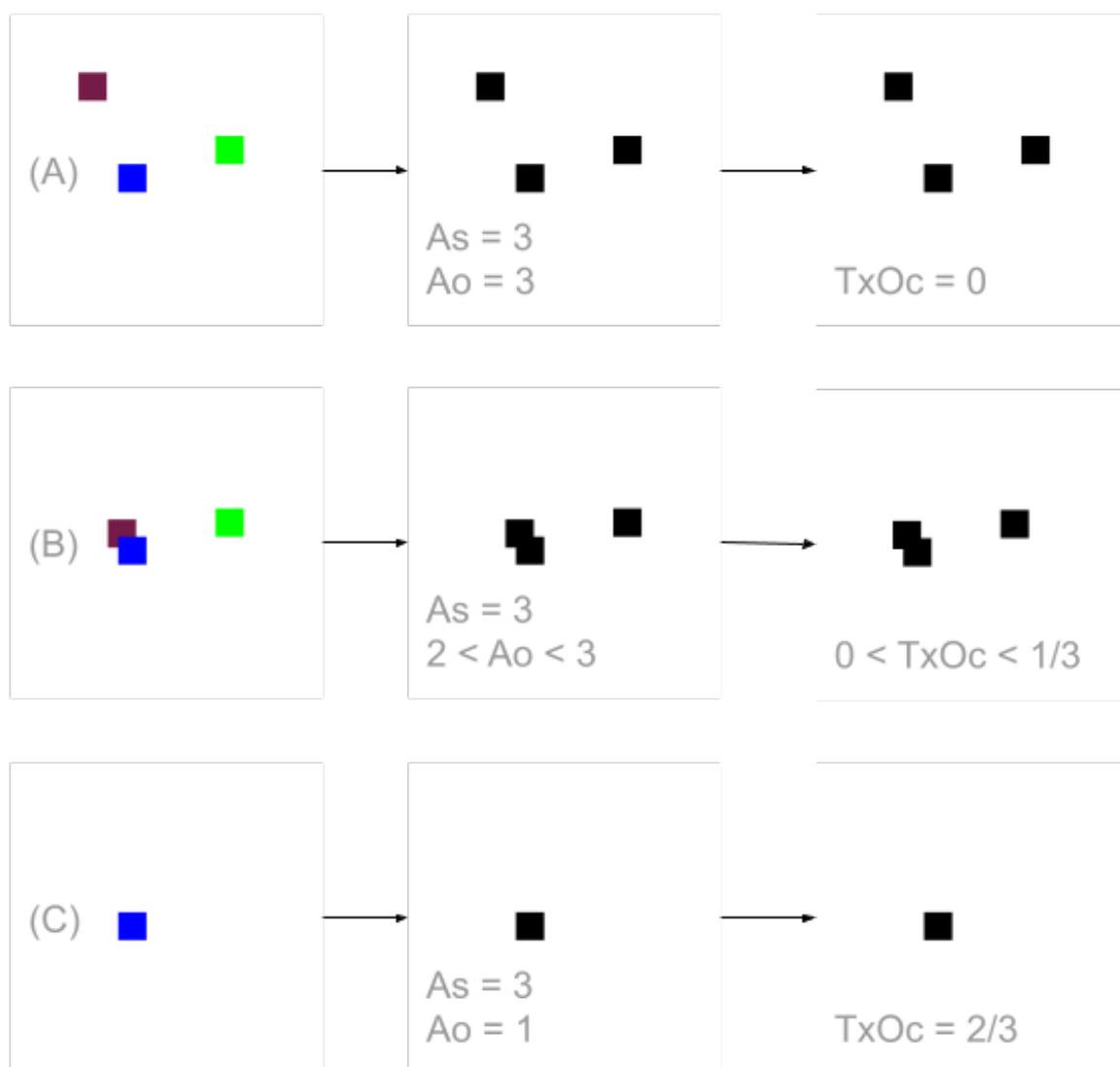


Figura 16 – Exemplos de oclusão

Nota – A_s = Área suposta, A_o = Área ocupada, $TxOc$ = Taxa de Oclusão

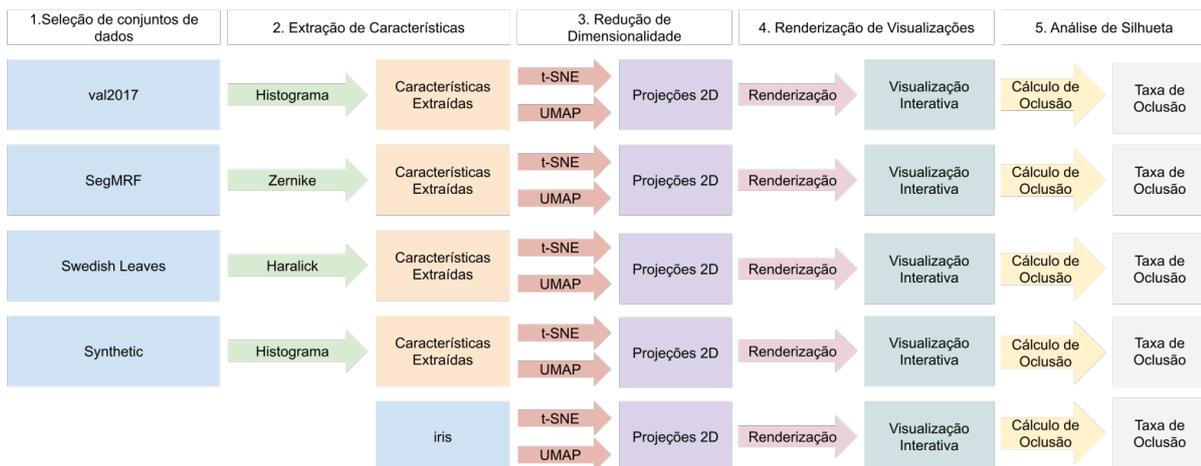


Figura 17 – Etapas da implementação do projeto

interpretação de formatos de objetos do mundo real. Todas estas análises são apresentadas em conjunto a métrica desenvolvida para avaliação de visualizações.

4.7.1 Parâmetros na redução de dimensionalidade

Um conjunto de casos teste foi desenvolvido apresentando as variações máximas e mínimas nos parâmetros das técnicas tsne e umap, além do caso padrão para cada técnica que oferece saídas balanceadas para a maioria dos casos.

Para a técnica t-SNE, foram variados os parâmetros de *perplexity*, número de iterações (*n_iter*) e taxa de aprendizado (*learning_rate*), e para a técnica UMAP foram variados o número de vizinhos (*n_neighbors*), distância mínima (*min_dist*) e *spread*. Esta seleção entre as variações disponíveis foi feita com o objetivo de variar apenas os parâmetros mais básicos de cada técnica, criando assim um número menor de casos teste para posterior processamento. Conforme disponível na documentação do t-SNE ⁶, os valores de casos teste foram variados entre os máximos e mínimos recomendados, sendo alterado um parâmetro por vez, e mantendo os outros nos valores default. Os valores são apresentados na Tabela 3. O procedimento foi realizado de forma análogo para a técnica UMAP, cuja documentação ⁷ também descreve seus parâmetros básicos e recomendações.

Dentre os parâmetros selecionados, existem aqueles que são inadequados para os dados escolhidos, no caso do t-SNE, o número de iterações 250, é o mínimo recomendado utilizado na implementação da biblioteca para a convergência do método, não trazendo convergência para o volume de dados trabalhado neste caso (Figura 18), e no caso do UMAP, o número de vizinhos 2, força o algoritmo a se concentrar em estruturas excessivamente locais (múltiplas vizinhanças

⁶ <<https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html>>

⁷ <<https://umap-learn.readthedocs.io/en/latest/index.html>>

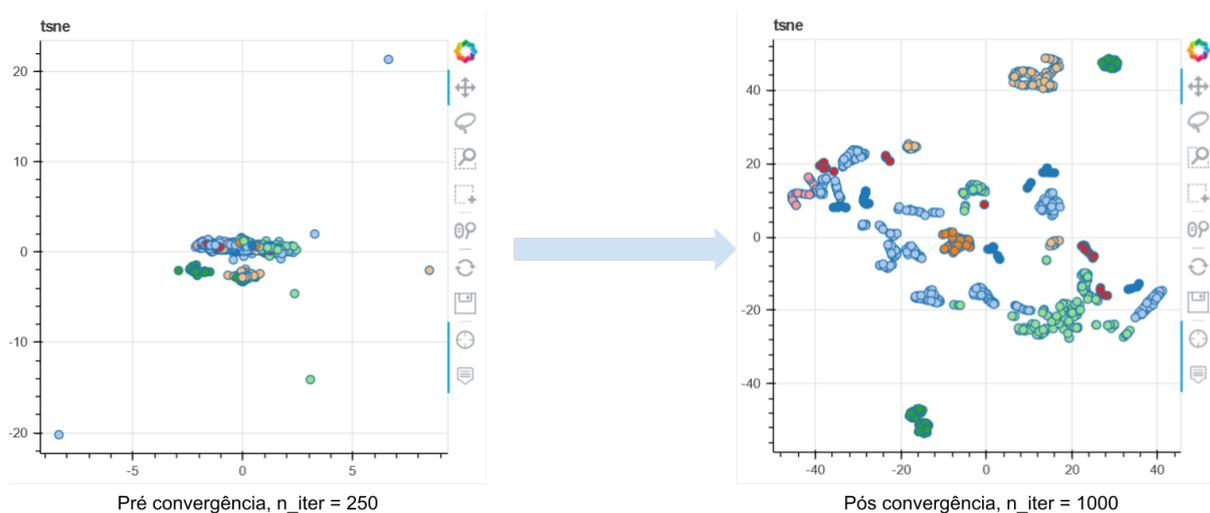


Figura 18 – Estado da visualização com variação no número de iterações

	default	mínimo recomendado	máximo recomendado
perplexity	30	5	50
learning_rate	auto	10	1000
n_iter	1000	250	não mencionado

Tabela 3 – Parâmetros recomendados na documentação da técnica t-SNE

Nota – $auto = \max(\text{samplesize} / \text{early_exaggeration}(\text{default} = 12.0) / 4, 50)$

	number of neighbors	minimal distance	spread
umap_default	15	0.1	1
umap_n_neighbors_2	2	0.1	1
umap_n_neighbors_50	50	0.1	1
umap_n_neighbors_100	100	0.1	1
umap_min_dist_0_01	15	0.01	1
umap_min_dist_0_5	15	0.5	1
umap_spread_0_5	15	0.1	0.5
umap_spread_3	15	0.1	3

Tabela 4 – Casos teste técnica UMAP

de dois elementos/vizinhos) por imagem não traduz as características de nenhum dos conjuntos de dados. Desta forma estes parâmetros servem a função de comparação com aqueles que seriam mais adequados, como o próprio caso padrão de cada algoritmo.

4.7.2 Aplicação

Na Figura 19 observa-se a tela inicial da aplicação Web que oferece as possibilidades de seleção entre conjuntos de dados na barra superior, e nos dois menus do tipo *dropdown* para cada

	perplexity	learning rate	number of iterations
tsne_default	30	auto	1000
tsne_perplexity_5	5	auto	1000
tsne_perplexity_50	50	auto	1000
tsne_perplexity_100	100	auto	1000
tsne_learning_rate_10	30	10	1000
tsne_learning_rate_500	30	500	1000
tsne_learning_rate_1000	30	1000	1000
tsne_n_iter_250	30	auto	250
tsne_n_iter_5000	30	auto	5000
tsne_n_iter_10000	30	auto	10000

Tabela 5 – Casos Teste t-SNE



Figura 19 – Tela inicial da aplicação

conjunto, a seleção dos parâmetros previamente calculados para ambas as técnicas de redução de dimensionalidade. Diretamente abaixo de cada visualização é exibida a taxa de oclusão calculada para cada visão, sempre aplicadas as mesmas configurações de tamanho de elementos em tela nos dois casos exibidos, proporcionando uma comparação de valores semelhantes com parâmetros semelhantes. Nas imagens exclusivamente ilustrativas da interface é utilizado o dataset iris.

4.7.3 Análise

Foram realizados dois tipos de análise nas visualizações geradas, primeiramente uma análise qualitativa, cujo objetivo foi identificar visualmente indícios da tradução das características visuais da imagem para elementos visuais, ou seja, verificar a adequação das visualizações ao objetivo proposto. Em seguida foi realizada uma avaliação quantitativa a partir da métrica de oclusão apresentada (ver seção 4.6), buscando colher características numéricas nas visualizações.

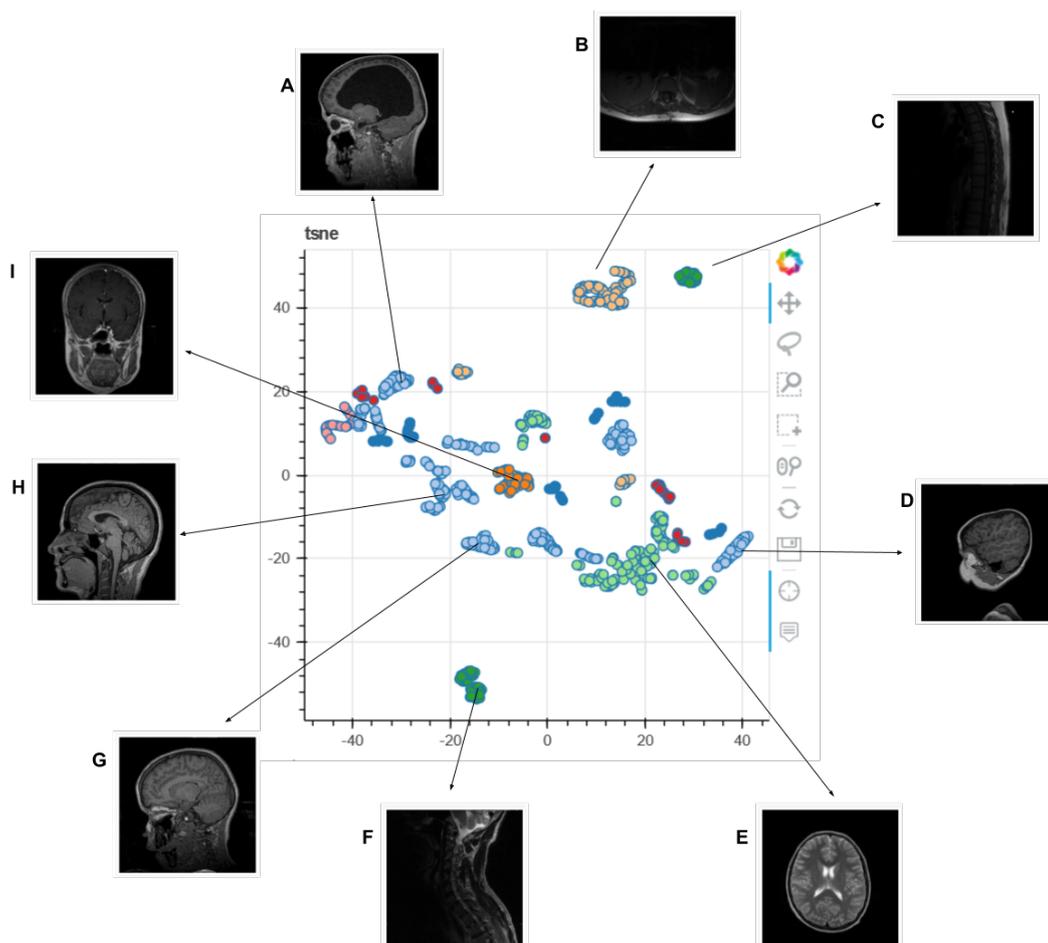


Figura 20 – Diferentes imagens e classes representados na visualização do conjunto SegMRF

Os principais achados relevantes na análises das visualizações geradas são apresentados nas figuras 21, 22 24, 25, e descritos a seguir.

Na Figura 21 é possível observar na seleção em laço do canto superior a formação de agrupamentos, que se apresentam separados do resto dos dados em ambas as técnicas de redução. Conforme é possível observar pela marcação de cor nos pontos da visualização, tais agrupamentos são compostos de pontos da mesma classe. Nas visões deste conjunto de dados é possível também observar pontos de suposta mesma classe distribuídos em mais de um agrupamento, como no caso do azul claro no centro. Foi constatado que estes pontos, apesar de atribuídos à mesma classe, são visualmente distintos na observação das imagens originais, visto que representam etapas mais centrais ou mais externas dos cortes da ressonância magnética. O mesmo efeito é observado nas visualizações que aplicam a técnica UMAP.

A Figura 20 detalha a análise anterior, em uma ampliação de visualização gerada pelo t-SNE, apresentando os diferentes agrupamentos e exemplos dos tipos de imagens presentes em cada um. Como mencionado, é possível observar que **A, D, G e H** fazem parte da mesma classe (pontos da mesma cor) porém representam tipos de imagens visualmente distintas, apesar

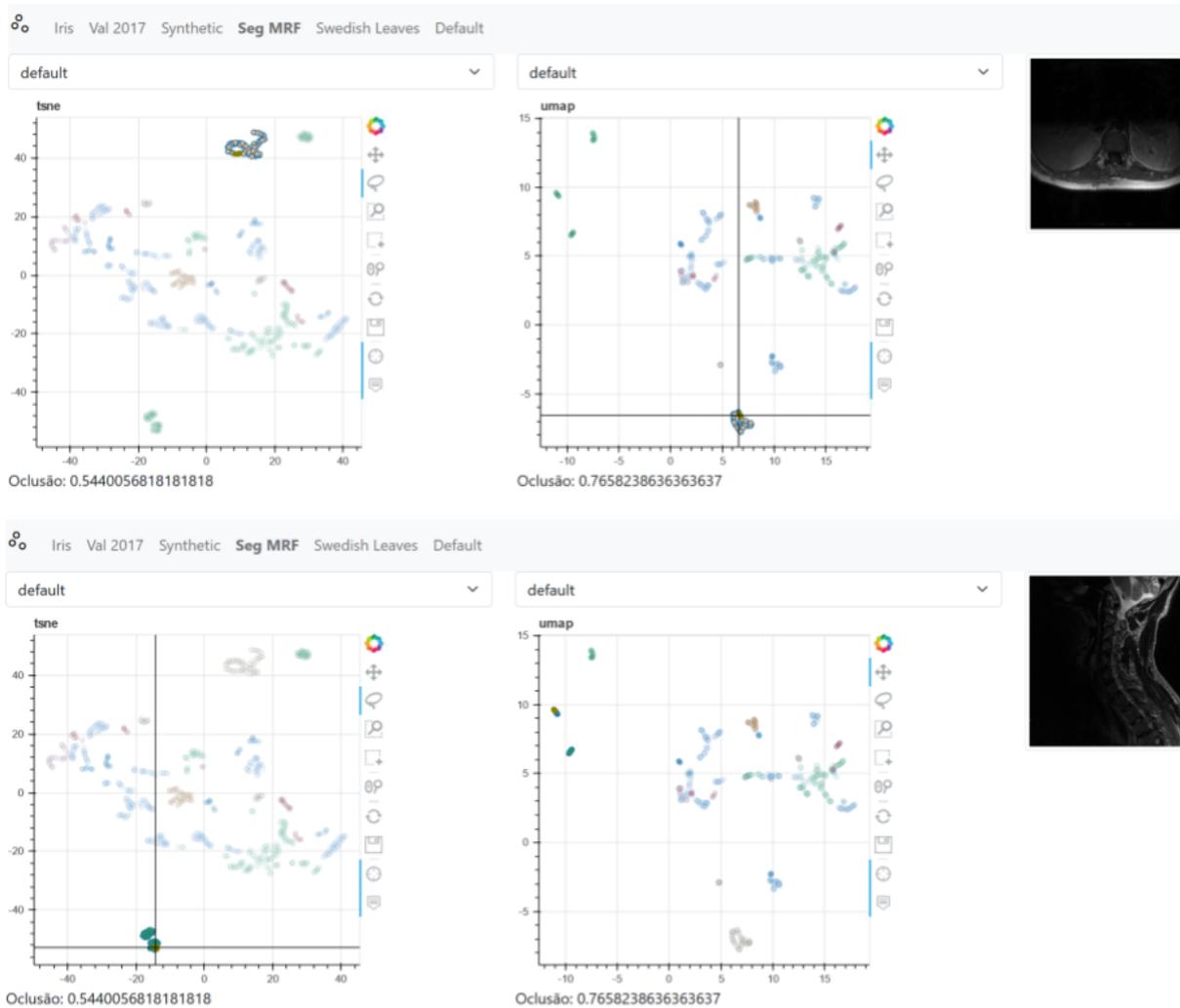


Figura 21 – Formação de Agrupamentos nas visualizações do conjunto SegMRF

do método da similaridade no contexto da semântica médica (tipo de corte). O mesmo efeito pode ser observado em **C** e **F**. Portanto, no caso deste conjunto de imagens foi possível observar algum nível de tradução visual dos dados complexos, para a projeção visual gerada.

No caso do conjunto de dados val2017, as características extraídas foram do tipo cor, por meio do histograma. Desta forma seria esperado que alguma percepção relacionada às cores das imagens fosse traduzida para a projeção. Como é possível observar na figura 22, determinados pontos do espaço correspondem a diferentes níveis de brilho nas imagens. Na ampliação da figura 23, são inclusos exemplos dos extremos de cor, com imagens mais escuras no canto superior e imagens mais claras na parte de baixo. É possível que a aplicação de um espaço métrico de coloração na redução de dimensionalidade (que em ambos os casos, t-SNE e UMAP, pode receber distintas métricas como parâmetro) neste caso pudesse trazer ainda mais percepção destas características, ficando esta sugestão para trabalhos futuros.

No caso das imagens de gradientes do cor do conjunto sintético desenvolvido, como

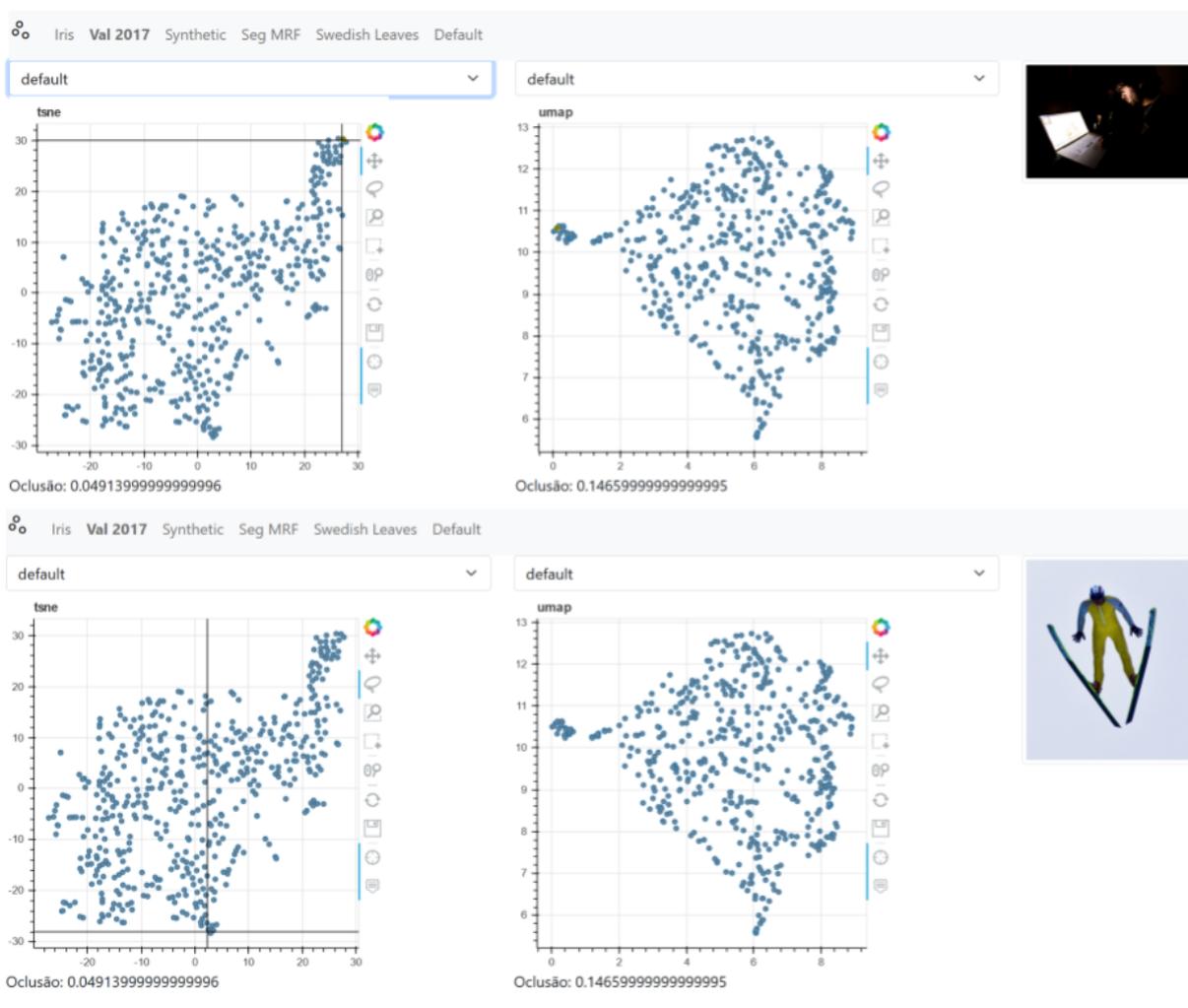


Figura 22 – Percepção de cor na visualização do dataset val2017

as cores geradas foram completamente aleatórias, observa-se o espalhamento das mesmas no espaço das projeções (Figura 24), de forma aleatória. Bem como no exemplo anterior, é possível que um espaço de distribuição de cores trouxesse uma dispersão distinta, mas a percepção de aleatoriedade e espalhamento confirma que não foram traduzidas características inexistentes de similaridade.

As características extraídas do conjunto de imagens Swedish Leaves, de distintas folhas de plantas, ao ser representadas nas visualizações (Figura 25) aparentaram formar um padrão que gradualmente muda as classes exibidas em torno de uma forma circular (Figura 26). Por meio da observação e comparação nos testes, foi possível observar ao percorrer o círculo que tanto o caule muda de tamanho, como o formato das folhas e tipo de rugosidade. Além disso houve uma separação entre folhas *únicas* e caules com múltiplas folhas pequenas. Tal separação clara pode evidenciar uma interpretação de textura neste caso (uma folha versus repetição de folhas), visto que o extrator aplicado foi de textura. No caso das aparentes características de forma (tamanho do caule, formato da folha), também é possível que sejam correlacionados a

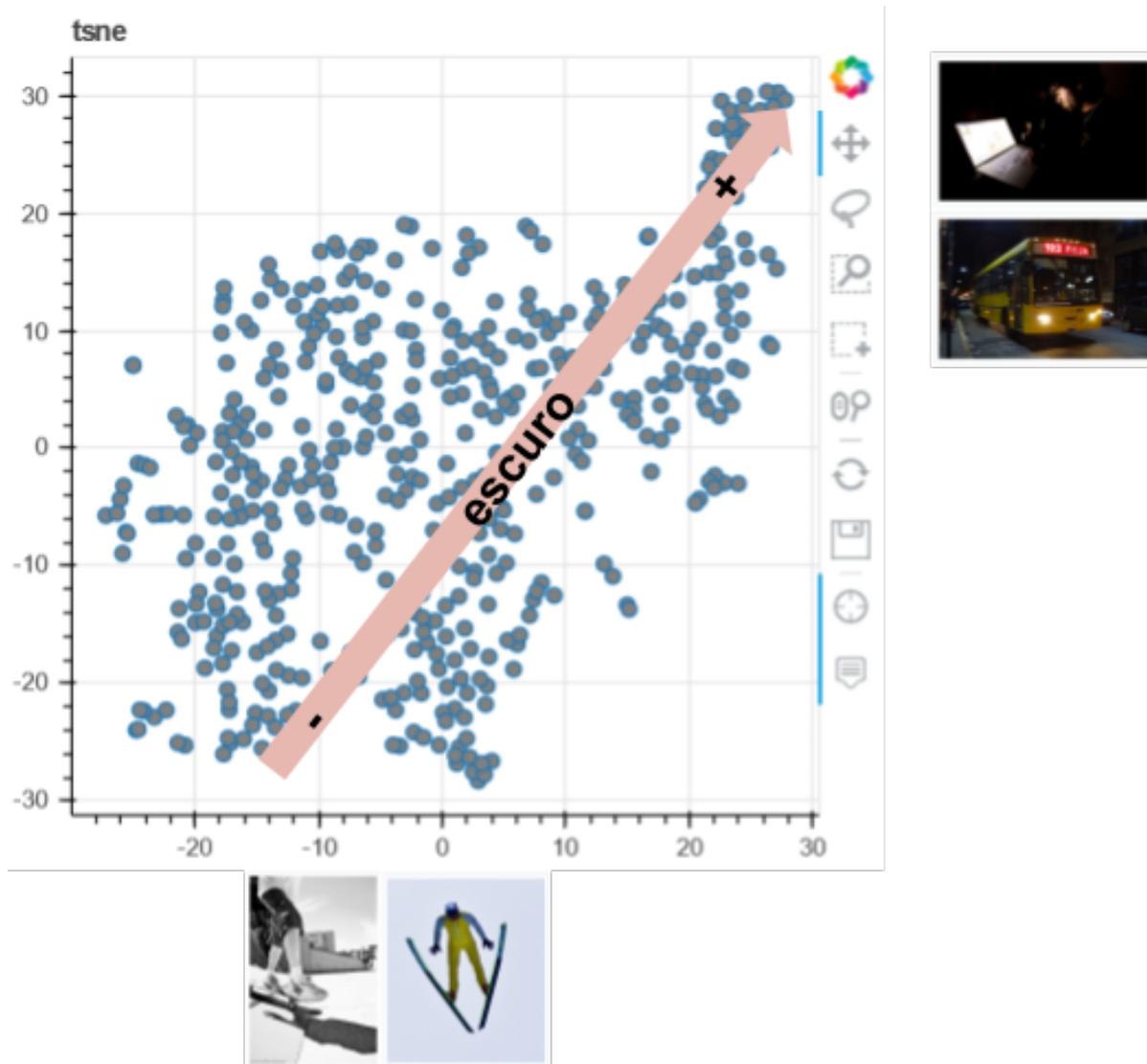


Figura 23 – Percepção de cor na visualização do dataset val2017 - Ampliação e exemplos

rugosidade e tipo das folhas, desta forma sendo representados indiretamente pela textura.

Foi realizada em paralelo uma análise de oclusão conforme o método apresentado na seção anterior 4.6. A partir da análise da tabela 6 foi possível observar que parâmetros propositalmente ruins (como aqueles planejados para não convergência), tenderam a taxas de oclusão mais altas, visto agrupamentos espúrios em estados pré-convergência, ou número muito inferior de vizinhos na UMAP gerando espalhamento de agrupamentos menores. Em datasets classificados com clusterização e imagens similares, foram observadas taxas de oclusão maiores, ao mesmo tempo que datasets não classificados com imagens aleatórias, apresentaram menores taxas de oclusão devido à menor similaridade e proximidade entre as imagens.

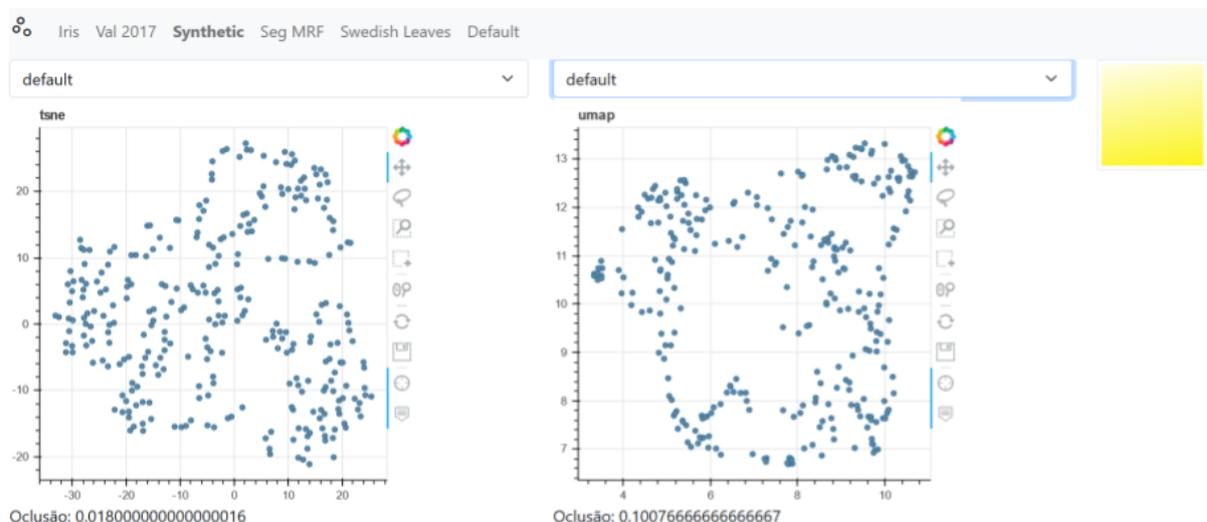


Figura 24 – Distribuição dos gradientes de cores aleatórias em espalhamento

	iris	val2017	SegMRF	Synthetic	Swedish Leaves	média
tsne_default	0,00%	4,91%	54,40%	1,80%	47,35%	21,69%
tsne_learning_rate_10	13,98%	5,43%	50,11%	4,29%	42,97%	23,36%
tsne_learning_rate_1000	0,00%	4,47%	52,56%	21,24%	52,21%	26,09%
tsne_learning_rate_500	59,59%	6,01%	57,61%	6,34%	44,59%	34,83%
tsne_n_iter_10000	21,11%	3,85%	52,05%	4,59%	62,67%	28,85%
tsne_n_iter_250	13,01%	76,51%	88,11%	44,94%	60,37%	56,59%
tsne_n_iter_5000	21,11%	3,85%	52,05%	4,59%	61,08%	28,54%
tsne_perplexity_100	5,73%	6,59%	34,13%	11,61%	57,85%	23,18%
tsne_perplexity_5	5,89%	18,32%	55,06%	14,32%	54,88%	29,69%
tsne_perplexity_50	14,65%	5,95%	39,65%	7,17%	43,51%	22,19%
umap_default	21,13%	14,66%	76,58%	10,08%	72,91%	39,07%
umap_min_dist_0_01	34,31%	19,22%	84,26%	17,75%	78,10%	46,73%
umap_min_dist_0_5	1,85%	5,03%	55,47%	3,14%	35,60%	20,22%
umap_n_neighbors_100	8,19%	8,39%	55,45%	4,28%	68,93%	29,05%
umap_n_neighbors_2	60,50%	69,41%	66,74%	62,94%	69,90%	65,90%
umap_n_neighbors_50	23,06%	7,89%	73,10%	5,51%	66,63%	35,24%
umap_spread_0_5	21,14%	13,95%	79,97%	10,46%	72,27%	39,56%
umap_spread_3	5,12%	10,23%	68,57%	5,47%	66,66%	31,21%

Tabela 6 – Taxas de oclusão classificadas

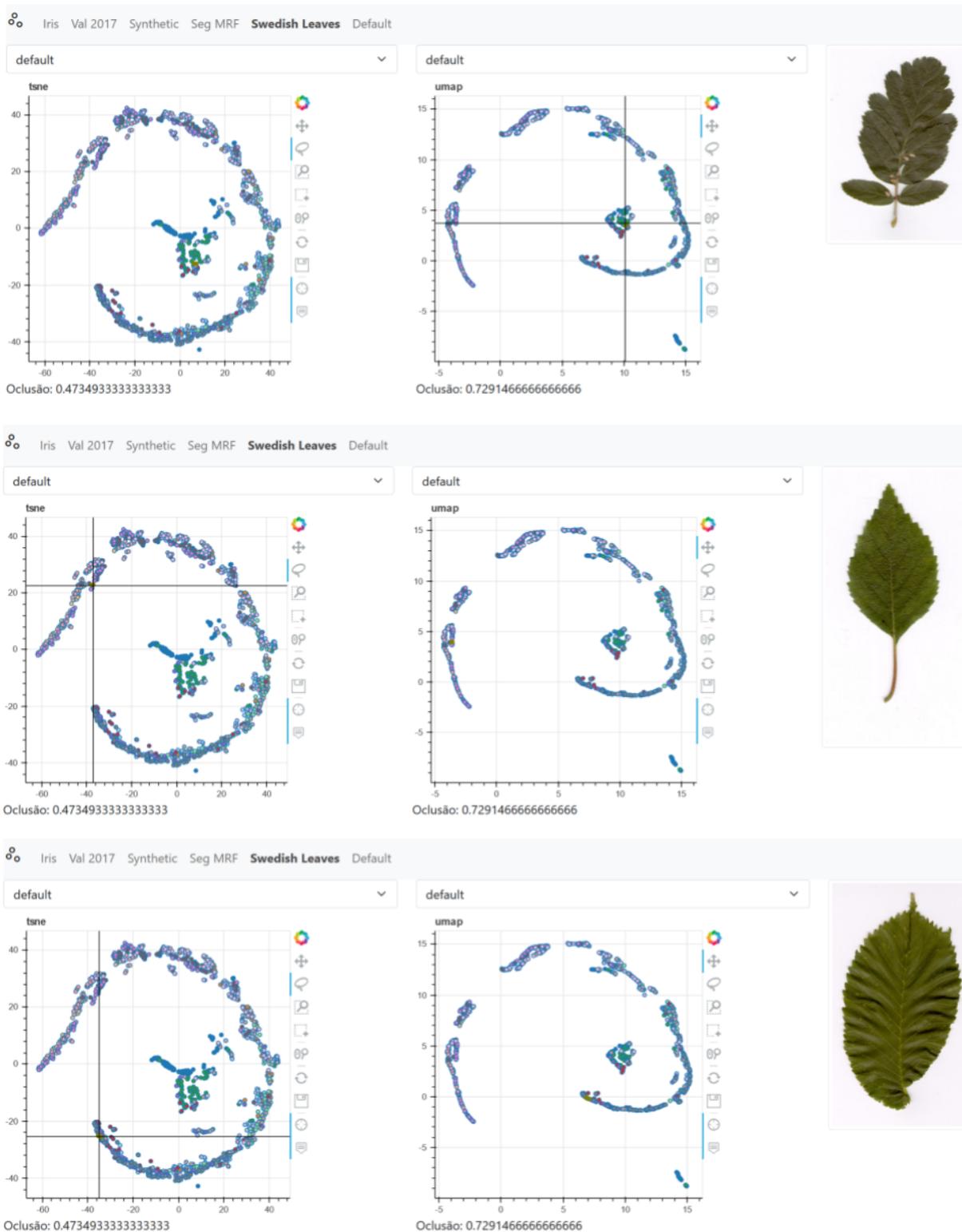


Figura 25 – Representação de tipos de folhas distribuídas

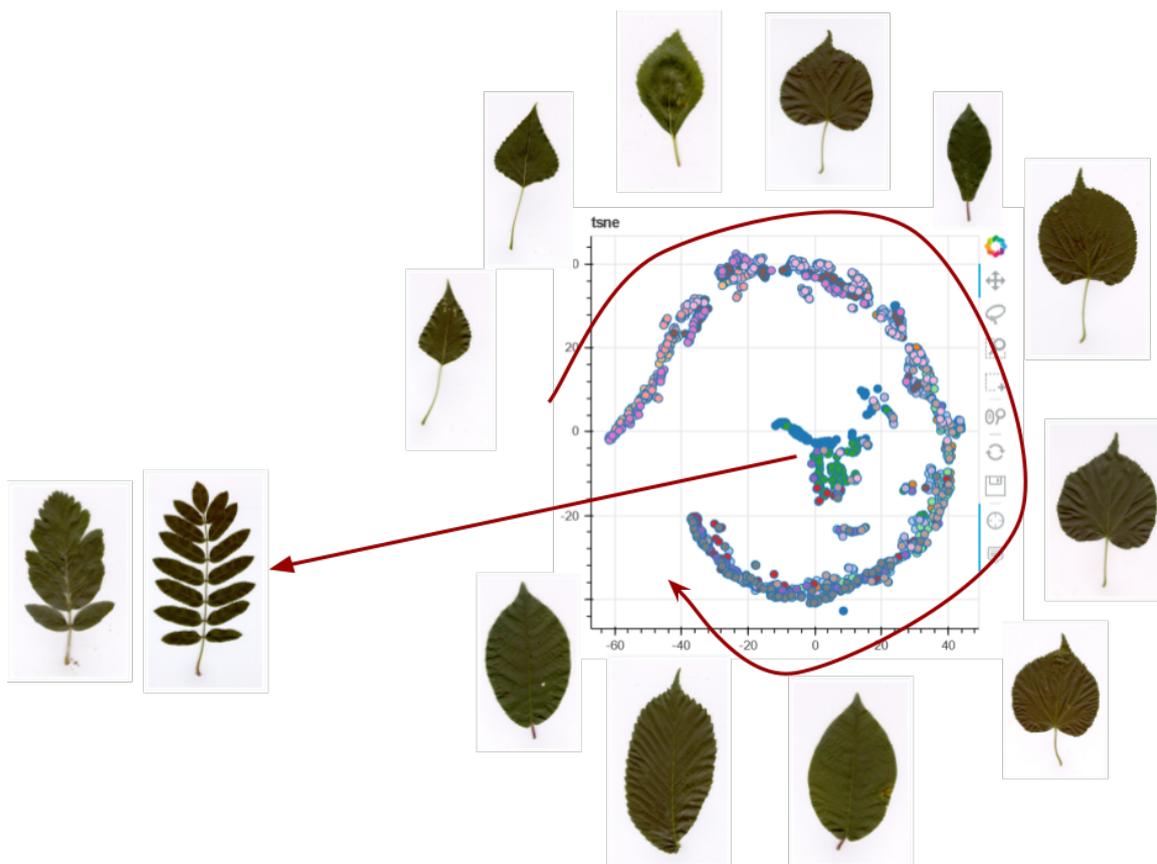


Figura 26 – Ampliação de tipos de folhas distribuídas

4.8 Considerações finais

A partir da metodologia implementada, as análises realizadas puderam trazer um nível promissor de observações no sentido da possibilidade de representação semântica características extraídas de imagens em visualizações. No próximo capítulo serão consolidadas estas observações.

CONCLUSÕES

5.1 Conclusões

Foi possível observar nos casos analisados que os vetores de características extraídos retiveram particularidades visuais, permitindo uma análise exploratória e observação do comportamento dos métodos de extração de características das imagens por meio das visualizações, e possibilitando a identificação destas similaridades visuais mesmo após a redução de dimensionalidade de seus respectivos dados complexos. Essa capacidade de diferenciação e apresentação visual das características que representam as imagens bem como a quantificação de quando não é possível diferenciá-las (taxa de oclusão) são os aspectos mais importantes do estudo realizado nessa pesquisa em nível de Mestrado. No tópico de análise de oclusão, foi observado que diferentes tipos de conjuntos de dados apresentam comportamentos distintos em relação ao agrupamento de pontos, sendo este um fator a ser levado em consideração, de forma que uma taxa de oclusão mais elevada no contexto de dados que representem imagens similares não pode ser diretamente considerado uma falha na visualização, mas sim que outras ferramentas e técnicas devem ser utilizadas conjuntamente.

5.2 Limitações

À vista das observações levantadas, a principal limitação encontrada foi na testagem e comparação de extratores ideais, ou seja, por não ser parte do escopo do projeto a extração de características das imagens propriamente ditas, e sim sua utilização e análise, os extratores aplicados não foram desenvolvidos e aprimorados visando cada um dos conjuntos analisados, desta forma este trabalho não apresenta comparações entre extratores específicos, que seriam considerados ideias e outras variedades destes. Porém, esse aspecto deverá ser objeto de estudo posterior.

5.3 Propostas futuras

Ficam como propostas para trabalhos futuros a realização de testes de redução de dimensionalidade em diferentes espaços métricos, buscando uma correspondência com os tipos de características extraídos. Outra sugestão é a aplicação em conjunto de sistemas CBIR (*Content-based image retrieval*), visando analisar a proposta de análise de extratores neste contexto, bem como a aplicação de seleção visual para a realimentação da visualização e da base de imagens, por meio das ferramentas de laço e seleção em caixa.

REFERÊNCIAS

AHMED, A. Implementing relevance feedback for content-based medical image retrieval. **IEEE Access**, v. 8, p. 79969–79976, 2020. Citado na página 28.

ALBUQUERQUE, G.; EISEMANN, M.; LEHMANN, D. J.; THEISEL, H.; MAGNOR, M. Improving the visual analysis of high-dimensional datasets using quality measures. In: **2010 IEEE Symposium on Visual Analytics Science and Technology**. [S.l.: s.n.], 2010. p. 19–26. Citado nas páginas 37, 39 e 40.

ANKERST, M. Visual data mining with pixel-oriented visualization techniques. In: **Proceedings of the ACM SIGKDD Workshop on Visual Data Mining**. [S.l.: s.n.], 2001. p. 23. Citado na página 38.

AZEVEDO-MARQUES, P. M. d.; RANGAYYAN, R. M. Content-based Retrieval of Medical Images: Landmarking, Indexing, and Relevance Feedback. **Synthesis Lectures on Biomedical Engineering**, v. 8, n. 1, p. 1–143, jan. 2013. ISSN 1930-0328. Disponível em: <<https://www.morganclaypool.com/doi/abs/10.2200/S00469ED1V01Y201301BME048>>. Citado na página 28.

AZEVEDO-MARQUES, P. M. de; ROSA, N. A.; TRAINA, A. J. M.; TRAINA, C.; KINOSHITA, S. K.; RANGAYYAN, R. M. Reducing the semantic gap in content-based image retrieval in mammography with relevance feedback and inclusion of expert knowledge. **International Journal of Computer Assisted Radiology and Surgery**, v. 3, n. 1, p. 123–130, jun. 2008. ISSN 1861-6429. Disponível em: <<https://doi.org/10.1007/s11548-008-0154-4>>. Citado na página 28.

BEDO, M.; RAMOS, J. S.; TRAINA, A. J.; TRAINA, C.; NOGUEIRA-BARBOSA, M. H.; AZEVEDO-MARQUES, P. M. Wia-spine: A cbir environment with embedded radiomic features to assess fragility fractures. In: **IEEE. 2022 IEEE 35th International Symposium on Computer-Based Medical Systems (CBMS)**. [S.l.], 2022. p. 72–77. Citado na página 19.

BEDO, M. V. N.; SANTOS, D. Pereira dos; PONCIANO-SILVA, M.; AZEVEDO-MARQUES, P. M. de; CARVALHO, A. P. d. L. Ferreira de; TRAINA, C. Endowing a Content-Based Medical Image Retrieval System with Perceptual Similarity Using Ensemble Strategy. **Journal of Digital Imaging**, v. 29, n. 1, p. 22–37, fev. 2016. ISSN 1618-727X. Disponível em: <<https://doi.org/10.1007/s10278-015-9809-1>>. Citado na página 19.

BEHRISCH, M.; BLUMENSCHNEIN, M.; KIM, N. W.; SHAO, L.; EL-ASSADY, M.; FUCHS, J.; SEEBACHER, D.; DIEHL, A.; BRANDES, U.; PFISTER, H.; SCHRECK, T.; WEISKOPF, D.; KEIM, D. A. Quality Metrics for Information Visualization. **Computer Graphics Forum**, v. 37, n. 3, p. 625–662, jun. 2018. ISSN 1467-8659. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13446>>. Citado nas páginas 9, 31, 32, 35, 37, 38, 39, 40 e 41.

BEHRISCH, M.; KORKMAZ, F.; SHAO, L.; SCHRECK, T. Feedback-driven interactive exploration of large multidimensional data supported by visual classifier. In: **2014 IEEE Conference**

on **Visual Analytics Science and Technology (VAST)**. [S.l.: s.n.], 2014. p. 43–52. Citado na página 40.

BERTINI, E.; TATU, A.; KEIM, D. Quality Metrics in High-Dimensional Data Visualization: An Overview and Systematization. **IEEE Transactions on Visualization and Computer Graphics**, v. 17, n. 12, p. 2203–2212, dez. 2011. ISSN 1077-2626. Citado na página 30.

BLAAS, J.; BOTHA, C. P.; POST, F. H. Interactive Visualization of Multi-Field Medical Data Using Linked Physical and Feature-Space Views. In: MUSETH, K.; MOELLER, T.; YNNERMAN, A. (Ed.). **Eurographics/ IEEE-VGTC Symposium on Visualization**. [S.l.]: The Eurographics Association, 2007. ISBN 978-3-905673-45-6. ISSN 1727-5296. Citado na página 41.

BORGO, R.; KEHRER, J.; CHUNG, D. H. S.; MAGUIRE, E.; LARAMEE, R. S.; HAUSER, H.; WARD, M.; CHEN, M. Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications. In: SBERT, M.; SZIRMAY-KALOS, L. (Ed.). **Eurographics 2013 - State of the Art Reports**. [S.l.]: The Eurographics Association, 2013. ISSN 1017-4656. Citado na página 39.

BORKIN, M. A.; BYLINSKII, Z.; KIM, N. W.; BAINBRIDGE, C. M.; YEH, C. S.; BORKIN, D.; PFISTER, H.; OLIVA, A. Beyond Memorability: Visualization Recognition and Recall. **IEEE Transactions on Visualization and Computer Graphics**, v. 22, n. 1, p. 519–528, jan. 2016. ISSN 1077-2626. Citado na página 41.

BOSTOCK, M.; OGIEVETSKY, V.; HEER, J. D³ data-driven documents. **IEEE transactions on visualization and computer graphics**, IEEE, v. 17, n. 12, p. 2301–2309, 2011. Citado na página 48.

BRATH, R. Metrics for effective information visualization. In: **Proceedings of VIZ '97: Visualization Conference, Information Visualization Symposium and Parallel Rendering Symposium**. [S.l.: s.n.], 1997. p. 108–111. Citado na página 41.

BRATH, R.; BANISSI, E. Evaluating lossiness and fidelity in information visualization. In: **Visualization and Data Analysis 2015**. International Society for Optics and Photonics, 2015. v. 9397, p. 93970H. Disponível em: <<https://www.spiedigitallibrary.org/conference-proceedings-of-spie/9397/93970H/Evaluating-lossiness-and-fidelity-in-information-visualization/10.1117/12.2083444.short>>. Citado na página 41.

BUGATTI, P. H.; RIBEIRO, M. X.; TRAINA, A. J. M.; JR, C. T. Feature Selection Guided by Perception in Medical CBIR Systems. In: **2011 IEEE First International Conference on Healthcare Informatics, Imaging and Systems Biology**. [S.l.: s.n.], 2011. p. 323–330. Citado na página 28.

CARD, M. **Readings in information visualization: using vision to think**. [S.l.]: Morgan Kaufmann, 1999. Citado na página 30.

CAZZOLATO, M. T.; SCABORA, L. C.; ZABOT, G. F.; GUTIERREZ, M. A.; JR., C. T.; TRAINA, A. J. M. Featset+: Visual features extracted from public image datasets. **J. Inf. Data Manag.**, v. 13, n. 1, 2022. Disponível em: <<https://doi.org/10.5753/jidm.2022.2328>>. Citado na página 24.

CHEN, M.; GOLAN, A. What May Visualization Processes Optimize? **IEEE Transactions on Visualization and Computer Graphics**, v. 22, n. 12, p. 2619–2632, dez. 2016. ISSN 1077-2626. Citado nas páginas 20 e 40.

CHUNG, D. H.; LEGG, P. A.; PARRY, M. L.; BOWN, R.; GRIFFITHS, I. W.; LARAMEE, R. S.; CHEN, M. Glyph sorting: Interactive visualization for multi-dimensional data. **Information Visualization**, v. 14, n. 1, p. 76–90, jan. 2015. ISSN 1473-8716. Disponível em: <<https://doi.org/10.1177/1473871613511959>>. Citado na página 39.

DATTA, R.; LI, J.; WANG, J. Z. Content-based image retrieval: approaches and trends of the new age. In: . [S.l.]: ACM, 2005. p. 253–262. ISBN 978-1-59593-244-0. Citado na página 25.

DECHEVA., D.; LINSEN., L. Data aggregation and distance encoding for interactive large multidimensional data visualization. In: INSTICC. **Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - IVAPP**. [S.l.]: SciTePress, 2018. p. 225–235. ISBN 978-989-758-289-9. ISSN 2184-4321. Citado nas páginas 33 e 41.

DHARANI, T.; AROQUIARAJ, I. L. A survey on content based image retrieval. In: IEEE. **2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering**. [S.l.], 2013. p. 485–490. Citado na página 19.

DUA, D.; GRAFF, C. **UCI Machine Learning Repository**. 2017. Disponível em: <<http://archive.ics.uci.edu/ml>>. Citado nas páginas 29 e 50.

ELLIS, G.; DIX, A. A Taxonomy of Clutter Reduction for Information Visualisation. **IEEE Transactions on Visualization and Computer Graphics**, v. 13, n. 6, p. 1216–1223, nov. 2007. ISSN 1077-2626. Citado na página 40.

GHOSH, P.; ANTANI, S.; LONG, L. R.; THOMA, G. R. Review of medical image retrieval systems and future directions. In: IEEE. **2011 24th International Symposium on Computer-Based Medical Systems (CBMS)**. [S.l.], 2011. p. 1–6. Citado na página 19.

HARALICK, R. M.; SHANMUGAM, K. *et al.* Textural features for image classification. **IEEE Transactions on systems, man, and cybernetics**, Ieee, n. 6, p. 610–621, 1973. Citado na página 25.

HAUPTMANN, A.; YAN, R.; LIN, W.; CHRISTEL, M.; WACTLAR, H. Can High-Level Concepts Fill the Semantic Gap in Video Retrieval? A Case Study With Broadcast News. **IEEE Transactions on Multimedia**, v. 9, n. 5, p. 958–966, ago. 2007. ISSN 1520-9210. Citado na página 26.

HOTELLING, H. Analysis of a complex of statistical variables into principal components. **Journal of educational psychology**, Warwick & York, v. 24, n. 6, p. 417, 1933. Citado na página 33.

HUMEAU-HEURTIER, A. Texture feature extraction methods: A survey. **IEEE Access**, v. 7, p. 8975–9000, 2019. Citado na página 25.

INSELBERG, A.; DIMSDALE, B. Parallel Coordinates for Visualizing Multi-Dimensional Geometry. In: KUNII, T. L. (Ed.). **Computer Graphics 1987**. [S.l.]: Springer Japan, 1987. p. 25–44. ISBN 978-4-431-68057-4. Citado na página 37.

JOHANSSON, J.; FORSELL, C. Evaluation of Parallel Coordinates: Overview, Categorization and Guidelines for Future Research. **IEEE Transactions on Visualization and Computer Graphics**, v. 22, n. 1, p. 579–588, jan. 2016. ISSN 1077-2626. Citado na página 37.

KALPATHY-CRAMER, J.; HERRERA, A. G. S. de; DEMNER-FUSHMAN, D.; ANTANI, S.; BEDRICK, S.; MÜLLER, H. Evaluating performance of biomedical image retrieval systems—an overview of the medical image retrieval task at imageclef 2004–2013. **Computerized Medical Imaging and Graphics**, Elsevier, v. 39, p. 55–61, 2015. Citado na página 19.

KEIM, D. A. Pixel-oriented visualization techniques for exploring very large data bases. **Journal of Computational and Graphical Statistics**, v. 5, n. 1, p. 58–77, 1996. Citado na página 38.

_____. Designing pixel-oriented visualization techniques: Theory and applications. **IEEE Transactions on visualization and computer graphics**, IEEE, v. 6, n. 1, p. 59–78, 2000. Citado na página 38.

_____. Information visualization and visual data mining. **IEEE transactions on Visualization and Computer Graphics**, v. 8, n. 1, p. 1–8, 2002. Citado nas páginas 30 e 32.

KITANOVSKI, I.; STREZOSKI, G.; DIMITROVSKI, I.; MADJAROV, G.; LOSKOVSKA, S. Multimodal medical image retrieval system. **Multimedia Tools and Applications**, v. 76, n. 2, p. 2955–2978, jan. 2017. ISSN 1573-7721. Disponível em: <<https://doi.org/10.1007/s11042-016-3261-1>>. Citado nas páginas 19 e 26.

KUO, F. Y.; SLOAN, I. H. Lifting the curse of dimensionality. **Notices of the AMS**, v. 52, n. 11, p. 1320–1328, 2005. Citado na página 33.

LI, J.; ALLINSON, N. M. Relevance Feedback in Content-Based Image Retrieval: A Survey. In: BIANCHINI, M.; MAGGINI, M.; JAIN, L. C. (Ed.). **Handbook on Neural Information Processing**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, (Intelligent Systems Reference Library). p. 433–469. ISBN 978-3-642-36657-4. Disponível em: <https://doi.org/10.1007/978-3-642-36657-4_13>. Citado na página 28.

LI, X.; YANG, J.; MA, J. Recent developments of content-based image retrieval (cbir). **Neurocomputing**, v. 452, p. 675–689, 2021. ISSN 0925-2312. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0925231220319044>>. Citado na página 27.

LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; HAYS, J.; PERONA, P.; RAMANAN, D.; DOLLÁR, P.; ZITNICK, C. L. Microsoft coco: Common objects in context. In: SPRINGER. **European conference on computer vision**. [S.l.], 2014. p. 740–755. Citado na página 51.

LIU, J.; TANG, T.; WANG, W.; XU, B.; KONG, X.; XIA, F. A Survey of Scholarly Data Visualization. **IEEE Access**, v. 6, p. 19205–19221, 2018. ISSN 2169-3536. Citado nas páginas 20 e 42.

MA, L.; LIU, X.; GAO, Y.; ZHAO, Y.; ZHAO, X.; ZHOU, C. A new method of content based medical image retrieval and its applications to ct imaging sign retrieval. **Journal of Biomedical Informatics**, v. 66, p. 148–158, 2017. ISSN 1532-0464. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1532046417300023>>. Citado na página 19.

MAATEN, L. v. d.; HINTON, G. Visualizing data using t-SNE. **Journal of machine learning research**, v. 9, n. Nov, p. 2579–2605, 2008. Citado nas páginas 9, 33 e 34.

MAATEN, L. V. D.; POSTMA, E.; HERIK, J. Van den. Dimensionality reduction: a comparative. **J Mach Learn Res**, v. 10, p. 66–71, 2009. Citado na página 33.

MATZEN, L. E.; HAASS, M. J.; DIVIS, K. M.; WANG, Z.; WILSON, A. T. Data Visualization Saliency Model: A Tool for Evaluating Abstract Data Visualizations. **IEEE Transactions on Visualization and Computer Graphics**, v. 24, n. 1, p. 563–573, jan. 2018. ISSN 1077-2626. Citado na página 40.

MCINNES, L.; HEALY, J.; MELVILLE, J. **UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction**. arXiv, 2018. Disponível em: <<https://arxiv.org/abs/1802.03426>>. Citado na página 34.

MICALLEF, L.; PALMAS, G.; OULASVIRTA, A.; WEINKAUF, T. Towards Perceptual Optimization of the Visual Design of Scatterplots. **IEEE Transactions on Visualization and Computer Graphics**, v. 23, n. 6, p. 1588–1599, jun. 2017. ISSN 1077-2626. Citado na página 40.

MOLCHANOV, V.; CHITIBOI, T.; LINSEN, L. Visual Analysis of Medical Image Segmentation Feature Space for Interactive Supervised Classification. In: BÜHLER, K.; LINSEN, L.; JOHN, N. W. (Ed.). **Eurographics Workshop on Visual Computing for Biology and Medicine**. [S.l.]: The Eurographics Association, 2015. ISBN 978-3-905674-82-8. ISSN 2070-5786. Citado nas páginas 41 e 42.

MÜLLER, H.; UNAY, D. Retrieval From and Understanding of Large-Scale Multi-modal Medical Datasets: A Review. **IEEE Transactions on Multimedia**, v. 19, n. 9, p. 2093–2104, set. 2017. ISSN 1520-9210. Citado na página 26.

NONATO, L. G.; AUPETIT, M. Multidimensional Projection for Visual Analytics: Linking Techniques with Distortions, Tasks, and Layout Enrichment. **IEEE Transactions on Visualization and Computer Graphics**, p. 1–1, 2018. ISSN 1077-2626. Citado nas páginas 33 e 35.

PANDEY, A. V.; KRAUSE, J.; FELIX, C.; BOY, J.; BERTINI, E. Towards understanding human similarity perception in the analysis of large sets of scatter plots. In: **Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems**. New York, NY, USA: Association for Computing Machinery, 2016. (CHI '16), p. 3659–3669. ISBN 9781450333627. Disponível em: <<https://doi.org/10.1145/2858036.2858155>>. Citado na página 41.

PEDRONETTE, D. C. G.; TORRES, R. d. S. Unsupervised rank diffusion for content-based image retrieval. **Neurocomputing**, Elsevier, v. 260, p. 478–489, 2017. Citado na página 19.

PIRAS, L.; GIACINTO, G. Information fusion in content based image retrieval: A comprehensive overview. **Information Fusion**, Elsevier, v. 37, p. 50–60, 2017. Citado na página 19.

POBLETE, B.; BUSTOS, B.; MENDOZA, M.; BARRIOS, J. M. Visual-semantic graphs: Using queries to reduce the semantic gap in web image retrieval. In: **Proceedings of the 19th ACM International Conference on Information and Knowledge Management**. New York, NY, USA: Association for Computing Machinery, 2010. (CIKM '10), p. 1553–1556. ISBN 9781450300995. Disponível em: <<https://doi.org/10.1145/1871437.1871670>>. Citado na página 26.

POWERS, D. M. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. **arXiv preprint arXiv:2010.16061**, 2020. Citado na página 27.

RAUBER, P. E.; FALCÃO, A. X.; TELEA, A. C. Projections as visual aids for classification system design. **Information Visualization**, p. 1473871617713337, jun. 2017. ISSN 1473-8716. Disponível em: <<https://doi.org/10.1177/1473871617713337>>. Citado na página 42.

REDA, K.; GONZÁLEZ, A.; LEIGH, J.; PAPKA, M. E. Tell me what do you see: Detecting perceptually-separable visual patterns via clustering of image-space features in visualizations. In: **2015 IEEE Conference on Visual Analytics Science and Technology (VAST)**. [S.l.: s.n.], 2015. p. 211–212. Citado na página 41.

REES, D.; LARAMEE, R. S. A survey of information visualization books. In: WILEY ONLINE LIBRARY. **Computer Graphics Forum**. [S.l.], 2019. v. 38, n. 1, p. 610–646. Citado na página 33.

RODRIGUES, D. S.; NUNES, F. L. Visualização de Informação aplicada à compreensão de resultados de Recuperação de Imagens Baseada em Conteúdo. **iSys-Revista Brasileira de Sistemas de Informação**, v. 7, n. 1, p. 89–111, 2014. Citado na página 41.

SACHA, D.; SEDLMAIR, M.; ZHANG, L.; LEE, J. A.; PELTONEN, J.; WEISKOPF, D.; NORTH, S. C.; KEIM, D. A. What you see is what you can change: Human-centered machine learning by interactive visualization. **Neurocomputing**, v. 268, p. 164–175, dez. 2017. ISSN 09252312. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0925231217307609>>. Citado nas páginas 20 e 42.

SALAU, A. O.; JAIN, S. Feature extraction: A survey of the types, techniques, applications. In: **2019 International Conference on Signal Processing and Communication (ICSC)**. [S.l.: s.n.], 2019. p. 158–164. Citado na página 25.

SAMMON, J. W. A nonlinear mapping for data structure analysis. **IEEE Transactions on computers**, Ieee, v. 100, n. 5, p. 401–409, 1969. Citado na página 33.

SHNEIDERMAN, B. The eyes have it: A task by data type taxonomy for information visualizations. In: **The Craft of Information Visualization**. [S.l.]: Elsevier, 2003. p. 364–371. Citado na página 31.

SMEULDERS, A. W. M.; WORRING, M.; SANTINI, S.; GUPTA, A.; JAIN, R. Content-based image retrieval at the end of the early years. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 22, n. 12, p. 1349–1380, dez. 2000. ISSN 0162-8828. Citado nas páginas 25, 26 e 27.

SÖDERKVIST, O. **Computer vision classification of leaves from swedish trees**. 2001. Citado na página 51.

SUPIANTO, A. A.; CHRISTYAWAN, T. Y.; HAFIS, M.; HAYASHI, Y.; HIRASHIMA, T.; HASANAH, N. Feature dimensionality reduction for visualization and clustering on learning process data. In: IEEE. **2019 International Conference on Sustainable Information Engineering and Technology (SIET)**. [S.l.], 2019. p. 84–89. Citado na página 33.

TANG, J.; LIU, J.; ZHANG, M.; MEI, Q. Visualizing large-scale and high-dimensional data. In: **Proceedings of the 25th International Conference on World Wide Web**. [S.l.]: International World Wide Web Conferences Steering Committee, 2016. p. 287–297. Citado na página 33.

- TATU, A.; S, F. M.; FÄRBER, I.; BERTINI, E.; SCHRECK, T.; SEIDL, T.; KEIM, D. Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In: **Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on**. [S.l.]: IEEE, 2012. p. 63–72. Citado nas páginas 35 e 42.
- TORRES, R. da S.; FALCAO, A. X. Content-based image retrieval: theory and applications. **RITA**, v. 13, n. 2, p. 161–185, 2006. Citado nas páginas 19, 23, 24 e 26.
- TRAINA, A.; BALAN, A.; BORTOLOTTI, L.; TRAINA, C. Content-based image retrieval using approximate shape of objects. In: **Proceedings. 17th IEEE Symposium on Computer-Based Medical Systems**. [S.l.: s.n.], 2004. v. 1, n. 1, p. 91–96. Citado na página 51.
- TRAINA, A.; MARQUES, J.; TRAINA, C. Fighting the semantic gap on cbir systems through new relevance feedback techniques. In: **19th IEEE Symposium on Computer-Based Medical Systems (CBMS'06)**. [S.l.: s.n.], 2006. p. 881–886. Citado nas páginas 27 e 28.
- TRAINA, A. J. M.; BÊDO, M. V. N.; SANTOS, L. F. D.; CARVALHO, L. O.; PEDROSA, G. V.; COSTA, A. F.; JUNIOR, C. T. Making content-based medical image retrieval systems worth for computer-aided diagnosis: from theory to application. **Medical image analysis and informatics: computer-aided diagnosis and therapy**, 2017. Citado na página 24.
- WANG, Y.; HUANG, H.; RUDIN, C.; SHAPOSHNIK, Y. Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization. **J. Mach. Learn. Res.**, v. 22, n. 201, p. 1–73, 2021. Citado na página 33.
- WARD, M. O.; GRINSTEIN, G.; KEIM, D. **Interactive Data Visualization: Foundations, Techniques, and Applications, Second Edition**. 2nd. ed. Natick, MA, USA: A. K. Peters, Ltd., 2015. ISBN 978-1-4822-5737-3. Citado nas páginas 9, 29, 30, 31, 32, 35, 36 e 37.
- WATTENBERG, M.; VIÉGAS, F.; JOHNSON, I. How to use t-sne effectively. **Distill**, 2016. Disponível em: <<http://distill.pub/2016/misread-tsne>>. Citado nas páginas 9 e 34.
- WILKINSON, L. **The grammar of graphics**. [S.l.]: Springer Science & Business Media, 2013. Citado na página 48.
- ZEZULA, P.; AMATO, G.; DOHNAL, V.; BATKO, M. **Similarity search: the metric space approach**. [S.l.]: Springer Science & Business Media, 2006. v. 32. Citado nas páginas 25 e 26.
- ZHANG, J.; YANG, Y.; ZHUO, L.; DIAO, M. Image retrieval method using visual query suggestion and relevance feedback. In: **2012 International Conference on Wireless Communications and Signal Processing (WCSP)**. [S.l.: s.n.], 2012. p. 1–6. Citado na página 26.

CÓDIGOS-FONTE

Código-fonte 1 – Extração de características do conjunto val2017

```
1: import math
2: import os
3:
4: import numpy as np
5: import pandas as pd
6: from skimage import exposure, img_as_float
7: from skimage import io
8: from tqdm import tqdm
9:
10: imgs_path = 'static/img/val2017'
11: static_path = 'val2017'
12: all_images = os.listdir(imgs_path)
13:
14: n_bins = 5
15: n_images = 500
16:
17: def to_color_img(img):
18:     return np.array([[cell, cell, cell] for cell in row] for
19:                     row in img)
20:
21: def four_divide(img):
22:     shape = img.shape
23:     center_x = math.floor(shape[0] / 2)
24:     center_y = math.floor(shape[1] / 2)
25:     return [
```

```
26:         img[:center_x, :center_y, :],
27:         img[center_x:, :center_y, :],
28:         img[:center_x, center_y:, :],
29:         img[center_x:, center_y:, :]
30:     ]
31:
32:
33: def get_histogram(img):
34:     red_hist = exposure.histogram(img_as_float(img[:, :, 0]),
35:     nbins=n_bins, normalize=True)[0]
36:     green_hist = exposure.histogram(img_as_float(img[:, :, 1]),
37:     nbins=n_bins, normalize=True)[0]
38:     blue_hist = exposure.histogram(img_as_float(img[:, :, 2]),
39:     nbins=n_bins, normalize=True)[0]
40:     return red_hist, green_hist, blue_hist
41:
42:
43: def get_dimensions(img_path):
44:     img = io.imread(os.path.join(imgs_path, img_path))
45:     if img.ndim == 2:
46:         img = to_color_img(img)
47:
48:     features = {
49:         'image': static_path + '/' + img_path
50:     }
51:
52:     quadrants = four_divide(img)
53:     for q, quadrant in enumerate(quadrants):
54:         red_hist, green_hist, blue_hist = get_histogram(img)
55:         for i in range(n_bins):
56:             features[f'hist_q{q}_red_{i}'] = red_hist[i]
57:             features[f'hist_q{q}_green_{i}'] = green_hist[i]
58:             features[f'hist_q{q}_blue_{i}'] = blue_hist[i]
59:
60:     return features
61:
62: print("Extracting Features")
63: data = pd.DataFrame([get_dimensions(img_path) for img_path in
64:     tqdm(all_images[:n_images])])
65: data.to_csv('data/2_extracted_features/val.csv', index=False)
```

Código-fonte 2 – Extração de características do conjunto Swedish Leaves

```
1: import os
2:
3: import mahotas.demos
4: import pandas as pd
5: from tqdm import tqdm
6:
7: imgs_path = "static/img/SwedishLeaves"
8: static_path = 'SwedishLeavesJPG'
9: all_images = os.listdir(imgs_path)
10: n_images = 1125
11: zernike_radius = 128
12:
13:
14: def get_dimensions(img_path):
15:     # img_path = os.path.join(imgs_path, img_path)
16:     full_img_path = imgs_path + "/" + img_path
17:
18:     # loading image
19:     img = mahotas.imread(full_img_path, as_grey=True)
20:
21:     dict_feature = get_haralick_features(img)
22:
23:     dict_feature['class'] = img_path.split('\n')[0]
24:     dict_feature['image'] = f"{static_path}/{img_path[:-3]}.jpg"
25:     return dict_feature
26:
27:
28: def get_haralick_features(img):
29:     # adding gaussian filter
30:     img = mahotas.gaussian_filter(img, 4)
31:
32:     # setting threshold
33:     threshed = (img > img.mean())
34:
35:     # making is labeled image
36:     labeled, n = mahotas.label(threshed)
37:
38:     # getting haralick features
39:     h_feature = mahotas.features.haralick(labeled)
40:
```

```
41:     dict_feature = dict(enumerate(h_feature.flatten(), 1))
42:     return dict_feature
43:
44:
45: print("Calculating Dimensions")
46: data = pd.DataFrame([get_dimensions(img_path) for img_path in
47:     tqdm(all_images[:n_images])])
48: data.to_csv('data/2_extracted_features/swedishleaves.csv',
49:     index=False)
```

Código-fonte 3 – Extração de características do conjunto SegMRF

```
1: import os
2: import mahotas.demos
3: import pandas as pd
4: from tqdm import tqdm
5:
6: imgs_path = "static/img/SegMRF"
7: static_path = 'SegMRF'
8: all_images = os.listdir(imgs_path)
9: n_images = 704
10: zernike_radius = 128
11:
12:
13: def get_dimensions(img_path):
14:     # img_path = os.path.join(imgs_path, img_path)
15:     full_img_path = imgs_path + "/" + img_path
16:
17:     # loading image
18:     img = mahotas.imread(full_img_path, as_grey=True)
19:
20:     dict_feature = get_zernike_features(img)
21:
22:     dict_feature['class'] = img_path.split('_')[0]
23:     dict_feature['image'] = f"{static_path}/{img_path}"
24:     return dict_feature
25:
26:
27: def get_zernike_features(img):
28:     features = mahotas.features.zernike_moments(img,
29:         zernike_radius)
```

```
29:     dict_feature = dict(enumerate(features, 1))
30:     return dict_feature
31:
32:
33: def get_haralick_features(img):
34:     # adding gaussian filter
35:     img = mahotas.gaussian_filter(img, 4)
36:
37:     # setting threshold
38:     threshed = (img > img.mean())
39:
40:     # making is labeled image
41:     labeled, n = mahotas.label(threshed)
42:
43:     # getting haralick features
44:     h_feature = mahotas.features.haralick(labeled)
45:
46:     dict_feature = dict(enumerate(h_feature.flatten(), 1))
47:     return dict_feature
48:
49:
50: print("Calculating Dimensions")
51: data = pd.DataFrame([get_dimensions(img_path) for img_path in
52:                      tqdm(all_images[:n_images])])
53: data.to_csv('data/2_extracted_features/segmrf.csv', index=False
54:            )
```

Código-fonte 4 – Definição de casos teste na redução de dimensionalidade

```
1: from models.dataset_definition import DatasetDefinition
2: from models.tsne_config import TsneConfig
3: from models.umap_config import UmapConfig
4: from typing import List
5:
6: umap_variations = {
7:     "default": UmapConfig(
8:         umap_n_neighbors=15,
9:         umap_metric='euclidean',
10:        umap_min_dist=0.1,
11:        umap_spread=1.0,
12:        umap_random_state=42
```

```
13:     ),
14:     "umap_n_neighbors_2": UmapConfig(
15:         umap_n_neighbors=2,
16:         umap_metric='euclidean',
17:         umap_min_dist=0.1,
18:         umap_spread=1.0,
19:         umap_random_state=42
20:     ),
21:     "umap_n_neighbors_50": UmapConfig(
22:         umap_n_neighbors=50,
23:         umap_metric='euclidean',
24:         umap_min_dist=0.1,
25:         umap_spread=1.0,
26:         umap_random_state=42
27:     ),
28:     "umap_n_neighbors_100": UmapConfig(
29:         umap_n_neighbors=100,
30:         umap_metric='euclidean',
31:         umap_min_dist=0.1,
32:         umap_spread=1.0,
33:         umap_random_state=42
34:     ),
35:     "umap_min_dist_0_01": UmapConfig(
36:         umap_n_neighbors=15,
37:         umap_metric='euclidean',
38:         umap_min_dist=0.01,
39:         umap_spread=1.0,
40:         umap_random_state=42
41:     ),
42:     "umap_min_dist_0_5": UmapConfig(
43:         umap_n_neighbors=15,
44:         umap_metric='euclidean',
45:         umap_min_dist=0.5,
46:         umap_spread=1.0,
47:         umap_random_state=42
48:     ),
49:     "umap_spread_0_5": UmapConfig(
50:         umap_n_neighbors=15,
51:         umap_metric='euclidean',
52:         umap_min_dist=0.1,
53:         umap_spread=0.5,
54:         umap_random_state=42
```

```
55:     ),
56:     "umap_spread_3": UmapConfig(
57:         umap_n_neighbors=15,
58:         umap_metric='euclidean',
59:         umap_min_dist=0.1,
60:         umap_spread=3.0,
61:         umap_random_state=42
62:     ),
63: }
64:
65: tsne_variations = {
66:     "default": TsneConfig(
67:         tsne_perplexity=25.0,
68:         tsne_learning_rate=500.0,
69:         tsne_n_iter=1000,
70:         tsne_random_state=42,
71:     ),
72:     "tsne_perplexity_5": TsneConfig(
73:         tsne_perplexity=5.0,
74:         tsne_learning_rate=200.0,
75:         tsne_n_iter=1000,
76:         tsne_random_state=42
77:     ),
78:     "tsne_perplexity_50": TsneConfig(
79:         tsne_perplexity=50.0,
80:         tsne_learning_rate=200.0,
81:         tsne_n_iter=1000,
82:         tsne_random_state=42
83:     ),
84:     "tsne_perplexity_100": TsneConfig(
85:         tsne_perplexity=100.0,
86:         tsne_learning_rate=200.0,
87:         tsne_n_iter=1000,
88:         tsne_random_state=42
89:     ),
90:     "tsne_learning_rate_10": TsneConfig(
91:         tsne_perplexity=30.0,
92:         tsne_learning_rate=10.0,
93:         tsne_n_iter=1000,
94:         tsne_random_state=42
95:     ),
96:     "tsne_learning_rate_500": TsneConfig(
```

```
97:         tsne_perplexity=30.0,
98:         tsne_learning_rate=500.0,
99:         tsne_n_iter=1000,
100:        tsne_random_state=42
101:    ),
102:    "tsne_learning_rate_1000": TsneConfig(
103:        tsne_perplexity=30.0,
104:        tsne_learning_rate=1000.0,
105:        tsne_n_iter=1000,
106:        tsne_random_state=42
107:    ),
108:    "tsne_n_iter_250": TsneConfig(
109:        tsne_perplexity=30.0,
110:        tsne_learning_rate=200.0,
111:        tsne_n_iter=250,
112:        tsne_random_state=42
113:    ),
114:    "tsne_n_iter_5000": TsneConfig(
115:        tsne_perplexity=30.0,
116:        tsne_learning_rate=200.0,
117:        tsne_n_iter=5000,
118:        tsne_random_state=42
119:    ),
120:    "tsne_n_iter_10000": TsneConfig(
121:        tsne_perplexity=30.0,
122:        tsne_learning_rate=200.0,
123:        tsne_n_iter=10000,
124:        tsne_random_state=42
125:    ),
126: }
127:
128: definitions: List[DatasetDefinition] = [
129:     DatasetDefinition(
130:         name="iris",
131:         exhibition_name="Iris",
132:         has_class=True,
133:         umap_variations=umap_variations,
134:         tsne_variations=tsne_variations,
135:     ),
136:     DatasetDefinition(
137:         name="val",
138:         exhibition_name="Val 2017",
```

```
139:         has_class=False ,
140:         umap_variations=umap_variations ,
141:         tsne_variations=tsne_variations ,
142:     ),
143:     DatasetDefinition(
144:         name="synthetic",
145:         exhibition_name="Synthetic",
146:         has_class=False ,
147:         umap_variations=umap_variations ,
148:         tsne_variations=tsne_variations ,
149:     ),
150:     DatasetDefinition(
151:         name="segmrf",
152:         exhibition_name="Seg MRF",
153:         has_class=True ,
154:         umap_variations=umap_variations ,
155:         tsne_variations=tsne_variations ,
156:     ),
157:     DatasetDefinition(
158:         name="swedishleaves",
159:         exhibition_name="Swedish Leaves",
160:         has_class=True ,
161:         umap_variations=umap_variations ,
162:         tsne_variations=tsne_variations ,
163:     ),
164:     DatasetDefinition(
165:         name="default",
166:         exhibition_name="Default",
167:         has_class=False ,
168:         umap_variations=umap_variations ,
169:         tsne_variations=tsne_variations ,
170:     ),
171: ]
```

Código-fonte 5 – Definição do dataset sintético de imagens coloridas

```
1: import math
2: import numpy as np
3: import pandas as pd
4: from imageio import imsave
5: from numpy import zeros, uint8
6:
```

```
7: image_width = 100
8: image_height = 100
9: number_of_images = 300
10: number_of_bins = 10
11: imgs_path = 'static/img/synthetic'
12: static_path = 'synthetic'
13: gradient_radius = math.sqrt((image_width / 2) ** 2 + (
    image_height / 2) ** 2)
14:
15:
16: def calculate_my_histogram(img, n_bins):
17:     h = len(img)
18:     w = len(img[0])
19:     bucket_size = uint8(255 / n_bins)
20:     buckets = np.zeros(n_bins)
21:     for y in range(h):
22:         for x in range(w):
23:             pixel = img[y, x]
24:             bucket = min(uint8(pixel / bucket_size), n_bins -
1)
25:             buckets[bucket] += 1
26:
27:     buckets = buckets / np.sum(buckets)
28:     return buckets
29:
30:
31: def gen_image_with_one_color(color):
32:     img = zeros([image_width, image_height, 3], dtype=uint8)
33:     for y in range(image_height):
34:         for x in range(image_width):
35:             img[y, x] = color
36:     return img
37:
38:
39: def gen_image_with_gradient(color1, color2, degree):
40:     direction_x, direction_y = math.cos(degree), math.sin(
degree)
41:     lerp_start_x, lerp_start_y = -direction_x * gradient_radius
+ image_width / 2, -direction_y * gradient_radius +
image_height / 2
42:     img = zeros([image_width, image_height, 3], dtype=uint8)
43:     for y in range(image_height):
```

```
44:         for x in range(image_width):
45:             lerp_factor = ((x - lerp_start_x) * direction_x + (
                y - lerp_start_y) * direction_y) / (2 * gradient_radius)
46:             img[y, x] = lerp_factor * color2 + (1 - lerp_factor
                ) * color1
47:         return img
48:
49:
50: def random_color():
51:     return np.trunc(np.random.random(3) * 255).astype('uint8')
52:
53:
54: def four_divide(img):
55:     shape = img.shape
56:     center_x = math.floor(shape[0] / 2)
57:     center_y = math.floor(shape[1] / 2)
58:     return [
59:         img[:center_x, :center_y, :],
60:         img[center_x:, :center_y, :],
61:         img[:center_x, center_y:, :],
62:         img[center_x:, center_y:, :]
63:     ]
64:
65:
66: def get_histogram(img):
67:     red_hist = calculate_my_histogram(img[:, :, 0],
        number_of_bins)
68:     green_hist = calculate_my_histogram(img[:, :, 1],
        number_of_bins)
69:     blue_hist = calculate_my_histogram(img[:, :, 2],
        number_of_bins)
70:     return red_hist, green_hist, blue_hist
71:
72:
73: def get_dimensions(name, img):
74:     features = {
75:         'image': static_path + '/' + name + '.png'
76:     }
77:
78:     quadrants = four_divide(img)
79:     for q, quadrant in enumerate(quadrants):
80:         red_hist, green_hist, blue_hist = get_histogram(
```

```

    quadrant)
81:         for i in range(number_of_bins):
82:             features[f'hist_q{q}_red_{i}'] = red_hist[i]
83:             features[f'hist_q{q}_green_{i}'] = green_hist[i]
84:             features[f'hist_q{q}_blue_{i}'] = blue_hist[i]
85:
86:         return features
87:
88:
89: def create_image_with_features(name, color1):
90:     color2 = random_color()
91:     degree = np.random.rand() * 2 * math.pi
92:     img = gen_image_with_gradient(color1, color2, degree)
93:     #img = gen_image_with_one_color(random_color())
94:     imsave(f'{imgs_path}/{name}.png', img)
95:     return get_dimensions(name, img)
96:
97:
98: all_images = [f'img_{i}' for i in range(number_of_images)]
99:
100: print("Calculating Dimensions")
101: data = pd.DataFrame([create_image_with_features(name, np.array
    ([255, 255, 255], uint8)) for name in all_images])
102: data.to_csv('data/2_extracted_features/synthetic.csv', index=
    False)

```

Código-fonte 6 – Cálculo de silhueta

```

1: import json
2: import pandas as pd
3: import matplotlib.pyplot as plt
4: import imageio as iio
5: import matplotlib as mpl
6: import numpy as np
7:
8: mpl.rcParams['text.antiialiased'] = False
9:
10: from models.definitions import definitions
11:
12: def silhouette_measure(data: pd.DataFrame, prefix: str) -> dict
    :
13:     n_points = data.shape[0]

```

```
14:     square_side = 10
15:     square_area = square_side * square_side
16:     ideal_visual_elements = square_area * n_points
17:
18:     fig = plt.figure(dpi=1000, figsize=(1, 1))
19:     plt.scatter(data[f"{prefix}_x"], data[f"{prefix}_y"], s=((
square_side * (72. / fig.dpi)) ** 2), edgecolors='none',
20:                 antialiased=False, marker='s', c='black')
21:
22:     plt.axis('off')
23:
24:     plt.savefig("temp.png", dpi=fig.dpi)
25:     plt.close(fig)
26:     im = iio.imread("temp.png", as_gray=True)
27:
28:     visual_elements = np.sum(np.array(im) < 254)
29:     return {
30:         "total_contado": int(visual_elements),
31:         "total_estimado": ideal_visual_elements,
32:         "total_pixels": im.shape[0] * im.shape[1],
33:         "occlusion_rate": 1 - (visual_elements /
ideal_visual_elements),
34:     }
35:
36:
37: for definition in definitions:
38:     print(f"processing {definition.name}")
39:     csv = f"data/3_reduced_dimensions/{definition.name}.csv"
40:     data = pd.read_csv(csv)
41:
42:     tsne_variations = [f"tsne_{key}" for key in definition.
tsne_variations]
43:     umap_variations = [f"umap_{key}" for key in definition.
umap_variations]
44:     all_variations = umap_variations + tsne_variations
45:     metadata = {}
46:     for variation in all_variations:
47:         metadata[variation] = silhouette_measure(data,
variation)
48:
49:     output_filename = f"data/4_metadata/{definition.name}.json"
50:     with open(output_filename, 'w') as f:
```

```
51:         json.dump(metadata, f)
```

APÊNDICE
B

DADOS BRUTOS

Tabela 7 – Valores brutos de taxas de oclusão

	iris	val2017	SegMRF	Synthetic	Swedish Leaves
tsne_default	0.0	0.04913999999999996	0.5440056818181818	0.018000000000000016	0.4734933333333333
tsne_perplexity_5	0.05893333333333328	0.18322000000000005	0.5506107954545454	0.14316666666666666	0.5487555555555556
tsne_perplexity_50	0.14653333333333333	0.05954000000000004	0.39651988636363633	0.07169999999999999	0.43509333333333333
tsne_perplexity_100	0.05726666666666669	0.06591999999999998	0.34129261363636365	0.11606666666666665	0.57845333333333333
tsne_learning_rate_10	0.13980000000000004	0.054300000000000015	0.5011079545454545	0.04293333333333338	0.42965333333333333
tsne_learning_rate_500	0.59593333333333333	0.06008000000000002	0.5760795454545454	0.06340000000000001	0.44594666666666667
tsne_learning_rate_1000	0.0	0.04468000000000005	0.5255681818181819	0.21240000000000003	0.52207111111111112
tsne_n_iter_250	0.13006666666666666	0.7651	0.8811363636363636	0.4494	0.60365333333333334
tsne_n_iter_5000	0.21106666666666662	0.03849999999999998	0.5205113636363636	0.04586666666666661	0.61081777777777778
tsne_n_iter_10000	0.21106666666666662	0.03849999999999998	0.5205113636363636	0.04586666666666661	0.62667555555555555
umap_default	0.21126666666666667	0.14659999999999995	0.7658238636363637	0.10076666666666667	0.72914666666666666
umap_n_neighbors_2	0.605	0.69408	0.6674431818181819	0.62943333333333333	0.69901333333333333
umap_n_neighbors_50	0.23060000000000003	0.07889999999999997	0.7310227272727272	0.05506666666666671	0.66626666666666667
umap_n_neighbors_100	0.08193333333333333	0.08386000000000005	0.5545170454545454	0.04279999999999995	0.68930666666666667
umap_min_dist_0_01	0.34306666666666663	0.19221999999999995	0.8425568181818182	0.1775	0.78099555555555556
umap_min_dist_0_5	0.01846666666666663	0.05034000000000005	0.5547301136363636	0.031399999999999983	0.35598222222222222
umap_spread_0_5	0.21140000000000003	0.13946000000000003	0.7997159090909091	0.10463333333333336	0.72272888888888889
umap_spread_3	0.05120000000000002	0.10231999999999997	0.6857102272727273	0.05473333333333333	0.66664

	iris	val2017	SegMRF	Synthetic	Swedish Leaves
tsne_default	0,00%	4,91%	54,40%	1,80%	47,35%
tsne_perplexity_5	5,89%	18,32%	55,06%	14,32%	54,88%
tsne_perplexity_50	14,65%	5,95%	39,65%	7,17%	43,51%
tsne_perplexity_100	5,73%	6,59%	34,13%	11,61%	57,85%
tsne_learning_rate_10	13,98%	5,43%	50,11%	4,29%	42,97%
tsne_learning_rate_500	59,59%	6,01%	57,61%	6,34%	44,59%
tsne_learning_rate_1000	0,00%	4,47%	52,56%	21,24%	52,21%
tsne_n_iter_250	13,01%	76,51%	88,11%	44,94%	60,37%
tsne_n_iter_5000	21,11%	3,85%	52,05%	4,59%	61,08%
tsne_n_iter_10000	21,11%	3,85%	52,05%	4,59%	62,67%
umap_default	21,13%	14,66%	76,58%	10,08%	72,91%
umap_n_neighbors_2	60,50%	69,41%	66,74%	62,94%	69,90%
umap_n_neighbors_50	23,06%	7,89%	73,10%	5,51%	66,63%
umap_n_neighbors_100	8,19%	8,39%	55,45%	4,28%	68,93%
umap_min_dist_0_01	34,31%	19,22%	84,26%	17,75%	78,10%
umap_min_dist_0_5	1,85%	5,03%	55,47%	3,14%	35,60%
umap_spread_0_5	21,14%	13,95%	79,97%	10,46%	72,27%
umap_spread_3	5,12%	10,23%	68,57%	5,47%	66,66%

Tabela 8 – Taxas de oclusão por caso de teste em cada conjunto de dados em ordem de casos

