

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



FEUP

Sistema de apoio à mobilidade de utilizadores da bicicleta

Marco André Ferreira da Costa

Mestrado Integrado em Engenharia Informática e Computação

Orientador: António Coelho (Professor Auxiliar FEUP)

Co-orientador: Rosado Rossetti (Professor Auxiliar FEUP)

Julho de 2011

Sistema de apoio à mobilidade de utilizadores da bicicleta

Marco André Ferreira da Costa

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo Júri:

Presidente: Maria Eduarda Silva Mendes Rodrigues (Professora Auxiliar FEUP)

Vogal Externo: Fernando Augusto e Silva Mouta (Professor Coordenador ISEP)

Orientador: António Fernando Vasconcelos Cunha Castro Coelho (Professor Auxiliar FEUP)

15 de Julho de 2011

Resumo

A bicicleta é um meio de transporte com bastantes vantagens mas a informação sobre a rede ciclável bem como o desenvolvimento de tecnologias de navegação e roteamento específicos para utilizador de bicicletas são insuficientes.

Esta dissertação tem como objectivo criar formas de obter a rede ciclável através de fontes de dados *open source* e de informação obtida directamente dos utilizadores através de dados GPS. Outro objectivo é classificar a rede ciclável segundo vários critérios e adaptar os algoritmos de roteamento de forma a serem utilizados para bicicletas

Neste relatório é feita a revisão bibliográfica sobre cartografia digital, algoritmos de *map-matching*, algoritmos de roteamento e formas de actualizar a rede ciclável a partir de informação obtida de dados GPS. É apresentada a metodologia proposta para a criação da rede ciclável e sua gestão, a sua classificação da rede segundo o comprimento das vias, altitude acumulada, velocidade e tempo médio de percurso. Por fim é abordada a implementação efectuada mostrando a arquitectura geral e as funções de cada módulo, caracterizando a importância de cada um.

A criação da rede ciclável através de dados *open source*, nomeadamente do *OpenStreetMap*, é exequível obtendo uma rede ciclável base já bastante completa através da remoção de vias onde não é possível a passagem da bicicleta. A actualização da rede ciclável com informação dos utilizadores através dos percursos guardados pelo aparelho GPS é praticável com resultados satisfatórios, nomeadamente ao adicionar trilhos frequentemente utilizados por ciclistas.

Classificar a rede segundo os vários critérios propostos é conveniente para o roteamento, principalmente a combinação entre a altitude acumulada e o comprimento das vias, devido ao maior ou menor esforço a que os utilizadores da bicicleta estão sujeitos.

Abstract

The bicycle is a means of transport with many benefits but the information on the cycling network and the development of navigation technologies and user-specific routing for bicycles are still insufficient.

This dissertation aims to create methods for obtaining a cycling network through *open source* data sources and information obtained directly from users via GPS data. Another objective is to classify the cycling network according to various criteria and adapt the routing algorithms for the bicycle user.

In this report we present a literature review on digital mapping, map-matching algorithms, routing algorithms and ways to update the cycling network with information obtained from GPS data. The methodology proposed for the creation of the cycling network and its management is also presented. In order to be used by specific routing algorithms, a classification system is also described according to the length of the streets, accumulated altitude, speed, and average time of journey. Finally we approached the implementation carried out showing the general architecture and functionality of each module, featuring the importance of each one.

The creation of the cycling network through open source data, in particular *OpenStreetMap*, allows us to obtain a complete data set by removing routes where it is not possible to cycle. The update of the cycling network with user information through the GPS tracks is feasible and with satisfactory results, particularly when adding trails used frequently by cyclists.

The classification of the road network according to the proposed criteria is suitable for routing, especially the combination of accumulated height and the length of the cycleways, since it can improve the effort that users are subjected when cycling.

Agradecimentos

Aos meus orientadores António Coelho e Rossaldo Rossetti pela disponibilidade e orientação durante a realização da dissertação.

A Daniel Almeida que realizou a dissertação em colaboração comigo.

A todos os colegas e amigos do Laboratório de Computação Gráfica pela ajuda prestada.

Aos amigos e colegas que realizaram a dissertação no Departamento de Engenharia Informática.

À família por toda a paciência e colaboração.

A todas as outras pessoas que de alguma forma contribuíram para a realização desta dissertação.

Marco André Ferreira da Costa

Índice

1	Introdução	1
1.1	Enquadramento e motivação.....	1
1.2	Descrição do Problema	1
1.3	Objectivos	2
1.4	Resultados Esperados	3
1.5	Estrutura do Documento	3
2	Revisão Bibliográfica	5
2.1	Informação Geográfica	5
2.1.1	Fontes de informação	5
2.1.2	Base de dados espacial	6
2.1.3	Aquisição de dados GPS	7
2.1.3.1	Formato GPX	7
2.1.3.2	Formato KML	8
2.2	Algoritmos de <i>map-matching</i>	9
2.2.1	Análise Geométrica	10
2.2.2	Point-to-point.....	10
2.2.3	Point-to-curve	11
2.2.4	Curve-to-curve.....	12
2.2.5	Análise Topológica.....	12
2.2.6	Estatísticos	12
2.3	Algoritmos de roteamento	13
2.3.1	Dijkstra	13
2.3.2	A*	13
2.4	Correcção de mapas.....	14
2.4.1	Detecção de novas ruas	14
2.4.2	Detecção de ruas com geometria errada	17
2.4.3	Correcção dos cruzamentos das ruas	18
3	Metodologia	21
3.1	Modelo de dados.....	22

3.2	Criação da rede ciclável inicial	24
3.3	Actualização da rede ciclável	27
3.3.1	Actualizar a rede a partir das fontes <i>open source</i>	27
3.3.2	Detectar e criar novas vias	28
3.4	Classificação da rede	32
3.4.1	Altitude acumulada	32
3.4.2	Tempo e velocidade média	33
3.5	Roteamento	35
4	Implementação e resultados	39
4.1	Arquitectura	39
4.1.1	Base de dados	40
4.1.2	Interface com a base de dados	40
4.1.3	Gestor da rede ciclável	41
4.1.4	Roteamento	41
4.2	Visualização	41
4.3	Resultados	42
4.3.1	Estatísticas	42
4.3.2	Rede Ciclável inicial	43
4.3.3	Detecção de novas vias	44
4.3.4	Roteamento	46
5	Conclusões e Trabalho Futuro	49
5.1	Conclusões	49
5.2	Trabalho Futuro	49
	Referências	51
A	Modelo Relacional	55

Lista de Figuras

Figura 2.1: Exemplo de <i>query</i> à base de dados espacial [YHA07]	6
Figura 2.2: Informação disponível em KML [KML11b]	8
Figura 2.3: Problema de <i>map-matching</i> [WBK00]	10
Figura 2.4: Corresponder o ponto GPS ao nó mais próximo [GRE02]	11
Figura 2.5: Corresponder um ponto GPS ao segmento mais próximo [GRE02]	11
Figura 2.6: Usar informação topológica [BKO96]	12
Figura 2.7: Detecção de nova rua – primeira fase [FCR09b]	15
Figura 2.8: Detecção de nova rua – segunda fase [FCR09b]	16
Figura 2.9: Algoritmo de detecção da geometria da rua – primeira fase [FCR09a]	17
Figura 2.10: Algoritmo de detecção da geometria da rua – segunda fase [FCR09a]	18
Figura 2.11: Diagrama do algoritmo para correcção dos cruzamentos [FCR09a]	19
Figura 3.1: Funcionamento geral da metodologia	21
Figura 3.2: Modelo de dados - rede ciclável	23
Figura 3.3: Modelo de dados - <i>Tracks</i> GPS	24
Figura 3.4: Criação da rede inicial	24
Figura 3.5: Obter informação do <i>OpenStreetMap</i>	25
Figura 3.6: Filtragem de dados do <i>OpenStreetMap</i>	26
Figura 3.7: Actualização da rede	27
Figura 3.8: Actualizar rede através de fonte <i>open source</i>	28
Figura 3.9: Possível nova via	29
Figura 3.10: Detectar possíveis novas vias	30
Figura 3.11: Actualização da rede com nova via	31
Figura 3.12: Altitude de uma via	32
Figura 3.13: Classificar a rede segundo as altitudes	33
Figura 3.14: Classificar a rede segundo a velocidade média e tempo	34
Figura 3.15: Roteamento	35
Figura 3.16: Caminho de menor custo	36
Figura 4.1: Arquitectura do sistema	40
Figura 4.2: Visualização da rede no <i>uDig</i>	42
Figura 4.3: Rede obtida directamente do <i>OpenStreetMap</i>	43

Figura 4.4: Rede filtrada obtida a partir do <i>OpenStreetMap</i>	44
Figura 4.5: Detecção de novas vias - exemplo 1	45
Figura 4.6: Detecção de novas vias - exemplo 2	45
Figura 4.7: Detecção de novas vias a partir de vários <i>tracks</i> GPS	46
Figura 4.8: Roteamento tendo em conta o caminho mais curto	46
Figura 4.9: Caminho mais curto	47
Figura 4.10: Caminho com menor esforço	48

Lista de Tabelas

Tabela 4.1: Informação obtida do *OpenStreetMap*

42

Abreviaturas e Símbolos

GPS	Global Positioning System
BTT	Bicicleta Todo o Terreno
GPX	GPS Exchange Format
DBMS	Database Management System
XML	eXtensible Markup Language

Capítulo 1

Introdução

1.1 Enquadramento e motivação

A bicicleta é um meio de transporte ecológico, saudável e económico, sendo utilizado tanto para lazer como para deslocamentos pendulares. A inexistência de ciclovias em número suficiente e adequadas para este tipo de deslocamento, torna a adopção deste meio de transporte baixa.

A tecnologia responsável para permitir a navegação e roteamento para automóvel estão em um estado de desenvolvimento bastante avançado. No entanto, os desenvolvimentos dos sistemas pedonais e adaptados a bicicletas estão algo atrasados, devido, em parte, à especificidade e flexibilidade destes meios de locomoção.

Nesta dissertação pretende-se desenvolver um sistema especialmente dedicado aos utilizadores de bicicleta para colmatar um pouco esta desigualdade existente entre a bicicleta e o automóvel.

1.2 Descrição do Problema

Os sistemas de navegação específicos para bicicleta têm de ser implementados de forma muito particular, de forma a otimizar a sua utilização, evitando o trânsito automóvel e minimizando o esforço despendido pelo ciclista.

Em países em que não existem suficientes vias dedicadas à bicicleta, será necessário classificar a rede viária para que os algoritmos de roteamento possam seleccionar as rotas mais adequadas para os utilizadores da bicicleta. Esta classificação pode ser feita através dos seguintes atributos:

Introdução

- Declive;
- Trânsito;
- Velocidade média;
- Tipologia.

O principal problema científico desta dissertação é a determinação de uma metodologia que possibilite a classificação da rede viária existente em fontes de dados de acesso livre, e o acrescento de trilhos e vias não classificadas mas que possibilitam a utilização da bicicleta, através da utilização dos trajectos (obtidos por GPS) partilhados pela comunidade ciclista.

Por outro lado os algoritmos de roteamento não estão adaptados às especificidades da bicicleta e não permitem personalizar os parâmetros dos algoritmos de forma a calcular caminhos específicos tendo em consideração características como inclinação, esforço necessário ou segundo o perfil de um utilizador. Pretende-se também encontrar uma solução para este problema.

1.3 Objectivos

Nesta dissertação o objectivo principal é desenvolver um sistema que permita a criação de rotas adaptadas para cada perfil de utilizador de bicicleta.

Será portanto necessário concretizar os seguintes objectivos específicos:

- Fazer um relatório do estado da arte ao nível de bases de dados espaciais, algoritmos de roteamento e dos métodos de levantamento de dados viários com base em trajectórias GPS;
- Desenvolver um modelo de dados para a incorporação de toda a informação sobre vias cicláveis, bem como da sua classificação segundo diversos itens, tais como velocidade média, declive ou tipo de via (estrada, rua, caminho, trilho, etc.);
- Implementar uma base de dados espacial de acordo com o modelo definido;
- Seleccionar e implementar um algoritmo para extracção da rede de vias cicláveis e sua caracterização a partir de trajectórias GPS e sua integração com a rede viária obtida de fontes de dados *open source*;
- Seleccionar e implementar um algoritmo de roteamento para a rede ciclável e que funcione de acordo com as preferências do utilizador.

1.4 Resultados Esperados

Espera-se que no final da dissertação o sistema permita autonomamente fazer a gestão da rede ciclável, adicionando e sincronizando novos percursos através de dados obtidos de fontes *open source* e de dados submetidos pela comunidade. Deve também permitir ao utilizador pedir ao sistema percursos adaptadas ao perfil do utilizador e do tipo de bicicleta que utiliza (estrada ou BTT).

1.5 Estrutura do Documento

Este documento é composto por três capítulos, para além deste capítulo introdutório e do capítulo final de conclusões e trabalho futuro.

No segundo capítulo, "Revisão Bibliográfica" é feito o levantamento do estado da arte relativo à cartografia digital na forma como é criada e de como pode ser organizada, aos algoritmos de roteamento mais utilizados e de *map-matching* e por fim uma abordagem de algoritmos que permitam actualizar a rede ciclável a partir de dados GPS. No capítulo "Metodologia" é abordada a metodologia aplicada nesta dissertação na obtenção da rede ciclável pelas várias vias estudadas bem como à solução de roteamento adaptada à utilização da bicicleta. No capítulo "Implementação e resultados" são apresentados os aspectos de implementação da solução adoptada, detalhando algumas das partes mais importantes, os resultados da implementação feita dos algoritmos bem como informação sobre os dados obtidos a partir de fontes *open source* e dos utilizadores.

Capítulo 2

Revisão Bibliográfica

Neste capítulo será realizada a análise do estado da arte relativamente a quatro principais temas: cartografia digital, algoritmos de *map-matching*, algoritmos de roteamento e algoritmos que permitam actualizar mapas a partir de dados GPS.

2.1 Informação Geográfica

Nesta secção serão apresentadas fontes de dados disponibilizando informação geográfica sobre a rede de estradas e a forma como se estrutura esta informação, a forma de a armazenar e aceder de forma eficiente, através das bases de dados espaciais e a forma de aquisição de dados GPS.

2.1.1 Fontes de informação

OpenStreetMap é um projecto colaborativo para a disponibilização gratuita de informação geográfica sobre a rede viária e tem um enquadramento mundial. O projecto foi iniciado porque a maioria dos mapas disponíveis possuem restrições legais ou técnicas relativas à sua utilização [OSM11a]. Em Portugal o *OpenStreetMap* cobre no geral o país todo mas ainda existem zonas que não estão bem cobertas faltando muita informação da rede de vias, estradas, auto-estradas entre outros.

A informação guardada no *OpenStreetMap* é composta por três tipos de elementos básicos:

- Nós (*nodes*) – que é o elemento mais básico do *OpenStreetMap* que representam um ponto geospacial. Pode ser adicionado ao nó outras informações através da utilização de *tags*;

Revisão Bibliográfica

- Vias (*ways*) – que são compostas por um conjunto ordenado de elementos do tipo nó. Este elemento é usado para representar por exemplo estradas, auto-estradas, linhas de comboio, rios, entre outros. As formas podem ser estruturas que formam áreas quando são fechadas;
- Relações (*relations*) – permitem agrupar os elementos como os nós, as formas e outras relações. Este elemento é utilizado por exemplo na construção de polígonos.

O *OpenStreetMap* permite importar e exportar informação a partir de API fornecida [OSM11b]. Os dados podem ser obtidos através de *webservice REST*, sendo que a informação a receber é no formato XML.

2.1.2 Base de dados espacial

A utilização das capacidades das bases de dados espaciais (SDBMS) facilita a gestão de informação espacial, garantindo segurança, integridade, *backup* e recuperação da informação [YHA07].

Devido à utilização de base de dados, e devido às características da mesma, é possível assim fazer *queries* aos dados contidos na base de dados forma fácil e rápida devido ao tipo de indexação. As *queries* espaciais permitem usar os dados geométricos como pontos, linhas e polígonos e consideram a relação espacial destas geometrias. Exemplo deste tipo de *query* está na Figura 2.1 (versão adaptada) que ilustra a procura dos nomes das parcelas que pertençam geograficamente à subdivisão com o nome “Trofa”.

```
SELECT parcela.nome
FROM parcela, subdivisão
WHERE within(parcela.localização, subdivisão.localização)
AND subdivisão.nome="Trofa"
```

Figura 2.1: Exemplo de *query* à base de dados espacial [YHA07]

Actualmente praticamente todas os DBMS suportam nativamente dados espaciais ou a partir de extensões instaladas.

O *PostgreSQL* é uma poderosa base de dados objecto-relacional *open source* que tem mais de 15 anos de desenvolvimento activo e com uma forte reputação de confiabilidade, correcção e integridade de dados [POS11a]. A extensão espacial *PostGIS* adiciona suporte para objectos geográficos a base de dados objecto-relacional *PostgreSQL* de forma a permitir que seja usado como um *backend* de base de dados espaciais para sistemas de informação geográfica [POS11b].

2.1.3 Aquisição de dados GPS

Os dados geospaciais são representados segundo dois dos principais padrões o GPX e KML que são utilizados para fins diferentes.

2.1.3.1 Formato GPX

O esquema GPX 1.1 que foi lançado em 2004 tem sido o padrão de XML para intercâmbio de dados GPS desde a versão 1.0 em 2002. GPX é usado por dezenas de programas de *software* e serviços Web para troca de dados GPS e mapeamento [GPX11] [TUR06].

O formato GPX é composto por três tipos elementos:

- Pontos de interesse;
- Caminhos;
- Rotas.

Os caminhos e as rotas são compostos por pontos de interesse, tendo cada um destes como informação obrigatória a latitude e longitude apesar de poder conter outra informação como a altitude e tempo.

No código é visível que num ficheiro GPX é possível ter um *track* com o respectivo nome e o conjunto de pontos representando pela latitude e longitude, a elevação e a data e hora em que foi registado o ponto.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
  <gpx xmlns="http://www.topografix.com/GPX/1/1" creator="GPS TrackMaker"
  version="1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.topografix.com/GPX/1/1
  http://www.topografix.com/GPX/1/1/gpx.xsd">
  <metadata> (...) </metadata>
  <trk>
    <name>
      ACTIVE LOG
    </name>
    <trkseg>
      <trkpt lat="41.399574289" lon="-8.667590563">
        <ele>
          54.451780
        </ele>
        <time>
```

Revisão Bibliográfica

2006-02-08T14:21:43Z

</time>

</trkpt>

(...)

2.1.3.2 Formato KML

No caso do KML que é uma linguagem XML focada na visualização geográfica incluindo a anotação dos mapas e imagens. Alguns dos elementos principais deste padrão são os pontos, linhas, anéis lineares, polígonos entre outros, como visível na Figura 2.2. Este padrão pretende ser um padrão internacional na linguagem par expressar anotações e visualizações geográficas em aplicações *online*, aplicações de mapas móveis. [KML11a].

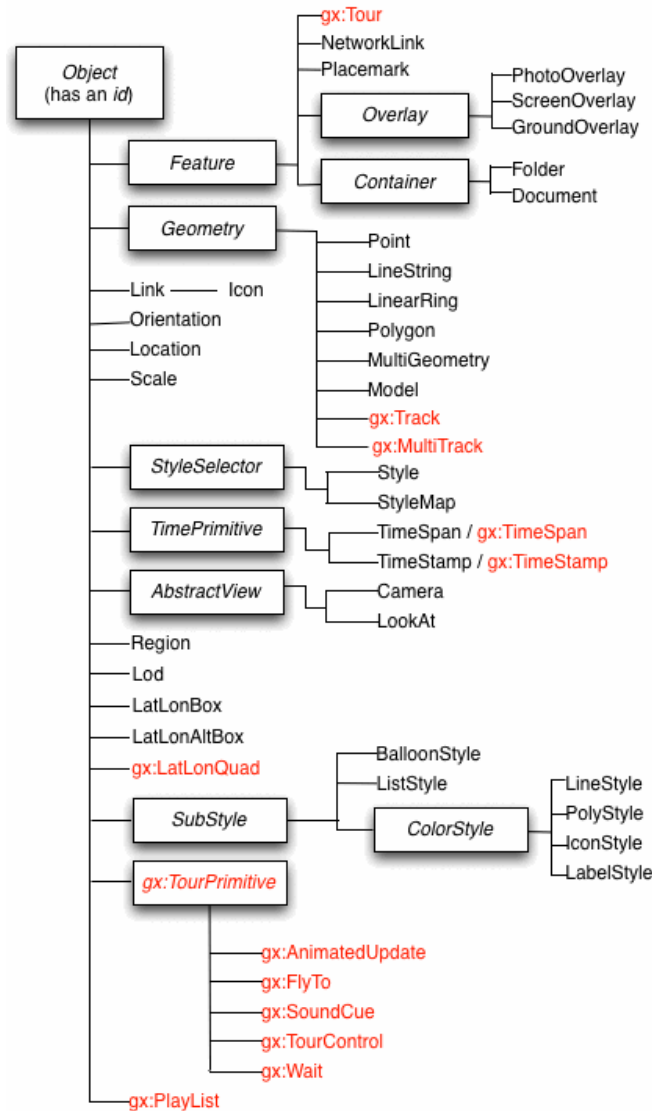


Figura 2.2: Informação disponível em KML [KML11b]

Revisão Bibliográfica

Exemplo de código KML é o seguinte onde se pode observar que é composto neste caso por um marcador (*placemark*) que tem um nome (*name*) uma descrição (*description*) e neste caso um ponto (*point*) com as suas coordenadas.

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
  <Placemark>
    <name>
      casa1
    </name>
    <description>
      Casa de campo
    </description>
    <Point>
      <coordinates>
        -8.7724220811056,41.7689908978,0
      </coordinates>
    </Point>
  </Placemark>
</kml>
```

2.2 Algoritmos de *map-matching*

Os algoritmos de *map-matching* têm como objectivo determinar a posição no mapa a partir de uma localização estimada por GPS como se pode observar na Figura 2.3, onde se verifica o ponto estimado por GPS e a sua localização estimada na via em “(b)” comparando com a posição real em “(a)”. Estes tipos de algoritmos serão necessários para a utilização nos algoritmos de correcção de mapas.

Nesta secção serão analisadas os vários tipos de algoritmos de *map-matching*, desde os algoritmos estritamente geométricos, os que utilizam informação topológica e por fim os estatísticos.

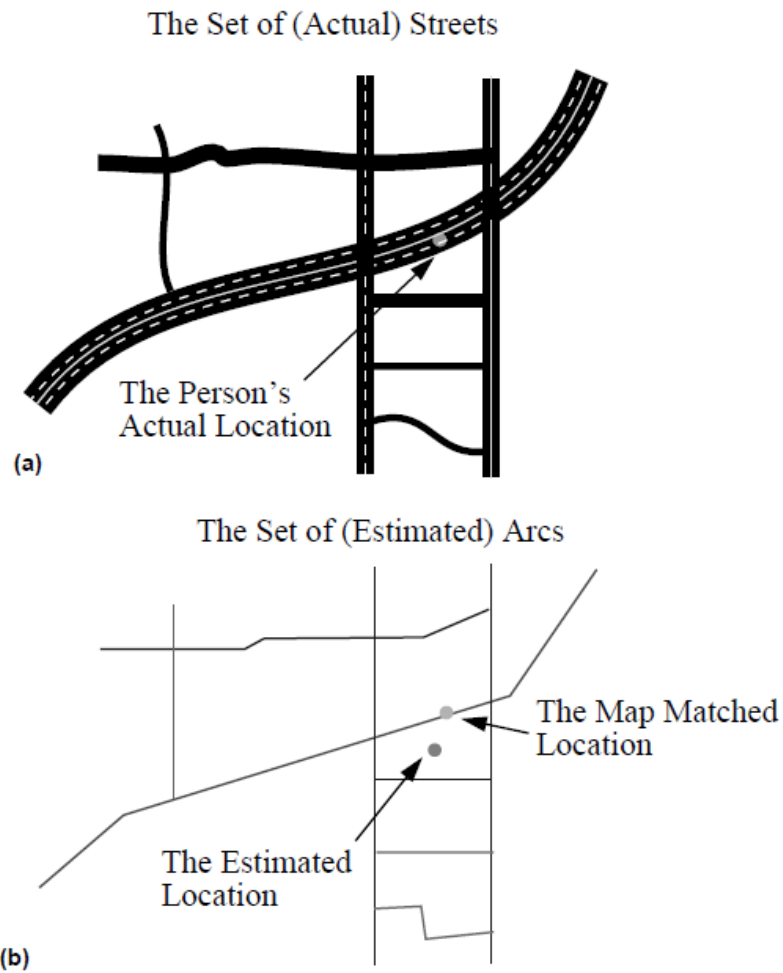


Figura 2.3: Problema de *map-matching* [WBK00]

2.2.1 Análise Geométrica

A forma mais simples de se implementar um algoritmo de *map-matching* é considerar apenas aspectos geométricos entre a rede e o ponto que se pretende fazer *matching*. Em seguida serão apresentadas três abordagens diferentes.

2.2.2 Point-to-point

A abordagem mais intuitiva para resolver o problema de *map-matching* é combinar o ponto pretendido com o nó ou ligação mais próximo. Embora sendo uma solução simples, pode em certos casos produzir resultados indesejados [GRE02].

Na Figura 2.4 pode-se observar que os pontos P^2 a P^4 com a utilização deste algoritmo correspondem a nós em segmentos que não são o correcto.

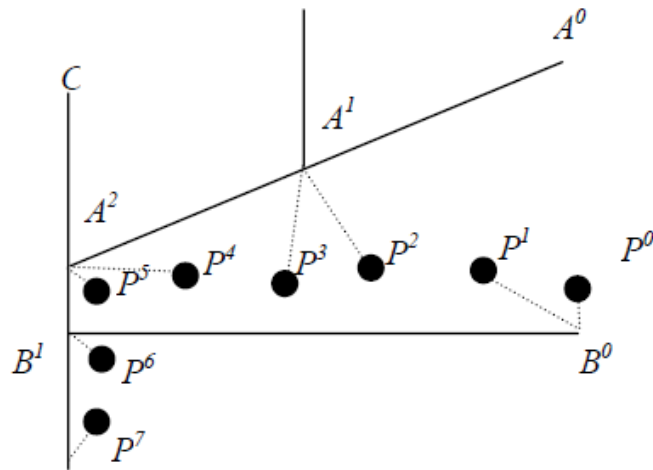


Figura 2.4: Corresponder o ponto GPS ao nó mais próximo [GRE02]

2.2.3 Point-to-curve

Uma outra forma intuitiva de fazer *map-matching* é encontrar o segmento mais próximo do ponto GPS. O arco mais próximo pode ser encontrado através da menor distância entre um ponto e uma linha que é a distância medida na perpendicular a essa linha. [GRE02]

Como se pode observar na Figura 2.5, este algoritmo pode apresentar erros nas soluções. Os pontos P^4 e P^5 deveriam ser correspondidos ao segmento B, mas isso não acontece porque o ponto P^4 está mais próximo do segmento A e o ponto P^5 mais próximo de B.

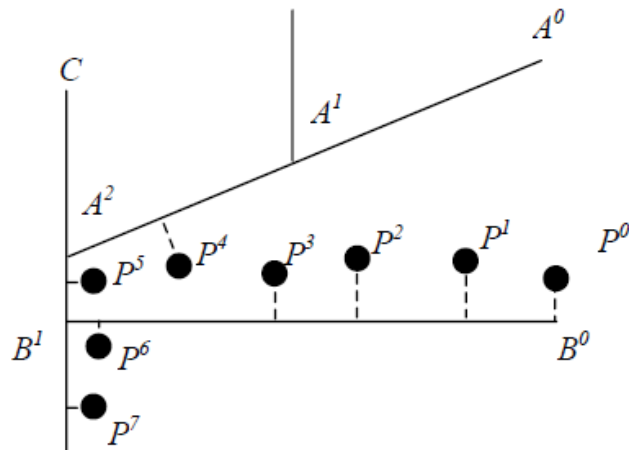


Figura 2.5: Corresponder um ponto GPS ao segmento mais próximo [GRE02]

2.2.4 Curve-to-curve

Outra forma de fazer *map-matching* é considerando vários pontos de um segmento e verificar qual o segmento que se aproxima mais desse.

Para se calcular a distância entre as curvas pode-se usar várias técnicas [BKO96].

2.2.5 Análise Topológica

A performance dos algoritmos somente geométricos podem ser melhorados se usarem também a informação topológica, principalmente através dos pontos iniciais e com informação topológica da rede, reduzir o número de possíveis segmentos drasticamente [BKO96].

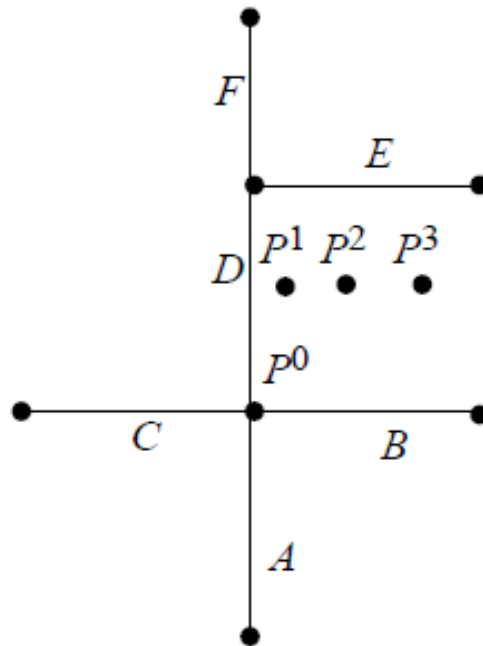


Figura 2.6: Usar informação topológica [BKO96]

Na Figura 2.6 caso se use unicamente informação geométrica os pontos p1, p2, p3 corresponderiam ao segmento E. No caso de utilização de informação topológica e sabendo que a posição inicial é p0 o ponto p1 só poderá pertencer ao segmento A, B, C ou D, prevenindo que o ponto p3 pertença a E.

2.2.6 Estatísticos

O algoritmo probabilístico tem como base a construção de uma região elíptica ou rectangular na região próxima da posição. Esta zona representada determina o possível erro

existente de forma a determinar a posição correcta na rede através dos segmentos que se encontram dentro dessa região. Caso existam mais do que um segmento usa-se critérios vários como a proximidade, conectividade, entre outros. [QON07]

2.3 Algoritmos de roteamento

Nesta secção serão introduzidos dois algoritmos, *Dijkstra* e *A**, que permitam calcular rotas de caminho mais curto ou de esforço necessário menor.

2.3.1 Dijkstra

O algoritmo de *Dijkstra* [DIJ59] baseia-se no facto de quando se pretende calcular o caminho mínimo entre dois nós P e Q e sendo o nó R um nó no caminho mínimo entre eles, o conhecimento de R implica o conhecimento do caminho mínimo de P para R. Assim os caminhos mínimos de P para os outros nós são construídos em ordem crescente de comprimento até que Q seja atingido.

O algoritmo funciona guardando em cada nó visitado o custo do menor caminho encontrado até ao momento do nó de início S e o de destino T. Este custo tem como valor $d(T)$ [SGL07].

Assim os passos necessários são:

- Inicializar todos os nós com $d(T) = \infty$;
- Para cada nó ligado a S adicionar o custo de S e o custo da ligação. Se o custo for menor que o custo de $d(T)$ trocar pelo menor custo;
- Escolher o nó com menor valor de $d(T)$ e seleccionar o nó como sendo a localização actual;
- Iterar entre os passos 2 e 3 enquanto não for atingido o nó de destino ou todos os nós sejam visitados.

2.3.2 A*

O algoritmo *A** visita os nós numa ordem que que será preferencialmente mais rápida do que a pesquisa em todos os nós no algoritmo de *Dijkstra*.

Para seleccionar o nó que será visitado usa tipicamente uma heurística do custo que falta para chegar ao final e a distância actualmente guardada.

Tipicamente o algoritmo A* visitará menos nós para um caminho específico origem/destino exacto em relação a outros algoritmos e será em média mais rápido [SGL07].

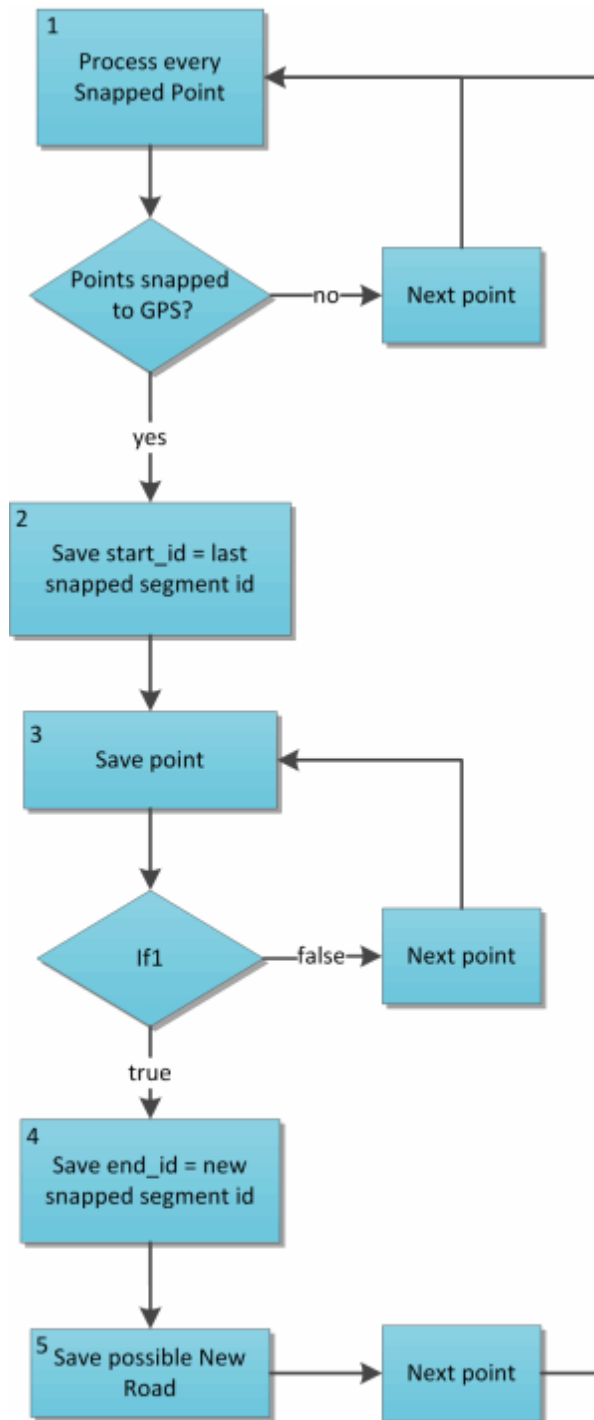
2.4 Correção de mapas

Devido a alterações na rede física ou a incorrecções criadas na passagem da rede para formato digital, a informação deve ser actualizada para corresponder à realidade. Assim uma forma de se actualizar a rede é através de dados de pontos GPS.

Nesta sessão será analisado três algoritmos diferentes. Em primeiro lugar um algoritmo que permita adicionar novos percursos à rede que não existiam previamente, que pode ocorrer por a rede estar incompleta ou por construção de uma nova rua. Será ainda necessário recorrer ao dois outros algoritmos, um que permita actualizar a posição das ruas e outro que permita actualizar a posição dos cruzamentos.

2.4.1 Detecção de novas ruas

Para determinar uma nova rua é necessário recorrer a um algoritmo com duas fases. Numa primeira fase determinar as possíveis novas ruas através da detecção de pontos que não façam *matching* com nenhum dos percursos existentes. A segunda fase consiste em utilizar a informação da primeira fase e determinar a geometria da nova rua. [FCR09b]



If1: Point snapped to a segment and distance from GPS to location is acceptable

Figura 2.7: Detecção de nova rua – primeira fase [FCR09b]

A primeira fase do algoritmo presente na Figura 2.7, para todos os pontos de um segmento (1), guarda-se o ponto inicial (2), obtém-se todos os pontos que não pertençam a uma rua já

existente (3), quando algum ponto pertencer a uma rua existente para-se o processo (4) e guarda-se como sendo uma possível nova rua (5) para ser processado na segunda fase.

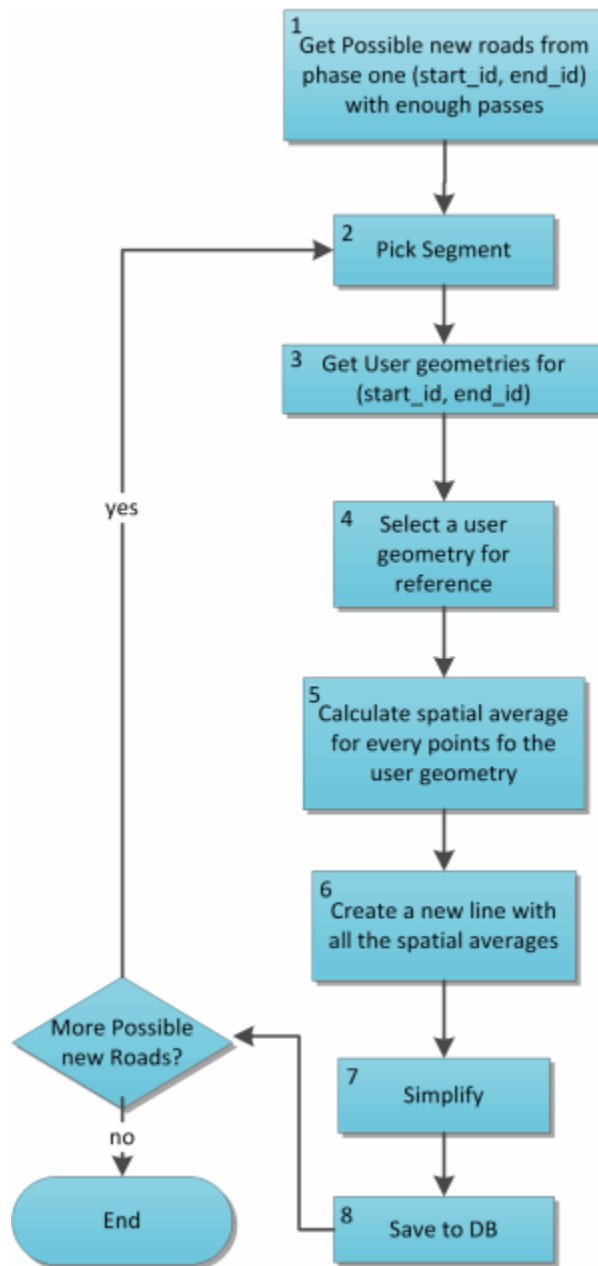


Figura 2.8: Detecção de nova rua – segunda fase [FCR09b]

A segunda fase do algoritmo de detecção de novas ruas na Figura 2.8 consiste em que para cada segmento encontrado na primeira fase do algoritmo, obter a geometria submetida pelo utilizador (3), calcular a média espacial para todos os pontos (5), onde se obtém uma nova linha (6).

2.4.2 Detecção de ruas com geometria errada

Para a detecção e correcção da geometria das ruas é necessário numa primeira fase detectar as ruas que não têm a geometria correcta e numa segunda fase calcular a posição correcta [FCR09a].

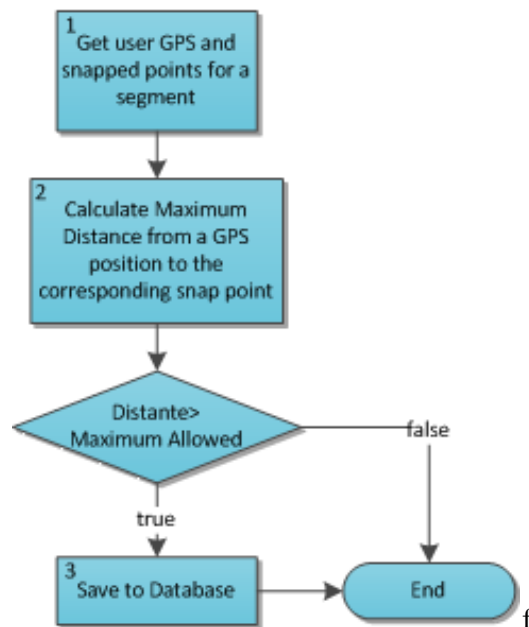


Figura 2.9: Algoritmo de detecção da geometria da rua – primeira fase [FCR09a]

Numa primeira fase encontram-se as possíveis ruas geometria errada como mostra na Figura 2.9 obtendo os pontos GPS do utilizador e os pontos correspondentes no segmento (1), calcular a distância máxima existente entre os pontos do utilizador e o segmento (2), caso essa distancia seja suficiente para ser considerado uma nova rua essa informação será guardada para ser processada pela segunda fase do algoritmo.

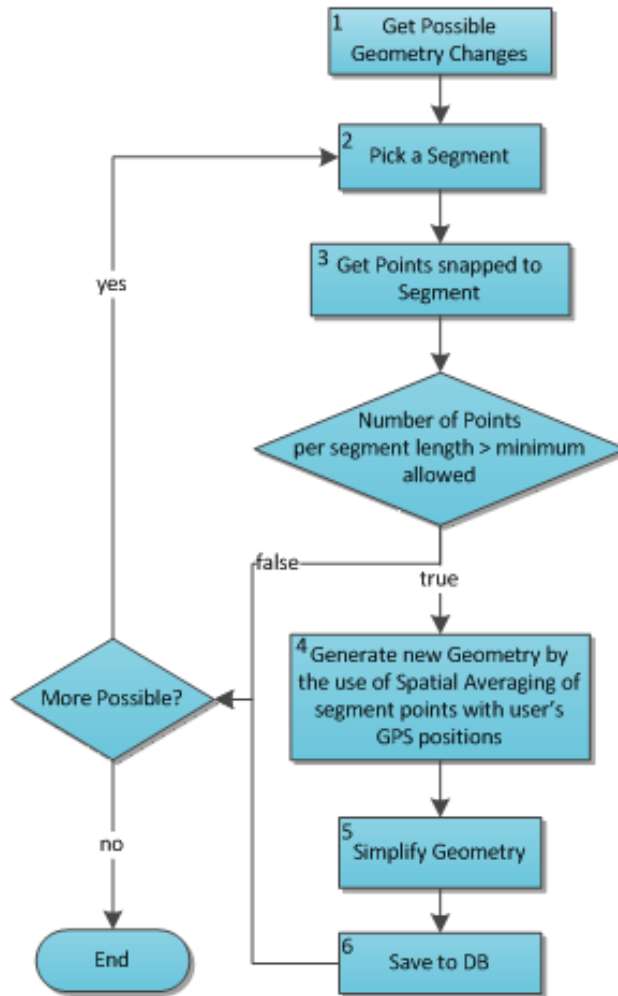


Figura 2.10: Algoritmo de detecção da geometria da rua – segunda fase [FCR09a]

Na segunda fase, como se pode observar na Figura 2.10, será criada a nova localização da rua. Esta operação é feita através do cálculo da média espacial entre o segmento existente e os dados do utilizador (4).

2.4.3 Correção dos cruzamentos das ruas

Para corrigir a localização de um cruzamento entre ruas é necessário, segundo [FCR09a], calcular a média dos possíveis percursos que passam pelo cruzamento pretendido, obtidos dos pontos GPS.

Revisão Bibliográfica

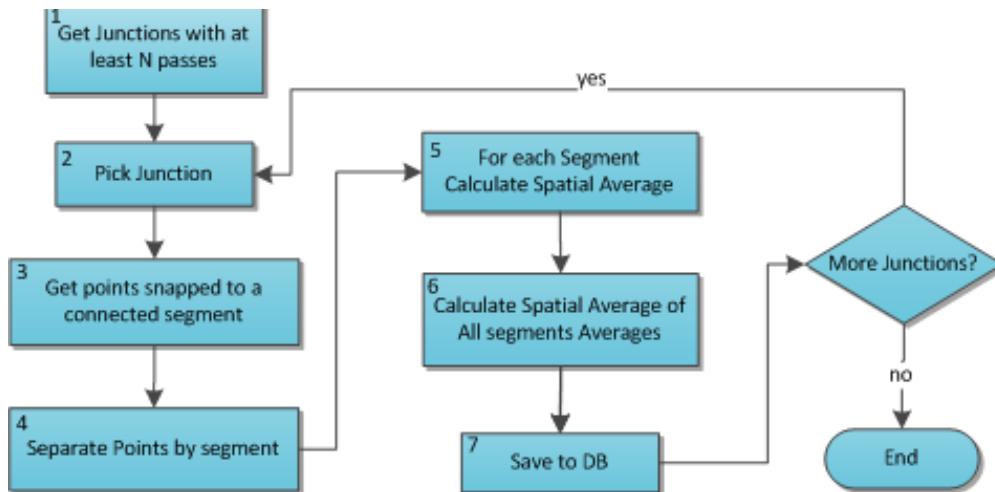


Figura 2.11: Diagrama do algoritmo para correcção dos cruzamentos [FCR09a]

Como se pode ver no diagrama da Figura 2.11, obtém-se os pontos que poderão pertencer aos percursos que fazem parte do cruzamento (3), separam-se os pontos encontrados nos segmentos correspondentes (4), para cada segmento encontrado calcula-se a média espacial (5), por fim, calcula-se a média espacial de todos os percursos obtidos do passo anterior. Assim obtém-se uma estimativa ponderada do local correcto do cruzamento.

Capítulo 3

Metodologia

A metodologia desenvolvida consiste numa primeira fase, na obtenção e processamento da rede existente proveniente de fontes *open source* e numa segunda fase e através da recolha de informação dos utilizadores da bicicleta a actualização da rede a partir dos trajectos GPS, bem como a sua classificação quanto a parâmetros relevantes para a sua utilização por bicicletas como refere na Figura 3.1.

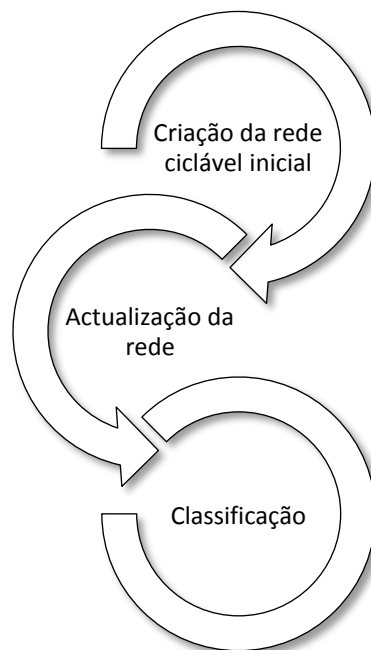


Figura 3.1: Funcionamento geral da metodologia

Para além da metodologia, irá também ser abordado o modelo de dados que foi desenvolvido para permitir a criação da rede ciclável e sua classificação, forma de criação da rede ciclável inicial, actualização da rede já existente a partir de dados dos utilizadores e por fim a sua classificação.

3.1 Modelo de dados

A rede ciclável será guardada na forma de grafo, como se pode observar na Figura 3.2, sendo portanto composta por nós (*node*) e arestas (*edge*).

Os nós são representados pela sua geometria, o que corresponde à longitude/latitude/altitude desse mesmo ponto. No caso das arestas, que representam a ligação entre dois nós, têm uma geometria do tipo polilinha, composta por todos os vértices que a compõem. É ainda guardada a informação relativamente ao comprimento, velocidade média, tempo médio de percurso e altitude acumulada.

As arestas podem ser extraídas de dois locais:

- *OpenStreetMap* – onde será necessário guardar o identificador único bem como a sua versão, esta informação permite que sejam feitas actualizações à rede caso seja feita alguma actualização no *OpenStreetMap*;
- *Tracks GPS* dos utilizadores de bicicleta – a partir desta informação submetida pelos utilizadores pode ser adicionada mais informação à rede, adicionando informação que permite fazer novos tipos de classificação que só com a informação do *OpenStreetMap* não seria possível.

Cada aresta que em conjunto com outras possa formar uma nova via, pode ser classificada segundo vários tipos: ciclovias, estradas, caminhos, entre outros. Caso seja um trilho (*track*) ou caminho (*path*) poderá ainda ser adicionada informação acerca de ciclismo de montanha que está de acordo com o modelo usado pelo *OpenStreetMap*:

- *scale* – Classifica numa escala de 0 a 5 a dificuldade em subir ou descer este troço;
- *type* – Classificação do tipo de trilho ou caminho na perspectiva de uma bicicleta de montanha. Pode usar-se *crosscountry*, *allmountain*, *downhill*, *trial*, *freeride*;
- *uphill* – Classificação da dificuldade para ser subida por bicicleta de montanha.

Metodologia

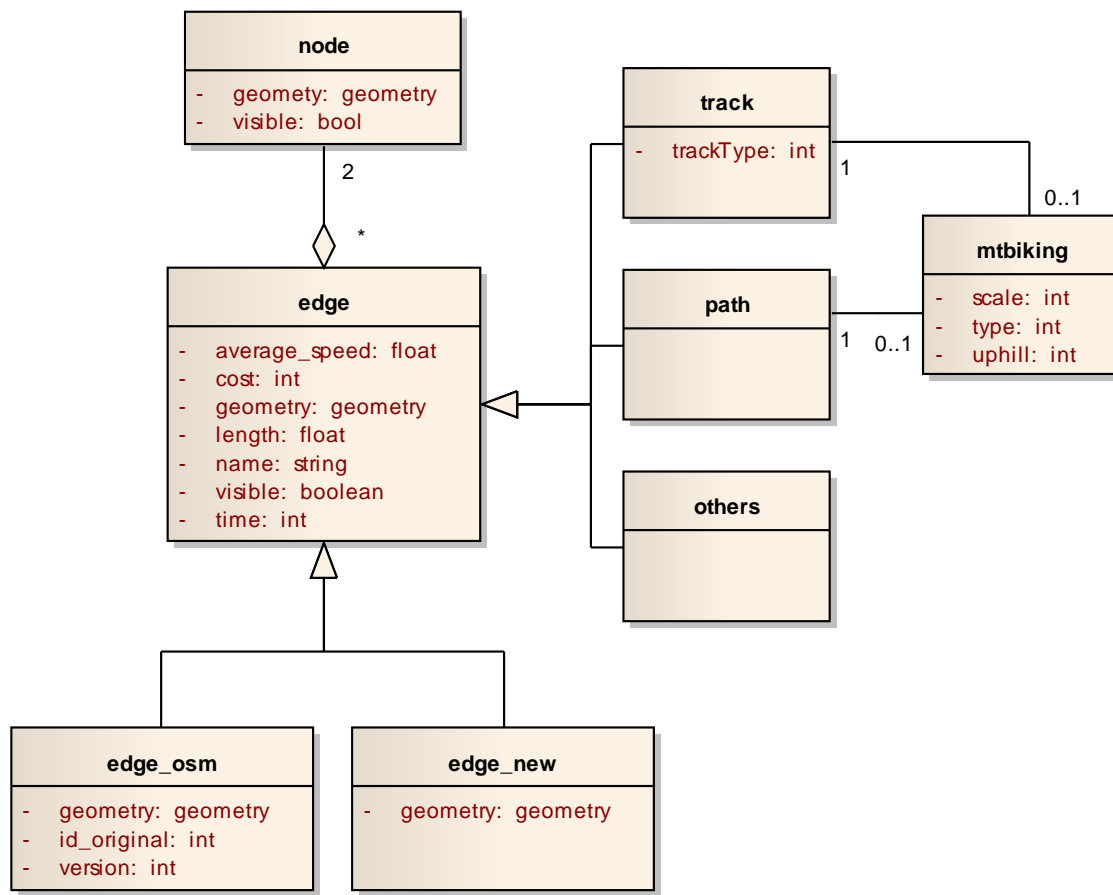


Figura 3.2: Modelo de dados - rede ciclável

Para guardar a informação relativa aos *tracks* GPS dos utilizadores, são armazenados a data e o nome, bem como um identificador que refere se a informação nele guardado é válida para cálculo de velocidades (por exemplo devido à falta da data nos pontos GPS). Para cada *track* é guardado o conjunto de pontos GPS com a sua localização e a data e hora que foi obtido. Existe também a referência ao utilizador que submeteu os *tracks* GPS. O módulo do diagrama relacional referente a esta parte está presente na Figura 3.3.

Metodologia

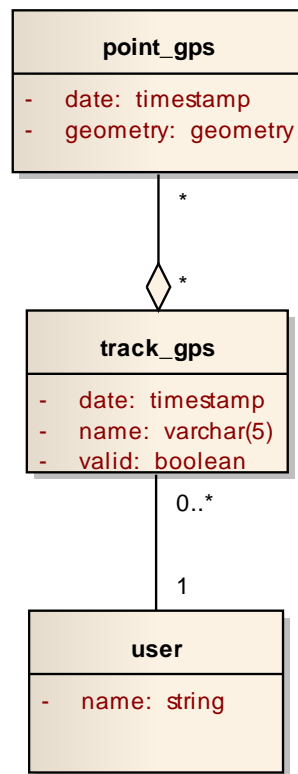


Figura 3.3: Modelo de dados - *Tracks* GPS

3.2 Criação da rede ciclável inicial

A rede inicial para a base de toda a rede ciclável pode ser obtida através do *OpenStreetMap*. Nesta secção será demonstrada a forma de criação desta mesma rede a partir da fonte seleccionada.

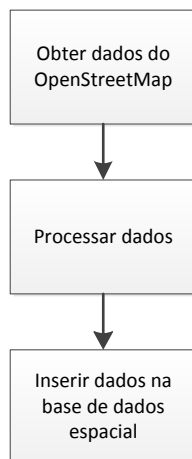


Figura 3.4: Criação da rede inicial

Metodologia

Para a criação da rede é necessário obter os dados da fonte já existente, neste caso do *OpenStreetMap*, fazer o processamento, a sua conversão e a inserção da informação para a base de dados desenvolvida como representado na

Figura 3.4.

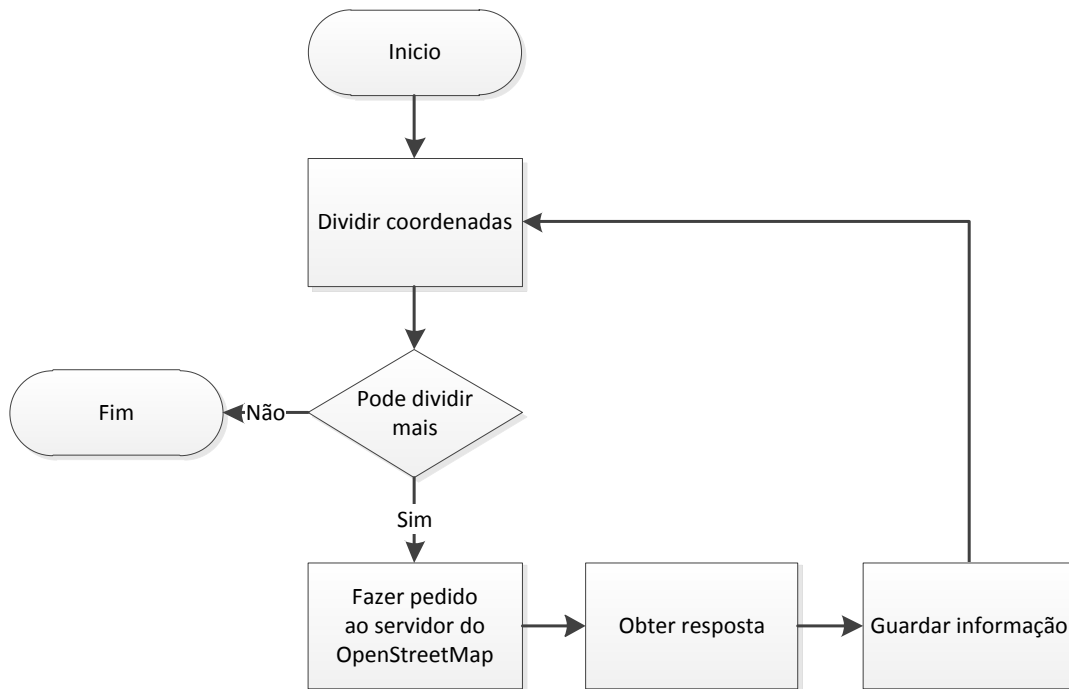


Figura 3.5: Obter informação do *OpenStreetMap*

Para a obtenção de dados do *OpenStreetMap* e devido a limitações na API do *OpenStreetMap* só é permitido obter a informação da rede para áreas de 0,25 graus quadrados. Assim, como é visível no fluxograma da Figura 3.5, é necessário dividir a área que se pretende obter em áreas mais pequenas com dimensões inferiores à imposta. Para cada subárea tem de se fazer o pedido ao servidor para obtenção do mapa, aguardar a resposta e por fim guardar a informação para tratamento posterior.

Metodologia

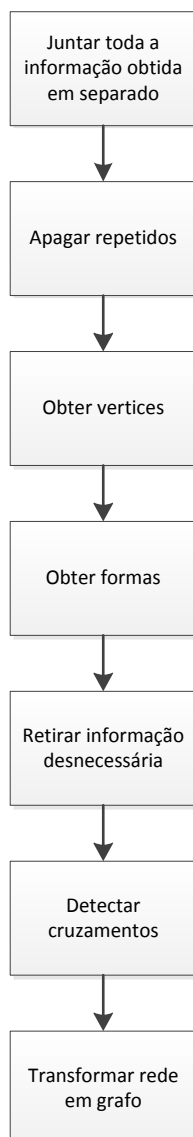


Figura 3.6: Filtragem de dados do *OpenStreetMap*

Devido à informação obtida no *OpenStreetMap* ter de ser feita a partir de áreas mais pequenas e sendo que a informação obtida numa das áreas obtém também informação de outras, devido a que todas as geometrias serão enviadas de forma completa. Assim tem de ser apagar a informação repetida, como se verifica no fluxograma da Figura 3.6.

Seguidamente é necessário separar a informação referente a nós e a formas, removendo também locais onde a bicicleta não poderá andar devido a restrições legais sendo exemplo disso as auto-estradas e vias equiparadas. Tem também de se fazer filtragem removendo nós e formas não pertençam a ruas ou estradas como as fronteiras, delimitações de edifícios, entre outros.

Como a rede será representada e guardada em forma grafo, é necessário detectar onde se localizam os cruzamentos. Os cruzamentos são obtidos verificando quais os pontos que aparecem em mais que uma rua, bem como o ultimo ponto de uma rua sem saída.

Metodologia

Determinados os nós de cruzamento é necessário calcular as arestas do grafo. Isto é possível dividindo as ruas pelos nós dos cruzamentos e fazer as respectivas ligações entre nós e arestas.

3.3 Actualização da rede ciclável

A rede ciclável já existente tem de ser actualizada devido à inexistência de certas vias, principalmente trilhos na montanha. Para que isto seja possível, a informação será inferida a partir de informação de *tracks* obtidos dos aparelhos GPS submetidos pelos utilizadores.

Os ficheiros dos *tracks*, normalmente no formato *gpx*, armazenam informação relativamente à localização do ponto com a sua latitude e longitude, tendo por vezes a altitude. Para cada ponto pode existir ainda a data e hora em que foi obtido.

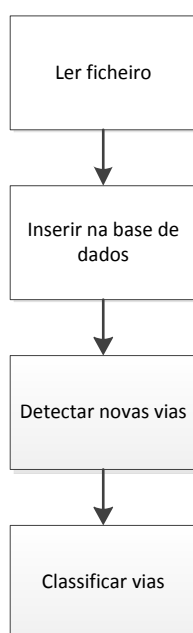


Figura 3.7: Actualização da rede

Os dados submetidos pelos utilizadores depois de serem lidos, interpretados e filtrados são inseridos na base de dados. Os dados existentes na base de dados permitiram detectar novas vias e posteriormente ajudar a classificar a rede, como apresentado no fluxograma da Figura 3.7.

3.3.1 Actualizar a rede a partir das fontes *open source*

Visto que as fontes de onde se obteve a rede ciclável inicialmente estão em constante actualização é necessário obter essas actualizações e submetê-las na base de dados existente.

Metodologia

Para isso é necessário obter a informação do *OpenStreetMap*, como se verifica no fluxograma da Figura 3.8. Em seguida é necessário filtrar retirar a informação que foi actualizada depois da última vez em que se obteve a informação da fonte *open source* e por fim é necessário verificar as vias que são afectadas na base de dados e actualizar.

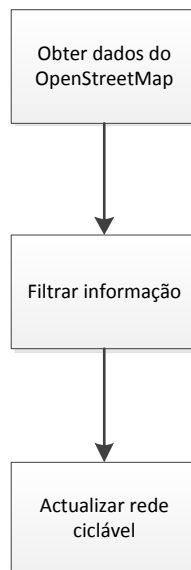


Figura 3.8: Actualizar rede através de fonte *open source*

3.3.2 Detectar e criar novas vias

Para determinar uma nova via é necessário ter a rede inicial já criada e os dados submetidos pelos utilizadores. O problema da detecção de novas vias encontra-se representado de forma gráfica na Figura 3.9, sendo que as cruces vermelhas representam ponto GPS por onde o utilizador passou.



Figura 3.9: Possível nova via

Para determinar a nova via a partir de um *track* GPS (

Figura 3.10), é necessário determinar quais os pontos do *track* que pertencem e que não pertencem a uma via já existente, podendo isto ser feito através de um algoritmo de *map-matching*.

Metodologia

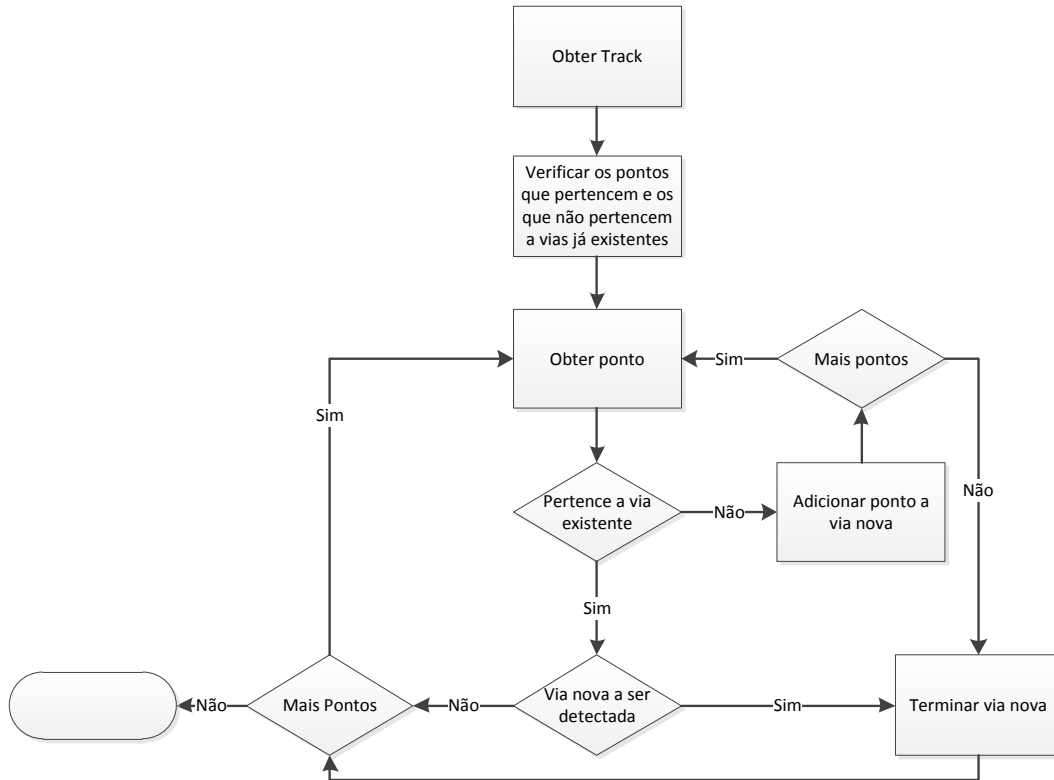


Figura 3.10: Detectar possíveis novas vias

Para cada ponto encontrado, caso não pertença a uma determinada via é referido como sendo uma possível nova via até que, os pontos comecem a pertencer a uma via já existente. O último e primeiro que pertençam a uma via imediatamente depois e antes, respectivamente, de uma possível nova via também são contemplados para a geometria da via, uma vez que facilita a determinação com maior precisão dos cruzamentos. Este algoritmo é apresentado no fluxograma da Figura 3.10.

Metodologia

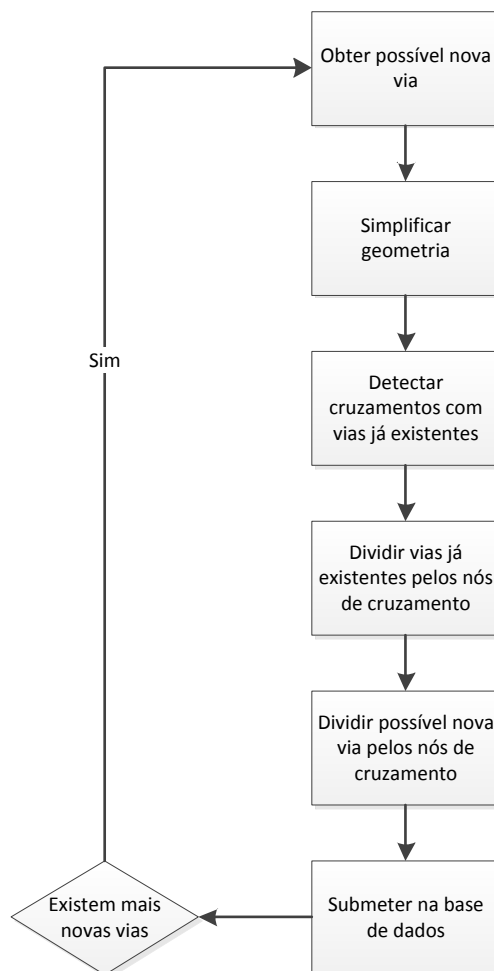


Figura 3.11: Actualização da rede com nova via

Como é referido no fluxograma da Figura 3.11, depois de obtida uma possível nova via faz-se uma simplificação da geometria usando por exemplo o algoritmo de *Douglas-Peucker* [PMI97]. Esta simplificação vai permitir remover informação desnecessária para que as geometrias das vias guardadas na base de dados ocupem menos espaço e sejam mais facilmente manipuláveis.

Depois de simplificada a geometria é necessário calcular os cruzamentos ou pontos de ligação destas novas vias às já existentes na rede. Caso existam cruzamentos terá de ser refeita a rede existente, adicionando o nó correspondente ao cruzamento detectado e partir a via já existente em que esse cruzamento foi obtido para que o grafo representativo da rede esteja correcto.

Da mesma forma que se partiu as vias da rede já existente nos cruzamentos com a nova via, o mesmo terá de ser feito para a nova via detectada e que se pretende inserir.

3.4 Classificação da rede

A classificação a vários níveis da rede tem especial relevância para permitir fazer roteamento na rede.

A classificação é feita segundo quatro tipos:

- Altitude acumulada.
- Velocidade média;
- Tempo médio;
- Comprimento das vias;

3.4.1 Altitude acumulada

O princípio para o cálculo da altitude acumulada será tentar medir o esforço necessário para percorrer uma certa via. Uma vez que para o utilizador da bicicleta o maior problema são os desníveis de altitude, para isso uma aproximação a ser feita é o cálculo da área do desnível de altitude ao longo da via como se pode observar na Figura 3.12. Aqui é possível verificar que a azul estão marcadas as partes da via que são a subir e a avermelhado partes da via que são a descer.

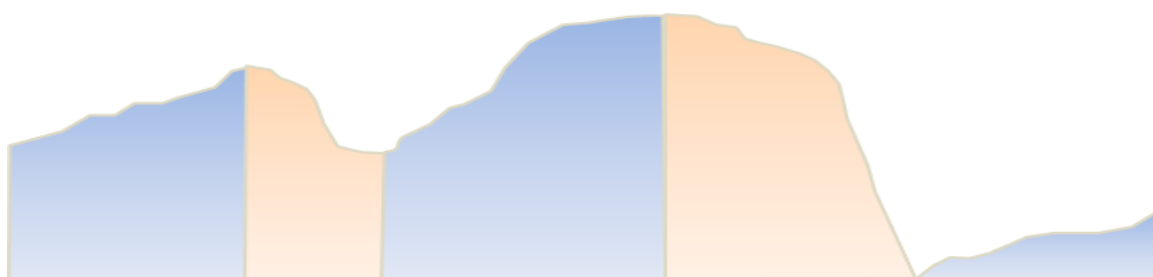


Figura 3.12: Altitude de uma via

Para classificar uma via segundo seu desnível de altitude, Figura 3.13, é necessário verificar se é uma via já existente na rede original ou se trata-se de uma nova via encontrada a partir de informação submetida pelo utilizador.

Caso seja uma via nova a classificação é determinada pela soma acumulada de todas as altitudes da via nas subidas numa direcção e as descidas que representarão o esforço para o quem se desloca em sentido contrário. No caso de ser uma via já existente o cálculo da classificação da via segundo a sua altitude é pelo cálculo da altitude acumulada pelo início e fim que descrevem cada via, caso os pontos tenham altitude definida inerentes da criação de novas vias.

Metodologia

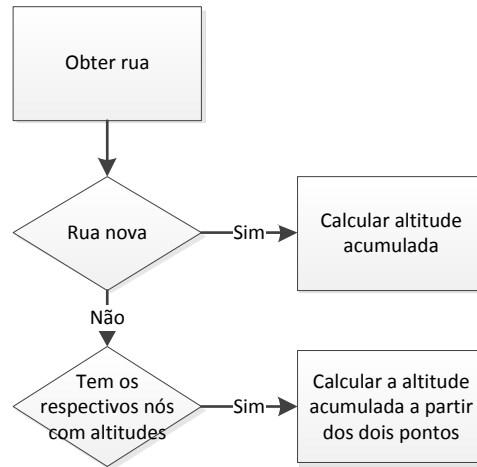


Figura 3.13: Classificar a rede segundo as altitudes

3.4.2 Tempo e velocidade média

Para classificar cada via segundo a velocidade média praticada ou tempo médio é necessário que existam *tracks* GPS que passem por essa via com informação sobre a hora de passagem em cada ponto. Assim, depois de obter o *track* que passa na via determina-se o instante de tempo em que entrou na via, bem como, o instante de tempo em que saiu da via obtendo assim o tempo que foi gasto para percorrer a via.

Metodologia

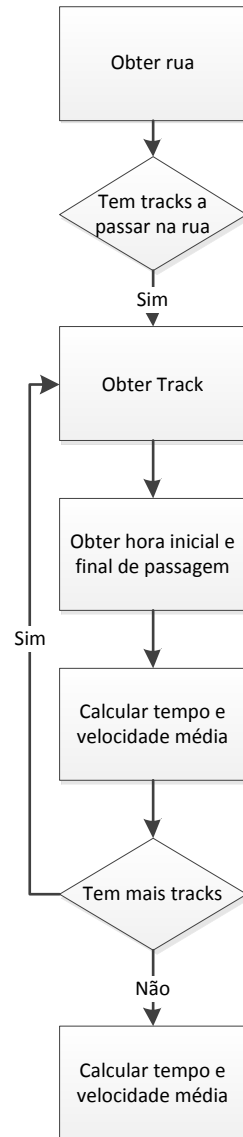


Figura 3.14: Classificar a rede segundo a velocidade média e tempo

A partir do comprimento calculado através do comprimento da geometria da via pode assim calcular-se a velocidade média para esse *track* nessa via.

Depois de calculado o tempo e velocidade para cada *track* que passa na via, calcula-se a média geral de todos os *tracks* nessa mesma via e actualiza-se a classificação na base de dados.

3.5 Roteamento

O roteamento permite encontrar caminhos mais adaptados às preferências de cada utilizador utilizando a classificação feita na rede ciclável.

Para efectuar o roteamento, como é visível na Figura 3.15, é necessário, em primeiro lugar, seleccionar um ponto de partida e um ponto de destino, seguidamente é necessário obter a rede ciclável que se encontra guardada na base de dados excluindo vias que não tenham certos parâmetros de classificação quando esses serão necessários na tarefa de roteamento. Uma vez obtida a rede ciclável é necessário adaptar de forma a facilitar a tarefa de roteamento e por fim calcular o caminho pretendido através do algoritmo de *Dijkstra*. Este algoritmo foi utilizado como base pois é um algoritmo de simples implementação e permite obter resultados de forma correcta e de forma relativamente eficiente.

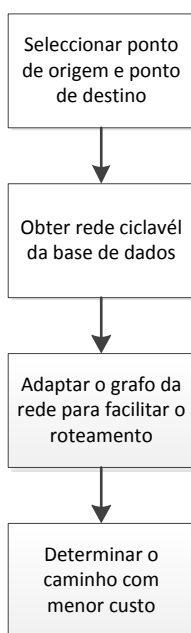


Figura 3.15: Roteamento

O algoritmo para cálculo do percurso com menor custo representado na Figura 3.16, é baseado no algoritmo de *Dijkstra* onde se preserva uma lista de nós que deveram ser visitados. Inicialmente é adicionado à lista de nós para visitar o nó onde se pretende iniciar o percurso.

Quando a lista de nós para visitar estiver vazia é porque já foram encontrados todos os caminhos de custo menor. Será necessário calcular o percurso do nó inicial ao final através das informações presente em cada nó que são alteradas pelo algoritmo e indicam o caminho que permitem ligar de um nó ao outro.

Sempre que a lista de nós para visitar não se encontrar vazia é necessário obter dessa lista o nó que apresenta um menor custo (custo necessário até chegar ao nó desde o inicio do percurso). Para esse nó obtido é necessário calcular para cada nó que se liga a ele o valor do custo total

Metodologia

necessário para lá chegar. Caso o valor do custo do caminho para lá chegar seja menor que o actualmente existente nesse mesmo nó, o valor desse custo será actualizado, referido que para se chegar a esse nó é necessário passar pelo nó que estamos actualmente a visitar. Por fim é adicionado o nó que se actualizou à lista de nós para serem visitados.

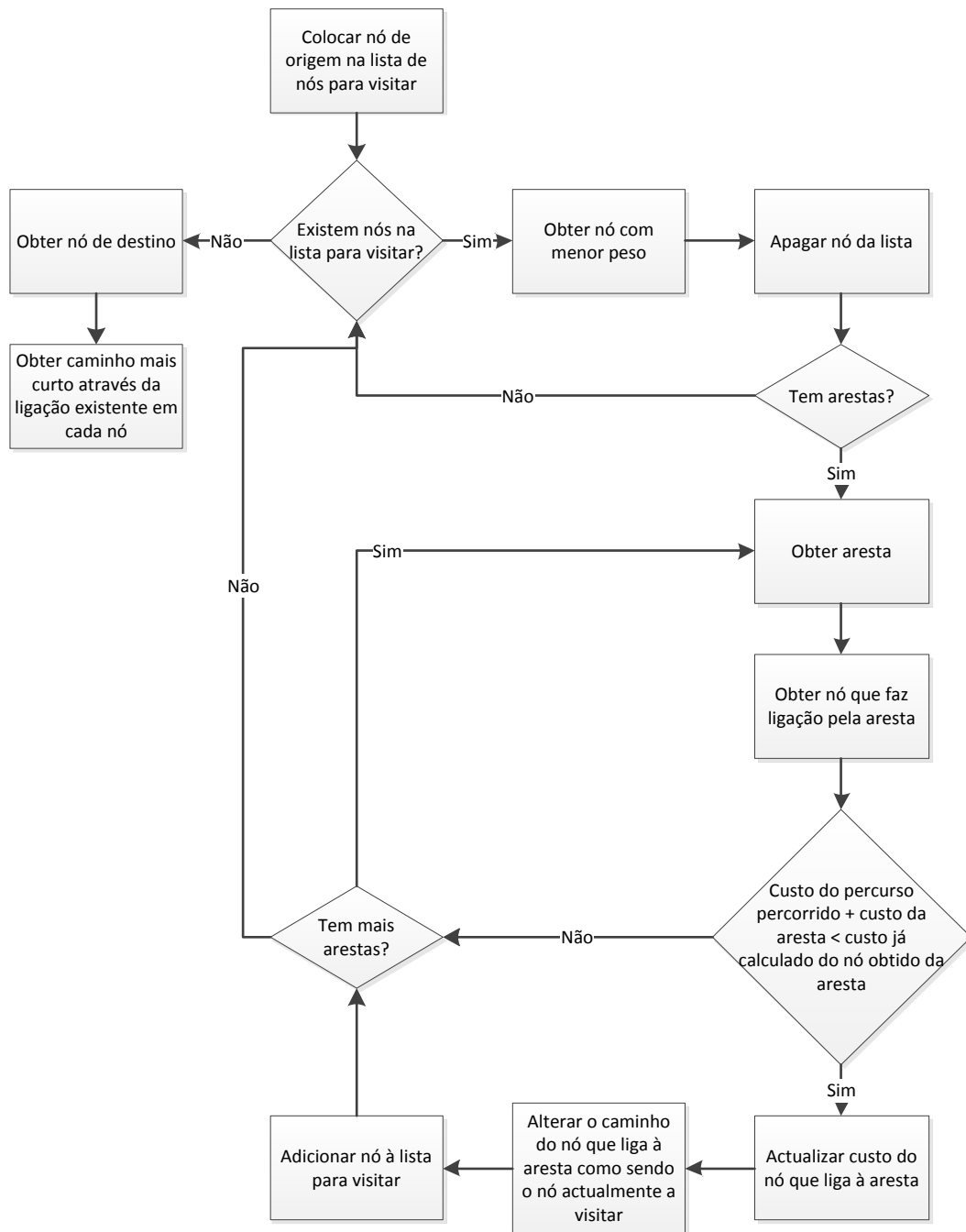


Figura 3.16: Caminho de menor custo

O cálculo do custo para cada nó é calculado tendo em conta os quatro parâmetros disponíveis da classificação da rede, o comprimento da via, a altitude acumulada, o tempo

Metodologia

médio necessário para o percorrer e a velocidade média praticada. A cada parâmetro é atribuído um peso dependendo das preferências do utilizador segundo o cálculo efectuado através de uma média pesada:

$$(peso1 * comprimento + peso2 * altitude + peso3 * tempo + peso4 * velocidade) / (peso1 + peso2 + peso3 + peso4);$$

Capítulo 4

Implementação e resultados

Este capítulo baseia-se na metodologia apresentada no capítulo 3 e define o processo de implementação da solução desenvolvida. É apresentada a arquitectura do sistema, algumas considerações relacionadas com a implementação bem como o método utilizado para a visualização. É também apresentado os resultados da implementação realizada, os dados que foram usados nos testes, bem como os resultados da rede ciclável inicial, detecção de novas vias e do roteamento.

4.1 Arquitectura

O sistema implementado foi architectado de forma a ser separado por vários módulos para facilitar a actualização e separação de funcionalidades segundo a sua lógica como visível na Figura 4.1.

Implementação e resultados

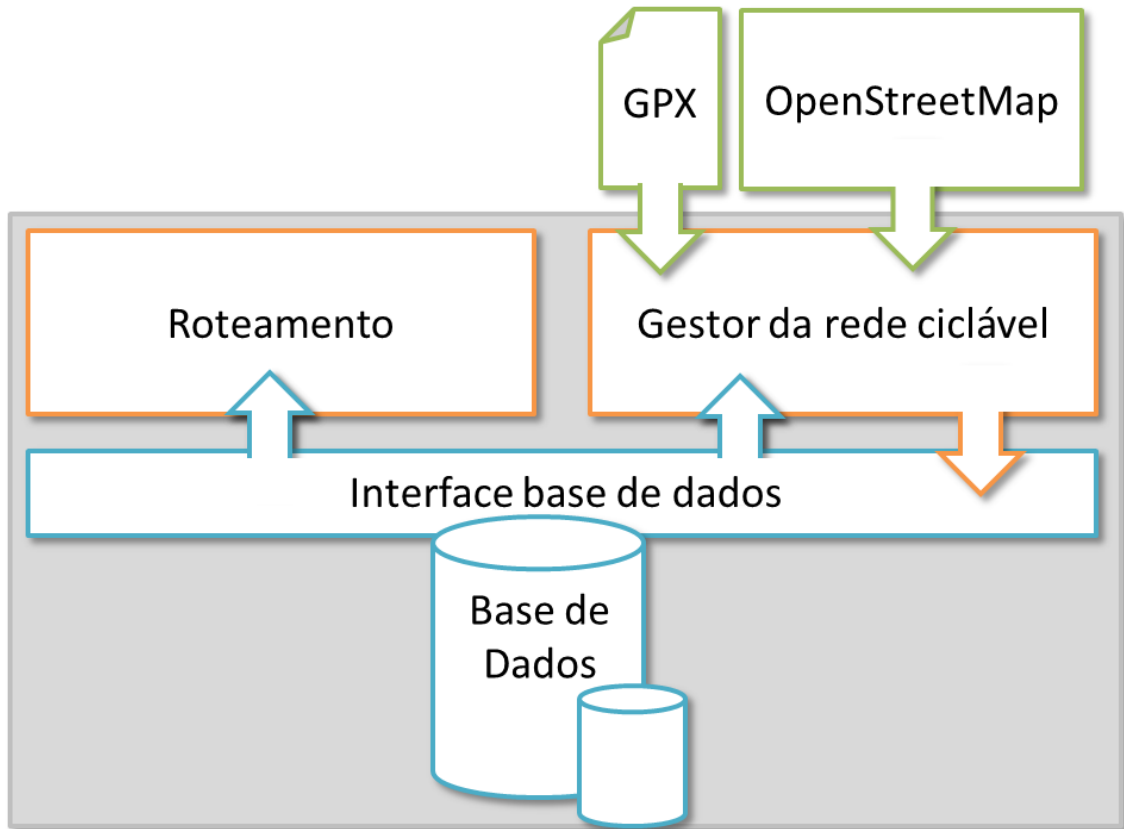


Figura 4.1: Arquitectura do sistema

4.1.1 Base de dados

A base de dados foi implementada em *PostgreSQL* com *PostGis*, permitindo assim a estruturação, armazenamento e pesquisa de forma eficiente dos dados espaciais. Na base de dados é guardada toda a rede ciclável, bem como, os dados dos ficheiros dos *tracks* dos utilizadores. O esquema relacional da base de dados está presente no anexo Modelo Relacional e baseia-se no modelo de dados apresentado no capítulo Metodologia.

4.1.2 Interface com a base de dados

Foi implementada uma interface onde existem as funções que permitem o acesso à base de dados de forma uniformizada e correcta. Nesta interface existem métodos que permitem fazer a ligação, criação e apagamento da base de dados, bem como as funções de obtenção, inserção, e actualização dos dados existentes na base de dados. Este módulo foi implementado em *c#* e comunica com a base de dados através do *driver ODBC*.

4.1.3 Gestor da rede ciclável

O gestor da rede ciclável é responsável pela obtenção da informação a partir do *OpenStreetMap*, tratamento da informação e sua filtragem. Neste módulo são também recebidos os ficheiros com os *tracks* dos utilizadores, é feita a sua filtragem de forma a possibilitar a detecção de novas vias bem como permitir a sua classificação.

O gestor foi implementado em *c#* sendo que obtém a informação do *OpenStreetMap* a partir da sua *API* através de *webservice*. O *parser* da informação obtida que é numa especificação em *XML* própria do *OpenStreetMap* é feita recorrendo à tecnologia *LINQ to XML* que permite de forma rápida, com código de fácil leitura e facilmente actualizável a leitura da informação e seu tratamento. Os ficheiros submetidos também são lidos recorrendo a esta tecnologia.

4.1.4 Roteamento

O módulo de roteamento é responsável pela criação de percursos com um menor custo segundo os vários critérios disponíveis permitindo dar pesos diferentes a cada um dos tipos de classificação disponíveis.

Como esta dissertação foi pensada e efectuada em colaboração com outra dissertação realizada na mesma faculdade a fim de incorporar o sistema num só, como uma aplicação a disponibilizar numa rede social, que no caso é o *Facebook*, este modulo foi criado com especial cuidado de forma a permitir que seja interligado de forma fácil, permitindo obter o melhor percurso que a partir das referencias do ponto de origem e destino, bem como o valor dos pesos a dar a cada um dos custos disponíveis de forma a adaptar-se melhor ao utilizador de destino, com as suas preferências.

4.2 Visualização

Para a visualização pode ser utilizado qualquer editor *GIS* mas neste caso foi utilizado *uDig*, visível na Figura 4.2, que permite visualizar a geometria da rede e de todas as vias que foram adicionadas, e no caso do *Google Earth*, também utilizado no projecto, permite a visualização da rede em cima de uma imagem de satélite.

No caso do *Google Earth* foi feito um exportador para permitir exportar no formato de ficheiro *KML* de forma a ser possível a sua visualização.

Implementação e resultados

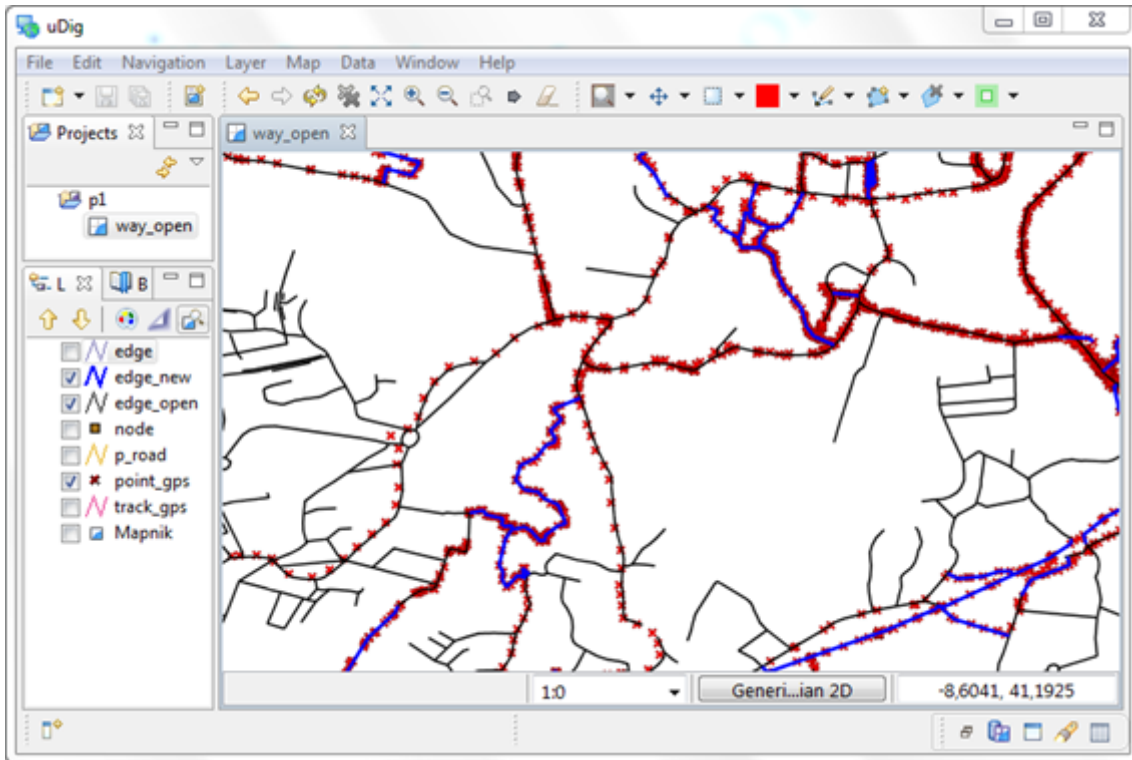


Figura 4.2: Visualização da rede no *uDig*

4.3 Resultados

4.3.1 Estatísticas

A partir do *OpenStreetMap* foi possível obter da zona do Porto e arredores um total de 40043 arcos separados pelos pontos de cruzamento com outras e um total de 29207 cruzamentos como referido na Tabela 4.1.

Depois de feita a filtragem para obter somente o que é ciclável chegou-se ao número de 28304 formas e 21605 cruzamentos.

Tabela 4.1: Informação obtida do *OpenStreetMap*

	Antes da filtragem	Depois da filtragem
Número de formas	40043	28304
Número de cruzamentos	29207	21605

Implementação e resultados

Para efectuar testes foram utilizados 112 ficheiros GPX com os respectivos *tracks* registados entre os anos 2004 e 2011 na zona do Porto e arredores. Estes ficheiros totalizam um total de 119664 pontos GPS.

Com a utilização destes ficheiros foi possível detectar 2127 novas vias, considerando que uma via é o percurso entre dois cruzamentos ou entroncamento com outras vias, aos 28304 anteriormente importadas do *OpenStreetMap* que devido aos novos cruzamentos detectados entre as vias novas e as iniciais totalizam 28910 vias.

4.3.2 Rede Ciclável inicial

Os dados obtidos a partir do *OpenStreetMap* têm informação que não é necessária para criar a rede ciclável com é observável na Figura 4.3. Nesta situação é visível que existem por exemplo auto-estradas onde não é permitido a utilização de bicicletas, bem como delimitações de edifícios, fronteiras, rios vedações entre outros.

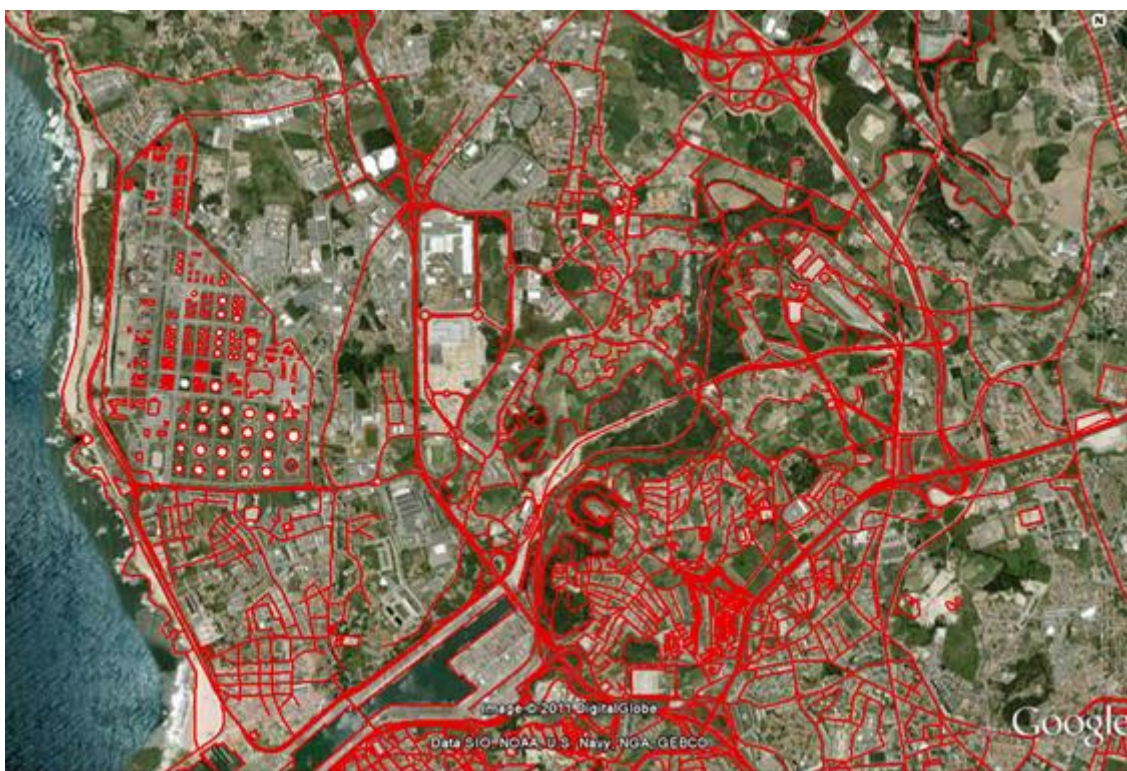


Figura 4.3: Rede obtida directamente do *OpenStreetMap*

Retirada toda a informação irrelevante para a utilização da bicicleta, obtém-se assim a rede ciclável presente na Figura 4.4.



Figura 4.4: Rede filtrada obtida a partir do *OpenStreetMap*

4.3.3 Detecção de novas vias

Como é possível observar em Figura 4.5 e Figura 4.6, foram detectadas novas vias a partir de pontos GPS (a amarelo na figura). As novas vias são representadas pela cor azul.

Caso a na mesma via passem mais de um track é considerado só o que foi inserido pela primeira vez como é visível na Figura 4.7.

Implementação e resultados



Figura 4.5: Detecção de novas vias - exemplo 1



Figura 4.6: Detecção de novas vias - exemplo 2



Figura 4.7: Detecção de novas vias a partir de vários *tracks* GPS

4.3.4 Roteamento



Figura 4.8: Roteamento tendo em conta o caminho mais curto

Implementação e resultados

O roteamento permite obter o caminho mais curto entre dois pontos do mapa. Assim na Figura 4.8 é apresentado o resultado de um percurso mais curto entre dois pontos seleccionados. Na imagem está representado a vermelho a rede que é considerada e a preto o percurso que foi calculado pela aplicação.



Figura 4.9: Caminho mais curto

Na Figura 4.9 onde representa um caminho para um utilizador que pretende um caminho com menor comprimento entre dois pontos do mapa sem qualquer tipo de restrição, em comparação com a Figura 4.10 para um utilizador que não deseja um caminho mais curto mas sim um caminho onde não se tenha de esforçar muito devido às diferenças de altitude mesmo que o comprimento do percurso seja maior.

Implementação e resultados



Figura 4.10: Caminho com menor esforço

Capítulo 5

Conclusões e Trabalho Futuro

5.1 Conclusões

A análise do estado da arte permitiu adquirir competências e criar uma metodologia que permitisse a criação de uma rede ciclável, a sua classificação e o roteamento nessa mesma rede.

Foi possível verificar que apesar da rede de estradas existente no *OpenStreetMap* estar bastante completa ainda existe algumas vias e trilhos em falta e que estes podem ser actualizados, a partir de dados dos *tracks* GPS submetidos pelos utilizadores, de forma satisfatória como foi possível verificar pelos resultados obtidos.

Os vários tipos de classificação aplicados à rede permitiram fazer um roteamento mais diversificado podendo-se assim controlar os parâmetros de forma a se obterem caminhos mais específicos para as necessidades de um utilizador de bicicletas.

Os objectivos propostos para este projecto foram atingidos com resultados satisfatórios através da análise dos resultados obtidos, apesar não ter sido testado exaustivamente existindo sempre possibilidade de ser melhorado a nível de precisão e de performance.

5.2 Trabalho Futuro

Os dados obtidos a partir dos *tracks* GPS incluem informação como por exemplo situações onde a bicicleta se encontra parada, o erro do GPS é maior e começa a dispersar à volta do ponto em que se encontra parado, que trazem problemas na detecção de vias originando vias que não existem. Um melhoramento seria na parte da filtragem dos ficheiros GPX remover os pontos em que essa situação se verifica.

Na fase de roteamento é tido em conta vários tipos de classificação, futuramente podia ser usada outras classificações como os tipos de via, estado do pavimento, tipo de bicicleta

Conclusões e Trabalho Futuro

utilizado, entre outros. Seria necessário também fazer um estudo das várias formas de dar peso a cada um dos parâmetros do roteamento especificamente para as especificidades da bicicleta.

Por fim seria necessário integrar todo o sistema numa aplicação Web para poder ser utilizada pelos utilizadores, integrando todos as funcionalidades do sistema. Neste caso a aplicação Web seria introduzida no *Facebook*, aplicação essa realizada em outra dissertação nesta mesma faculdade, onde é possível detectar o perfil de cada utilizador da bicicleta através da classificação dos percursos realizados.

Referências

- [BKO96] David Bernstein, Alain Kornhauser. An Introduction to Map Matching for Personal Navigation Assistants *in* Transportation Research Part C-emerging Technologies, pp 91-108, 1996.
- [DIJ59] Dijkstra, E. W. “A note on two problems in connexion with graphs”. *Numerische Mathematik* 1, pp 269-271, 1959
- [FCR09a] Tiago R. M. Freitas, António Coelho, Rosaldo J. F. Rossetti. Correcting Routing Information through GPS Data Processing, *in* 13th International IEEE Conference on Intelligent Transportation Systems, pp. 706-711, 2009.
- [FCR09b] Tiago R. M. Freitas, António Coelho, Rosaldo J. F. Rossetti. Improving Digital Maps Through GPS Data Processing, *in* ITSC'09, pp.480-485, 2009
- [FHU00] Kenneth E. Foote, Donald J. Huebner. Database Concept, The Geographer's Craft Project. http://www.colorado.edu/geography/gcraft/notes/datacon/datacon_f.html, 2000 ultima consulta em Janeiro de 2011
- [GPX11] GPX: The GPS Exchange Format. <http://www.topografix.com/gpx.asp>, ultimo acesso Fevereiro de 2011

- [GRE02] Greenfeld, J.S. Matching GPS observations to locations on a digital map, *in* proceedings of the 81st Annual Meeting of the Transportation Research Board, Washington D.C.,2002.
- [KML11a] KML OGC, <http://www.opengeospatial.org/standards/kml/>, ultimo acesso Junho de 2011
- [KML11b] KML Reference – KML – Google Code, <http://code.google.com/intl/pt-PT/apis/kml/documentation/kmlreference.html>, ultimo acesso Junho de 2011
- [OSM11a] OpenStreetMap wiki page, http://wiki.openstreetmap.org/wiki/Main_Page, ultimo acesso Janeiro de 2011
- [OSM11b] OpenStreetMap API, <http://wiki.openstreetmap.org/wiki/API>, ultimo acesso Janeiro de 2011
- [PMI97] Paul S. Heckbert, Michael Garland. Survey of Polygonal Surface Simplification Algorithms, 1997.
- [POS11a] PostgreSQL web site, <http://www.postgresql.org/about/>, ultimo acesso Janeiro de 2011
- [POS11b] Postgis web site, <http://postgis.refrations.net/>, ultimo acesso Janeiro de 2011
- [QON07] Mohammed A. Quddus, Washington Y. Ochieng, Robert B. Noland. Current map-matching algorithms for transport applications: State-of-the art and future research directions, in Transportation Research - Part C 15 (2007), pp. 312–328, Maio 2007
- [SGL07] Michael John De Smith, Michael F. Goodchild, Paul Longley. Geospatial analysis: a comprehensive guide to principles, techniques and software tools, pp 344-?? Troubador Publishing Ltd, 2007
- [TUR06] Andrew Turner. Introduction to Neogeography. pp 5-6. O'Reilly Media, Inc., 2006
- [WBK00] Christopher E. White, David Bernstein, Alain L. Kornhauser. Some map matching algorithms for personal navigation assistants. In Transportation Research - Part C 8 (2000), pp. 91 – 108, 2000.

[YHA07] Albert K. W. Yeung, G. Brent Hall. Spatial database systems: design, implementation and project management. Springer, 2007.

Anexo A

Modelo Relacional

