Águeda Augusta Fortes Piedade Ramos
Licenciada em Engenharia Informática

Parametrização da Estrutura de Dados Métrica RLC

Dissertação para obtenção do Grau de Mestre em Engenharia Informática

Orientadora:
Margarida Paula Neves Mamede, Professora Auxiliar, FCT/UNL

Júri:

Presidente:	Prof. Doutor Pedro Abílio Duarte de Medeiros
Arguente:	Prof. Doutor João Pedro Guerreiro Neto
Vogal:	Prof. Doutora Margarida Paula Neves Mamede

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Junho de 2012

Parametrização da Estrutura de Dados Métrica RLC

AGRADECIMENTOS

À minha orientadora, Professora Margarida Mamede, que esteve sempre presente e atenciosa. Agradeço-lhe a paciência, o apoio, a ajuda e os conselhos dados ao longo deste trabalho. Há pessoas que passam nas nossas vidas e que deixam um "rasto de coisas boas". A Professora Margarida é com certeza uma dessas pessoas na minha vida.

Ao Humberto e à Rita, obrigada pelo companheirismo. Não foram tempos fáceis, principalmente devido à falta de tempo para gerir tudo, mas vocês estiveram sempre presentes a apoiar. Digo-vos que seria menos feliz sem a vossa presença na minha vida!

Aos meus pais e irmãos que, mesmo estando a milhas de distância, me apoiaram e me motivaram nas horas de desânimo e cansaço.

À Professora Fernanda Barbosa pelo material que disponibilizou e pelas críticas construtivas durante o processo de preparação desta dissertação, que ajudaram na sua elaboração.

Ao Pedro Chambel por ter cedido o protótipo da tese para extracção dos vectores de pesos para a base de dados de imagens de rostos.

RESUMO

Em muitas aplicações, existe a necessidade de pesquisar objectos semelhantes ou próximos de um objecto dado. Exemplos desses objectos incluem imagens médicas ou de rostos, sequências de proteínas ou de ADN, palavras de uma língua ou trajectórias de furacões. As pesquisas por proximidade podem ser formalizadas no contexto de espaços métricos, onde a semelhança entre dois elementos do domínio é medida através da função de distância. Como, em geral, as bases de dados possuem muitos elementos e o cálculo da distância entre dois objectos é uma operação cara, foram desenvolvidas estruturas de dados que tentam minimizar o número de distâncias calculadas durante as pesquisas deste tipo, designadas por estruturas de dados métricas.

Nesta tese, faz-se um levantamento dos espaços métricos mais frequentemente usados nos testes de desempenho das estruturas de dados métricas. Depois, descreve-se a evolução da estrutura de dados métrica *Recursive Lists of Clusters* (RLC), caracterizando-se as suas variantes.

O desempenho da RLC, tal como o de qualquer estrutura de dados métrica parametrizada, depende fortemente dos valores dos seus parâmetros. O problema é que os valores mais adequados a cada espaço métrico têm sido encontrados por observação de resultados experimentais, tornando o processo de parametrização pouco fiável e muito moroso. Para atacar esta questão, propõe-se uma nova variante da RLC cujos valores dos parâmetros dependem de valores extraídos do espaço métrico. Os resultados experimentais, que envolvem quinze espaços métricos de diferentes domínios, mostram que a nova variante é mais eficiente do que a anterior.

Termos chave: estruturas de dados, espaços métricos, pesquisas por proximidade, métodos de indexação.

ABSTRACT

In many applications, there is the need to search objects that are similar or close to a given one. Examples of these objects include medical or face images, protein or DNA sequences, natural language words or hurricane trajectories. Proximity searches can be formalised in the metric space setting, where similarity between two elements of the domain is measured through the distance function. As, in general, databases have large amounts of information and the cost of evaluating distances is very high, several data structures, called metric data structures, have been developed in order to minimise the number of distance computations performed in searches of this type.

In this thesis, we survey the metric spaces that are most commonly used to evaluate the performance of metric data structures. Then, we describe the evolution of the Recursive Lists of Clusters (RLC) metric data structure, characterising its variants.

The RLC performance, like that of any parameterized metric data structure, depends strongly on the values of its parameters. The problem is that the most suitable values for each metric space have been found by observation of experimental results, which makes this process unreliable and very time consuming. To tackle this issue, a new RLC version is proposed, where the parameter values depend on values extracted from the metric space. The experimental results, which involve fifteen metric spaces of different domains, show that the new variant outperforms the previous one.

Keywords: data structures, metric spaces, proximity searching, indexing methods.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABELAS

1 INTRODUÇÃO

Os sistemas de gestão de bases de dados foram desenvolvidos, inicialmente, com o objectivo de facilitar e agilizar pesquisas exactas. Com a evolução da informação e das tecnologias da informação, tornou-se necessário armazenar e consultar, adequadamente, dados mais complexos e não estruturados, tais como imagens, trechos de áudio, informações genéticas e séries temporais, entre outros. Tornou-se então necessário desenvolver novos algoritmos de pesquisa, não utilizados nos modelos anteriores de bases de dados, onde o utilizador pode não pretender um valor exacto mas sim um valor aproximado, a partir de um outro, dado como entrada. Por exemplo, em aplicações envolvendo imagens médicas, pode ser necessário descobrir imagens parecidas com uma outra e assim comparar casos semelhantes que possam já ter ocorrido.

1.1 CONTEXTO

Um tipo de consulta que se aplica de maneira geral a muitos tipos de dados complexos é a consulta por semelhança, também denominada por pesquisa por proximidade. A proximidade ou semelhança entre elementos de uma base de dados é medida através de uma função de distância, que calcula a semelhança entre pares de elementos e retorna um valor que é tanto maior quanto mais distante um elemento estiver do outro [Mamede 2005].

A *consulta por proximidade* (*range query*), também designada por *consulta por abrangência*, e a *consulta dos k-vizinhos mais próximos* (*k-nearest neighbour query* ou *k-NN*) são os tipos mais comuns de pesquisas por semelhança [Chávez et al. 2001]. Neste tese serão abordadas as consultas por proximidade.

A consulta por proximidade recebe um elemento q do domínio dos dados, chamado centro da consulta, e um limite máximo de semelhança r_q. Retorna todos os elementos da base de dados cujas distâncias a q não excedem o limite máximo r_q especificado [Amato et al. 2003]. Um exemplo de consulta por proximidade é "Seleccionem-se todas as imagens que distam da imagem A no máximo dez unidades".

Já uma consulta aos k-vizinhos mais próximos recebe um elemento do domínio de dados e o número de vizinhos desejados, obtendo-se como resposta k elementos da base de dados que são os

mais próximos do elemento dado [Amato el al. 2003]. Um exemplo deste tipo de consulta é "Seleccionem-se as dez imagens mais semelhantes à imagem A".

Para que uma função seja utilizada para medir a semelhança entre objectos, é necessário que ela tenha determinadas propriedades, fazendo com que seja classificada como uma *função de distância* ou *métrica* [Amato et al. 2003]. Ao conjunto dos elementos com uma função de distância chama-se *espaço métrico*.

O problema das pesquisas por proximidade em espaços métricos tem aplicação em inúmeros domínios, como: multimédia (por exemplo, na pesquisa de imagens, vídeo e áudio), bioinformática (como, por exemplo, na pesquisa de sequências de ADN ou de proteínas), séries temporais, reconhecimento da fala (onde se pesquisam padrões vocais semelhantes) ou, ainda, na detecção de cópias (pesquisando padrões semelhantes em bases de dados de documentos).

O resultado de uma pesquisa por proximidade é um conjunto de objectos, guardados numa base de dados, que mais se aproximam do objecto pesquisado. Uma vez que as pesquisas são feitas num domínio onde existe uma função que determina o grau de semelhança entre os objectos, o resultado de uma pesquisa poderia ser processado, calculando, para cada elemento da base de dados, a sua distância ao objecto pesquisado. No entanto, como, normalmente, as bases de dados apresentam grandes dimensões e os dados são complexos, este processo seria muito pouco eficiente.

Para evitar processamentos exaustivos, e agilizar as pesquisas por proximidade, surgiram estruturas de dados, designadas por *estruturas de dados métricas* [Zezula et al. 2006]. Estas procuram minimizar o número de distâncias calculadas aquando de uma pesquisa. Para tal, particionam o espaço em regiões, organizando os elementos da base de dados com base na função de distância, para que a pesquisa só seja realizada em algumas regiões [Zezula et al. 2006].

A primeira proposta de estrutura de dados métrica foi feita em 1973 por Burkhard e Keller [Burkhard e Keller 1973]. Seguiram-se muitas outras. Apesar das suas muitas diferenças, todas se baseiam nas propriedades da função de distância para, durante uma pesquisa, seleccionar ou descartar elementos da base de dados sem calcular as suas distância ao objecto dado na consulta.

As estruturas de dados métricas são classificadas de várias formas, tendo em conta alguns aspectos, como, por exemplo, se asseguram actualizações à base de dados após o carregamento da estrutura. Se tal acontece, são classificadas como *dinâmicas*; no caso contrário, dizem-se *estáticas*. Podem ser implementadas em memória central ou em memória secundária. As estruturas de dados que requerem parâmetros de entrada aquando da sua construção são *parametrizadas*.

1.2 MOTIVAÇÃO

Ao longo dos últimos anos, foram propostas diversas estruturas de dados métricas, todas com o objectivo de minimizar o número de distâncias calculadas nas operações de pesquisa. Isto porque, num espaço métrico, a semelhança entre objectos é medida através da função de distância, cuja

complexidade depende da natureza dos dados envolvidos. Essa função pode ser computacionalmente muito cara, como, por exemplo, a que calcula a distância entre dois romances.

A RLC – *Recursive Lists of Clusters* – é uma estrutura de dados métrica, desenvolvida no Departamento de Informática da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa. É parametrizada, dinâmica e implementada em memória central e em memória secundária. Para efeitos desta tese, considera-se a implementação em memória central. Lida com qualquer tipo de dados e qualquer tipo de função de distância.

O desempenho das estruturas de dados métricas é medido através da realização de testes experimentais com espaços métricos. Quase sempre, nos trabalhos de investigação onde se propõem estruturas de dados métricas, compara-se o desempenho da estrutura de dados em estudo com o desempenho de outras estruturas de dados métricas.

Nos testes realizados à RLC, o seu desempenho tem sido comparado com o de outras estruturas de dados métricas, como se pode verificar em [Mamede 2005], [Rodrigues 2006], [Mamede 2007], [Mamede e Barbosa 2007], [Barbosa 2009], [Chambel 2009], [Costa 2009] e [Sarmento 2010], tendo sempre apresentado bons desempenhos com os parâmetros escolhidos.

Regra geral, os valores dos parâmetros das estruturas de dados métricas (parametrizadas) influenciam fortemente os seus desempenhos. Um parâmetro mal escolhido pode conduzir a um mau desempenho. Acresce que os "bons" valores geralmente dependem do espaço métrico, nomeadamente, das características dos dados envolvidos e da função de distância.

Como já foi dito anteriormente, as bases de dados associadas aos espaços métricos apresentam grandes dimensões e o cálculo da distância entre objectos é, normalmente, uma operação muito cara. Para escolher os valores óptimos dos parâmetros de uma estrutura de dados para um espaço métrico, é necessário realizar vários testes experimentais. Se a estrutura tiver mais de um parâmetro, torna-se necessário variar os diferentes parâmetros de forma a tirar conclusões. Ora, como, com determinados espaços métricos, os testes realizados demoram horas ou, algumas vezes, dias, seria extremamente vantajoso conhecer os valores dos parâmetros que conduzem a bons desempenhos.

Em relação à RLC, são apresentadas, nos trabalhos de investigação que envolvem a estrutura, as parametrizações efectuadas para cada espaço métrico utilizado na realização dos testes de desempenho, obtidos por observação dos resultados. Nalguns casos ([Mamede 2005], [Rodrigues 2006] e [Mamede 2007]), os valores de alguns parâmetros foram definidos através de fórmulas, mas essas fórmulas dependem de constantes escolhidas para cada caso. Em [Mamede e Barbosa 2007] e [Barbosa 2009], são apresentadas as parametrizações em espaços métricos formados por dicionários de línguas. Os espaços métricos são caracterizados a partir da média e da variância das distâncias, tentando ir ao encontro do estudo apresentado em [Chávez et al. 2001] sobre a dimensionalidade intrínseca dos espaços métricos. Noutros trabalhos, como [Chambel 2009], [Costa 2009], [Barbosa e Rodrigues 2009] e [Sarmento 2010], são apenas apresentadas as parametrizações escolhidas, que conduziram a bons desempenhos da estrutura.

A motivação deste trabalho consiste no facto de, nos diversos trabalhos de investigação onde se propõem estruturas de dados métricas parametrizadas, não existirem referências sobre como calcular os valores dos parâmetros, de forma a obter um bom desempenho. E, uma vez que os espaços métricos influenciam esses parâmetros, as relações que poderão existir entre os valores dos parâmetros e as características dos espaços métricos, muito particularmente, na RLC.

Por estes motivos, nesta tese propõe-se mais uma variante da estrutura de dados métrica RLC, cujos valores dos parâmetros dependem de valores extraídos do espaço métrico.

1.3 PRINCIPAIS CONTRIBUIÇÕES

A principal contribuição desta dissertação é a proposta de uma nova variante da estrutura de dados métrica RLC, cujos valores dos parâmetros dependem das características do espaço métrico. Mais precisamente, os valores dos parâmetros dependem da média e do desvio padrão das distâncias entre os elementos distintos do universo. Assim, o processo de parametrização da RLC passa a ser semi-automático, bastando conhecer aqueles dois valores.

São estabelecidas algumas relações entre os desempenhos obtidos com a nova variante e as características do espaço métrico. Esse relacionamento é outra contribuição desta tese.

Outra contribuição é um levantamento dos espaços métricos mais frequentemente utilizados nos testes de desempenho de estruturas de dados métricas.

Para avaliar a eficiência da nova variante da RLC, foram realizados testes experimentais com quinze espaços métricos diferentes, todos caracterizados. Logo, outra contribuição deste trabalho é a caracterização desses espaços métricos.

1.4 ESTRUTURA DO DOCUMENTO

Este documento está organizado em nove capítulos. Neste primeiro capítulo é apresentado o tema deste trabalho, o seu contexto, a sua motivação e as suas principais contribuições.

No capítulo dois são apresentadas algumas definições, importantes para a compreensão deste trabalho. Seguidamente são apresentadas algumas funções métricas e descritos alguns espaços métricos utilizados nos testes de desempenho de estruturas de dados métricas. Tanto as funções como os espaços métricos encontram-se divididos em categorias.

O terceiro capítulo é direccionado para as estruturas de dados métricas. São classificadas, descritas as técnicas que utilizam no particionamento dos dados e apresentados os métodos utilizados por estas aquando das pesquisas por proximidade, para descartar ou seleccionar elementos da base de dados sem calcular distâncias.

O quarto capítulo é dedicado à RLC. Primeiramente são apresentados alguns conceitos básicos relacionados com a estrutura, é dada a sua definição original e são descritos os algoritmos de inserção,

remoção e pesquisa por proximidade. A seguir, são analisadas as variantes da RLC, são referidas as complexidades temporais dos seus algoritmos e é apresentada uma lista com os testes realizados à estrutura, onde consta a variante da RLC utilizada, o espaço métrico e os valores escolhidos para os seus parâmetros.

A nova variante da estrutura é definida no capítulo cinco.

No capítulo seis descrevem-se e caracterizam-se os espaços métricos que foram utilizados na fase de testes.

O capítulo sete é dedicado aos testes experimentais: descrevem-se os testes realizados e apresentam-se e analisam-se os resultados obtidos.

No capítulo oito encontram-se as conclusões extraídas da elaboração deste trabalho e no capítulo nove está a bibliografia.

2 ESPAÇOS MÉTRICOS

Os espaços métricos introduzem uma formulação matemática propícia às consultas por semelhança, pois, com base na função de distância e nas suas propriedades, é possível elaborar técnicas de indexação eficientes, capazes de responder a estas consultas. Uma característica importante dos espaços métricos é não imporem restrições ao universo. Existem universos multidimensionais, como, por exemplo, conjuntos de imagens, e universos sem uma dimensão associada, como é o caso de dicionários ou conjuntos de sequências de ADN.

As estruturas de dados métricas têm sido, nos últimos anos, amplamente estudadas, tal como o têm sido os seus algoritmos e as áreas de aplicação das pesquisas por semelhança. A par destes estudos, a definição de semelhança entre objectos vem sendo um desafio, na questão de avaliar se é apropriada para o domínio de dados em questão. Em [Chen 2005] é apresentada a distância ERP, que é avaliada em séries temporais, e em [Fuad e Marteau 2008] é proposta a distância de edição estendida (*extended edit distance*), testada com séries temporais e com textos.

Em determinados tipos de dados, como, por exemplo, imagens, para que seja possível armazená-las numa base de dados e posteriormente efectuar pesquisas por semelhança através de uma função de distância, é necessário que os objectos sejam primeiramente processados tendo em conta um conjunto de características. Para a extracção dessas características são utilizados métodos automáticos que têm como resultado vectores de características [Chambel 2009]. Estes vectores são armazenados na base de dados e, posteriormente, utilizados no cálculo da distância. Por exemplo, para o domínio das imagens, atributos como a forma, a textura e a cor são extraídos das imagens, formando vectores de características. Normalmente, as métricas mais apropriadas dependem das características que foram extraídas [Corel Features].

Neste capítulo, são apresentadas algumas definições básicas, importantes no contexto deste trabalho e baseadas em [Chávez et al. 2001] e [Zezula et al. 2006]. Em seguida, é apresentada uma lista de funções métricas e é feita uma descrição de espaços métricos utilizados para testar estruturas de dados métricas.

Os espaços métricos aqui apresentados foram recolhidos da bibliografia consultada. Vão ser seleccionados alguns, de diferentes domínios, para os testes a realizar nesta tese.

2.1 DEFINIÇÕES BÁSICAS

ESPAÇO MÉTRICO

Um *espaço métrico* é composto por um conjunto de objectos e uma função de distância definida entre eles. Define-se como um par (U, d()), em que U representa o conjunto de todos os objectos, ou seja, o *universo* dos elementos, e d() é uma função de distância.

Uma função de *distância* ou *métrica* está definida em $d : U \times U \rightarrow \mathbb{R}$ e satisfaz as seguintes propriedades, $\forall$ x,y,z $\in$ U:

(p1). Não negatividade: $d(x,y) \geq 0$;

(p2). Simetria: $d(x, y) = d(y, x)$;

(p3). Identidade: $d(x, x) = 0$;

(p4). Positividade estrita: $x \neq y \Rightarrow d(x, y) > 0$;

(p5). Desigualdade triangular: $d(x, y) \leq d(x, z) + d(z, y)$.

A função de distância d() mede a semelhança entre pares de objectos de um domínio e retorna zero se os dois objectos forem iguais, valores próximos de zero para objectos muito similares e valores superiores para objectos mais diferentes.

A figura 2.1 mostra algumas das propriedades das funções métricas, onde as setas representam a distância entre as imagens ligadas.

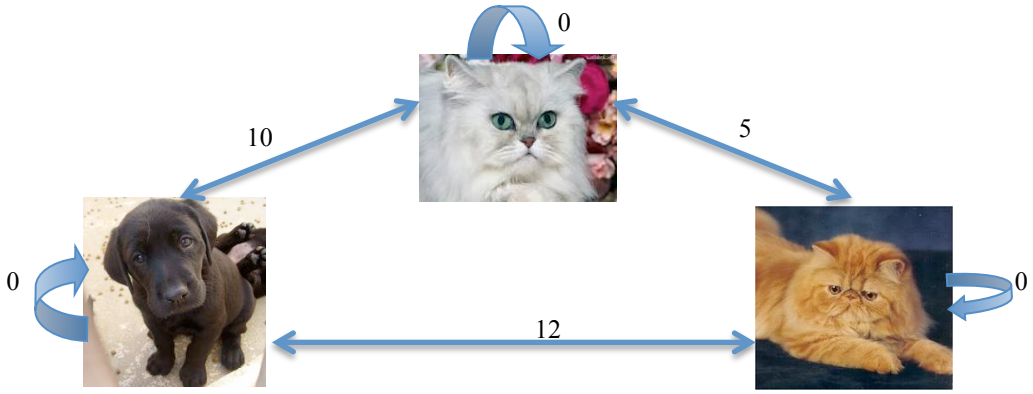


Figura 2.1 - Ilustração de métrica para imagens de animais.

BASE DE DADOS

Uma *base de dados* X de um espaço métrico (U, d()) é um subconjunto finito de U ($X \subseteq U$).

OPERAÇÕES USUAIS

No contexto de pesquisas em espaços métricos, a *pesquisa por proximidade* (*range query*), do *vizinho mais próximo* (*nearest neighbour query* ou *NN*) e dos *k vizinhos mais próximos* (*k nearest neighbour query* ou *k-NN*) são os três tipos mais utilizados.

Nas definições que se seguem, X representa uma base de dados de (U, d()).

1. **PESQUISA POR PROXIMIDADE**: Consiste em obter todos os objectos que estão a uma distância não superior a r do *objecto pergunta* q (*query point*). Formalmente, seja (q, r) uma *pergunta*, em que $q \in U$ e r é um número não negativo que representa o *raio* da pesquisa. O problema da *pesquisa por proximidade* consiste em calcular o conjunto dos elementos da base de dados cujas distâncias a q não excedem r, ou seja, $\{x \in X \mid d(x, q) \leq r\}$.

A figura 2.2 exemplifica uma pesquisa por proximidade. Os elementos contidos na região delimitada pela curva compõem a resposta.

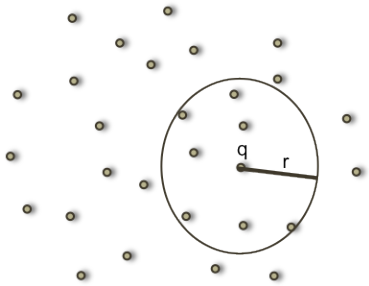


Figura 2.2 - Exemplo de uma pesquisa por proximidade.

2. **PESQUISA DO VIZINHO MAIS PRÓXIMO:** Esta consulta retorna os elementos de X mais próximos do objecto pesquisado. Formalmente, seja $q \in U$ o *objecto pergunta*. A *pesquisa do vizinho mais próximo* retorna o conjunto de objectos $\{x \in X \mid \forall\ v \in X, d(q, x) \leq d(q, v)\}$.

A figura 2.3 mostra um exemplo deste tipo de consulta; o objecto o1 é o vizinho mais próximo do objecto pergunta q.

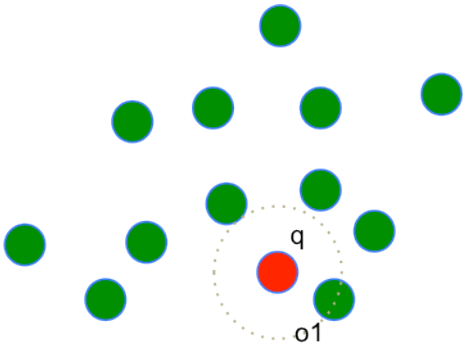


Figura 2.3 - Exemplo de uma pesquisa do vizinho mais próximo.

3. **PESQUISA DOS K VIZINHOS MAIS PRÓXIMOS:** Esta consulta retorna k elementos de X mais próximos do objecto pesquisado. Formalmente, seja q ∈ U o *objecto pergunta* e k um número inteiro positivo que não excede #X. A *pesquisa dos k vizinhos mais próximos* retorna um conjunto A que satisfaz as três seguintes propriedades:

   (p1). $A \subseteq X$;

   (p2). $\#A = k$;

   (p3). $\forall u \in A, \forall v \in X - A, d(q, u) \leq d(q, v)$.

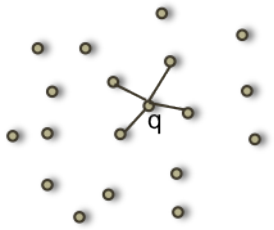


Figura 2.4 - Exemplo de uma pesquisa dos k vizinhos mais próximos.

A figura 2.4 ilustra um exemplo deste tipo de consulta, que tem como entrada o objecto q e o valor k igual a quatro.

2.2 FUNÇÕES MÉTRICAS

Para que uma função forme, juntamente com um conjunto de dados, um espaço métrico, é necessário que tenha as propriedades acima referidas: não negatividade, simetria, identidade, positividade estrita e desigualdade triangular. Nestes casos, as funções são designadas por funções de distância ou métricas. No entanto, por vezes algumas funções não métricas são chamadas funções de distância. Isso não será feito nesta tese. Note-se que uma função não métrica não deve ser usada em estruturas de dados métricas por não ter as propriedades desejadas.

Quando a função não satisfaz a propriedade p4 (positividade estrita), o espaço é chamado *pseudo-métrico*. O espaço é *quase-métrico (quasi-metric)* quando não se verifica a propriedade de simetria. Nos espaços *super-métricos* ou *ultra-métricos*, na desigualdade triangular, o triângulo deve ter pelo menos dois lados iguais [Zezula et al. 2006].

Dependendo do contradomínio, as funções métricas são classificadas como *discretas* ou *contínuas* [Chávez et al. 2001]. As funções métricas são discretas quando o contradomínio é um conjunto finito (pequeno) de valores, enquanto que, numa função contínua, o contradomínio é infinito. Por exemplo, a distância euclidiana é uma métrica contínua e a distância de edição é discreta [Zezula et al. 2006].

Nesta secção é apresentada uma lista de métricas, dividida em três categorias: funções para cadeias de caracteres, funções para vectores e funções para outros tipos de dados. Na próxima secção indicam-se espaços métricos que as usam e alguns trabalhos relevantes que as utilizaram.

2.2.1 FUNÇÕES PARA CADEIAS DE CARACTERES

As funções para cadeias de caracteres são utilizadas em espaços métricos cujos universos são simples palavras ou grandes textos (considerados sem estrutura).

Nas três próximas definições, $x = x_1, ..., x_m$ e $y = y_1, ..., y_n$, com $m \geq 1$ e $n \geq 1$, são duas cadeias de caracteres.

DISTÂNCIA DE HAMMING

A *distância de Hamming* representa o menor número de elementos que precisam de ser modificados para transformar uma palavra na outra. Por exemplo, d(string, strong) = 1.

É definida para cadeias de caracteres de igual comprimento (ou seja, quando m = n) por:

$d(x,y) = \#\{i \mid 1 \leq i \leq n, x_i \neq y_i\}$.

DISTÂNCIA DE LEVENSHTEIN OU DE EDIÇÃO

A *distância de Levenshtein* também é conhecida como *distância de edição* ou, simplesmente, L_{Edit}. Retorna o número mínimo de operações de edição (inserções, remoções e substituições de caracteres) necessárias para transformar uma sequência de caracteres na outra sequência de caracteres [Amato et al. 2003]. É definida por:

$$d(x,y) = d'_{xy}(m,n)$$

$$d'_{xy}(i,j) = \begin{cases} i, & \text{se } i \geq 0 \text{ e } j = 0 \\ j, & \text{se } i = 0 \text{ e } j > 0 \\ \min(\ d'_{xy}(i-1,j-1) + \text{diff}(x_i,y_j), & \\ \quad 1 + d'_{xy}(i,j-1), & \\ \quad 1 + d'_{xy}(i-1,j)\ ) & \text{se } i > 0 \text{ e } j > 0 \end{cases}$$

$$\text{diff}(a,b) = \begin{cases} 0, & \text{se } a = b \\ 1, & \text{se } a \neq b \end{cases}$$

onde $d'_{xy}(i,j)$ representa a distância de edição entre $x_1, \ldots, x_i$ e $y_1, \ldots, y_j$.

DISTANCIA DE EDIÇÃO ESTENDIDA OU EED

A *distância de edição estendida* ou *EED* (*extended edit distance*) foi proposta em [Fuad e Marteau 2008]. Recorre à distância de edição (denotada por L_{Edit}) e a um número real não negativo λ. Nesta definição, assume-se que as cadeias de caracteres são palavras sobre um alfabeto finito A e que f_a^x e f_a^y representam a frequência do carácter a ∈ A em, respectivamente, x e y. Define-se da seguinte forma:

$$d(x,y) = L_{\text{Edit}}(x,y) + \lambda \left(m + n - 2 \sum_{a \in A} \min(f_a^x, f_a^y) \right).$$

E imediato verificar que, quando λ é zero, a distância de edição estendida é a distância de edição.

2.2.2 FUNÇÕES PARA VECTORES

Estas funções são definidas para dados do tipo vector de comprimento fixo. No que se segue, $x=x_1...x_n$ e $y = y_1...y_n$ representam dois vectores, com $n \geq 1$.

DISTÂNCIAS DE MINKOWSKI

As *distâncias de Minkowski* são as funções de distância mais utilizadas para dados vectoriais. São definidas em [Amato et al. 2003] por:

$$L_p(x,y) = \left(\sum \left(x_i - y_i \right)^p \right)^{1/p}, \; p \geq 1.$$

Três casos particulares das medidas de Minkowski são muito usados: a distância de Manhattan, a distância euclidiana e a distância de Chebychev.

DISTÂNCIA DE MANHATTAN OU L1

A *distância de Manhattan*, também designada por *L1* ou *city block*, deriva da distância de Minkowski quando p = 1. É definida em [Amato et al. 2003] por:

$$d(x,y) = \sum_{i=1}^{n} \left| x_i - y_i \right|.$$

DISTÂNCIA EUCLIDIANA OU L2

A *distância euclidiana* ou *L2* é a distância de Minkowski quando p = 2. É definida em [Amato el al. 2003] por:

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}\,.$$

DISTÂNCIA DE CHEBYCHEV OU L∞

A *distância de Chebychev* ou *L∞* corresponde à distância de Minkowski quando p tende para mais infinito. É a maior diferença, em valor absoluto, entre elementos dos dois vectores com o mesmo índice. Está definida em [Chambel 2009] por:

$$d(x,y) = \max_{i=1}^{n} |x_i - y_i|\,.$$

As próximas três distâncias recorrem a uma matriz ou a um vector que, de alguma forma, indicam correlações entre elementos dos vectores x e y.

DISTÂNCIA QUADRÁTICA

A *distância quadrática* é definida em [Zezula et al. 2006] e [Amato et al. 2003] por:

$$d(x,y) = \sqrt{(x-y)^T \times Q \times (x-y)},$$

onde Q = [q_{ij}] é uma matriz de n x n cujos valores q_{ij} representam a correlação entre x_i e y_j.

A expressão $(x - y)^T$ denota a transposta de (x - y). Para ser uma métrica, é necessário que a matriz Q seja simétrica ($q_{ij} = q_{ji}$, para i, j = 1, ..., n) e que a diagonal principal só tenha uns ($q_{ii} = 1$, para i = 1, ..., n).

DISTÂNCIA DE MAHALANOBIS-L1

A *distância de Mahalanobis-L1* é uma métrica que corresponde à distância de Manhattan projectada no espaço de Mahalanobis. É definida em [Chambel 2009]:

$$d_{ML1}(x,y) = \sum_{i=1}^{n} \left| \frac{x_i - y_i}{\sqrt{\lambda_i}} \right|,$$

onde $\lambda = \lambda_1, \ldots \lambda_n$ é um vector de números reais positivos.

DISTÂNCIA DE MAHALANOBIS-L2

A *distância de Mahalanobis-L2* corresponde à distância euclidiana projectada no espaço de Mahalanobis [Chambel 2009]. A sua definição pode ser encontrada em [Chambel 2009]:

$$d_{ML2}(x,y) = \sqrt{\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\lambda_i}},$$

onde $\lambda = \lambda_1, \ldots \lambda_n$ é um vector de números reais positivos.

2.2.3 OUTRAS FUNÇÕES

Estas funções estão definidas para dados que não são vectores nem cadeias de caracteres. São definidas para quantificar semelhanças entre documentos (que contêm palavras), conjuntos e séries temporais.

DISTÂNCIA DOS CO-SENOS

A *distância dos co-senos* é utilizada na semelhança entre documentos. Basicamente, mede o peso dos termos partilhados pelos documentos [SISAP].

Sejam $d_1, \ldots, d_n$ os documentos e $\{t_1, \ldots, t_k\}$ o conjunto dos *termos* (as palavras do vocabulário) que ocorrem nos documentos. Cada documento d_i é representado por um vector, num espaço de dimensão k, onde a coordenada $w_{r,i}$ traduz o peso do termo t_r no documento.

$$d_i = (w_{1,i}, w_{2,i}, \ldots, w_{k,i}), \text{ para } i = 1, \ldots, n.$$

O peso $w_{r,i}$ é definido por:

$$w_{r,i} = \frac{\phi_{r,i} \log \frac{n}{n_r}}{\sqrt{\sum_{s=1}^{k} \left(\phi_{s,i} \log \frac{n}{n_s} \right)^2}},$$

onde $\phi_{r,i}$ é o número de vezes que o termo t_r aparece no documento d_i e n_r é o número de documentos onde t_r ocorre.

A definição da distância entre os documentos d_i e d_j é:

$$d(d_i, d_j) = \arccos\left(\sum_{r=1}^{k} w_{r,i}\, w_{r,j} \right).$$

DISTÂNCIA DE JACARD

A *distância de Jacard* é também conhecida por *métrica de semelhança entre conjuntos*. Sendo A e B dois conjuntos, é definida em [Amato et al. 2003] por:

$$d(A,B) = 1 - \frac{\#(A \cap B)}{\#(A \cup B)}.$$

Um exemplo de utilização desta métrica é referido em [Zezula et al. 2006] no contexto de uma base de dados onde são armazenadas as páginas web visitadas por determinados utilizadores de um espaço público. As áreas de interesse de um utilizador podem ser representadas pelo conjunto de páginas visitadas e a distância de Jacard permite averiguar indivíduos com interesses semelhantes.

DISTÂNCIA DE HAUSDORFF

A *distância de Hausdorff* também é aplicada a conjuntos. De modo informal, dados dois conjuntos A e B, a função h(A,B) calcula a distância de cada elemento do conjunto A a todos os elementos do conjunto B, de modo a determinar, para cada elemento de A, a distância deste ao elemento mais próximo de B [Chambel 2009]. De seguida, deste conjunto de valores é retornada a maior distância. Como o valor de h(A, B) pode ser diferente do valor de h(B, A), a distância de Hausdorff é definida como o máximo destes dois valores.

É definida em [Chambel 2009] por:

$$d(A,B) = \max(h(A,B), h(B,A))$$

$$h(A,B) = \max_{a \in A} \min_{b \in B} d(a,b)$$

Apesar de estar referenciada na bibliografia consultada, não se encontraram referências ao seu uso em testes de estruturas de dados métricas.

DISTÂNCIA ERP

A *distância ERP* (*edit distance with real penality*), introduzida em [Chen 2005], é aplicada em domínios relacionados com séries temporais.

Formalmente, sejam

$$R = < (t_{1r}, x_{1r}, y_{1r}), \ldots , (t_{mr}, x_{mr}, y_{mr}) > \text{ e } S = < (t_{1s}, x_{1s}, y_{1s}), \ldots, (t_{ns}, x_{ns}, y_{ns}) >$$

duas trajectórias cujos comprimentos são m e n, respectivamente. A distância ERP entre R e S pode ser definida por:

$$d(R,S) = d'_{RS}(1,1)$$

$$d'_{RS}(i,j) = \begin{cases} 0 & \text{se } i = m+1 \text{ e } j = n+1 \\ d'_{RS}(i+1,j) + d((x_{ir},y_{ir}),(0,0)) & \text{se } i \leq m \text{ e } j = n+1 \\ d'_{RS}(i,j+1) + d((0,0),(x_{js},y_{js})) & \text{se } i = m+1 \text{ e } j \leq n \\ \min(\ d'_{RS}(i+1,j+1) + d((x_{ir},y_{ir}),(x_{js},y_{js})), & \\ \quad d'_{RS}(i+1,j) + d((x_{ir},y_{ir}),(0,0)), & \\ \quad d'_{RS}(i,j+1) + d((0,0),(x_{js},y_{js}))\ ) & \text{se } i \leq m \text{ e } j \leq n \end{cases}$$

onde d((x, y), (x', y')) representa a distância entre (x, y) e (x', y'). Chen recorre à distância de Manhattan [Chen 2005], enquanto Barbosa e Rodrigues fazem uso da distância euclidiana [Barbosa e Rodrigues 2009].

2.3 ESPAÇOS MÉTRICOS

Os testes de desempenho de estruturas de dados métricas são realizados utilizando espaços métricos, com universos formados por pontos gerados aleatoriamente ou objectos existentes de determinados domínios. No primeiro caso, os espaços métricos chamam-se *sintéticos* e, no segundo, *reais*.

Alguns espaços não têm uma dimensão associada (são *adimensionais*), porque os elementos do universo não possuem uma dimensão fixa. Um dicionário de uma língua natural é um exemplo deste tipo de universo, porque as palavras não têm todas o mesmo número de caracteres [Pola 2010]. Mas, por exemplo, quando o universo é $\mathbb{R}^n$, considera-se que o espaço tem *dimensão* n. Por exemplo, as características de uma imagem são representadas através de um vector de números, cujo comprimento é igual para todas as imagens do universo. Porém, a composição destes vectores pode ser oriunda de diferentes domínios, como descritores diferentes de texturas. Nestes casos, os espaços são *multi-dimensionais* e métricas como as de Minkowski podem ser utilizadas para comparar os elementos [Pola 2010].

Em [Chávez e Navarro 2000] propõe-se que a *dimensionalidade intrínseca* de uma espaço métrico seja definida por $\rho = \frac{\mu^2}{2.\sigma^2}$, em que μ e σ^2 representam, respectivamente, a média e a variância do histograma de distâncias. Os autores referem, no entanto, que estes valores não devem ser considerados como exactos.

Foram recolhidos alguns espaços métricos presentes na bibliografia, que são apresentados nesta secção. Estão divididos nas seguintes categorias: sintéticos, de imagens, de textos ou documentos e de outros tipos, como sequências de proteínas ou de ADN e séries temporais. Esses espaços métricos

foram usados para testar estruturas de dados métricas. Por isso, a seguir à apresentação dos espaços métricos, encontra-se uma lista de estruturas de dados métricas.

2.3.1 ESPAÇOS MÉTRICOS SINTÉTICOS

O universo destes espaços métricos é caracterizado pelo conjunto dos pontos num hiper-cubo, por vezes unitário, num espaço de dimensão k. Genericamente, é representado por $[a,b]^k$, sendo k a dimensão do espaço e a < b.

Com este universo são formados vários espaços métricos, destacando-se os seguintes: pontos com distância de Manhattan, pontos com distância euclidiana e pontos com distância de Chebychev.

Os pontos podem estar distribuídos uniformemente no espaço, ou então segundo uma outra distribuição, por vezes formando agrupamentos que poderão ser de tamanho fixo. Os centros destes agrupamentos podem estar uniformemente distribuídos.

Estes espaços métricos foram muito utilizados em testes de desempenho de diversas estruturas de dados, como se pode verificar na lista a seguir apresentada.

- **Espaço métrico ($[0,1]^k$, L2):** VT [Dehne e Nolteimer 1987]; Kd-Tree e VPT [Yianilos 1993]; M-tree [Zezula et al. 1998]; SAT [Navarro 1999]; SAT, FQA, LAESA e GNAT [Chávez e Navarro 2000]; DSA-Tree, HDSAT1 e HDSAT2 [Arroyuelo et al. 2003]; LC, BKT, GNAT, SAT, FQA e LAESA [Chávez e Navarro 2005]; IAESA e AESA [Figueroa et al. 2006]; RLC, VPT, LC, LAESA, HDSAT e GNAT [Mamede 2007].
- **Espaços métricos ($[-1, 1]^D$, L2), para $D \in [4, 24]$:** t-AESA (t-Spanners) e AESA [Navarro et al. 2007].
- **Espaços métricos ($[0,1]^k$, L1) e ($[0,1]^k$, L∞):** VT [Dehne e Nolteimer 1987]; Kd-Tree e VPT [Yianilos 1993].

2.3.2 ESPAÇOS MÉTRICOS DE IMAGENS

Existem várias formas de analisar a semelhança entre imagens, que dependem da categoria do resultado desejado. Estas categorias referem-se aos atributos considerados relevantes, que podem ser atributos visuais como a cor e a forma; atributos lógicos como a identificação de elementos (por exemplo, a pesquisa de imagens que contêm uma flor); atributos semânticos como a identificação de emoções humanas (pesquisa de imagens que expressam alegria) [Pola 2010].

Os vectores de características, anteriormente referidos, representam a descrição matemática de características (como a forma e a cor) e têm como objectivo representar os aspectos significativos de uma imagem. Em [Thomasian et al. 2008], uma imagem *P* é especificada pelo seu vector de características como um ponto num espaço de dimensão n.

Nesta secção são apresentados alguns métodos utilizados na extracção de características de imagens, centrados nos utilizados no âmbito desta tese: histogramas de cores e imagens de rostos. Em

relação às imagens de rostos, todos os conceitos apresentados encontram-se no trabalho elaborado por Pedro Chambel [Chambel 2009].

A identificação de uma imagem através da *característica cor* é geralmente realizada pela construção de um *histograma de cores*, em que são calculados os números de pixéis da imagem com cada cor [Corel Features]. Existem vários domínios onde são utilizados os histogramas. Por exemplo, a análise de tons cinza, conhecida por *gray level histogram* ou como *brightness histogram*, é utilizada em imagens médicas. Uma métrica muito utilizada na comparação de histogramas é a distância euclidiana.

A técnica mais utilizada na construção de histogramas foi proposta em 1991 por Swain e Ballard, tendo-se seguido outros métodos, como o uso de *histogramas de cores cumulativos*, proposto por Stricker e Orengo em 1995, o uso de *análise de cor baseada em regiões* e o uso de *histogramas métricos* [Pola 2010].

Os problemas que podem decorrer da comparação de imagens através dos seus histogramas de cores são o facto de duas imagens bem distintas poderem possuir histogramas de cores semelhantes e o facto de frequentemente o número de cores ser elevado (normalmente maior que 256), o que gera vectores de características de dimensão alta.

Uma outra área onde as imagens são utilizadas é na pesquisa de imagens de rostos. Os métodos utilizados na extracção de características de imagens de rostos podem basear-se nas características globais, onde a representação da imagem de rosto é utilizada em toda a região do rosto. Nestes casos, as imagens de dimensão A x L são representadas por vectores unidimensionais com tamanho A x L, contendo a informação de cada pixel. No entanto, este método apresenta informação redundante no processo de reconhecimento e a representação da informação tem uma dimensão muito elevada para permitir um reconhecimento facial rápido. Assim sendo, foram propostas outras técnicas que permitem reduzir a dimensão dos dados, entre as quais o método *eigenfaces*.

O método *eigenfaces* é baseado nas características globais. Cada imagem de rosto com largura L e altura A, em píxeis, é inicialmente representada por um vector unidimensional de inteiros de dimensão D, em que D = L x A e onde cada elemento desta matriz representa um pixel da imagem.

A ideia principal deste método consiste em reduzir a dimensão dos dados e, desta forma, executar o reconhecimento de rostos num espaço de menor dimensão. Desta forma, as imagens de rostos são projectadas no espaço de rostos (*feature space*) que melhor descreve a variação em relação às imagens conhecidas da base de dados. Para isso, é necessário extrair as características principais do conjunto de treino. Estas designam-se por vectores próprios (*eigenvectors*) e são extraídas por meio do método matemático *principal component analysis*. Estes vectores podem ser vistos como um conjunto de características que juntos conseguem caracterizar a variação entre as imagens de rostos.

Uma outra área onde são utilizadas imagens é no reconhecimento de impressões digitais. Nestes casos, as características extraídas de uma impressão digital dizem respeito às minúcias pertencentes à mesma e são as coordenadas *x* e *y* da minúcia, a direcção da minúcia e uma lista de minúcias vizinhas.

Todas estas informações são armazenadas num vector de características, como é descrito em [Jardini 2007].

Imagens, juntamente com métricas como a euclidiana, quadrática, Mahalanobis-L1 ou Mahalanobis-L2, formam espaços métricos de imagens que são amplamente utilizados em vários trabalhos de investigação onde se propõem estruturas de dados métricas. A seguir são listados alguns espaços métricos cujos universos são imagens, bem como estruturas de dados onde estes espaços métricos foram utilizados para avaliar os seus desempenhos.

- **Histogramas de cores com L2:** DSAT [Navarro e Reyes 2002]; LC e M-Tree [Bustos e Navarro 2009]; M-Tree, Slim-Tree, DF-Tree e RLC [Sarmento 2010][1].
- **Histogramas de cores com distância quadrática:** D-Index e M-Tree [Dohnal et al. 2003].
- **Histogramas de cores, em tons cinza, com L1:** VPT e MVPT [Bozkaya e Ozsoyoglu 1997].
- **Histogramas de cores, em tons cinza, com L2:** VPT e MVPT [Bozkaya e Ozsoyoglu 1997].
- **Imagens de rostos com L1:** LAESA, VPTree, DSAT, HDSATI, HDSAT2, GNAT, LC e RLC [Chambel 2009]; M-Tree, Slim-Tree, DF-Tree e RLC [Sarmento 2010].
- **Imagens de rostos com L2:** Slim-Tree e M-Tree [Traina et al. 2002 a]; DF-Tree, Slim-Tree e M-Tree [Traina et al. 2002 b]; AESA e IAESA [Figueroa et al. 2006]; LAESA, VPTree, DSAT, HDSAT1, HDSAT2, GNAT, LC e RLC [Chambel 2009][2].
- **Imagens de rostos com distâncias Mahalanobis-L1 e Mahalanobis-L2:** LAESA, VPTree, DSAT, HDSAT1, HDSAT2, GNAT, LC e RLC [Chambel 2009].
- **Impressões digitais com L2:** Slim-Tree [Jardini 2007][3].

2.3.3 ESPAÇOS MÉTRICOS DE TEXTOS OU DOCUMENTOS

Na bibliografia consultada, os universos destes espaços métricos são dicionários de línguas, documentos, linhas de texto retiradas de documentos e conjuntos de endereços URL. O espaço métrico dicionário com distância de Levenshtein é muito utilizado.

- **Dicionário com distância de Levenshtein:** VPT e GNAT [Brin 1995]; D-SAT [Navarro e Reyes 2002]; Slim-tree e M-Tree [Traina et al. 2002 a]; DF-Tree, Slim-Tree e M-Tree [Traina et al. 2002 b]; D-SAT e HDSAT [Arroyuelo et al. 2003]; BVPT e VPT [Fredriksson 2005]; t-AESA (t-Spanners) e AESA [Navarro et al. 2007]; GNAT, HDSAT, LAESA, RLC e VPT

---

1 Os 112.682 histogramas encontram-se em http://www.dbs.informatik.uni-muenchen.de/~seidl/DATA/histo112.112682.gz

2 A base de dados de imagens de rostos encontra-se em http://cswww.essex.ac.uk/mv/allfaces/index.html

3 É referido que bases de dados de impressões digitais podem ser obtidas neste endereço: http://bias.csr.unibo.it/fvc2006/download.asp

[Mamede e Barbosa 2007]; LC e M-Tree [Bustos e Navarro 2009]; M-Tree, Slim-Tree, DF-Tree e RLC [Sarmento 2010][4].

- **Dicionário com distância EED:** VPT e RLC [Barbosa 2009].
- **Documentos com distância dos co-senos:** D-SAT e SAT [Navarro e Reyes 2002][5]; IAESA e AESA [Figueroa et al. 2006]; t-AESA (t-spanners) e AESA [Navarro et al. 2007].
- **Linhas de texto com distância de Levenshtein:** VP-Tree, GH-Tree, OPT-Tree e GNAT [Brin 1995].
- **Endereços URL com distância de Jacard:** D-Index e M-Tree [Dohnal et al. 2003].

2.3.4 OUTROS ESPAÇOS MÉTRICOS

Nesta secção são apresentados espaços métricos referentes a outros domínios de aplicação das pesquisas por semelhança. No que se refere às sequências de ADN ou de proteínas, estas podem ser interpretadas como fragmentos de textos, formando bases de dados genéticas, uma vez que as quatro bases que compõem as sequências de ADN podem ser representadas pelas letras A, C, G e T.

Uma série temporal consiste num conjunto de observações ordenadas no tempo e que apresentam dependência. Ocorrem em muitas áreas como finanças, marketing, seguros e meteorologia. Em relação ao espaço métrico apresentado, cujo universo é um conjunto de trajectórias de furacões, cada trajectória é uma sequência de triplos da forma (x,y,t), sendo x e y coordenadas no plano e t o tempo.

- **Sequências de ADN ou de proteínas com distância de Hamming:** BVPT e VPT [Fredriksson 2005][6].
- **Trajectórias de furacões com distância ERP:** RLC [Barbosa e Rodrigues 2009][7].

[4] Bases de dados obtidas em http://www.sisap.org/Metric_Space_Library.html

[5] Bases de dados para espaços de documentos podem ser obtidas no endereço http://trec.nist.gov

[6] Bases de dados de proteínas podem ser encontradas em: http://pizzachili.dcc.uchile.cl/texts/protein/, http://aug.csres.utexas.edu/mobios-workload/ e http://corpus.canterbury.ac.nz/descriptions/

[7] É referido que a base de dados pode ser acedida a partir de http://weather.unisys.com/hurricane/atlantic/.

3 ESTRUTURAS DE DADOS MÉTRICAS

Um dos vectores de intervenção para melhorar as pesquisas por proximidade tem sido a procura de algoritmos que reduzam, de alguma forma, o cálculo de distâncias entre objectos de um determinado domínio. Neste sentido, têm sido investigadas e propostas diversas estruturas de dados para espaços métricos, todas com o objectivo de agilizar as consultas por proximidade em espaços métricos, ou seja, pretende-se que retornem resultados no mais curto espaço de tempo, uma vez que as bases de dados apresentam grandes dimensões.

Os métodos propostos por Burkhard e Keller, em 1973 [Burkhard e Keller 1973], foram o ponto de partida, introduzindo a técnica de particionamento do espaço métrico. Os autores definiram algoritmos para realizar a pesquisa em estruturas em forma de árvore, usando representantes (pivots) para guiar a pesquisa através dos nós. Introduziram a *BK-tree* (ou, simplesmente, *BKT* de Burkhard e Keller *tree*), para métricas discretas, em que um elemento arbitrário da base de dados $p \in X$ é seleccionado como raiz da árvore. Para cada distância $i > 0$, define-se $Xi = \{x \in X \mid d(x, p) = i\}$ como o conjunto de todos os elementos à distância i da raiz p. Depois, para qualquer conjunto não vazio Xi, são recursivamente construídas as sub-árvores de p [Chávez e Navarro 2000]. A figura 3.1, retirada de [Zezula et al. 2006], mostra a construção de uma árvore BK-tree.

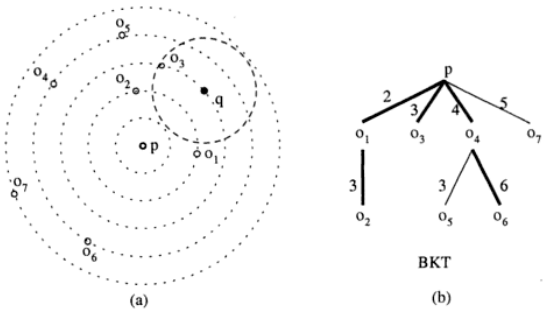


Figura 3.1 - Exemplo de uma árvore BK-tree.

Seguiram-se outras propostas, tais como: BST (*Bisector Tree*) definida por Kalantari e McDonald em 1983 [Chávez et al. 2001]; AESA (*Approximating and Eliminating Search Algorithm*) em 1986 [Ruiz 1986]; VT (*Voronoi Tree*) em 1987 por Dehne e Noltemeier [Dehne e Nolteimer 1987]; GHT (*Generalized-Hyperplane Tree*) apresentada por Uhlmann em 1991 [Chávez et al. 2001]; VP-tree

(*Vantage-Point Tree*), proposta por Yianilos em 1993 [Yianilos 1993]; FQT (*Fixed Queries Tree*) proposta em 1994 em [Baeza-Yates et al. 1994]; LAESA (*Linear Approximating and Eliminating Search Algorithm*) em 1994 [Micó et al. 1994]; GNAT (*Geometric Near-neighbor Access Tree*) em 1995 por Brin [Brin 1995]; MT ou M-tree (*Metric Tree*) em 1997 [Ciaccia et al. 1997]; MVPT ou MVP-tree (*Multi-Vantage Point Tree*) apresentada em 1997 [Bozkaya e Ozsoyoglu 1997]; FHQT (Fixed-Height FQ-Tree) apresentada em [Baeza-Yates e Navarro 1998]; FQA (Fixed Query Array) descrita em [Chávez et al. 1999]; SAT (*Spatial Approximation Tree*) em 1999 [Navarro 1999]; VPF (Vantage-Point Forest) por Yianilos em 1999; LC (List of Clusters) em 2000 [Chávez e Navarro 2000]; RLC (Recursive Lists of Clusters) e variantes em 2005 [Mamede 2005] e 2010 [Sarmento 2010].

Todas as estruturas de dados métricas dividem os elementos da base de dados em subconjuntos e são construídas de forma a se determinar o conjunto de subconjuntos nos quais um determinado objecto se enquadra. Aquando das consultas, é feita uma pesquisa nos subconjuntos candidatos, calculando-se assim o resultado. A figura 3.2, adaptada de [Chávez et al. 2001], exemplifica o particionamento dos dados e a pesquisa nos subconjuntos candidatos.

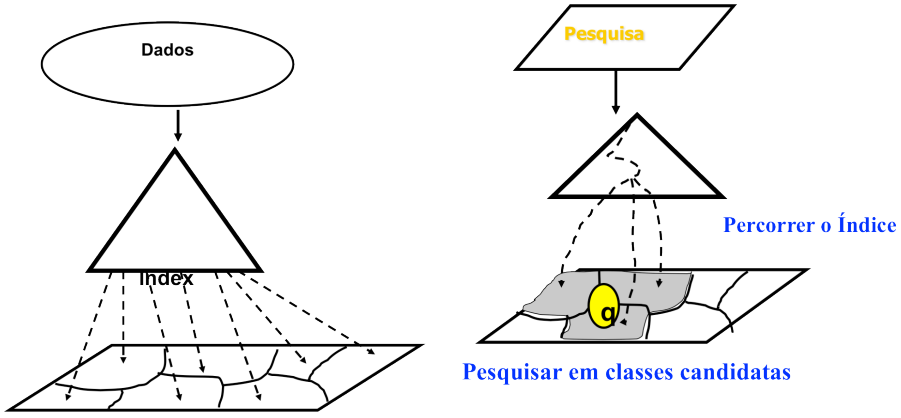


Figura 3.2 - Exemplo do funcionamento de uma estrutura de dados métrica.

Neste capítulo apresentam-se as classificações das estruturas de dados métricas, dá-se uma visão geral sobre as técnicas utilizadas no particionamento dos dados e explica-se como a desigualdade triangular é usada para diminuir o número de distâncias computadas. As definições e conceitos, quando não referenciados, encontram-se em [Chávez et al. 2001], [Amato et al. 2003] e [Zezula et al. 2006].

3.1 CLASSIFICAÇÕES DAS ESTRUTURAS DE DADOS MÉTRICAS

Há estruturas de dados métricas que só aceitam funções de distância *discretas*, cujos contradomínios são conjuntos pequenos de valores, e outras preparadas para lidar com funções de distância *contínuas*, cujos contradomínios são infinitos. Das estruturas acima apresentadas, as seguintes lidam somente com funções de distância discretas: BKT, FQT, FHQT e FQA.

As estruturas de dados podem ser implementadas em *memória central* ou em *memória secundária*, sendo que as primeiras sofrem das limitações inerentes ao espaço disponível [Sarmento e Mamede 2010]. A eficiência de uma estrutura de dados métrica implementada em memória secundária depende de mais factores, conforme apresentado em [Chávez et al. 2001].

- O número de acessos a disco necessários para processar consultas e inserções, pois tais métodos organizam os dados em páginas de disco de tamanho fixo e os acessos aos mesmos implicam a leitura/gravação de uma página do disco para a memória central.
- O custo computacional da função de distância pode ser muito alto, de tal forma que o cálculo das distâncias pode ter impacto na eficiência, mesmo com acessos a disco.
- A utilização do espaço de armazenamento devido ao número de acessos a disco necessário para responder a grandes consultas.

As estruturas de dados métricas podem ainda ser classificadas como: *estáticas*, quando não suportam actualizações ao seu conteúdo após o carregamento inicial dos dados, ou *dinâmicas,* em caso contrário; *genéricas*, se aceitam qualquer tipo de objecto e qualquer função de distância, ou *não genéricas,* em caso contrário; *parametrizadas,* quando a construção da estrutura requer parâmetros de entrada, ou *não parametrizadas*, em caso contrário. A forma como particionam os objectos da base de dados, abordada na próxima secção, é outro critério de classificação.

3.2 TÉCNICAS DE PARTICIONAMENTO

Basicamente, existem duas técnicas de particionamento dos dados: baseadas em *pivots* ou baseadas em *agrupamentos*. As estruturas de dados baseadas em pivots guardam a distância entre cada elemento da base de dados e alguns elementos pré-seleccionados, chamados *pivots*, enquanto que as baseadas em agrupamentos dividem o espaço em regiões, em que cada região tem um objecto especial chamado *centro*. Os objectos são guardados nas regiões. Alguns exemplos de estruturas de dados métricas baseadas em agrupamentos são VT, GNAT, M-tree, Slim-tree e SAT, enquanto que BKT, AESA e LAESA são baseadas em pivots.

O particionamento em agrupamentos foi proposto por Uhlmann em 1991, que definiu duas formas para estruturar um domínio métrico: através da divisão em *regiões com raio de cobertura* (*ball decomposition*) e através da divisão em *hiper-planos generalizados* (*generalized hyperplane decomposition*). Posteriormente, em 1999, Yianilos sugeriu a técnica de particionamento *por exclusão do meio* (*excluded middle partitioning*) [Zezula et al. 2006]. Têm sido propostas algumas variantes destas técnicas, como a apresentada em [Navarro e Uribe-Paredes 2011] com os *ghost hyperplanes.*

Nesta secção encontra-se a descrição sumária das técnicas utilizadas, baseada nos trabalhos de [Zezula et al. 2006] e [Amato et al. 2003]. No final é apresentada uma figura que ilustra as técnicas, retirada da primeira referência. Uma vez que o âmbito deste trabalho incide sobre uma estrutura de dados métrica que utiliza agrupamentos com raio de cobertura, será dada mais ênfase a esta técnica.

3.2.1 PARTICIONAMENTO BASEADO EM AGRUPAMENTOS

As estruturas de dados baseadas em agrupamentos dividem o espaço em regiões, em que cada região tem um objecto especial chamado *centro* [Chávez et al. 2001]. Numa pesquisa é possível descartar objectos comparando a distância da pergunta ao centro do agrupamento [Navarro e Uribe-Paredes 2011].

Existem basicamente dois tipos de particionamento: por *regiões com raio de cobertura* (*ball* ou *covering radius*) e por *hiper-planos* [Navarro e Uribe-Paredes 2011].

PARTICIONAMENTO POR REGIÕES COM RAIO DE COBERTURA

Esta técnica, proposta por Uhlmann, divide um conjunto $S \subseteq U$ em dois subconjuntos S1 e S2, escolhendo um elemento p para ser o *centro* da região S1. Seja d_m o *raio* da região. Então, qualquer objecto o_j vai pertencer às regiões S1 ou S2, de acordo com as seguintes regras [Zezula et al. 2006]:

- $S1 \leftarrow \{ o_j \mid d(o_j, p) \leq d_m \}$,
- $S2 \leftarrow \{ o_j \mid d(o_j, p) > d_m \}$.

A figura 3.3 – caso (a) ilustra este tipo de particionamento.

Esta técnica foi estendida por Chávez e Navarro, em 2000, que propuseram *agrupamentos* (*clusters*), em que cada agrupamento é definido por um objecto (o seu centro) e por um número positivo que define o raio de cobertura da região. No interior de cada agrupamento estão todos os objectos (excepto o centro) cujas distâncias ao centro não excedem o valor do raio de cobertura.

A LC e a RLC são dois exemplos de estruturas de dados que utilizam a partição por agrupamentos, sendo a primeira constituída por uma lista de agrupamentos e a segunda por uma lista de listas de agrupamentos.

Uma questão importante é a forma como os centros e os raios são escolhidos, quando os agrupamentos são construídos. Os autores da LC [Chávez e Navarro 2000] discutem no referido artigo cinco critérios para a selecção dos centros e dois para a selecção de raios.

Os centros podem ser: (1) escolhidos aleatoriamente; (2) o objecto mais próximo do centro anterior dos objectos restantes; (3) o objecto mais afastado do centro anterior dos objectos restantes; (4) o objecto que minimiza a soma das distâncias aos centros anteriores; (5) o objecto que maximiza a soma das distâncias aos centros anteriores.

Os raios dos agrupamentos podem ser todos iguais ou pode-se fixar o número de elementos no interior dos agrupamentos. A primeira escolha leva a listas de agrupamentos de *raio fixo*, enquanto a segunda conduz a listas de agrupamentos de *tamanho fixo.* A RLC é um exemplo da utilização de listas de agrupamentos de raio fixo.

Para estruturas de dados que utilizam listas de agrupamentos (LC e RLC), se os raios são todos iguais, normalmente os primeiros agrupamentos contêm muitos elementos, enquanto que os últimos contêm muitas vezes somente o centro [Mamede 2007]. O tamanho das listas afecta o desempenho das

referidas estruturas de dados. Se, por um lado, valores muito pequenos para o raio dos agrupamentos conduzem a que a estrutura fique comprida, por outro lado, raios com valores elevados diminuem o comprimento da lista, mas aumentam o número de elementos no interior dos agrupamentos, cujo processamento pode tornar-se muito pesado [Mamede 2007]. Portanto, neste tipo de estruturas, é importante efectuar-se uma escolha adequada para o valor do raio.

PARTICIONAMENTO POR HIPER-PLANOS GENERALIZADOS

Assim como a técnica anterior, esta técnica divide um conjunto S $\subseteq U$ em subconjuntos S1 e S2. Desta vez, dois objectos p1 e p2 são aleatoriamente escolhidos para serem os *centros*. Todos os outros objectos são atribuídos a S1 ou a S2 dependendo da sua distância aos centros, de acordo com as seguintes regras [Zezula et al. 2006]:

- S1 $\leftarrow \{ o_j \mid d(p1, o_j) \leq d(p2, o_j)\}$,
- S2 $\leftarrow \{ o_j \mid d(p1, o_j) > d(p2, o_j)\}$.

Em contraste com o particionamento por regiões com raio de cobertura, esta técnica divide de uma forma mais equilibrada os objectos pelos planos criados. A figura 3.3 – caso (c) exemplifica este tipo de particionamento.

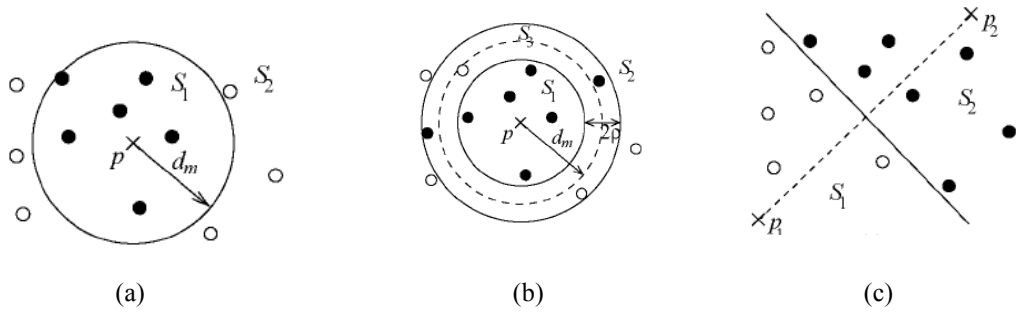
As estruturas GHT e SAT são dois exemplos de estruturas baseadas em hiper-planos generalizados.

PARTICIONAMENTO POR EXCLUSÃO DO MEIO

Ao contrário das duas técnicas anteriormente apresentadas, esta divide S em três subconjuntos, S1, S2 e S3, e é uma extensão da técnica de particionamento por regiões com raio de cobertura. A modificação tem a seguinte motivação. Se o objecto da pergunta está perto do limiar de particionamento, a pesquisa normalmente requer o acesso a ambos os subconjuntos S1 e S2. Ter um subconjunto S3 como uma região de exclusão faz com que a execução de tais consultas possa descartar um ou ambos os subconjuntos S1 e S2. Sendo 2ρ a espessura da região de exclusão, a partição é definida por:

- S1 $\leftarrow \{ o_j \mid d(o_j, p) \leq d_m - \rho\}$,
- S2 $\leftarrow \{ o_j \mid d(o_j, p) > d_m + \rho\}$,
- S3 $\leftarrow nos\ outros\ casos.$

O caso (b) da figura 3.3 ilustra este tipo de particionamento.



(a) (b) (c)

Figura 3.3 - Exemplos de tipos de particionamento.

3.2.2 PARTICIONAMENTO BASEADO EM PIVOTS

Esta técnica selecciona um número de pivots da base de dados e todos os outros elementos são classificados de acordo com as suas distâncias aos pivots. As distâncias entre os elementos e os pivots e entre o objecto da pergunta q e os pivots são utilizadas, em conjunto com a desigualdade triangular, para seleccionar ou descartar elementos da base de dados sem calcular as suas distâncias a q [Chávez e Navarro 2000]. Estruturas métricas como BKT, AESA, LAESA e variantes, e VPF particionam o espaço métrico utilizando pivots.

Nesta técnica, dois elementos estão na mesma região se estiverem à mesma distância em relação a todos os pivots definidos. O espaço pode ser dividido tendo em conta um número arbitrário de pivots, que podem ser escolhidos de diversas formas [Bustos et al. 2003]. Na figura 3.4, retirada de [Chávez et al. 2001], o espaço é particionado com base em dois pivots, u8 e u11. Tendo apenas em conta o pivot u8, os elementos u2 e u4 podem estar próximos entre si. A pesquisa de elementos próximos de um dado objecto q consiste em procurar os elementos nas intersecções de algumas "coroas circulares" centradas nos pivots. Tendo ainda em consideração a mesma imagem, os candidatos ao resultado da pesquisa de q (com o raio assinalado) seriam u5 e u12.

Quando o número de pivots é baixo, esta técnica não garante a proximidade de elementos da mesma zona [Chambel 2009]. Para garantir a proximidade dos elementos na mesma zona, utiliza-se um número maior de pivots. Mas, se este aumento melhora a eficiência das pesquisas, quanto mais pivots forem utilizados, maior será a quantidade de memória gasta [Zezula et al. 2006].

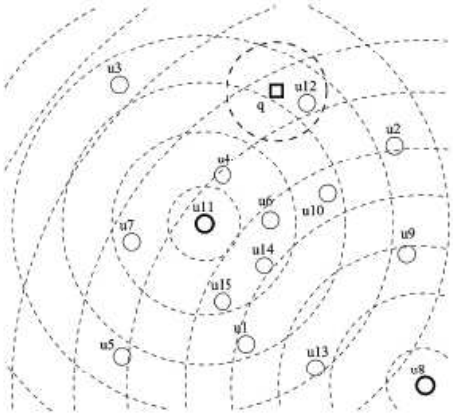


Figura 3.4 - Particionamento do espaço com base em dois pivots.

Nas técnicas apresentadas, verifica-se que os centros e os pivots desempenham papéis semelhantes; no entanto, a diferença está no facto de que um dado elemento x está associado a um pivot p através da distância entre p e x e não porque p seja um pivot próximo de x.

3.3 AS PESQUISAS POR PROXIMIDADE NAS ESTRUTURAS DE DADOS MÉTRICAS

A desigualdade triangular é utilizada pelas estruturas de dados para incluir ou descartar elementos aquando de uma pesquisa por proximidade (q, rq). Nesta secção é descrito como é utilizada a referida propriedade para seleccionar ou descartar elementos, sem calcular distâncias.

Se o particionamento é baseado em agrupamentos, os elementos da base de dados vão encaixar-se nos agrupamentos tendo em conta as distâncias aos centros dos agrupamentos. Um elemento pode encaixar-se num agrupamento se a sua distância ao centro do mesmo não exceder o valor do raio de cobertura. A figura 3.5 mostra o espaço particionado em quatro agrupamentos, cujos centros são os objectos c1, c2, c3 e c4. São ilustradas duas pesquisas, cujos objectos são q1 e q2, respectivamente, com raios de cobertura r1 e r2. Apenas são pesquisados os agrupamentos que interceptam a região da pergunta. No caso da pergunta q1, pesquisa-se o agrupamento com centro c2, retornando um elemento nele contido; para a pergunta q2, são pesquisados os agrupamentos com centros c3 e c4.

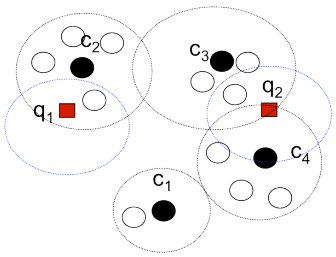


Figura 3.5 - Particionamento do espaço em quatro agrupamentos.

Sejam (q, rq) a região definida pela pergunta e (c, r) um agrupamento.

- Todos os objectos que pertencem ao agrupamento estão à distância máxima r do centro c. Portanto, se d(q,c) – r > rq, todos os elementos do agrupamento podem ser descartados. A figura 3.6 – caso (a) ilustra este caso. Por outro lado, se d(q, c) + r ≤ rq, todos os elementos do agrupamento fazem parte do conjunto resposta (figura 3.6 – caso (b)).

- Seja agora x um objecto do agrupamento. Se d(q,c) – d(c,x) > rq, então d(q, x) > rq. Logo, x pode ser descartado do resultado da pesquisa sem que d(q, x) seja calculada (figura 3.6 – caso (a)). Quando d(q, c) + d(c, x) ≤ rq, então d(q, x) ≤ rq e x pertence ao conjunto resposta (figura 3.6 – caso (b)).

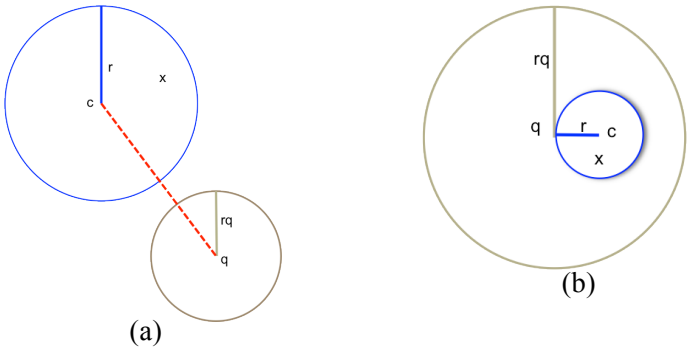


Figura 3.6 - Descarte e selecção de elementos de um agrupamento.

Quando a técnica utilizada é baseada em pivots, o descarte ou selecção de elementos é baseado nas distâncias entre os pivots e os elementos. Dados uma pergunta (q, rq) e um pivot p, um elemento x pode ser descartado através de p da seguinte forma.

(1) Pela desigualdade triangular, d(p, x) ≤ d(p, q) + d(q, x) ⇨ d(p, x) – d(p, q) ≤ d(q, x);

(2) Pela desigualdade triangular, d(p, q) ≤ d(p, x) + d(x, q) ⇨ d(p, q) – d(p, x) ≤ d(x, q);

(3) Pela simetria e d(p, x) – d(p, q) ≤ d(q, x) ⇨ d(p, x) – d(p, q) ≤ d(x, q);

(4) d(p, q) – d(p, x) ≤ d(x, q) e d(p, x) – d(p, q) ≤ d(x, q) ⇨ |d(p, q) – d(p, x)| ≤ d(x, q).

Se |d(p, q) – d(p, x)| > rq, então d(x, q) > rq e o elemento pode ser descartado. As regras aqui apresentadas encontram-se em [Chambel 2009].

Por exemplo, se um pivot p1 guardar a distância em relação a um elemento x, d(p1, x), verifica-se que x pode ser descartado do conjunto resultado quando |d(p1, q) – d(p1, x) | > rq (figura 3.7 – caso (a)). Noutros casos (figura 3.7 – caso (b)), d(p1, x) não permite descartar o elemento. Generalizando, verifica-se que um elemento pode ser descartado da pesquisa, se existir um pivot p tal que | d(p, q) – d(p, x) | > rq. Na figura 3.7 – caso (c), verifica-se que o elemento x pode ser descartado (não por p1

mas) através do pivot p2. A figura 3.7 – caso (d) mostra um exemplo onde se pode seleccionar um elemento x, sem calcular d(q, x). Como d(q, p1) + d(p1, x) ≤ rq, pela desigualdade triangular, garante-se que d(q, x) ≤ rq.

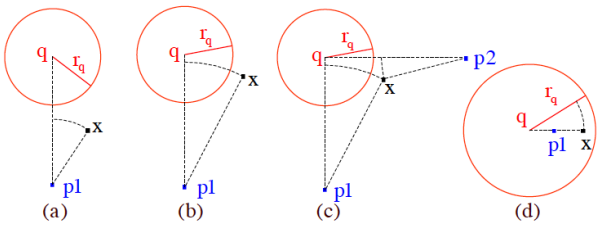


Figura 3.7 - Descarte e selecção de elementos utilizando pivots.

4 A ESTRUTURA DE DADOS MÉTRICA RLC

Neste capítulo é descrita a estrutura de dados métrica RLC. São apresentadas algumas definições, os algoritmos de inserção, remoção e pesquisa de objectos, as suas complexidades, as variantes da RLC e as parametrizações efectuadas durante os testes realizados à estrutura. Todos os conceitos fundamentais e os algoritmos encontram-se descritos em [Mamede 2005] e [Mamede 2007].

4.1 DEFINIÇÕES BÁSICAS

Nas definições seguintes, X representa uma base de dados do espaço métrico (U, d()).

Um *agrupamento* de X é um triplo representado por (c,r,I), em que c ∈ X é designado por *centro* do agrupamento, r é um número real positivo chamado *raio* do agrupamento e I é o *interior* do agrupamento. No interior do agrupamento estão objectos de X (excepto c) cujas distâncias ao centro não excedem o valor do raio, ou seja $I \subseteq \{ x \in X \mid 0 < d(x,c) \leq r\}$.

Graficamente, um agrupamento pode ser representado por uma *região (ball)* definida a partir do seu centro, delimitada pelo raio, onde estarão objectos (figura 4.1). Um objecto x pode *pertencer* a um agrupamento (c, r, I) se a sua distância a c não exceder o raio, ou seja d(x, c) ≤ r.

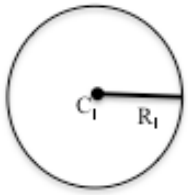


Figura 4.1 - Agrupamento de centro C_1 e raio R_1.

Uma *lista de agrupamentos* de X é uma sequência de agrupamentos representada por L = ((c_1, r_1, I_1), ... (c_n, r_n, I_n)), onde n corresponde ao número de agrupamentos, e $\bigcup_{i=1,\dots n} I_i \cup \{c_i\} = X$. Os agrupamentos são disjuntos dois a dois, ou seja, agrupamentos diferentes não têm elementos em comum, e cada objecto x pertence ao primeiro agrupamento (c_j, r_j, I_j) que o pode conter, isto é: $d(x, c_j) \leq r_j$ e $\forall_{i=1,\dots,j-1}\, d(x, c_i) > r_i$.

A figura 4.2 ilustra uma lista de agrupamentos onde L = <(c_1, r_1, {x_1,x_2,x_3}), (c_2, r_2, {x_4,x_5}), (c_3, r_3, {x_6, x_7, x_8})>.

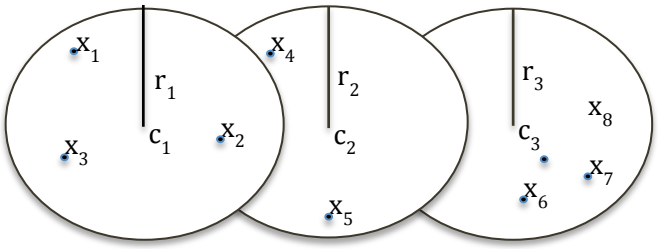


Figura 4.2 –– Lista de agrupamentos.

4.2 DEFINIÇÃO ORIGINAL DA RLC

A RLC - *Recursive Lists of Clusters* - é apresentada em [Mamede 2005] como uma estrutura de dados métrica parametrizada, genérica, dinâmica, implementada em memória central e que trata o problema da pesquisa por proximidade. Esta estrutura indexa os espaços métricos dividindo os objectos em vários níveis de listas de agrupamentos de raios fixos.

Os interiores dos agrupamentos podem ser uma lista de agrupamentos (ou seja, a própria estrutura) ou uma *folha*, que é implementada em vector. As folhas têm uma capacidade definida a priori que determina qual a forma do interior. Se o número de elementos do interior exceder a capacidade das folhas, então este é uma nova lista de agrupamentos. Por sua vez, cada nova lista de agrupamentos é a própria estrutura, com um determinado nível, considerando-se que a primeira lista de agrupamentos é uma RLC de nível zero. A figura 4.3 exemplifica uma RLC com três níveis.

Como os agrupamentos se encontram organizados em níveis, um objecto pode pertencer a vários agrupamentos, de vários níveis. Cada objecto tem associado um vector com as suas distâncias aos centros dos agrupamentos a que pertence. Esse vector está ordenado do centro do agrupamento mais interior em que o objecto está contido ao mais exterior. Nas folhas, os objectos estão ordenados de forma decrescente pelo primeiro elemento deste vector. Como se irá verificar aquando da descrição do algoritmo de pesquisa por proximidade, esta organização dos elementos do agrupamento e a existência do vector acima referido permitem optimizar a operação de pesquisa por proximidade, procurando minimizar o número de distâncias calculadas.

A figura 4.3, retirada de [Mamede 2005], esquematiza uma RLC de raio ρ cuja capacidade das folhas é igual a cinco. Cada agrupamento na imagem é identificado por um quíntuplo (c, r, l, s e I), em que c representa o centro, r o raio, l o nível, s o número de elementos do interior e I um apontador para o interior do agrupamento.

No primeiro agrupamento de nível zero, o centro é c_1. Como este agrupamento possui dezasseis elementos no seu interior, este é uma RLC de nível um, em que o seu primeiro agrupamento contém seis elementos (um centro e cinco elementos no seu interior). Por sua vez, o segundo agrupamento contém dez elementos.

O centro c_1" (que se encontra na RLC de nível dois) possui a seguinte sequência de distâncias associada: <d(c_1", c_2'), d(c_1", c_1)>. Se x for um objecto do interior desse agrupamento, o vector a si associado contém: <d(x, c_1"), d(x,c_2'), d(x, c_1)>.

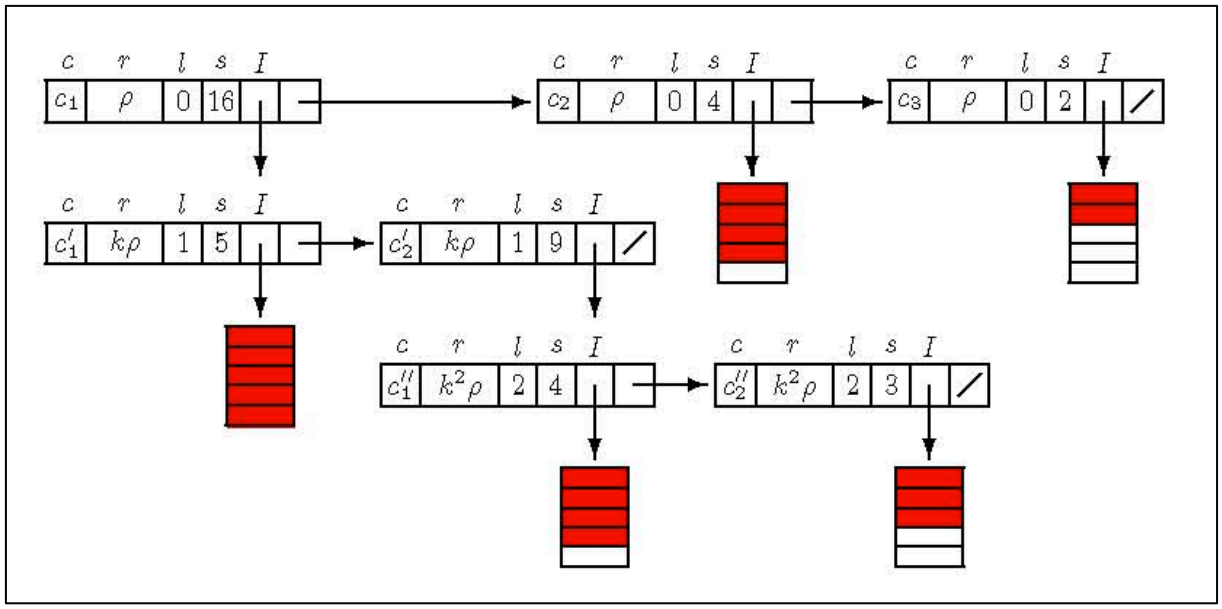


Figura 4.3 - RLC com três níveis, de raio ρ e capacidade das folhas igual a 5.

A RLC tem três parâmetros: a capacidade das folhas, o valor inicial do raio e a função do raio $\psi: R^+ \rightarrow R^+$. O raio de um agrupamento de nível n é $\psi^n(\rho)$, onde $\psi^0(\rho) = \rho$ e $\psi^n(\rho) = \psi\left(\psi^{n-1}(\rho)\right)$ para n = 1,2,...

Na figura 4.3, a RLC esquematizada apresenta os seguintes parâmetros: a capacidade das folhas é igual a cinco; o valor inicial do raio é ρ e a função do raio é definida por $\psi\,(r) = kr$. Assim sendo, o raio de um agrupamento de nível l é $k^l\rho$.

4.3 DESCRIÇÃO DOS ALGORITMOS

4.3.1 INSERÇÃO

A construção da RLC é feita por sucessivas operações de inserção de elementos.

A inserção de um novo objecto x é feita percorrendo a lista de agrupamentos até se encontrar um agrupamento ao qual esse objecto possa pertencer. Podem existir vários agrupamentos aos quais x pode pertencer. No entanto, x é inserido no primeiro agrupamento encontrado.

Se o objecto não se encaixar em nenhum dos agrupamentos existentes, é criado um novo agrupamento que é adicionado à cauda da lista de agrupamentos. Nesse novo agrupamento, x é o centro e o interior é vazio. Se, pelo contrário, x se encaixar num agrupamento, pode verificar-se uma das seguintes situações.

1. O interior do agrupamento é uma folha. Esta folha pode:

   1.1. Ter capacidade para mais um elemento. Então é feita a inserção na folha e o novo objecto é inserido numa posição que garanta a manutenção da ordenação atrás mencionada.

   1.2. Estar cheia. Neste caso, a folha é substituída por uma nova lista de agrupamentos. Os objectos da folha e o novo objecto são inseridos na lista por sucessivas operações de inserção.

2. O interior do agrupamento é uma lista de agrupamentos. Vai-se iterar a lista e inserir o objecto no primeiro agrupamento em que ele se encaixe.

A figura 4.4 descreve os vários passos do algoritmo recursivo para a inserção do objecto x.

```
RLC_Insert(L, x)
  se L = <>
    A ← (x, r, ∅); // Cria novo agrupamento
    Retorna <A | L>;
  senão
    L = <(c, r, I) | L'>
    distOC ← d(x,c);
    se distOC ≤ r
      se distOC = 0 // c = x
        Retorna L;
      senão
        A ←(c, r, I U {x}); // Insere dentro do agrupamento
        Retorna <A | L'>;
    senão
      Retorna <(c, r, I) | RLC_Insert(L', x)>;
```

Figura 4.4 - Inserção de um novo objecto na RLC.

4.3.2 REMOÇÃO

A remoção começa por pesquisar o agrupamento ao qual o objecto a remover pode pertencer. Essa pesquisa é efectuada iterando a lista de agrupamentos e calculando, em cada iteração, a distância do elemento a remover ao centro do agrupamento corrente (c,r,I). Se essa distância for menor ou igual a r, então esse agrupamento deverá conter o elemento a remover. Se não for encontrado nenhum agrupamento que verifique esta condição, então o objecto não existe e o algoritmo termina.

Ao encontrar esse agrupamento, podem ocorrer duas situações:

1. O objecto a remover é o centro do agrupamento. Neste caso, o agrupamento é removido e é feita a reinserção de todos os objectos do seu interior na lista de agrupamentos corrente, a partir da posição seguinte à do agrupamento removido.

2. O objecto encontra-se no interior do agrupamento. O elemento é removido do seu interior.

Após a remoção de um objecto do interior de um agrupamento implementado em lista, se esse interior tiver tantos elementos quanto a capacidade das folhas, a lista é removida e os seus objectos guardados numa folha.

A figura 4.5 descreve os vários passos do algoritmo de remoção do elemento x da RLC.

```
RLC-Delete(L, x)
  se L = <>
    Retorna L;
  senão
    L = <(c, r, I) | L'>
    distOC ← d(x,c);
    se distOC ≤ r
      se distOC = 0 // c = x
        L'' ← Reinserir cada elemento de I em L';
        Retorna L'';
      senão
        A ← (c, r, I-{x}) // Remove dentro do agrupamento
        Retorna <A | L'>;
    senão
      Retorna <(c, r, I) | RLC-Delete(L', x)>;
```

Figura 4.5 – Remoção de um elemento da RLC.

4.3.3 PESQUISA POR PROXIMIDADE

O algoritmo de pesquisa vai iterar a lista de agrupamentos à procura dos elementos que pertencem à região da pergunta, formando assim o conjunto resposta. Em cada iteração é calculada a distância da pergunta ao centro do agrupamento corrente e é determinada a relação entre eles. Esta relação é baseada nessa distância, podendo surgir duas situações: ou a região da pergunta contém o centro do agrupamento corrente ou não o contém. As figuras 4.6 e 4.7 exemplificam cada uma destas situações.

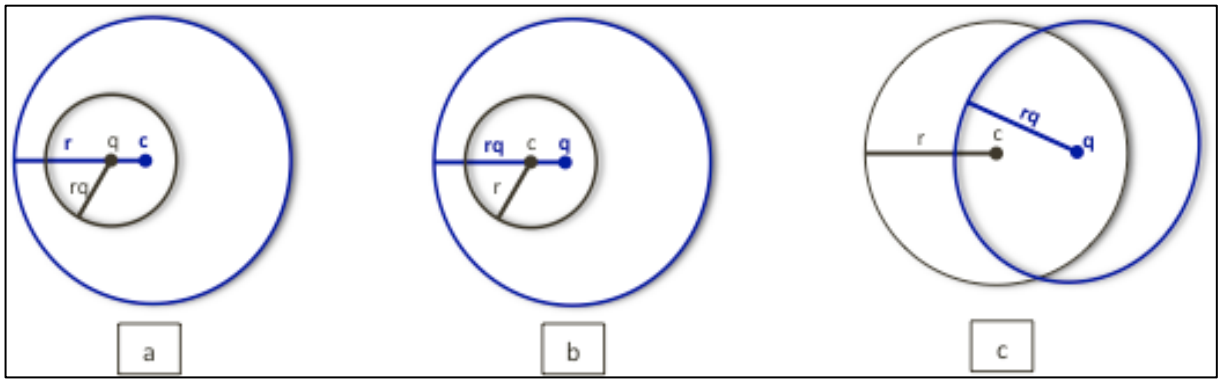


Figura 4.6 - Região da pergunta contém o centro do agrupamento

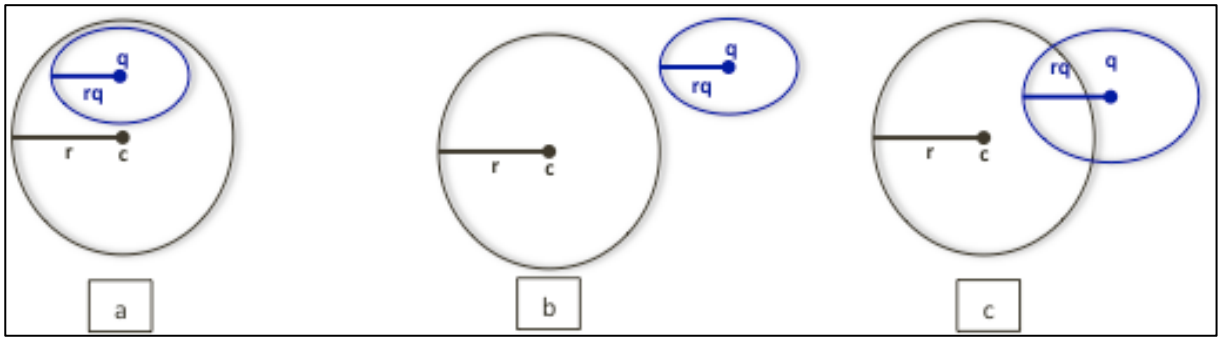


Figura 4.7 - Região da pergunta não contém o centro do agrupamento.

Como se pode verificar na figura 4.6, quando a região da pergunta contém o centro do agrupamento, ela pode estar completamente contida no agrupamento (caso (a)), conter o agrupamento (caso (b)) ou interceptar o agrupamento sem o conter ou estar contida nele (caso (c)). Se, ao contrário, ela não contém o centro, então podem ocorrer os seguintes casos: a região da pergunta está completamente contida no agrupamento, as duas regiões são disjuntas ou então interceptam-se. Estas situações estão exemplificadas nos casos (a), (b) e (c) da figura 4.7.

Os casos a seguir apresentados mostram as acções decorrentes do algoritmo de pesquisa em cada uma das situações referidas.

1. A região da pergunta (q, r_q) contém o centro do agrupamento (c, r, I), ou seja $d(q, c) \leq r_q$. O centro c é automaticamente adicionado ao conjunto resposta. Nestes casos, a região da pergunta pode:

A. Estar contida no agrupamento (figura 4.6 – caso (a)).

   É necessário efectuar a pesquisa no interior do agrupamento, sendo coleccionados os objectos nesse interior que pertencem à região da pergunta. Como a região da pergunta está completamente contida no agrupamento, a pesquisa termina.

B. Conter o agrupamento (figura 4.6 – caso (b)).

   Todos os objectos do interior do agrupamento são automaticamente adicionados ao conjunto resposta, sem que seja necessário calcular a distância entre a pergunta e os objectos do interior

do agrupamento. A pesquisa continua na restante lista de agrupamentos, pois a região da pergunta pode interceptar mais agrupamentos.

C. Interceptar o agrupamento, sem o conter nem estar contida nele (figura 4.6 – caso (c)).

São adicionados ao conjunto resposta os elementos do agrupamento que pertencem à região da pergunta, através de uma pesquisa no seu interior. Como a região da pergunta pode interceptar mais agrupamentos, a pesquisa prossegue.

2. A região da pergunta (q,r_q) não contém o centro do agrupamento (c, r, I). Sucede quando a distância de q a c é maior que o raio da pergunta, $d(q, c) > r_q$. Podem ocorrer os seguintes casos:

A. A região da pergunta está contida no agrupamento (figura 4.7 – caso (a)).

É efectuada uma pesquisa no interior do agrupamento. São adicionados os objectos que estão no agrupamento e que pertencem à região da pergunta. A pesquisa termina.

B. A região da pergunta é disjunta do agrupamento (figura 4.7 – caso (b)).

O interior do agrupamento é ignorado. A pesquisa continua.

C. A região da pergunta intercepta o agrupamento, sem o conter ou estar contida nele (figura 4.7 – caso (c)).

É efectuada uma pesquisa no interior do agrupamento e adicionados ao conjunto resposta os objectos que pertencem à região da pergunta. A pesquisa continua na lista de agrupamentos.

Uma das estratégias utilizadas para a optimização das pesquisas é, sempre que possível, descartar ou seleccionar objectos sem calcular distâncias. Isto é feito por aplicação das regras a seguir apresentadas, que se encontram demonstradas em [Mamede 2005]. Sendo x um objecto, (c, r, I) um agrupamento, (q, r_q) uma pergunta e $m = d(q,c) – r_q$ então:

- Se $d(x,c) < m$, então $d(x,q) > r_q$. Neste caso x não pertence ao conjunto resposta.
- Se $d(x,c) \leq -m$, então $d(x, q) \leq r_q$. Concluiu-se que x pertence ao conjunto resposta.

Quando m é positivo, o seu valor é o limite mínimo para a distância entre os objectos do agrupamento e o seu centro, abaixo do qual os objectos podem ser descartados sem calcular outras distâncias. Se m é negativo, o seu simétrico pode ser usado de forma semelhante para incluir objectos.

Sempre que se entra num agrupamento é guardado o limite mínimo. É construído um vector com estas distâncias, designado por *minDists*, ordenado do agrupamento mais interior ao mais exterior e que acompanha todo o processo de pesquisa. Quando a pesquisa entra no interior de um agrupamento, se o mesmo estiver implementado em vector, é necessário descobrir os pontos nele contidos que pertencem ao conjunto resposta. Para isso, percorre-se o vector. Então, fazendo uso das regras acima, do vector de distâncias associado a cada ponto (*dists*) e do vector das distâncias mínimas acima referido, decide-se se o ponto é descartado, adicionado à resposta ou nenhuma das duas situações

anteriores. Neste último caso, há que recorrer ao cálculo da distância do ponto ao objecto da pergunta para determinar se este pertence ou não ao conjunto resposta. Essa decisão é feita aplicando as seguintes regras:

- Se dist[0] < minDists[0], nem o ponto, nem nenhum dos pontos seguintes pertencem ao conjunto resposta e a iteração termina.
- Se dist[0] + minDists[0] ≤ 0, todos os pontos são adicionados ao conjunto resposta e a iteração termina.
- Se dist[i] < minDists[i], para algum i ≥ 1, o ponto não pertence ao conjunto resposta e a iteração prossegue com o próximo ponto.
- Se dist[i] + minDists[i] ≤ 0, para algum i ≥ 1, então o ponto pertence ao conjunto resposta e a iteração prossegue com o próximo ponto.
- Nos restantes casos, calcula-se a distância do ponto ao objecto da pergunta para determinar se este pertence ou não ao conjunto resposta.

Os passos do algoritmo recursivo encontram-se na figura 4.8.

4.4 VARIANTES DA RLC

Como já foi dito, a versão original da RLC tem três parâmetros: a capacidade das folhas, o valor inicial do raio e a função do raio. Esta definição da RLC é designada neste documento por *RLC_2005*. Na secção 4.6 apresenta-se uma função do raio que foi utilizada nos testes de desempenho realizados à estrutura.

RLC_2006

A primeira adaptação da RLC para memória secundária, denotada por *RLC_2006*, é apresentada por Carlos Rodrigues [Rodrigues 2006]. Essa adaptação é realizada a partir da RLC_2005, efectuando um mapeamento da estrutura para um modelo onde o endereçamento se faz por páginas (ou blocos). As listas de agrupamentos e as folhas são transformadas em sequências de páginas em que cada página contém uma parte da lista ou da folha.

```
RLC-Search(L, (q, rq), minDists, R)
 se L = <>
  Retorna R;
 senão
  L = <(c, r, I) | L'>
  distQC ← d(q,c);
  se distQC ≤ rq //Região da pergunta contém centro
   se distQC + rq ≤ r //Região da pergunta contida no agrupamento
    R' ← C-Search((c,r,I),(q,rq),minDists,distQC,RU{c}); //Pesquisa em I
    Retorna R';
   senão
    se distQC + r ≤ rq //Região da pergunta contém agrupamento
     Retorna RLC-Search(L', (q, rq), minDists, R U {c} U I);
    senão //Região da pergunta intercepta agrupamento
     R' ← C-Search((c,r,I),(q,rq),minDists,distQC,RU{c}); //Pesquisa em I
     Retorna RLC-Search(L', (q, rq), minDists, R');
  senão //Região da pergunta não contém centro
   se distQC + rq ≤ r //Região da pergunta contida no agrupamento
    R' ← C-Search((c,r,I),(q,rq),minDists,distQC,R); //Pesquisa em I
    Retorna R';
   senão
    se distQC > rq + r //Regiões da pergunta e do agrupamento disjuntas
     Retorna RLC-Search(L', (q, rq), minDists, R);
    senão //Região de pergunta intercepta agrupamento
     R' ← C-Search((c,r,I),(q,rq),minDists,distQC,R); //Pesquisa em I
     Retorna RLC-Search(L', (q, rq), minDists, R');
```

Figura 4.8 – Pesquisa por proximidade na RLC.

Para simplificar a implementação, o mapeamento é acompanhado de algumas regras relacionadas com a ocupação das páginas [Rodrigues 2006], considerando-se que:

- Não podem existir páginas com zero elementos.
- No caso de não estarem completamente ocupadas, não podem existir quaisquer posições vazias entre os elementos presentes na página.
- Nenhum elemento pode ocupar mais que o espaço disponível numa página.

A última regra restringe esta implementação, uma vez que todos os pontos guardados na RLC têm a eles associada uma sequência de distâncias aos centros dos agrupamentos onde estão contidos. Se a RLC crescer muito em profundidade, os comprimentos destas sequências de distâncias podem tornar os elementos das folhas demasiado grandes para caberem numa única página. Uma outra restrição desta adaptação consiste no facto da mesma não estar preparada para armazenar qualquer tipo de objectos, tornando-a assim não genérica (guarda apenas pontos de $\mathbb{R}^n$).

RLC_2007

Em 2007 [Mamede 2007] é apresentada uma variante da RLC, designada neste documento por *RLC_2007*. Nessa variante, a estrutura é simplificada passando a depender de menos um parâmetro, a função do raio. O raio é fixo, predefinido e igual para todos os agrupamentos, independentemente do nível a que se encontram.

Na RLC_2007, a estrutura cresce mais em profundidade e menos em largura porque, geralmente, sempre que uma folha é transformada numa nova lista, porque atingiu a sua capacidade, o primeiro agrupamento a ser criado nessa nova lista fica com a maior parte dos pontos do agrupamento do nível acima. Assim, os pontos não são distribuídos de forma equilibrada pelos agrupamentos da lista. A figura 4.9, retirada de [Sarmento 2010], exemplifica o interior de um agrupamento cujo centro é representado pela letra c e que é constituído por uma lista com dois agrupamentos, cujos centros são c1 e c2 respectivamente. Uma vez que todos os agrupamentos têm o mesmo raio, é muito provável que o primeiro agrupamento (cujo centro é c1) possua mais pontos que o segundo.

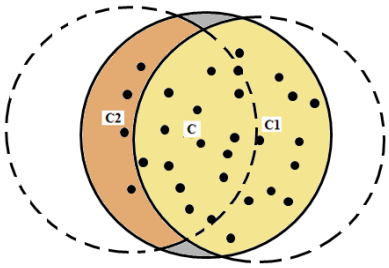


Figura 4.9 - Exemplo do interior de um agrupamento da RLC_2007.

RLC_2010

Dando seguimento ao trabalho de Carlos Rodrigues, Ângelo Sarmento apresenta em [Sarmento 2010] a segunda adaptação da estrutura a memória secundária, chamada *RLC_2010*. A RLC_2010 já é genérica, embora continue a exigir que qualquer elemento caiba numa única página.

Um dos objectivos da RLC_2010, dadas as suas limitações, era fazer a estrutura crescer mais em largura e menos em profundidade. Para isso, os agrupamentos não deviam ter todos o mesmo raio independentemente do seu nível. Assim, foi proposta a redução progressiva dos raios dos agrupamentos à medida que a profundidade aumenta. É apresentada uma função que permite calcular o raio de qualquer agrupamento da estrutura, definida por r = r' / (l + 1), em que r' representa o valor do raio dos agrupamentos de nível zero e l o nível do agrupamento.

A figura 4.10 (retirada de [Sarmento 2010]) mostra as vantagens da redução progressiva dos raios dos agrupamentos. Ilustra a distribuição dos objectos do interior de um agrupamento da RLC_2010, que consiste numa lista com sete agrupamentos. É possível observar-se uma distribuição mais equilibrada dos pontos pelos agrupamentos da lista.

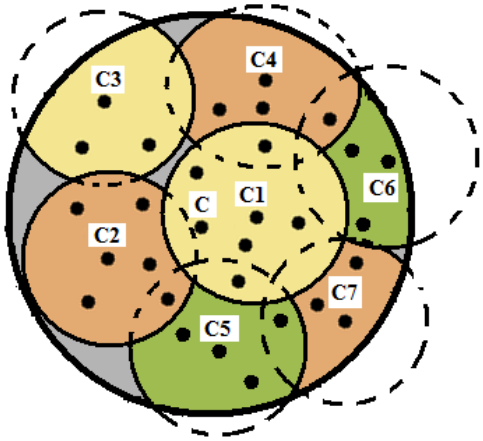


Figura 4.10 - Exemplo do interior de um agrupamento da RLC_2010.

4.5 COMPLEXIDADES

A análise das complexidades temporais dos algoritmos da RLC_2007 é realizada em [Mamede 2007], onde é provado que o número médio de distâncias calculadas para carregar uma base de dados com n objectos é O(n log n). Se a estrutura tiver n pontos (com n > 0), o número esperado de cálculos de distâncias entre objectos efectuados numa operação de inserção é $O(\log n)$, numa operação de remoção é $O(\log^2 n)$ e numa pesquisa por proximidade é $O(n^{\gamma})$, para algum $\gamma \in [0, 1]$.

4.6 PARAMETRIZAÇÕES E TESTES REALIZADOS

Nesta secção é apresentada a lista de parametrizações utilizadas nos testes de desempenho efectuados à RLC. As tabelas 4.1, 4.2, 4.3 e 4.4 indicam, para cada uma das variantes da RLC, os valores dos parâmetros usados com cada espaço métrico. Esses valores foram aqueles com os quais a RLC teve bons desempenhos, tendo sido obtidos através da realização de vários testes experimentais.

Tabela 4.1 - Parametrizações da RLC_2005.

$[0,1]^k$ com distribuição uniforme; distância euclidiana. [Mamede 2005] **k = 4, 6, 8, 10, 12, 14, 16, 18, 20.**		
Capacidade das folhas	**Raio**	**Função do raio**
16	$r = \frac{\sqrt{k}}{\eta_1}$ η_1- constante positiva dependente de k	$\psi(r) = \frac{r}{\eta_2}, r > 0$ η_2 - constante positiva dependente de k.

Tabela 4.2 - Parametrizações da RLC_2006.

$[-50, 50]^k$ com distribuição uniforme; distância euclidiana. [Rodrigues 2006] **$[a,b]^k$ com distribuição normal (média 0, desvio padrão 1); distância euclidiana.** [Rodrigues 2006] **k = 4, 8, 12.**		
Capacidade das folhas	**Raio**	**Dimensão das páginas**
80	$r = i * \frac{\sqrt{k}}{d}$ • i = b – a, para o universo $[a,b]^k$. • d = 4,5 para a distribuição uniforme; d = 2,5 para a distribuição normal.	4096 bytes

Na próxima tabela, o domínio "imagens de rostos" refere-se a quatro universos distintos, denotados por Faces94, JAFFE, AT&T e Yalefaces. Em relação aos espaços métricos de trechos de música, foram utilizados dois universos: um na dimensão melodia, com intervalos melódicos, e outro na dimensão timbre, baseado nos *Mel-Frequency Cepstral Coefficients* (MFCCs – ver [Costa 2009]).

Tabela 4.3 - Parametrizações da RLC_2007.

Capacidade das folhas	**Raio**
$[0,1]^k$ com distribuição uniforme; distância euclidiana. [Mamede 2007] **k = 10, 12, 14, 16, 18, 20.**	
16	$r = \frac{\sqrt{k}}{4,5}$
Dicionário de Alemão; distância de edição. [Mamede e Barbosa 2007]	
16	5
Dicionários de Espanhol, Francês, Inglês, Italiano e Português; distância de edição. [Mamede e Barbosa 2007]	
16	4
Dicionário de Alemão; distância EED. [Barbosa 2009]	
16	8

Dicionários de Espanhol, Francês, Inglês, Italiano e Português; distância EED. [Barbosa 2009]	
16	6
Imagens de rostos; distância euclidiana. [Chambel 2009]	
18 (Faces94)	2.776
16 (JAFFE)	5.790
6 (AT&T)	1.551
7 (Yalefaces)	10.310
Imagens de rostos; distância de Manhattan. [Chambel 2009]	
18 (Faces94)	10.445
18 (JAFFE)	17.486
7 (AT&T)	5.514
13 (Yalefaces)	35.067
Imagens de rostos; distância Mahalanobis-L1. [Chambel 2009]	
17 (Faces94)	6,19
10 (JAFFE)	10,34
7 (AT&T)	10,52
7 (Yalefaces)	15,32
Imagens de rostos; distância Mahalanobis-L2. [Chambel 2009]	
17 (Faces94)	1,58
11 (JAFFE)	2,76
5 (AT&T)	2,72
5 (Yalefaces)	4,06
Intervalos melódicos; distância de edição. [Costa 2009]	
29	30
Assinaturas de MFCCs; distância de Manhattan. [Costa 2009]	
38	6
Assinaturas de MFCCs; distância euclidiana. [Costa 2009]	
44	2,64

Tabela 4.4 - Parametrizações da RLC_2010.

Capacidade das folhas	Raio	Dimensão das páginas
Dicionário de Alemão; distância de edição. [Sarmento 2010]		
100	10	8.192 bytes
Dicionário de Inglês; distância de edição. [Sarmento 2010]		
100	6	8.192 bytes
Histogramas de imagens; distância euclidiana. [Sarmento 2010]		
50	0,71	16.384 bytes
Imagens de rostos; distância de Manhattan. [Sarmento 2010]		
25	23.160	4.096 bytes

4.7 IMPLEMENTAÇÃO DA RLC

A implementação da RLC utilizada nos testes efectuados nesta dissertação foi realizada na linguagem JAVA e foi gentilmente cedida pela Prof.ª Margarida Mamede. A seguir são apresentadas as interfaces e as classes que a compõem. A figura 4.11 mostra o respectivo diagrama.

METRICDS

Interface que representa uma estrutura de dados métrica. Disponibiliza as principais operações sobre estruturas de dados métricas: inserção de um ou vários pontos, remoção de um ponto e pesquisa por proximidade. Todos os métodos possuem um argumento chamado *results*, que contém, no final de cada operação, o número de distâncias calculadas.

RLC::METRICDS

Classe que implementa a estrutura de dados métrica RLC. Implementa as interfaces MetricDS e ClusterInterior.

CLUSTERINTERIOR

Interface que representa o interior de um agrupamento. Estende a interface *Iterable* e define todas as operações que um interior de um agrupamento deverá suportar.

BUCKET::CLUSTERINTERIOR

Classe que implementa uma folha. Implementa a interface ClusterInterior. Uma folha tem o vector de pontos, o número de pontos e o seu nível.

CLUSTER

Interface que representa um agrupamento. Herda da classe *Iterable* e define todas as operações que um agrupamento deverá suportar.

CLUSTERCLASS::CLUSTER

Classe que implementa um agrupamento. Implementa a interface Cluster. Um agrupamento tem um ponto da RLC, que é o centro do agrupamento, o raio, o nível e o seu interior.

CLUSTERITERATOR::ITERATOR

Classe que implementa o iterador de pontos de um agrupamento.

BUCKETITERATOR::ITERATOR

Classe que implementa o iterador de pontos de uma folha.

RLCITERATOR::ITERATOR

Classe que implementa o iterador de pontos de uma RLC.

POINT

Interface que representa um ponto. Define a operação de distância entre dois pontos.

POINTRLC::POINT

Classe que implementa um ponto guardado na RLC. Implementa a interface Point. Um ponto guardado na RLC tem um ponto e um vector de distâncias.

ARRAYOFDOUBLE

Classe auxiliar que implementa um vector extensível de números reais. É utilizada, por exemplo, para implementar o vector *minDists*.

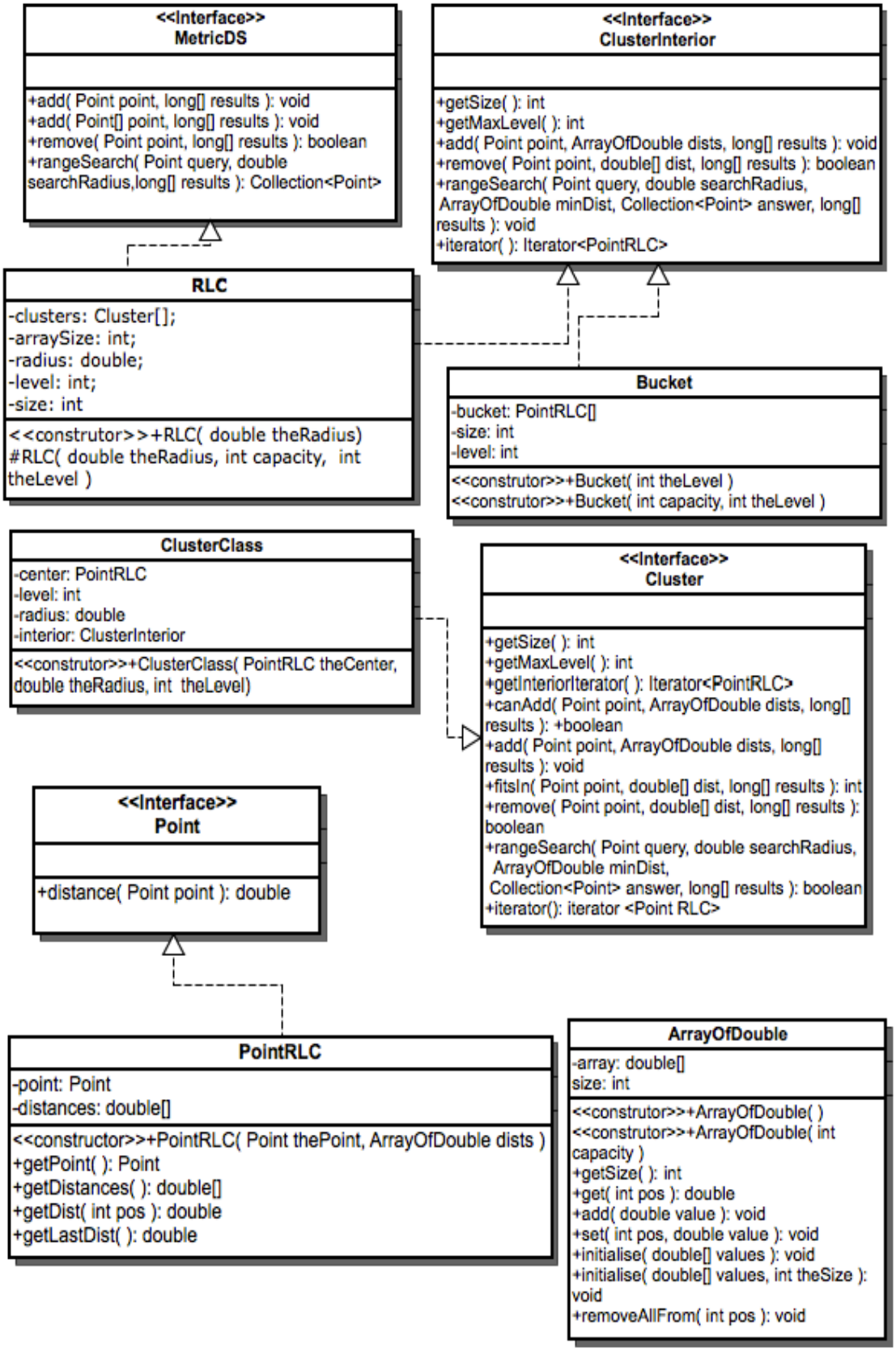


Figura 4.11 - Diagrama de interfaces e classes da RLC.

5 NOVA VARIANTE DA RLC

Como já foi referido no capítulo 4, a versão original da RLC tem três parâmetros: a capacidade das folhas, o valor inicial do raio dos agrupamentos e a função do raio. Nas variantes de 2006 e 2007, desaparece o terceiro parâmetro e todos os agrupamentos têm o mesmo raio. Consequentemente, a profundidade da estrutura aumenta. Com o objectivo de a diminuir, é proposta na RLC_2010 uma fórmula que reduz progressivamente os raios dos agrupamentos à medida que a profundidade destes aumenta.

Por outro lado, um dos problemas que podem surgir com as estruturas de dados métricas que utilizam listas de agrupamentos de raio fixo é o comprimento das listas ser muito grande. Verifica-se que os agrupamentos têm tendência a ficar vazios à medida que se avança na lista [Chávez e Navarro 2000]. Uma estratégia que pode ser usada para resolver este problema é aumentar os raios dos agrupamentos, à medida que se avança na lista. Como a área coberta pelos agrupamentos aumenta, o número de objectos não decresce tanto e o número de agrupamentos necessários diminui.

Mas a RLC é formada por várias listas de agrupamentos, organizadas em níveis. Coloca-se então a questão: deve-se aumentar os raios dos agrupamentos ao longo de qualquer lista, independentemente do nível da lista? Nesta proposta, o raio cresce na lista de agrupamentos de nível zero, para o comprimento da primeira lista não ser muito grande. Nas outras listas, para que a estrutura não cresça muito em profundidade, quer-se garantir que os raios são todos inferiores ao raio do agrupamento pai (como na variante de 2010, que deu bons resultados). Por esta razão, o raio não cresce à medida que se avança na lista. Como também não deve diminuir, para não aumentar o comprimento da lista, mantém-se constante.

Nesta secção é proposta uma nova variante da RLC. Para a sua caracterização é necessário definir a capacidade das folhas, o *raio inicial* (que é o raio do primeiro agrupamento de nível zero), a função que calcula o raio dos restantes agrupamentos de nível zero (que tenta impedir que o comprimento da lista de nível zero seja grande) e a função que calcula os raios dos agrupamentos dos restantes níveis (que tenta impedir que a profundidade da estrutura seja muito grande).

Para que a RLC não cresça demasiado no nível zero, é proposta uma função que determina o raio dos agrupamentos do nível zero. É definida por:

$$r0(posAgrupamento) = ri + posAgrupamento * \alpha,$$

onde ri é o raio inicial, posAgrupamento é a posição do agrupamento na lista de agrupamentos e α é um número real positivo. Como se considera que as posições dos agrupamentos na lista são 0, 1, 2, 3 ..., os raios dos primeiros quatro agrupamentos de nível zero são ri, ri + α, ri + 2α, ri + 3α. O valor α é chamado o *incremento do raio*.

Com a função r0 os raios dos agrupamentos vão crescendo à medida que se avança na lista. A figura 5.1 ilustra dois agrupamentos do nível zero, o primeiro com raio igual a dois e o segundo com raio igual a três.

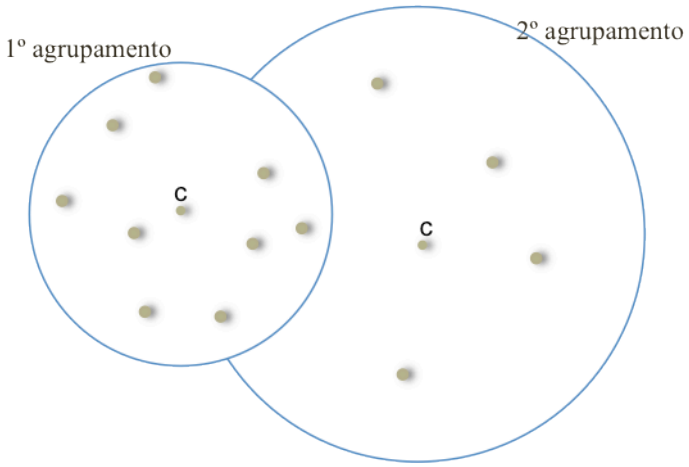


Figura 5.1 - Exemplo de dois agrupamentos do nível zero.

Para que a estrutura não cresça muito em profundidade, deve-se diminuir os raios dos agrupamentos, à medida que a profundidade destes aumenta. São consideradas três funções, r1, r2 e r3, definidas por:

$$r1(\text{nível}) = ri / \text{potência}(2, \text{nível}),$$

$$r2(\text{nível}) = ri / (\text{nível} + 1),$$

$$r3(\text{nível}) = ri / (0.5 * \text{nível} + 1),$$

onde ri é o raio inicial e nível é o nível do agrupamento. A primeira função, r1, divide o raio inicial pela potência de dois cujo expoente é o nível do agrupamento. As funções r2 e r3 dividem o raio inicial aproximadamente pelo nível e por metade do nível. A partir das definições, verifica-se que a redução do raio é mais acentuada na função r1 e mais suave na função r3. Convém referir que as duas primeiras funções foram estudadas em [Sarmento 2010], sendo r2 a função usada na RLC_2010. A função r3 é original.

Nesta nova variante, é usada a função r0 para calcular os raios dos agrupamentos do nível zero e deve-se escolher uma das funções r1, r2 ou r3 para calcular os raios dos agrupamentos dos restantes níveis.

Falta ainda responder às seguintes questões. Qual deve ser o valor do raio inicial (ri)? Qual deve ser o valor do incremento do raio (α)? Qual deve ser a função do raio para os agrupamentos de nível positivo (r1, r2 ou r3)? Qual deve ser a capacidade das folhas?

É natural que as respostas a estas questões dependam das características do espaço métrico, nomeadamente, da distribuição das distâncias entre os elementos (distintos) do universo. São então apresentadas as duas seguintes propostas.

- O raio inicial (ri) é a média das distâncias.

  Com esse valor, espera-se que o primeiro agrupamento de nível zero tenha muitos elementos.

- O incremento do raio (α) é o desvio padrão das distâncias.

  Espera-se que o comprimento da lista de nível zero seja sempre muito reduzido.

As duas últimas questões não serão respondidas de imediato. No entanto, podem-se fazer os seguintes comentários.

- Em relação à função do raio para os agrupamentos de nível positivo, sabe-se que, quando o histograma das distâncias é mais concentrado à volta da média, as pesquisas por proximidade são mais difíceis [Chávez e Navarro 2005]. Mais precisamente, a dificuldade aumenta quando a média é maior e aumenta quando a variância é menor. Ou seja, a dificuldade aumenta com a dimensionalidade intrínseca do espaço métrico (definida por $\frac{\mu^2}{2.\sigma^2}$). Assim, espera-se que os espaços métricos com maior dimensionalidade intrínseca requeiram reduções do raio mais acentuadas.
- Quanto à capacidade das folhas, todos os testes experimentais à RLC têm revelado que a influência deste parâmetro é menor do que a dos outros. Em memória central, o valor dezasseis tem sido muitas vezes o escolhido.

Os resultados experimentais, apresentados no capítulo sete, ajudarão a esclarecer as questões relacionadas com a capacidade das folhas e a escolha da função do raio, tendo em conta a dimensionalidade intrínseca do espaço métrico.

6 ESPAÇOS MÉTRICOS SELECCIONADOS

Neste capítulo são descritos os espaços métricos seleccionados para os testes realizados à RLC. São quinze espaços métricos, todos com dados reais, de diferentes domínios: oito de dicionários, três de imagens de rostos, dois de histogramas de imagens e dois de séries temporais (trajectórias de furacões e percursos de uma pessoa). Para cada um deles, é apresentada a sua descrição e feita uma análise da distribuição das distâncias entre os elementos.

Para analisar a distribuição das distâncias de cada espaço métrico, calcularam-se todas as distâncias entre elementos distintos do universo, sem repetições. Ou seja, se n for o número de elementos do universo, obtiveram-se n x (n-1) / 2 distâncias. A distribuição das distâncias (normalizada em relação ao número total de distâncias) é apresentada no respectivo histograma. Quando a função de distância é contínua, o histograma só tem cem intervalos de igual dimensão. Esses intervalos têm valores diferentes, para cada um dos espaços métricos, pois foram calculados tendo em conta as distâncias máxima e mínima e o número de intervalos desejados. Foi utilizada a seguinte fórmula para calcular a dimensão de cada intervalo: δ = (distância máxima – distância mínima) / 100. Assim, o primeiro intervalo corresponde às distâncias [distância mínima, distância mínima + δ[, o segundo intervalo [distância mínima + δ, distância mínima + 2 δ[, e assim sucessivamente, até ao centésimo intervalo, que é definido por [distância mínima + 99 δ, distância mínima + 100 δ].

Para além da média e da variância do conjunto de distâncias, também se calculou a dimensionalidade intrínseca do espaço métrico (definida, na secção 2.3, como o quociente entre o quadrado da média e o dobro da variância). Como se pode verificar, os espaços métricos têm dimensionalidades intrínsecas diferentes. Recorde-se que, quando o quociente é pequeno, a dimensionalidade intrínseca do espaço é baixa [Chávez et al. 2001].

6.1 DICIONÁRIOS

Os espaços métricos dicionários são todos formados por um dicionário de uma determinada língua e pela distância de edição. A métrica foi definida na secção 2.2.1 e os dicionários foram obtidos a partir de http://www.sisap.org/Home.html, excepto o dicionário de Português, que foi retirado de http://packages.debian.org/stable/text/. Por questões de eficiência, a função que calcula a distância foi implementada iterativamente, aplicando a técnica da programação dinâmica.

Os dicionários são de oito línguas: Alemão, Espanhol, Francês, Holandês, Inglês, Italiano, Norueguês e Português. Como se pode verificar pela tabela 6.1, têm tamanhos diferentes, sendo o maior o de Português, com 407.583 palavras, e o menor o de Inglês, com 69.069 palavras. Os comprimentos das menores palavras são um ou dois, e a maior palavra é holandesa e tem 38 caracteres. As médias das distâncias não variam muito (entre 8,35 e 11,74), mas há diferenças significativas nas variâncias (entre 3,88 e 10,00). Em relação à dimensionalidade intrínseca, o espaço de menor dimensionalidade é o de Norueguês (5,51) e o de maior dimensionalidade é o de Português (11,14).

Tabela 6.1 - Algumas estatísticas sobre os espaços métricos de dicionários.

	Número de palavras	Comprimento da maior palavra	Comprimento da menor palavra	Distância máxima	Distância mínima	Média	Variância	Dimens. intrínseca
Alemão	74.916	33	2	33	1	11,74	9,33	7,38
Espanhol	86.061	21	1	21	1	8,40	4,04	8,73
Francês	138.257	25	1	25	1	9,03	3,88	10,51
Holandês	229.328	38	2	38	1	10,38	7,52	7,16
Inglês	69.069	21	1	21	1	8,35	4,10	8,49
Italiano	116.879	24	2	24	1	9,10	3,97	10,44
Norueguês	85.560	32	1	32	1	10,49	10,00	5,51
Português	407.583	27	1	27	1	9,52	4,07	11,14

As figuras 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7 e 6.8 apresentam os respectivos histogramas. É fácil verificar que as curvas são, de certa forma, semelhantes.

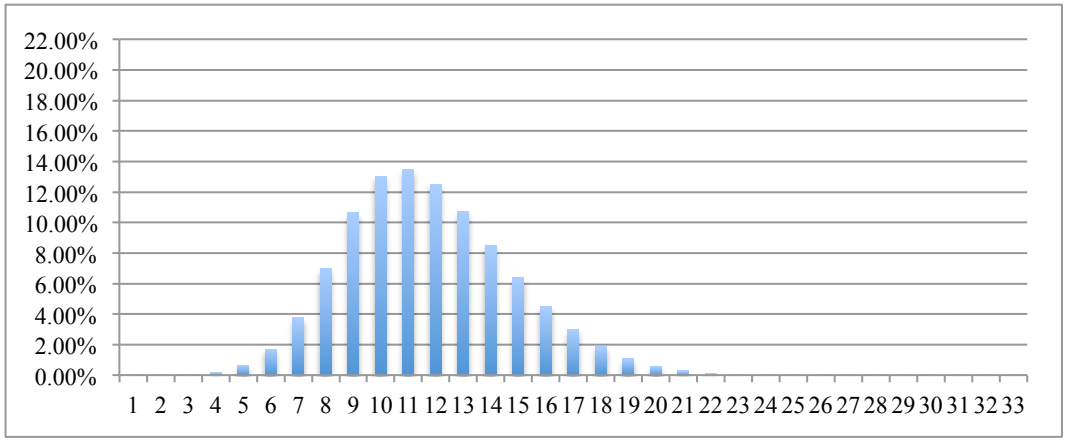


Figura 6.1- Histograma das distâncias do dicionário de Alemão.

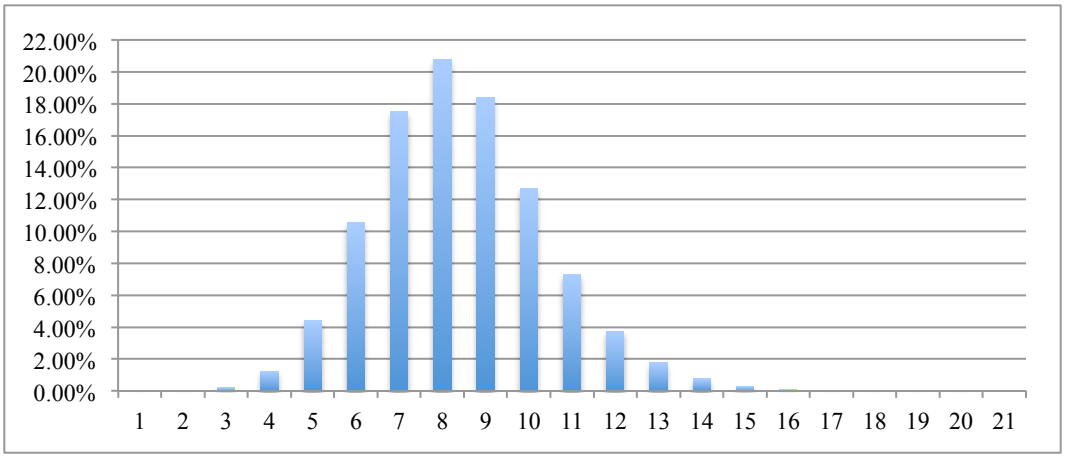


Figura 6.2 Histograma das distâncias do dicionário de Espanhol.

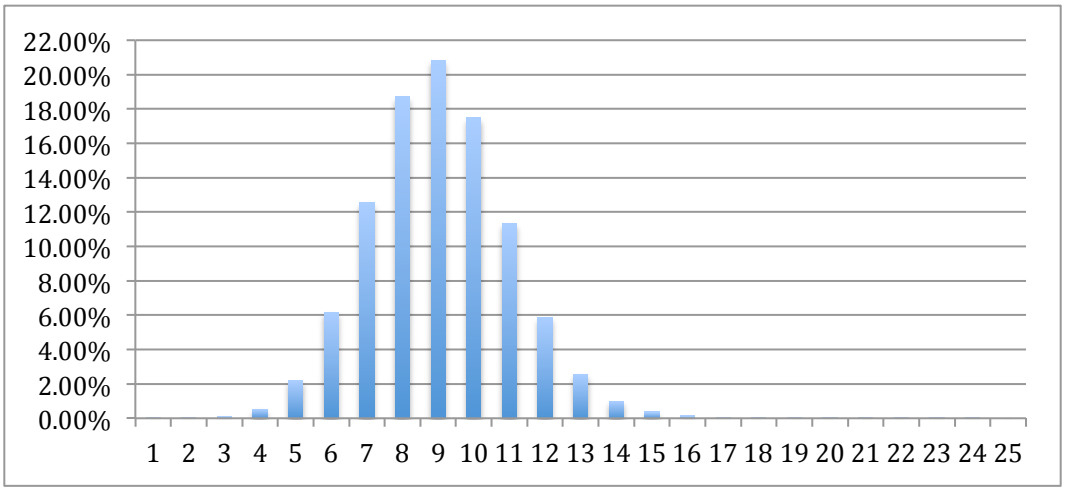


Figura 6.3 Histograma das distâncias do dicionário de Francês.

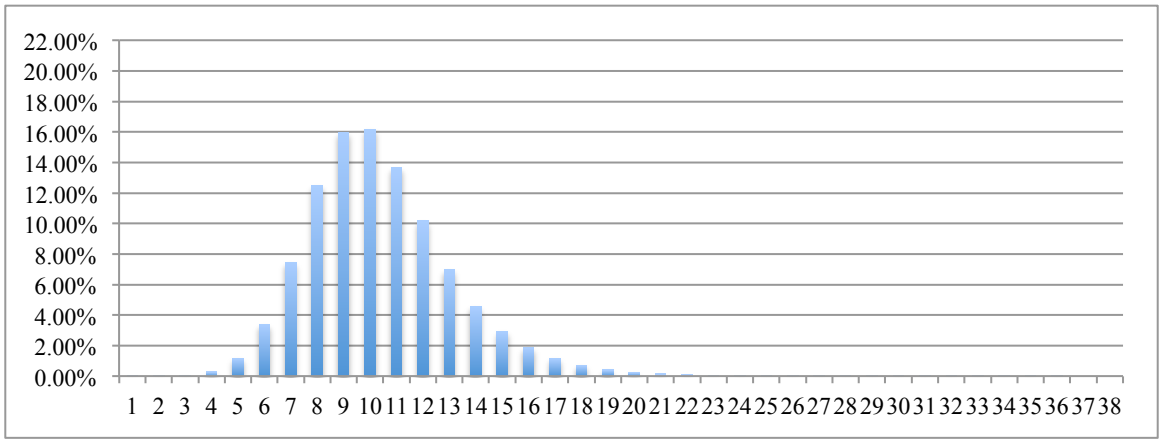


Figura 6.4 Histograma das distâncias do dicionário de Holandês.

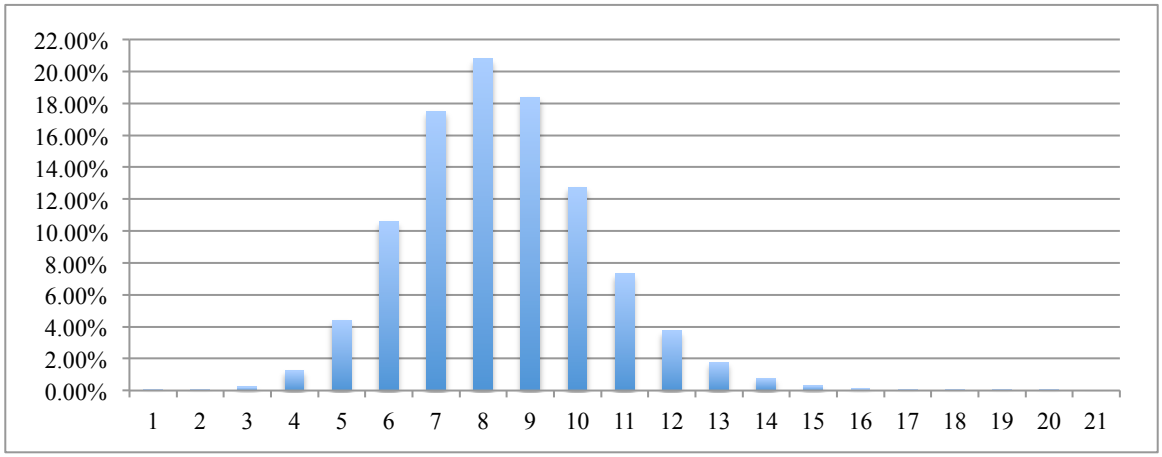


Figura 6.5 Histograma das distâncias do dicionário de Inglês.

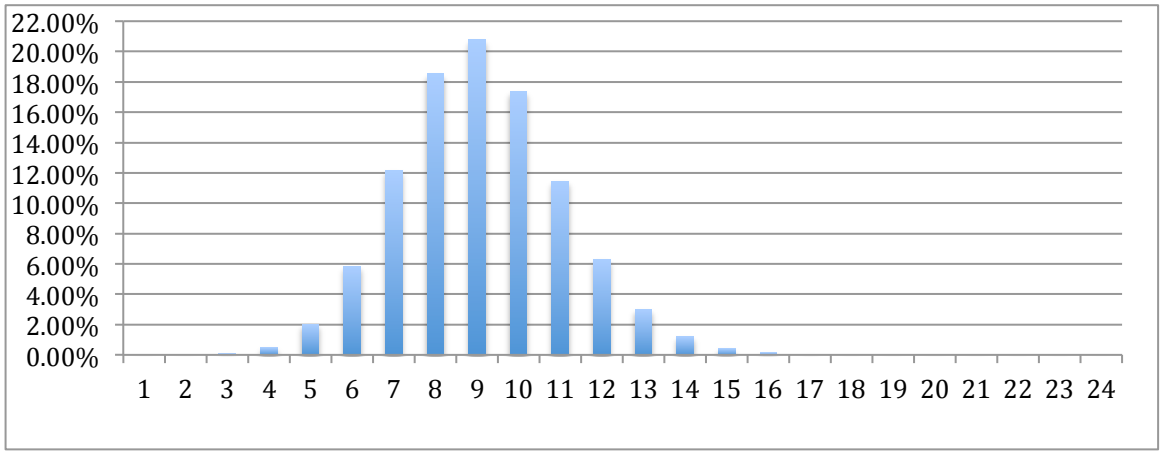


Figura 6.6 - Histograma das distâncias do dicionário de Italiano.

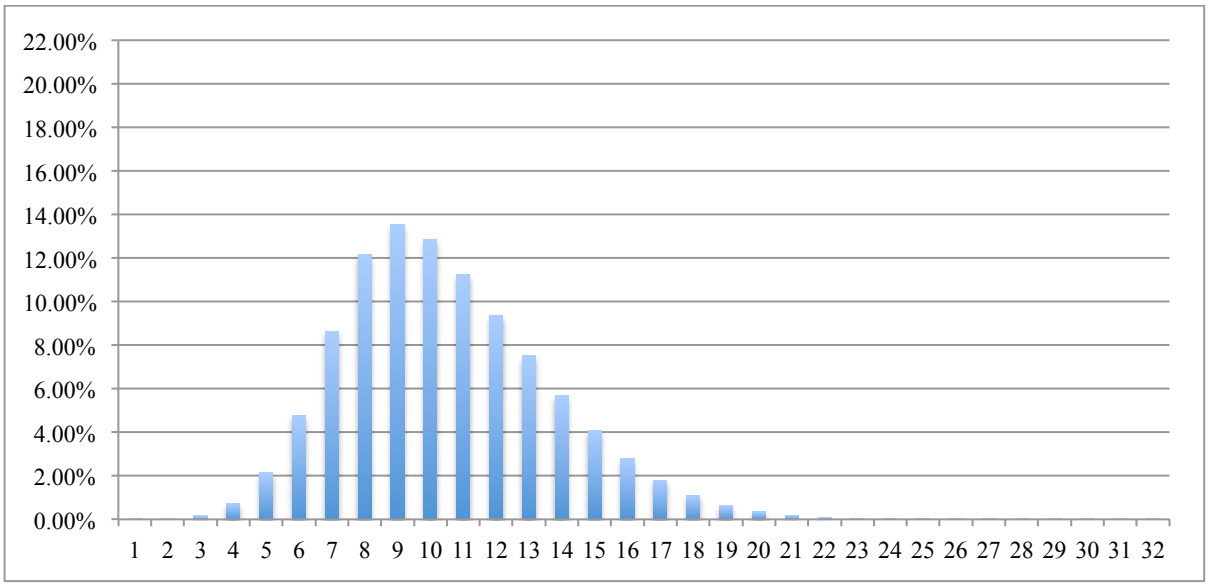


Figura 6.7 Histograma das distâncias do dicionário de Norueguês.

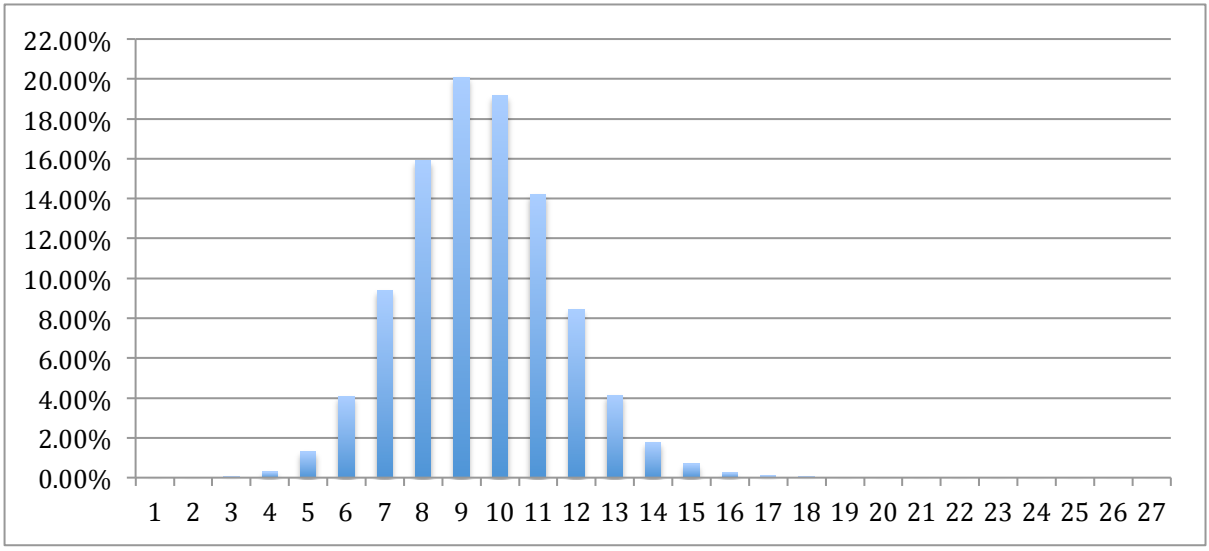


Figura 6.8 Histograma das distâncias do dicionário de Português.

6.2 CONJUNTOS DE IMAGENS

Nesta secção, são descritos os espaços métricos referentes a imagens, cujos universos estão divididos em duas categorias: histogramas de cores e imagens de rostos.

HISTOGRAMAS DE CORES

A base de dados dos histogramas de cores foi obtida em http://kdd.ics.uci.edu/databases/CorelFeatures/CorelFeatures.html e foi pré-processada, para se retirarem os duplicados. O conjunto

resultado é constituído por 68.030 histogramas de cores, em que cada um é uma sequência de trinta e dois números reais. Com este universo formaram-se dois espaços métricos: histogramas de cores com distância L1 e histogramas de cores com distância L2.

A Tabela 6.2 apresenta alguns dados sobre os espaços métricos. Refira-se que a dimensionalidade intrínseca do espaço com a distância euclidiana é quase metade da do outro.

Tabela 6.2 - Algumas estatísticas sobre os espaços métricos de histogramas de cores.

	Número de elementos do universo	Distância máxima	Distância mínima	Média	Variância	Dimens. intrínseca
Histogramas com distância L1	68.030	2,003948	1,00E-06	1,46	0,12	8,73
Histogramas com distância L2	68.030	1,413183	1,00E-06	0,56	0,03	4,79

As figuras 6.9 e 6.10 contêm os respectivos histogramas das distâncias. A dimensão dos intervalos com a distância de Manhattan é aproximadamente igual a 0,020 e com a distância euclidiana é cerca de 0,014. É interessante notar que, embora o universo seja o mesmo, as distribuições das distâncias são bastante diferentes.

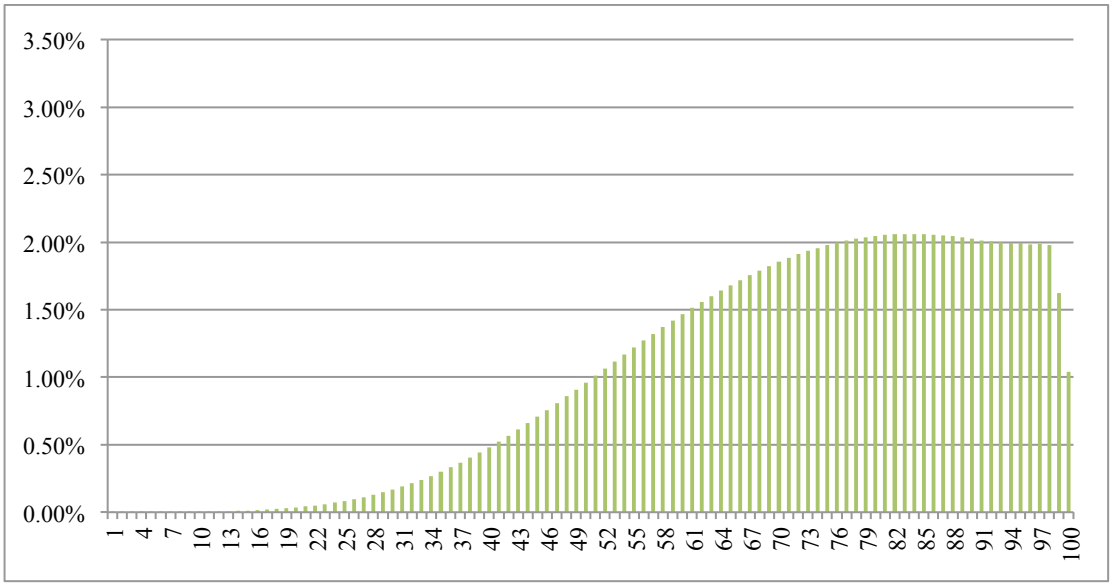


Figura 6.9 – Histograma das distâncias dos histogramas de cores com a distância L1.

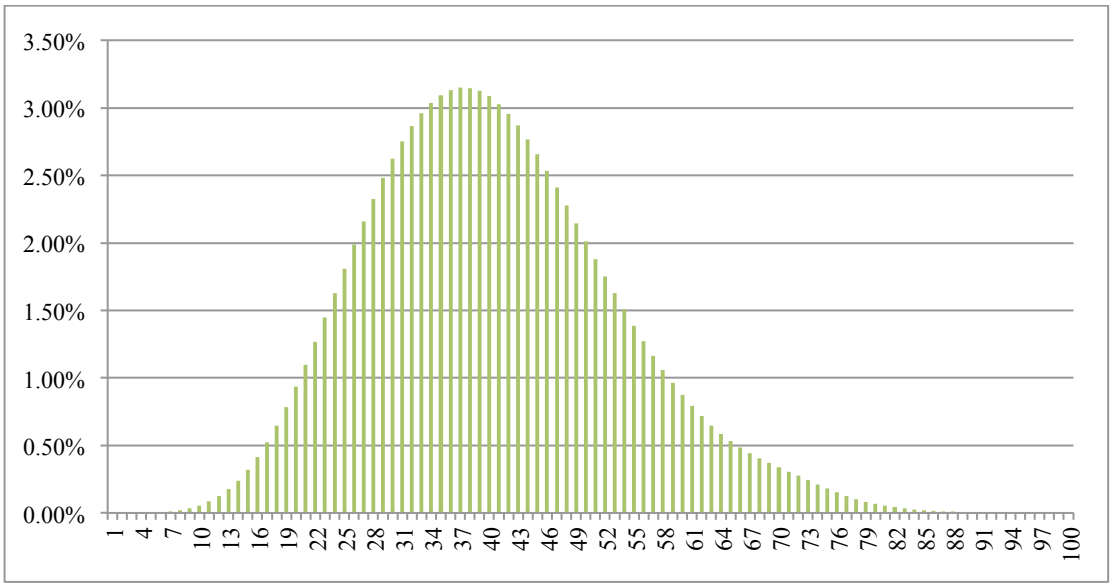


Figura 6.10 - Histograma das distâncias dos histogramas de cores com a distância L2.

IMAGENS DE ROSTOS

Foram seleccionados três espaços métricos de imagens de rostos: Rostos1 com distância de Manhattan, Rostos1 com distância euclidiana e Rostos2 com distância de Manhattan. A base de dados de Rostos1 foi obtida em http://sisap.org/library/dbs/faces/, enquanto que a de Rostos2 foi gentilmente cedida por Pedro Chambel.

O ficheiro original de Rostos1 foi pré-processado para eliminar repetições. Sem duplicados, o conjunto tem 760 elementos, em que cada um é um vector com 762 números reais. O universo Rostos2 tem 3.040 vectores, cada um com vinte e quatro números reais, extraídos de imagens de rostos humanos por Pedro Chambel, que utilizou o método *eigenface*s (referido na secção 2.3.2 e descrito com pormenor em [Chambel 2009]).

Na tabela 6.3 encontram-se as estatísticas habituais e as figuras 6.11, 6.12 e 6.13 têm os respectivos histogramas. Nestes casos, os comprimentos dos intervalos dos histogramas são, respectivamente e aproximadamente, 9,124; 9,050 e 600,701. As dimensionalidades intrínsecas dos espaços métricos com o universo Rostos1 são muito baixas.

Tabela 6.3 – Algumas estatísticas sobre os espaços métricos de imagens de rostos.

	Número de elementos do universo	Distância máxima	Distância mínima	Média	Variância	Dimens. intrínseca
Rostos1 e distância L1	760	912,48	0,045482	265,26	40.970,89	0,86
Rostos1 e distância L2	760	905,00	0,002271	258,31	40.974,56	0,81
Rostos2 e distância L1	3.040	60.231,05	160,9784	30.202,80	61.892.745,44	7,37

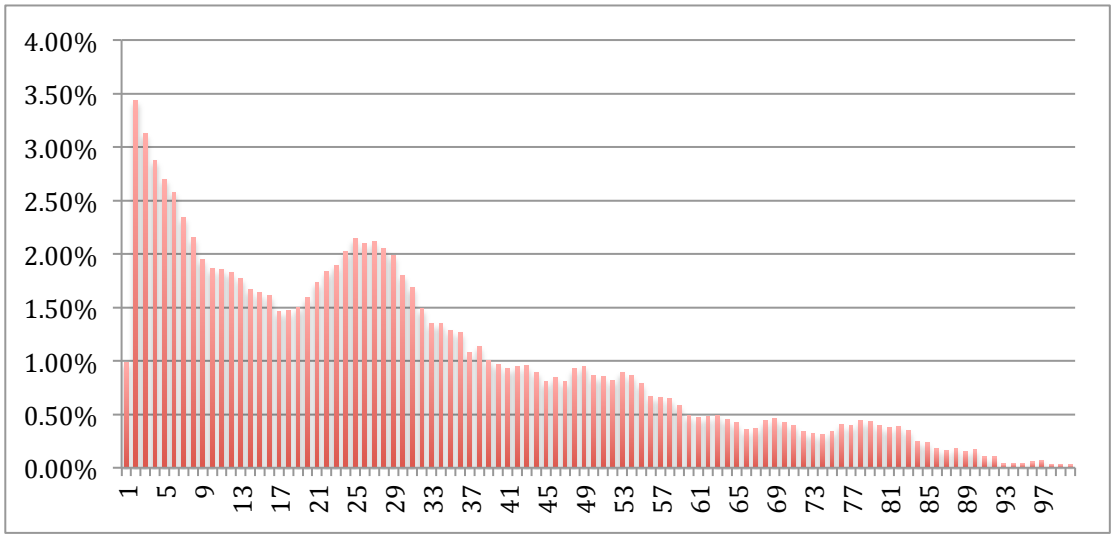


Figura 6.11 - Histograma das distâncias de Rostos1 com a distância L1.

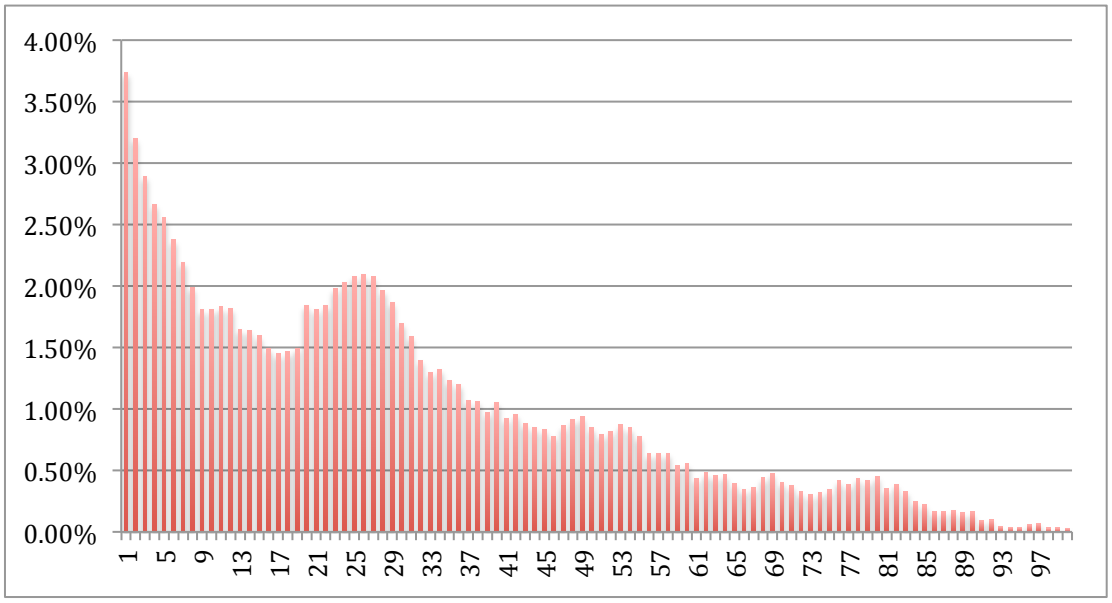


Figura 6.12 - Histograma das distâncias de Rostos1 com a distância L2.

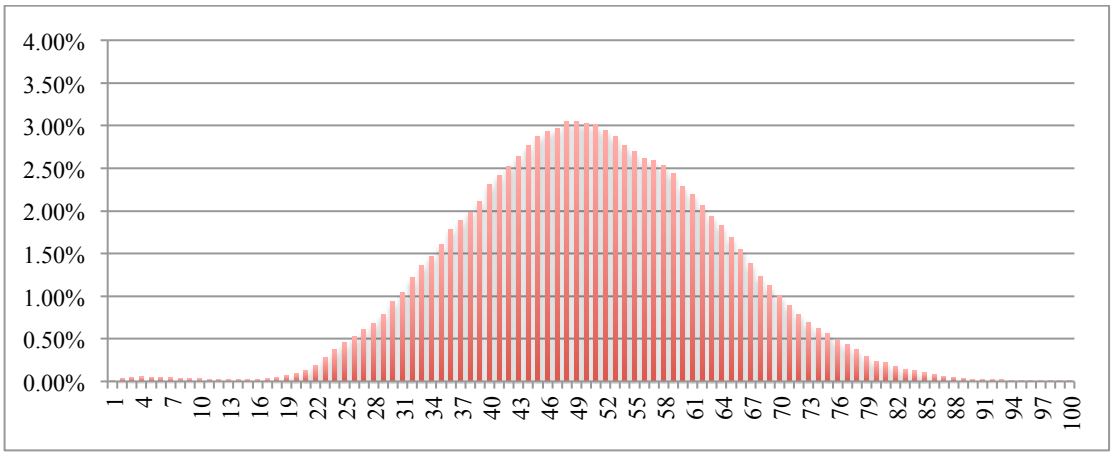


Figura 6.13 - Histograma das distâncias de Rostos2 com a distância L1.

6.3 SÉRIES TEMPORAIS

Agora são descritos dois espaços métricos de séries temporais, referentes a trajectórias de furacões e a percursos de uma pessoa. Os universos têm, respectivamente, 1331 trajectórias e 576 percursos, tendo sido obtidos a partir de http://weather.unisys.com/hurricane/atlantic/ e de http://location.e-2.org/.

Uma série temporal é uma sequência de observações e cada observação é um triplo da forma (x, y, tempo). Com estes universos, formaram-se dois espaços métricos: trajectórias com distância ERP e percursos com distância ERP.

A tabela 6.4 e as figuras 6.14 e 6.15 referem-se a estes espaços métricos. Para as trajectórias de furacões, o valor de cada intervalo do histograma de distâncias é aproximadamente 87,911 e para os percursos de uma pessoa esse valor é cerca de 122,999. O espaço métrico dos percursos é o que tem menor dimensionalidade intrínseca (dos quinze analisados).

Tabela 6.4 – Algumas estatísticas sobre os espaços métricos de séries temporais.

	Número de elementos do universo	Distância máxima	Distância mínima	Média	Variância	Dimens. intrínseca
Furacões	1.331	8795,69	4,59	1.602,29	986.198,81	1,30
Percursos	576	12.299,94	1.60E-04	311,47	1.010.655,55	0.05

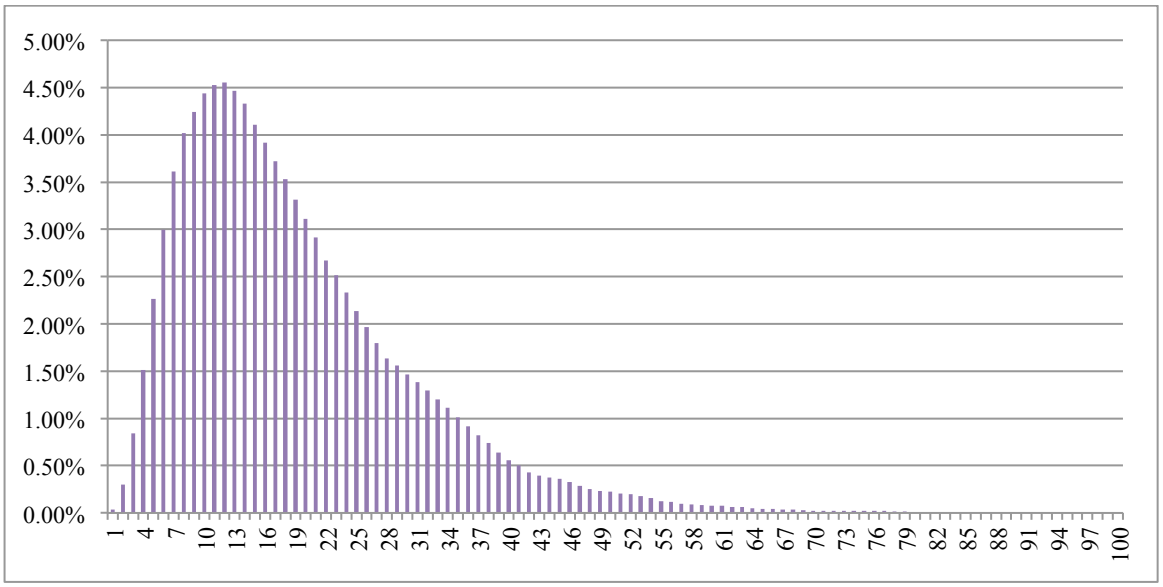


Figura 6.14 - Histograma das distâncias de trajectórias de furacões com a distância ERP.

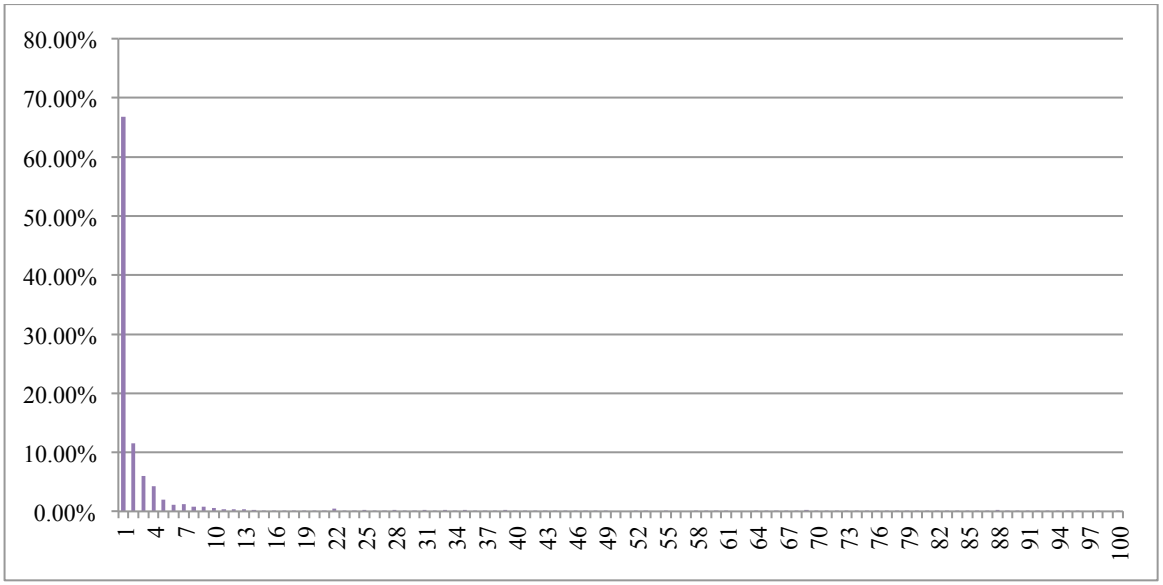


Figura 6.15 - Histograma das distâncias de percursos de uma pessoa com a distância ERP.

7 TESTES EXPERIMENTAIS

7.1 CARACTERIZAÇÃO DOS TESTES

Os testes foram realizados com os espaços métricos seleccionados e descritos no capítulo 6. Para cada universo, foram gerados cinco ficheiros, através de permutações de um ficheiro de base (com os objectos do ficheiro original, sem repetições). Três desses ficheiros (F-Ins1, F-Ins2 e F-Ins3) contêm todos os objectos e foram utilizados para carregar a base de dados. Os outros dois (F-Pesq e F-Rem) são mais pequenos e foram utilizados nas pesquisas e nas remoções. Cada teste consistiu na inserção de todos os objectos do universo (pela ordem em que se encontram num dos ficheiros F-Ins1, F-Ins2 ou F-Ins3), na pesquisa dos objectos de F-Pesq com três raios distintos e na remoção dos objectos de F-Rem. A justificação para o uso de três conjuntos equivalentes (F-Ins1, F-Ins2 e F-Ins3) reside no facto de que a forma final da estrutura de dados depende da ordem pela qual os objectos são inseridos. Portanto, para cada espaço métrico, cada teste consistiu exactamente nas mesmas inserções, nas mesmas pesquisas e nas mesmas remoções; apenas a ordem das inserções variou de teste para teste.

Para as pesquisas foram escolhidos três raios de pesquisa, dentro de um determinado intervalo, obtidos por observação.

A seguir, são caracterizados os testes realizados em cada espaço métrico, apresentando-se para cada um a totalidade de objectos inseridos, pesquisados e removidos, os raios usados nas pesquisas e os respectivos números médios de objectos retornados. Todos estes valores são independentes da ordem pela qual a base de dados é carregada.

7.1.1 DICIONÁRIOS

Com todos os espaços métricos de dicionários, foram pesquisadas e removidas 1000 palavras. As consultas foram realizadas com raios de pesquisa 1, 2 e 3.

A tabela 7.1 mostra, para cada dicionário, o total de palavras inseridas, o número médio de objectos retornados e a respectiva percentagem em relação à cardinalidade da base de dados. Verifica-se que, nas pesquisas de raio 1, o maior número médio de objectos é retornado pelo dicionário de Português, embora esse número corresponda à menor percentagem (0,001%), obtida também com o dicionário de Holandês. De notar que o dicionário de Português tem o maior número de palavras.

Tabela 7.1 - Número médio de objectos retornados nas pesquisas, com os dicionários.

	Palavras inseridas	Número médio de palavras retornadas numa pesquisa de raio R					
		R = 1		R = 2		R = 3	
Alemão	74.916	1,3	0,002%	4,3	0,006%	28,0	0,037%
Espanhol	86.061	3,0	0,004%	25,4	0,029%	215,5	0,250%
Francês	138.257	3,8	0,003%	22,0	0,016%	158,3	0,115%
Holandês	229.328	2,8	0,001%	18,8	0,008%	145,2	0,063%
Inglês	69.069	3,5	0,005%	30,5	0,044%	253,7	0,367%
Italiano	116.879	3,5	0,003%	20,3	0,017%	131,9	0,113%
Norueguês	85.560	1,8	0,002%	13,2	0,015%	119,1	0,139%
Português	407.583	5,2	0,001%	34,5	0,008%	249,8	0,061%

7.1.2 CONJUNTOS DE IMAGENS

Neste grupo estão os histogramas de cores e as imagens de rostos. Em todos estes casos foram pesquisados e removidos 100 objectos. Os raios de pesquisa dependeram do universo. Com os histogramas de imagens os raios foram 3,91E-256; 0,01 e 0,1; com Rostos1 foram 0,5; 2,5 e 6,5; e com Rostos2 foram 10; 2000 e 4000. De referir que os conjuntos Rostos1 e Rostos2 têm características diferentes, como se mostrou no capítulo 6.

A tabela 7.2 mostra o número médio de objectos retornados e a respectiva percentagem, para cada pesquisa. Comparando os dois espaços métricos de histogramas, verifica-se que os valores para as duas pesquisas de maior raio são superiores com a distância euclidiana. De referir que neste espaço métrico a média é menor (0,564 < 1,455) e o histograma é mais concentrado do lado esquerdo (figuras 6.10 e 6.9). Em relação a Rostos1, os valores da média são bastante próximos (265,26 com L1 e 258,31 com L2). No entanto, o histograma com L2 (figura 6.12) também é mais concentrado do lado esquerdo do que o histograma com L1 (figura 6.11), o que pode justificar o maior número médio de objectos retornados com os três raios de pesquisa.

Tabela 7.2 - Número médio de objectos retornados nas pesquisas, com os conjuntos de imagens.

	Objectos inseridos	Número médio de objectos retornados numa pesquisa de raio R					
		R = 3,91E-256		R = 0,01		R = 0,1	
Histogramas com distância L1	68.030	1,00	0,001%	1,93	0,003%	9,75	0,014%
Histogramas com distância L2		1,00	0,001%	2,58	0,004%	28,47	0,042%
		R = 0,5		R = 2,5		R = 6,5	
Rostos1 com distância L1	760	1,03	0,136%	1,11	0,146%	2,64	0,347%
Rostos1 com distância L2		3,00	0,395%	8,94	1,176%	20,80	2,737%
		R = 10		R = 2000		R = 4000	
Rostos2 com distância L1	3.040	1,00	0,033%	4,55	0,150%	9,68	0,318%

7.1.3 SÉRIES TEMPORAIS

Neste conjunto estão englobados percursos de uma pessoa e trajectórias de furacões. As bases de dados destes espaços métricos contêm poucos objectos, quando comparadas com as dos outros domínios.

As pesquisas e as remoções foram realizadas através de conjuntos com 100 objectos. Os raios de pesquisa dos percursos são muito inferiores aos das trajectórias (tabela 7.3). Assim, para os percursos foram efectuadas pesquisas com raios 0,02; 0,115 e 0,18 e para as trajectórias de furacões usaram-se os raios 150; 250 e 268.

Tabela 7.3 - Número médio de objectos retornados nas pesquisas, com as séries temporais.

	Objectos inseridos	Número médio de objectos retornados numa pesquisa de raio R					
Furacões	1.331	R = 150		R = 205		R = 268	
		4,89	0,367%	9,83	0,739%	20,37	1,530%
Percursos	576	R = 0,02		R = 0,115		R = 0,18	
		2,00	0,347%	10,20	1,771%	20,00	3,472%

7.2 RESULTADOS DOS TESTES

Muito antes de haver uma nova variante da RLC, efectuou-se a parametrização da versão anterior da estrutura (a RLC_2010), implementada em memória central, para cada um dos quinze espaços métricos seleccionados. Essa parametrização atribuiu dois valores (um para os raios dos agrupamentos de nível zero (r') e o outro para a capacidade das folhas (C) e seleccionou a função que calcula os raios dos agrupamentos de nível positivo. Foram consideradas as duas funções estudadas em [Sarmento 2010]:

$$r1(\text{nível}) = r' / \text{potência}(2, \text{nível}),$$

$$r2(\text{nível}) = r' / (\text{nível} + 1),$$

tendo-se verificado que os melhores resultados eram sempre obtidos com a primeira função. Os valores dos parâmetros foram obtidos por observação dos resultados de muitos testes, tendo sido utilizado o seguinte método. Para cada espaço métrico, começou-se por variar o raio dos agrupamentos de nível zero, mantendo a capacidade das folhas fixa. De seguida, com o melhor raio encontrado, variou-se a capacidade das folhas mantendo o raio fixo.

Os resultados dos testes apresentados neste capítulo comparam a variante anterior da RLC com a nova variante. Como já foi referido, os parâmetros da variante anterior (r' e C) foram seleccionados por observação e são os que apresentaram melhores resultados. Usou-se sempre a função r1. Para a nova variante foram utilizados os seguintes valores: o raio inicial (ri) é a média das distâncias e o incremento do raio (α) é o desvio padrão das distâncias. No cálculo do raio de um agrupamento de nível positivo, foram utilizadas separadamente as três funções descritas no capítulo 5,

$$r1(\text{nível}) = ri / \text{potência}(2, \text{nível}),$$

$$r2(\text{nível}) = ri / (\text{nível} + 1),$$

$$r3(\text{nível}) = ri / (0.5 * \text{nível} + 1),$$

e testaram-se dois valores para a capacidade das folhas: 16 e 128. Portanto, testaram-se seis alternativas da nova variante.

Apresenta-se a seguir uma tabela com os resultados obtidos para cada espaço métrico. Essa tabela contém o custo médio de uma operação de inserção, de uma operação de remoção e de cada uma das três operações de pesquisa. O campo "Total" é a soma destes cinco valores. O custo de uma operação é o número de distâncias (entre dois elementos do universo) computadas durante a operação. Todos os resultados apresentados são a média dos valores obtidos com as três permutações do espaço métrico, convertidos para uma percentagem do número de elementos do universo.

Para exemplificar, vamos calcular o custo médio de uma inserção na nova variante da RLC com folhas com capacidade 16 e a função r1, para a base de dados com o dicionário de Português. Os números totais de distâncias calculadas para inserir as 407.583 palavras com as três permutações geradas foram 307.386.002, 280.057.321 e 293.464.269. Ou seja, cada inserção requereu, em média, 754, 687 e 720 cálculos de distâncias, que correspondem, nos três casos, a 0,2% das palavras do universo. Portanto, o custo médio de uma inserção foi 0,2% dos objectos guardados na estrutura de dados.

7.2.1 DICIONÁRIOS

ALEMÃO

Como se pode ver na tabela 7.4, o processo de parametrização da variante anterior da RLC para o dicionário de Alemão resultou nos valores 12 e 16 para os raios dos agrupamentos de nível zero e a capacidade das folhas, respectivamente. Neste caso, é fácil concluir que a melhor alternativa é a nova variante da RLC com capacidade 16 e função r3. Para facilitar, assinala-se amarelo o valor mínimo de cada coluna.

Tabela 7.4 Número médio de distâncias por operação, com o dicionário de Alemão.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 12 CAPACIDADE DAS FOLHAS 16	0,5%	1,2%	4,0%	11,3%	21,7%	39%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,9%	3,6%	8,3%	16,5%	24,7%	55%
R2 = RI / (NÍVEL + 1)	1,8%	3,5%	8,0%	15,7%	23,2%	52%
R3 = RI / (0,5 * NÍVEL + 1)	0,5%	1,1%	3,0%	9,6%	18,9%	**33%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,9%	3,5%	8,4%	16,4%	24,5%	55%
R2 = RI / (NÍVEL + 1)	1,8%	3,5%	8,1%	15,9%	23,6%	53%
R3 = RI / (0,5 * NÍVEL + 1)	0,5%	1,1%	3,1%	10,1%	20,6%	35%

ESPANHOL

A tabela 7.5 mostra que, para o dicionário de Espanhol, embora nem todos os valores mínimos se encontrem numa mesma linha, a melhor alternativa volta a ser a nova variante da RLC com capacidade 16 e função r3.

Tabela 7.5 - Número médio de distâncias por operação, com o dicionário de Espanhol.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 16 CAPACIDADE DAS FOLHAS 16	0,2%	0,6%	4,4%	14,4%	28,3%	48%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,5%	1,1%	4,7%	13,4%	25,7%	45%
R2 = RI / (NÍVEL + 1)	0,5%	1,1%	4,7%	13,3%	25,6%	45%
R3 = RI / (0,5 * NÍVEL + 1)	0,2%	0,4%	2,5%	11,0%	25,9%	**40%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,5%	1,1%	4,8%	13,7%	27,0%	47%
R2 = RI / (NÍVEL + 1)	0,5%	1,1%	4,8%	13,7%	27,0%	47%
R3 = RI / (0,5 * NÍVEL + 1)	0,2%	0,4%	2,8%	13,2%	31,3%	48%

FRANCÊS

No caso do dicionário de Francês (veja-se a tabela 7.6), há duas alternativas muito competitivas da nova variante da RLC com capacidade 16: a que usa a função r2 e a que usa a função r3. No entanto, é com r2 que se minimiza a soma dos custos de cada tipo de operação.

Tabela 7.6 - Número médio de distâncias por operação, com o dicionário de Francês.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 8 CAPACIDADE DAS FOLHAS 16	0,2%	0,4%	2,5%	8,8%	18,3%	30%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,4%	0,8%	3,0%	8,1%	15,8%	28%
R2 = RI / (NÍVEL + 1)	0,3%	0,8%	2,7%	7,2%	14,3%	**25%**
R3 = RI / (0,5 * NÍVEL + 1)	0,1%	0,1%	1,6%	7,4%	17,9%	27%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,4%	0,8%	3,0%	8,3%	16,7%	29%
R2 = RI / (NÍVEL + 1)	0,3%	0,8%	2,8%	7,9%	16,4%	28%
R3 = RI / (0,5 * NÍVEL + 1)	0,1%	0,1%	1,6%	8,3%	21,1%	31%

HOLANDÊS

Os resultados com o dicionário de Holandês (na tabela 7.7) são de certa forma semelhantes aos obtidos com o de Alemão, no sentido em que a nova variante da RLC com capacidade 16 e função r3 é a que minimiza o número médio de distâncias calculadas em todas as operações.

Tabela 7.7 - Número médio de distâncias por operação, com o dicionário de Holandês.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 12 CAPACIDADE DAS FOLHAS 16	0,2%	0,4%	2,6%	9,0%	18,6%	31%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,5%	0,9%	3,7%	10,4%	19,3%	35%
R2 = RI / (NÍVEL + 1)	0,5%	0,8%	3,1%	8,7%	16,4%	29%
R3 = RI / (0,5 * NÍVEL + 1)	0,2%	0,4%	1,7%	7,1%	16,4%	**26%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,5%	0,9%	3,7%	10,3%	19,4%	35%
R2 = RI / (NÍVEL + 1)	0,5%	0,8%	3,1%	8,9%	17,3%	31%
R3 = RI / (0,5 * NÍVEL + 1)	0,2%	0,4%	1,8%	8,0%	19,3%	30%

INGLÊS

A tabela 7.8, referente ao dicionário de Inglês, apresenta desempenhos relativos parecidos com os encontrados para o dicionário de Espanhol. A nova variante da RLC com capacidade 16 e função r3 é a melhor alternativa, apesar de não minimizar as distâncias calculadas nas pesquisas de raio 3.

Tabela 7.8 - Número médio de distâncias por operação, com o dicionário de Inglês.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 12 CAPACIDADE DAS FOLHAS 16	0,3%	2,7%	4,9%	16,4%	32,3%	57%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,7%	1,3%	6,3%	15,7%	28,3%	52%
R2 = RI / (NÍVEL + 1)	0,7%	1,3%	6,3%	15,7%	28,3%	52%
R3 = RI / (0,5 * NÍVEL + 1)	0,2%	0,5%	3,3%	13,1%	28,8%	**46%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,7%	1,3%	6,3%	16,2%	29,8%	54%
R2 = RI / (NÍVEL + 1)	0,7%	1,3%	6,3%	16,2%	29,8%	54%
R3 = RI / (0,5 * NÍVEL + 1)	0,2%	0,6%	3,7%	15,5%	34,0%	54%

ITALIANO

No dicionário de Italiano (tabela 7.9), tal como no de Francês, é mais difícil escolher uma alternativa. A nova variante da RLC com capacidade 16 e função r3 tem melhores desempenhos nas inserções, remoções e pesquisas de raio 1. Mas, com a função r2, a eficiência das pesquisas de raios 2 e 3 aumenta, e o valor da última coluna é menor. Repare-se que nestes espaços métricos a maior diferença (em valor absoluto) entre as duas melhores alternativas está no custo da pesquisa com raio 3.

Tabela 7.9 - Número médio de distâncias por operação, com o dicionário de Italiano.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 16 CAPACIDADE DAS FOLHAS 16	0,2%	0,4%	2,4%	8,2%	17,4%	29%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,3%	0,8%	2,8%	7,4%	14,5%	26%
R2 = RI / (NÍVEL + 1)	0,3%	0,7%	2,6%	6,6%	13,1%	**23%**
R3 = RI / (0,5 * NÍVEL + 1)	0,1%	0,1%	1,4%	6,8%	16,5%	25%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,3%	0,8%	2,8%	7,6%	15,5%	27%
R2 = RI / (NÍVEL + 1)	0,3%	0,8%	2,7%	7,2%	15,1%	26%
R3 = RI / (0,5 * NÍVEL + 1)	0,1%	0,1%	1,6%	7,7%	19,6%	29%

NORUEGUÊS

O padrão dos melhores resultados com o dicionário de Norueguês é muito semelhante ao obtido com os dicionários de Alemão e Holandês. Como a tabela 7.10 mostra, a nova variante da RLC com capacidade 16 e função r3 é imbatível.

Tabela 7.10 - Número médio de distâncias por operação, com o dicionário de Norueguês.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 12 CAPACIDADE DAS FOLHAS 16	0,5%	1,3%	4,8%	14,5%	26,8%	48%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,0%	2,0%	7,2%	18,0%	30,4%	59%
R2 = RI / (NÍVEL + 1)	0,9%	1,8%	6,2%	15,7%	27,1%	52%
R3 = RI / (0,5 * NÍVEL + 1)	0,4%	1,0%	3,5%	12,7%	26,0%	**44%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,0%	2,0%	7,1%	17,7%	30,2%	58%
R2 = RI / (NÍVEL + 1)	0,9%	1,8%	6,3%	16,1%	28,1%	53%
R3 = RI / (0,5 * NÍVEL + 1)	0,4%	1,0%	3,8%	14,1%	29,0%	48%

PORTUGUÊS

Com o dicionário de Português (tabela 7.11), os resultados voltam a não permitir uma escolha fácil, dado que as pesquisas de raio 3 são bastante mais eficientes com a função r2 do que com a função r3 (com a nova variante da RLC com capacidade 16). Neste caso, há um valor mínimo que só se regista com a nova variante com capacidade 128. É também interessante notar que a ordem de grandeza dos custos das operações é menor neste dicionário do que nos restantes.

Tabela 7.11 - Número médio de distâncias por operação, com o dicionário de Português.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. RAIO 1	PESQ. RAIO 2	PESQ. RAIO 3	TOTAL
RAIO DOS AGRUPAMENTOS 16 CAPACIDADE DAS FOLHAS 16	0,1%	0,2%	1,3%	5,2%	11,5%	18%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,2%	0,4%	1,5%	4,2%	8,7%	15%
R2 = RI / (NÍVEL + 1)	0,2%	0,4%	1,3%	3,6%	7,7%	**13%**
R3 = RI / (0,5 * NÍVEL + 1)	0,0%	0,1%	0,7%	3,7%	10,0%	14%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,2%	0,4%	1,5%	4,3%	9,3%	16%
R2 = RI / (NÍVEL + 1)	0,2%	0,4%	1,4%	4,0%	9,0%	15%
R3 = RI / (0,5 * NÍVEL + 1)	0,0%	0,0%	0,8%	4,3%	12,2%	17%

7.2.2 CONJUNTOS DE IMAGENS

HISTOGRAMAS DE CORES COM A DISTÂNCIA L1

Para os histogramas de cores com a distância de Manhattan, foram seleccionados os valores 1,6 para o raio dos agrupamentos de nível zero e 256 para a capacidade das folhas. A tabela 7.12 mostra que existem duas alternativas cujos totais são equivalentes: a nova variante que usa a função r3, com capacidade 16 e com capacidade 128. Ambas não minimizam as distâncias calculadas numa operação: a primeira nas pesquisas de raio maior e a segunda nas remoções.

Tabela 7.12 – Número médio de distâncias por operação, com histogramas de cores e distância L1.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. R = 3,91E-256	PESQ. R = 0,01	PESQ. R = 0,1	TOTAL
RAIO DOS AGRUPAMENTOS 1,6 CAPACIDADE DAS FOLHAS 256	0,12%	0,15%	0,14%	0,18%	0,86%	1,5%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,15%	0,23%	0,16%	0,22%	1,23%	2,0%
R2 = RI / (NÍVEL + 1)	0,09%	0,16%	0,10%	0,12%	0,69%	1,2%
R3 = RI / (0,5 * NÍVEL + 1)	0,03%	0,07%	0,04%	0,05%	0,30%	**0,5%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,14%	0,26%	0,16%	0,20%	1,09%	1,8%
R2 = RI / (NÍVEL + 1)	0,09%	0,18%	0,10%	0,13%	0,65%	1,2%
R3 = RI / (0,5 * NÍVEL + 1)	0,03%	0,13%	0,04%	0,05%	0,27%	**0,5%**

HISTOGRAMAS DE CORES COM A DISTÂNCIA L2

O padrão dos resultados com o conjunto dos histogramas e a distância euclidiana (na tabela 7.13) é parecido com o anterior, uma vez que as duas alternativas consideradas com a nova variante da RLC e a função r3 são equivalentes nas inserções e nas duas pesquisas com menor raio, que as pesquisas de maior raio são mais eficientes com uma das capacidades e que, com a outra capacidade, as remoções têm menor custo. No entanto, a alternativa que minimiza a soma dos custos de cada uma das cinco operações é a que admite 128 objectos por folha.

Tabela 7.13 - Número médio de distâncias por operação, com histogramas de cores e distância L2.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. R = 3,91E-256	PESQ. R = 0,01	PESQ. R = 0,1	TOTAL
RAIO DOS AGRUPAMENTOS 1,413 CAPACIDADE DAS FOLHAS 256	0,06%	0,08%	0,07%	0,12%	3,07%	3,4%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,14%	0,21%	0,14%	0,29%	4,78%	5,6%
R2 = RI / (NÍVEL + 1)	0,07%	0,11%	0,07%	0,13%	3,10%	3,5%
R3 = RI / (0,5 * NÍVEL + 1)	0,03%	0,17%	0,04%	0,06%	2,40%	2,7%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,11%	1,14%	0,13%	0,25%	3,99%	5,6%
R2 = RI / (NÍVEL + 1)	0,07%	0,44%	0,08%	0,13%	3,02%	3,7%
R3 = RI / (0,5 * NÍVEL + 1)	0,03%	0,04%	0,04%	0,06%	2,43%	**2,6%**

IMAGENS DE ROSTOS (ROSTOS1) COM A DISTÂNCIA L1

No caso das imagens de Rostos1 com a distância de Manhattan (tabela 7.14), os resultados não permitem uma escolha fácil, pois as pesquisas são mais eficientes com a variante anterior da RLC e as distâncias mínimas nas inserções e remoções não estão numa única linha. No entanto, a nova variante que usa a função r2 com capacidade 128 minimiza a soma dos custos de cada tipo de operação.

Tabela 7.14 - Número médio de distâncias por operação, com Rostos1 e distância L1.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. R = 0,5	PESQ. R = 2,5	PESQ. R = 6,5	TOTAL
RAIO DOS AGRUPAMENTOS 468 CAPACIDADE DAS FOLHAS 128	0,67%	3,65%	1,07%	1,78%	3,29%	10,5%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,15%	5,50%	1,56%	2,21%	3,77%	14,2%
R2 = RI / (NÍVEL + 1)	1,55%	9,89%	1,96%	2,64%	4,10%	20,1%
R3 = RI / (0,5 * NÍVEL + 1)	2,06%	17,37%	2,59%	3,25%	4,74%	30,0%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,60%	1,92%	1,65%	2,36%	3,95%	10,5%
R2 = RI / (NÍVEL + 1)	0,61%	1,76%	1,67%	2,37%	3,98%	**10,4%**
R3 = RI / (0,5 * NÍVEL + 1)	0,69%	2,00%	1,87%	2,56%	4,14%	11,3%

IMAGENS DE ROSTOS (ROSTOS1) COM A DISTÂNCIA L2

Os resultados da tabela 7.15 mostram que, também para Rostos1 com a distância euclidiana, a escolha não é imediata. Se, por um lado, as duas pesquisas de menor raio são mais eficientes na variante anterior, as inserções e as remoções apresentam valores médios mínimos na nova variante com função r1 e capacidade 128, sendo que esta opção minimiza a coluna dos totais.

Tabela 7.15 - Número médio de distâncias por operação, com Rostos1 e distância L2.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. R = 0,5	PESQ. R = 2,5	PESQ. R = 6,5	TOTAL
RAIO DOS AGRUPAMENTOS 461 CAPACIDADE DAS FOLHAS 128	0,7%	4,3%	1,1%	1,8%	3,1%	10,8%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,1%	3,9%	1,5%	1,8%	2,0%	10,3%
R2 = RI / (NÍVEL + 1)	1,7%	10,1%	2,3%	2,7%	3,1%	19,9%
R3 = RI / (0,5 * NÍVEL + 1)	2,3%	23,2%	2,9%	3,4%	4,0%	35,8%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,6%	1,5%	1,7%	2,3%	3,6%	**9,7%**
R2 = RI / (NÍVEL + 1)	0,6%	1,6%	1,7%	2,3%	3,6%	9,8%
R3 = RI / (0,5 * NÍVEL + 1)	0,7%	2,2%	1,9%	2,5%	3,6%	10,9%

IMAGENS DE ROSTOS (ROSTOS2) COM A DISTÂNCIA L1

Na tabela 7.16 encontram-se os resultados para Rostos2 com a distância de Manhattan. É possível verificar-se que a nova variante com capacidade 16 e função r3 é a melhor alternativa, apesar de não minimizar as distâncias calculadas nas remoções.

Tabela 7.16 - Número médio de distâncias por operação, com Rostos2 e distância L1.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. R = 10	PESQ. R = 2000	PESQ. R = 4000	TOTAL
RAIO DOS AGRUPAMENTOS 60.231 CAPACIDADE DAS FOLHAS 16	0,7%	1,7%	0,7%	1,9%	3,6%	8,6%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,9%	2,2%	0,9%	2,2%	3,5%	9,5%
R2 = RI / (NÍVEL + 1)	0,9%	2,1%	0,9%	2,1%	3,3%	9,3%
R3 = RI / (0,5 * NÍVEL + 1)	0,5%	1,3%	0,5%	1,6%	3,1%	**7,1%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,9%	1,5%	1,1%	2,8%	4,5%	10,7%
R2 = RI / (NÍVEL + 1)	0,9%	1,5%	1,1%	2,8%	4,5%	10,7%
R3 = RI / (0,5 * NÍVEL + 1)	0,5%	0,9%	0,8%	2,5%	5,2%	9,8%

7.2.3 SÉRIES TEMPORAIS

TRAJECTÓRIAS DE FURACÕES

Nas trajectórias de furacões com a distância ERP, a soma dos custos de cada tipo de operação é mínima na nova variante com capacidade 16 e função r3 (tabela 7.17). Os resultados desta alternativa nem sempre são os mínimos; apresenta pesquisas eficientes embora não minimize as distâncias calculadas nas inserções e nas remoções.

Tabela 7.17 - Número médio de distâncias por operação, com trajectórias de furacões e distância ERP.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. R = 150	PESQ. R = 205	PESQ. R = 268	TOTAL
RAIO DOS AGRUPAMENTOS 5.000 CAPACIDADE DAS FOLHAS 16	0,9%	1,7%	3,9%	5,6%	7,9%	20%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,0%	2,1%	4,6%	6,8%	9,3%	24%
R2 = RI / (NÍVEL + 1)	0,8%	2,0%	3,6%	5,5%	7,8%	20%
R3 = RI / (0,5 * NÍVEL + 1)	0,7%	2,0%	3,1%	4,7%	7,0%	**18%**
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	0,9%	3,1%	4,8%	6,9%	9,5%	25%
R2 = RI / (NÍVEL + 1)	0,8%	2,5%	4,5%	6,7%	9,3%	24%
R3 = RI / (0,5 * NÍVEL + 1)	0,6%	2,4%	3,7%	5,5%	8.0%	20%

PERCURSOS

Os resultados com os percursos de uma pessoa com a distância ERP (tabela 7.18) mostram que a escolha da melhor opção não é directa. Existem duas alternativas da nova variante, que usam a função r1 com capacidades 16 e 128, onde estão situados os mínimos da tabela. Uma minimiza as pesquisas e

a outra minimiza as distâncias calculadas durante as actualizações. No entanto, a alternativa com capacidade 128 minimiza a soma dos custos de cada tipo de operação.

De notar que os valores percentuais da remoção são muito elevados em quatro casos. Esses valores são justificados pela remoção do centro do primeiro agrupamento de nível zero em duas das três permutações. Como já foi dito, esse primeiro agrupamento contém muitos objectos e a remoção do centro de um agrupamento implica a reinserção de todos os objectos contidos no seu interior e, como consequência, mais cálculos de distância. Como este espaço métrico contém poucos objectos, é provável que outros centros de agrupamentos muito populosos estejam também a ser removidos.

A remoção do centro do primeiro agrupamento de nível zero ocorre também com uma das permutações do domínio Rostos1.

Tabela 7.18 - Número médio de distâncias por operação, com percursos de uma pessoa e distância ERP.

VARIANTE ANTERIOR	INSERÇÃO	REMOÇÃO	PESQ. R = 0,02	PESQ. R = 0,115	PESQ. R = 0,18	TOTAL
RAIO DOS AGRUPAMENTOS 1.316,78 CAPACIDADE DAS FOLHAS 128	2,2%	22,2%	3,0%	5,0%	6,8%	39%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 16						
R1 = RI / POTÊNCIA(2, NÍVEL)	2,2%	30,7%	2,9%	4,5%	5,5%	46%
R2 = RI / (NÍVEL + 1)	9,6%	553,0%	16,0%	16,7%	17,1%	612%
R3 = RI / (0,5 * NÍVEL + 1)	12,0%	846,9%	20,3%	20,6%	21,0%	921%
NOVA VARIANTE CAPACIDADE DAS FOLHAS 128						
R1 = RI / POTÊNCIA(2, NÍVEL)	1,6%	18,2%	3,0%	5,2%	7,0%	**35%**
R2 = RI / (NÍVEL + 1)	3,5%	144,0%	5,0%	6,6%	8,2%	167%
R3 = RI / (0,5 * NÍVEL + 1)	13,3%	651,0%	17,8%	18,5%	19,1%	720%

7.2.4 CONCLUSÕES

A tabela 7.19 é um resumo dos resultados dos testes. Mostra, para cada espaço métrico, a média e o desvio padrão das distâncias, a parametrização usada na variante anterior da RLC, a soma dos custos médios de cada operação para as sete alternativas estudadas e a dimensionalidade intrínseca. Está ordenada decrescentemente pela última coluna. Tal como na secção anterior, o mínimo de cada linha encontra-se destacado a amarelo.

Verifica-se que alguma das alternativas da nova variante apresenta o melhor desempenho global, superando a RLC_2010 com a melhor parametrização encontrada. Verifica-se também que a nova variante com capacidade 16 e função r3 é a mais adequada para a maioria dos espaços métricos. As conclusões a seguir apresentadas dividem os 15 espaços métricos em três grandes grupos.

O primeiro grupo é formado pelos três primeiros espaços métricos da tabela 7.19: os dicionários de Português, Francês e Italiano com a distância de edição, cujas dimensionalidades intrínsecas são superiores a 10. Nestes espaços métricos os valores mínimos totais foram alcançados com a nova variante usando a função r2 e a capacidade 16. No entanto, com essa escolha obtiveram-se os mínimos

apenas nas pesquisas de raio 2 e de raio 3; nas inserções, nas remoções e nas pesquisas de raio 1, estes foram obtidos com a função r3 (vejam-se as tabelas 7.11, 7.6 e 7.9). Os resultados com r1 e r3 (e capacidade 16) para estes espaços métricos são próximos.

O segundo grupo é composto pelos nove espaços métricos seguintes, com dimensionalidades intrínsecas entre 1 e 9. Vai ser dividido em dois subgrupos.

Nos espaços métricos de histogramas de cores com a distância L1, e dicionários de Espanhol e de Inglês com a distância de edição (grupo 2 (a)), os melhores resultados em todas as operações excepto nas pesquisas de maior raio foram alcançados com a capacidade 16 e a função r3 (tabelas 7.12, 7.5 e 7.8).

O grupo 2 (b) inclui o dicionário de Alemão, Rostos2, os dicionários de Holandês e de Norueguês, os histogramas de cores com a distância L2 e as trajectórias de furacões. Para qualquer um destes espaços métricos a função r3 é a mais adequada, apresentado os mínimos em todas as operações, como no caso dos dicionários (tabelas 7.4, 7.7 e 7.10), ou exceptuando apenas algumas operações. Com Rostos2 (tabela 7.16) e os histogramas de cores (tabela 7.13), só não foi a mais eficiente nas remoções. Com os furacões (tabela 7.17) foi a mais eficiente em todas pesquisas.

De notar que, com todos os espaços métricos dos grupos 1 e 2, a capacidade 16 revelou-se uma escolha apropriada.

Tabela 7.19 - Resumo dos resultados dos testes.

Espaço Métrico	Média	Desvio	Variante anterior			Nova variante ri = Média; σ = Desvio; C = 16			Nova variante ri = Média; σ = Desvio; C = 128			Dim. intr.
			Raio	C	%	r1	r2	r3	r1	r2	r3	
Português, Edição	9,516	2.016	16	16	18	15	**13**	14	16	15	17	11,1
Francês, Edição	9.028	1.969	8	16	30	28	**25**	27	29	28	31	10,5
Italiano, Edição	9,102	1,992	16	16	29	26	**23**	25	27	26	29	10,4
Histogramas, L1	1,455	0,348	1,6	256	1,5	2.0	1,2	**0,5**	1.8	1.2	**0.5**	8,7
Espanhol, Edição	8,395	2,009	16	16	48	45	45	**40**	47	47	48	8,7
Inglês, Edição	8,346	2,025	12	16	57	52	52	**46**	54	54	54	8,5
Alemão, Edição	11,737	3,054	12	16	39	55	52	**33**	55	53	35	7,4
Rostos2, L1	30,203	7,867	60.231	16	9	10	9	**7**	11	11	10	7,4
Holandês, Edição	10,376	2,742	12	16	31	35	29	**26**	35	31	30	7,2
Norueguês, Edição	10,493	3,162	12	16	48	59	52	**44**	58	53	48	5,5
Histogramas, L2	0,564	0,182	1,413	256	3,4	5,6	3,5	**2,7**	5,6	3,7	**2,6**	4,8
Furacões, ERP	1,602	993	5.000	16	20	24	20	**18**	25	24	20	1,3
Rostos1, L1	265	202	468	128	**10,5**	14	20	30	**10,5**	**10,4**	11,3	0,9
Rostos1, L2	258	202	461	128	10,8	10.3	20	36	**9,7**	**9,8**	10,9	0,8
Percursos, ERP	311	1.005	1.316,78	128	39	46	612	921	**35**	167	720	0,0

O último grupo é formado por Rostos1 com as distâncias L1 e L2 e pelos percursos com a distância ERP. Têm em comum dimensionalidades intrínsecas inferiores a 1 e domínios muito pequenos. Os melhores resultados obtiveram-se com a capacidade 128. Com os espaços métricos das imagens de rostos (tabelas 7.14 e 7.15), a variante anterior da RLC revelou-se muito competitiva com a nova variante que utiliza as funções r1 e r2, tendo todas as pesquisas sido mais eficientes na anterior variante. No entanto, os mínimos totais são obtidos usando a função r2 com a distância L1 e usando a função r1 com a distância L2. Relativamente aos percursos (tabela 7.18), a função r1 é a única escolha adequada.

De acordo com os resultados obtidos, podem extrair-se as seguintes regras.

- Quando a dimensionalidade intrínseca está entre 10 e 12, deve ser adoptada a nova variante da RLC com capacidade 16 e a função r2.

- Quando a dimensionalidade intrínseca está entre 1 e 9, deve ser adoptada a nova variante da RLC com capacidade 16 e a função r3.

- Quando a dimensionalidade intrínseca é inferior a 1, deve ser adoptada a nova variante da RLC com capacidade 128 e a função r1.

Com estas regras, é possível baixar a soma dos custos médios das operações em 14 casos e manter em um caso (Rostos1 com a distância L1), em relação à melhor parametrização que tinha sido conseguida para a RLC_2010.

8 CONCLUSÕES

Neste trabalho foram realizadas diversas tarefas.

- Fez-se um levantamento dos espaços métricos mais utilizados nos testes de desempenho de estruturas de dados métricas e, posteriormente, a selecção e caracterização dos espaços utilizados neste trabalho.

- Descreveu-se a evolução da RLC e realizaram-se testes experimentais para seleccionar a melhor parametrização da variante mais recente (a RLC_2010, implementada em memória central) para todos os espaços métricos seleccionados.

- Fez-se uma nova proposta para a RLC, deixando dois pontos em aberto: a escolha da função do raio e da capacidade das folhas. Avaliou-se a sua eficiência, realizando mais testes experimentais que compararam os desempenhos da variante anterior e da nova proposta, considerando três funções e duas capacidades.

- Por último, analisaram-se os resultados obtidos e extraíram-se regras para determinar a função do raio e o valor da capacidade das folhas.

Na nova variante da RLC, os valores dos parâmetros dependem das características do espaço métrico, nomeadamente, da média e do desvio padrão. Esta variante é genérica, dinâmica, está implementada em memória central e revelou-se muito eficiente, quando comparada com a versão anterior.

Foram utilizados 15 espaços métricos, com características diferentes e com diferentes dimensionalidades intrínsecas. Seria interessante, num trabalho futuro, testar a RLC com espaços métricos cujas dimensionalidades intrínsecas sejam superiores às dos espaços deste trabalho. Também seria conveniente encontrar espaços métricos com dimensionalidade intrínseca inferior a um mas com cardinalidade maior, para se poder verificar se a função r1 é a mais adequada a estes casos.

A eficiência de uma estrutura de dados métrica implementada em memória secundária depende de outros factores para além do número de distâncias calculadas, destacando-se o número de acessos a disco. Outro trabalho futuro poderá ser testar a nova variante da RLC implementada em memória secundária.

9 BIBLIOGRAFIA

[Amato et al. 2003] G. Amato, F. Rabitti, P. Savino e P. Zezula. Region proximity in metric spaces and its use for approximate similarity search. *ACM Transactions on Information Systems*, 21(2):192–227, 2003.

[Arroyuelo et al. 2003] D. Arroyuelo, F. Muñoz, G. Navarro e N. Reyes. Memory-adaptative dynamic spatial approximation trees. *Proceedings of the 10th International Symposium on String Processing and Information Retrieval (SPIRE 2003)*. Lecture Notes in Computer Science, volume 2857, páginas 360–368. Springer-Verlag, 2003.

[Baeza-Yates et al. 1994] R. Baeza-Yates, W. Cunto, U. Manber e S. Wu. Proximity matching using fixed-queries trees. *Proceedings of the 5th Annual Symposium on Combinatorial Pattern Matching (CPM 1994)*. Lecture Notes in Computer Science, volume 807, páginas 198–212. Springer-Verlag, 1994.

[Baeza-Yates e Navarro 1998] R. Baeza-Yates e G. Navarro. Fast approximate string matching in a dictionary. *Proceedings of the 5th South American Symposium on String Processing and Information Retrieval (SPIRE 1998)*, páginas 14–22. IEEE CS Press, 1998.

[Barbosa 2009] F. Barbosa. Similarity-based retrieval in high dimensional data with recursive lists of clusters: a study case with natural language dictionaries. *Proceedings of the International Conference on Information Management and Engineering (ICIME 2009)*, páginas 432–436. IEEE Computer Society, 2009.

[Barbosa e Rodrigues 2009] F. Barbosa e A. Rodrigues. Range queries over trajectory data with recursive lists of clusters: a case study with hurricanes data. *Proceedings of Geographical Information Systems Research UK (GISRUK 2009)*, páginas 369–376. GISRUK conference series, Durham University, United Kingdom, 2009.

[Bozkaya e Ozsoyoglu 1997] T. Bozkaya e M. Ozsoyoglu. Distance-based indexing for high-dimensional metric spaces. *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data*, páginas 357–368. ACM Press, 1997.

[Brin 1995] S. Brin. Near neighbor search in large metric spaces. *Proceedings of the 21st International Conference on Very Large Data Bases (VLDB 1995)*, páginas 574–584. Morgan Kaufmann Publishers, 1995.

[Burkhard e Keller 1973] W. A. Burkhard e R. M. Keller. Some approaches to best-match file searching. *Communications of the ACM*, 16(4):230–236, 1973.

[Bustos et al. 2003] B. Bustos, G. Navarro e E. Chávez. Pivot selection techniques for proximity searching in metric spaces. *Pattern Recognition Letters*, 24(14):2357–2366, 2003.

[Bustos e Navarro 2009] B. Bustos e G. Navarro. Improving the space cost of k-NN search in metric spaces by using distance estimators. *Multimedia Tools and Applications*, 41(2):215–233, 2009.

[Chambel 2009] P. Chambel. Pesquisa de imagens de rosto. Dissertação de Mestrado em Engenharia Informática. Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, Portugal, 2009.

[Chávez et al. 1999] E. Chávez, J. Marroquín e G. Navarro. Overcoming the curse of dimensionality. *Proceedings of the European Workshop on Content-Based Multimedia Indexing (CBMI 1999)*, páginas 57–64. Tolouse, France, 1999.

[Chávez e Navarro 2000] E. Chávez e G. Navarro. An effective clustering algorithm to index high dimensional metric spaces. *Proceedings of the 7th Symposium on String Processing and Information Retrieval (SPIRE 2000)*, páginas 75–86. IEEE CS Press, 2000.

[Chávez et al. 2001] E. Chávez, G. Navarro, R. Baeza-Yates e J. L. Marroquín. Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321, 2001.

[Chávez e Navarro 2005] E. Chávez e G. Navarro. A compact space decomposition for effective metric indexing. *Pattern Recognition Letters*, 26(9):1363–1376, 2005.

[Chen 2005] L. Chen. Similarity-based search over time series and trajectory data. Ph.D. thesis. University of Waterloo, Canada, 2005.

[Ciaccia et al. 1997] P. Ciaccia, M. Patella e P. Zezula. M-tree: an efficient access method for similarity search in metric spaces. *Proceedings of the 23rd International Conference on Very Large Data Bases (VLDB 1997)*, páginas 426–435. Morgan Kaufmann Publishers, 1997.

[Corel Features] Corel Image Features (M. Ortega-Binderberger, K. Porkaew e S. Mehrotra). http://kdd.ics.uci.edu/databases/CorelFeatures/CorelFeatures.data.html.

[Costa 2009] F. Costa. Geração automática de "playlists" de músicas semelhantes. Dissertação de Mestrado em Engenharia Informática. Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, Portugal, 2009.

[Dehne e Nolteimer 1987] F. Dehne e H. Nolteimer. Voronoi trees and clustering problems. *Information Systems*, 12(2):171–175, 1987.

[Dohnal et al. 2003] V. Dohnal, C. Gennaro, P. Savino e P. Zezula. D-index: distance searching index for metric data sets. *Multimedia Tools and Applications*, 21(1):9–33, 2003.

[Figueroa et al. 2006] K. Figueroa, E. Chávez, G. Navarro e R. Paredes. On the least cost for proximity searching in metric spaces. *Proceedings of the 5th International Workshop on Experimental Algorithms (WEA 2006)*. Lecture Notes in Computer Science, volume 4007, páginas 279–290. Springer Verlag, 2006.

[Fredriksson 2005] K. Fredriksson. Exploiting distance coherence to speed up range queries in metric indexes. *Information Processing Letters*, 95(1):287–292, 2005.

[Fuad e Marteau 2008] M. Fuad, and P. Marteau. The extended edit distance metric. *Proceedings of the International Workshop on Content-Based Multimedia Indexing (CBMI 2008)*, páginas 242–248. IEEE Computer Society, 2008.

[Jardini 2007] E. Jardini. MFIS: algoritmo de reconhecimento e indexação em base de dados de impressões digitais em espaço métrico. Dissertação de Doutoramento em Engenharia Elétrica. Escola de Engenharia de São Carlos da Universidade de São Paulo, Brasil, 2007.

[Mamede 2005] M. Mamede. Recursive lists of clusters: a dynamic data structure for range queries in metric spaces. *Proceedings of the 20th International Symposium on Computer and Information Sciences (ISCIS 2005)*. Lecture Notes in Computer Science, volume 3733, páginas 843–853. Springer-Verlag, 2005.

[Mamede 2007] M. Mamede. A dynamic data structure for range queries in high dimensional metric spaces. http://ctp.di.fct.unl.pt/~mm/dynamic-07.pdf, 2007.

[Mamede e Barbosa 2007] M. Mamede e F. Barbosa. Range queries in natural language dictionaries with recursive lists of clusters. *Proceedings of the 22nd International Symposium on Computer and Information Sciences (ISCIS 2007)*, páginas 1–6. IEEE CS Press, 2007.

[Micó et al. 1994] M. L. Micó, J. Oncina e E. Vidal. A new version of the nearest-neighbour approximating and eliminating search algorithm (AESA) with linear preprocessing time and memory requirements. *Pattern Recognition Letters*, 15(1):9–17, 1994.

[Navarro 1999] G. Navarro. Searching in metric spaces by spatial approximation. *Proceedings of the 6th Symposium on String Processing and Information Retrieval (SPIRE 1999)*, páginas 141–148. IEEE CS Press, 1999.

[Navarro e Reyes 2002] G. Navarro e N. Reyes. Fully dynamic spatial approximation trees. *Proceedings of the 9th International Symposium on String Processing and Information Retrieval (SPIRE 2002)*. Lecture Notes in Computer Science, volume 2476, páginas 254–270. Springer-Verlag, 2002.

[Navarro et al. 2007] G. Navarro, R. Paredes e E. Chávez. t-Spanners for metric space searching. *Data & Knowledge Engineering*, 63(3): 820–854, 2007.

[Navarro e Uribe-Paredes 2011] G. Navarro e R. Uribe-Paredes. Fully dynamic metric access methods based on hyperplane partitioning. *Information Systems*, 36(4):734–747, 2011.

[Pola 2010] I. Pola. Explorando conceitos da teoria de espaços métricos em consultas por similaridade sobre dados complexos. Dissertação de Doutoramento em Ciências de Computação e Matemática Computacional. Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo, Brasil, 2010.

[Rodrigues 2006] C. Rodrigues. Implementação de sistemas de indexação para espaços métricos. Relatório do Projecto Final de Curso em Engenharia Informática. Departamento de Informática, Faculdade de Ciências e Tecnologia de Universidade Nova de Lisboa, Portugal, 2006.

[Ruiz 1986] E. V. Ruiz. An algorithm for finding nearest neighbours in (approximately) constant average time. *Pattern Recognition Letters*, 4(3):145–157, 1986.

[Sarmento 2010] A. Sarmento. Estruturas de dados métricas genéricas em memória secundária. Dissertação de Mestrado em Engenharia Informática. Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, Portugal, 2010.

[Sarmento e Mamede 2010] A. Sarmento e M. Mamede. Uma estrutura de dados métrica genérica, dinâmica, em memória secundaria. *Actas do II Simpósio de Informática (INForum 2010)*, páginas 79–90. Universidade de Minho, 2010.

[SISAP] Similarity Search and Applications (SISAP): Metric Spaces Library (K. Figueroa). http://www.sisap.org/Metric_Space_Library.html

[Thomasian et al. 2008] A. Thomasian, Y. Li e L. Zhang. Optimal subspace dimensionality for k-nearest-neighbor queries on clustered and dimensionality reduced datasets with SVD. *Multimedia Tools and Applications*, 40(2):241–259, 2008.

[Traina et al. 2002 a] C. Traina Jr., A. Traina, C. Faloutsos e B. Seeger. Fast indexing and visualization of metric data sets using slim-trees. *IEEE Transactions on Knowledge and Data Engineering*, 14(2):244–260, 2002.

[Traina et al. 2002 b] C. Traina Jr., A. Traina, R. Santos Filho, C. Faloutsos. How to improve the pruning ability of dynamic metric access methods. *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM 2002)*, páginas 219–226. ACM Press, 2002.

[Yianilos 1993] P. N. Yianilos. Data structures and algorithms for nearest neighbor search in general metric spaces. *Proceedings of the 4th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 1993)*, páginas 311–321. Society for Industrial and Applied Mathematics, 1993.

[Zezula et al. 1998] P. Zezula, P. Savino, G. Amato e F. Rabitti. Approximate similarity retrieval with M-trees. *The VLDB Journal*, 7(4):275–293, 1998.

[Zezula et al. 2006] P. Zezula, G. Amato, V. Dohnal e M. Batko. *Similarity search: the metric space approach*. Springer, 2006.