



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CAMPUS CHAPECÓ
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

RAFFAEL CICILIANO ROSSI

ANÁLISE COMPARATIVA DE TÉCNICAS DE *MAP MATCHING*

**CHAPECÓ
2018**

RAFFAEL CICILIANO ROSSI

ANÁLISE COMPARATIVA DE TÉCNICAS DE *MAP MATCHING*

Trabalho de conclusão de curso de graduação
apresentado como requisito para obtenção do
grau de Bacharel em Ciência da Computação da
Universidade Federal da Fronteira Sul.

Orientador: Prof. Dr. Marco Aurélio Spohn

CHAPECÓ

2018

RAFFAEL CICILIANO ROSSI

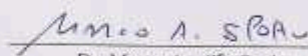
ANÁLISE COMPARATIVA DE TÉCNICA DE MAP MATCHING

Trabalho de conclusão de curso de graduação apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.

Orientador: Prof. Dr. Marco Aurélio Spohn

Este trabalho de conclusão de curso foi defendido e aprovado pela banca em: 09/07/2018

BANCA EXAMINADORA:



Dr. Marco Aurélio Spohn - UFFS



Dr. Claudio Pavão - UFFS



Dr. Emilio Wierges - UFFS

RESUMO

O algoritmo de map-matching (correspondência de mapa) consiste em identificar corretamente a posição da entidade no mapa. A partir de duas entradas, a posição da entidade obtida a partir de um sistema de geolocalização(GPS) e um mapa da região na qual ela se encontra, é possível determinar a posição física em que esta entidade se encontra em relação ao mapa. Para este fim, existem diversas abordagens, de uma simples busca pelo ponto mais próximo no mapa ao indicado pelo GPS, até a utilização de princípios matemáticos avançados para encontrar este ponto. Os desafios para correta identificação da posição física da entidade são a correta identificação do segmento (rua) em que a entidade se encontra, o tempo de resposta para esta identificação e certificar se este realmente é o ponto em que a entidade se encontra fisicamente. Neste contexto, este trabalho implementa uma técnica de map-matching a fim de validar a mesma com relação aos problemas acima mencionados.

Palavras-chave: Map-matching, Análise comparativa, Algoritmo Map-matching, map-matching pós processado, pos processing.

LISTA DE FIGURAS

Figura 3.1 – Elipse de erro [19]	15
Figura 3.2 – Y-junção [19]	17
Figura 4.1 – Angulo β [20]	21
Figura 4.2 – Distância perpendicular [20]	21
Figura 4.3 – Angulo α ponto ao segmento [20]	22
Figura 4.4 – Determinando a posição do veículo. [20]	24
Figura 5.1 – Interface aplicação desenvolvida para coleta dos dados	27
Figura 5.2 – Interface OpenStreetMap [1]	28
Figura 5.3 – Mapa da área da coleta.....	28
Figura 5.4 – Resultado map-matching. Azul é a saída do algoritmo e vermelho os traços reais.	31

LISTA DE TABELAS

Tabela 3.1 – Precisão das técnicas de map-matching. Tabela traduzida [19].	16
--	----

SUMÁRIO

1 INTRODUÇÃO	8
2 OBJETIVOS	9
2.1 Objetivos Geral	9
2.2 Justificativa	9
3 MAP-MATCHING	10
3.1 Técnicas de map-matching	10
3.1.1 Geométricas	10
3.1.2 Topológicas	11
3.1.3 Probabilístico	12
3.1.4 Avançadas	13
3.1.5 Performance das técnicas de map-matching existentes	15
3.2 Restrições e Limitações	15
3.2.1 Problemas com o início do map-matching	15
3.2.2 Problemas com valores limites	17
3.2.3 Problema das Y-junções	17
3.2.4 Considerando o desenho da via nos parâmetros no map-matching	17
3.2.5 Dados da altura dos sensores de navegação	17
3.2.6 Qualidade do mapa rodoviário	18
3.2.7 Técnicas usadas no processo de map-matching	18
3.2.8 Problemas de validação	18
3.2.9 Integridade (Nível de confiança)	18
4 TÉCNICA	20
4.1 Selecionando o segmento correto	20
4.1.1 Peso para o rumo do veículo e os segmentos	20
4.1.2 Pontuação para proximidade entre ponto e segmento	21
4.1.2.1 Pontuação para posição do ponto relativo ao segmento	22
4.1.3 Pontuação total de ponderação(TWS)	23
4.1.4 Determinando a posição física do veículo no segmento selecionado	23
4.1.5 Examinando se o veículo está no segmento atual	25
4.2 Identificação outlier	25
5 IMPLEMENTAÇÃO E RESULTADOS	26
5.1 Implementação aplicação móvel	26
5.2 Validação	27
5.3 Mapa	27
5.4 Traços Reais	28
5.5 Implementação	28
5.5.1 Parâmetros	29
5.5.2 Pontuações	29
5.5.3 Seleção do ponto dentro do segmento	29
5.5.4 Eliminação outliers	30
5.6 Resultados	30
6 CONCLUSÃO	32
REFERÊNCIAS	33

1 INTRODUÇÃO

A gama de sistemas de transporte inteligentes (ITS) - aplicações e serviços, como orientação de rota, gerenciamento de frota, taxas de circulação, acidentes e resposta de emergência, informações de chegada de ônibus e outros serviços baseados em localização (LBS) exigem informações de localização. Por exemplo, o sistema de navegação instalado em ônibus que pode determinar sua localização e enviar as informações de volta para um centro de controle, permite aos operadores prever a chegada dos veículos nas paradas de ônibus e, conseqüentemente, melhorar o nível de sistemas de transportes públicos de serviços. A precisão de posicionamento horizontal para essas aplicações está na gama de 1m a 40m (95%, isto é, ao nível 2R), com exigências relativamente elevadas sobre integridade, continuidade e a disponibilidade do sistema. Nos últimos anos, o Sistema de Posicionamento Global (GPS) estabeleceu-se como uma importante tecnologia de posicionamento por fornecer dados de localização para suas aplicações. Zito [28] fornece uma visão geral do uso de GPS como uma ferramenta para sistemas de veículo de estrada inteligentes. Deduzida Reckoning (comumente referido como DR 'Dead' Reckoning), consiste em sensores, um odômetro e um giroscópio rotineiramente utilizados para completar eventuais lacunas no posicionamento GPS. Esta informação é então usada em conjunto com dados da rede rodoviária para determinar a referência espacial da localização do veículo através de um processo conhecido como map-matching(MM) [19]. O presente trabalho implementa a técnica de map-matching apresentada no trabalho "A general map matching algorithm for transport telematics applications" [20], por Quddus, reproduzindo a técnica consumindo dados coletados (traços reais de mobilidade) na cidade de Chapecó/SC, Brasil, utilizando uma aplicação desenvolvida para este trabalho e um smarthphone.

2 OBJETIVOS

2.1 Objetivos Geral

Apresentar e discutir os resultados obtidos a partir da implementação da técnica de map-matching(MM) apresentada no trabalho "A general map matching algorithm for transport telematics applications" [20] num estudo de caso com dados coletados em Chapecó/SC, Brasil, a partir do uso de uma aplicação própria desenvolvida para este estudo e um smarthphone.

2.2 Justificativa

Através da replicação da técnica de map-matching apresentada por Quddus [20], verificar a precisão dos dados obtidos após a aplicação da técnica e a replicabilidade da mesma e outras aplicações.

3 MAP-MATCHING

A gama aplicações e serviços, como orientação de rota, gerenciamento de frotas, taxas de circulação, atendimento de emergência, informações de ônibus e outros baseados em localização exige informações de localização. Nos últimos anos, o Sistema Global de Posição (GPS) tem se estabelecido como uma importante tecnologia para prover dados de localização para as aplicações e serviços. Essa informação, juntamente com um mapa com boa resolução, é utilizada para indicar a posição, um processo chamado de map-matching(MM) [28]. O map-matching consiste em determinar a posição do veículo no mapa, como também, em alguns casos, na correção de uma leitura de um sensor de posição (GPS) [20]. Uma técnica pode ser desenvolvida de maneira genérica para aplicações ou de forma específica. O desempenho da técnica depende da aplicação que ela pretende atender, além do processo executado por ela [19].

3.1 Técnicas de map-matching

As técnicas de map-matching vão de simples buscas até algoritmos sofisticados, que usam teorias matemáticas. Na literatura, elas são divididas em 4 grupos [19]:

3.1.1 Geométricas

O map-matching geométrico faz uso de informação geométrica do mapa e considera como é a forma dos segmentos e não como eles se conectam um ao outro [7].

A técnica mais comum usa um algoritmo de busca simples, e a posição é associada ao nodo mais próximo no mapa. Essa forma é conhecida como map-matching ponto-a-ponto, sendo de simples e rápida implementação, no entanto, é muito sensível ao desenho das vias, o que acaba criando muitos problemas na prática [2].

Outra abordagem map-matching geométrica é a correspondência ponto-a-curva. Nesta abordagem, a posição fixa obtida a partir do sistema de navegação é combinada para a curva mais próxima na rede. Cada uma das curvas compreende segmentos de linha que são lineares por partes. A distância é calculada a partir da posição de correção de cada um dos segmentos de linha. O segmento de reta que dá a menor distância é selecionado como aquele em que o veículo está. Embora esta abordagem forneça melhores resultados do que a correspondência ponto-a-

ponto, ela possui várias deficiências que a tornam inadequada na prática. Por exemplo, uma saída pode ter resultados muito instáveis nas redes urbanas devido à alta densidade rodoviária. Além disso, a ligação mais próxima pode não ser sempre a ligação correta [24].

Taylor [23] propõem um novo método de map-matching referido como o algoritmo de filtro de redução de estrada (RRF), que utiliza GPS, altura (a partir dos dados da estrada) e diferencial de GPS virtual de correções (VDGPS). Devido ao uso de auxílio de altura, eles relatam que é necessário ao menos um satélite GPS para o cálculo da posição do veículo. O processo de combinação inicial deste algoritmo baseia-se na correspondência geométrica curva-a-curva, proposto por White [24], que é muito sensível a discrepâncias.

3.1.2 Topológicas

Em GIS, topologia refere-se a relação entre entidades (pontos, linhas e polígonos). Esta relação pode ser definida como adjacência (neste caso os polígonos), conectividade (no caso das linhas) ou contingência (no caso dos pontos no polígono). Com isso, o algoritmo faz uso da geometria dos segmentos e da conectividade e continuidade dos segmentos [20].

Greenfeld [7] analisa várias abordagens para resolver o problema do map-matching e propõe um algoritmo topológico ponderado. Este algoritmo baseia-se numa análise topológica de uma rede de estradas e visa a coordenação da informação sobre as posições observadas pelo utilizador.

Este algoritmo não considera qualquer informação sobre posição ou velocidade determinada a partir de GPS, sendo muito sensível à discrepâncias, uma vez que estas podem levar a um cálculo impreciso da posição do veículo [7].

Meng [13] também usa uma análise topológica da rede rodoviária para desenvolver um algoritmo simplificado, o qual fundamenta-se na correlação entre a trajetória do veículo e as características topológicas da estrada (no sentido do trânsito, a curvatura da estrada, e de ligação rodoviária). Uma série de testes condicionais é aplicada para eliminar segmentos de estrada que não preencham alguns limites predefinidos. Os limites são obtidos a partir da análise estatística dos dados de teste de campo. O algoritmo é implementado utilizando dados de navegação do GPS/DR e dados da rede rodoviária, incluindo informações sobre restrições de volta nos cruzamentos, a fim de melhorar o desempenho.

Contudo, o algoritmo não funciona bem nos cruzamentos em que a orientação das estradas não é semelhante. Nestas circunstâncias, o algoritmo alterna para um modo de pós-

processamento para identificar a ligação correta, tornando-se inadequado para aplicações em tempo real.

Por sua vez, uma decisão qualitativa pode se basear na velocidade do veículo, na direção do veículo a partir de um giroscópio, na distância percorrida pelo veículo, na qualidade do mapa digital e nos erros associados com os sensores de navegação para identificar o segmento correto em aplicações em tempo real para tais circunstâncias [13].

Quddus [20] desenvolveu um algoritmo map-matching topológico reforçado com base em vários critérios de similaridade entre a geometria da rede rodoviária e dados de navegação derivados.

Porém, buscando um algoritmo mais simples e rápido, mostra-se necessário aplicar também os critérios de similaridade desenvolvidos por Greenfeld [7]. Para melhorar o desempenho do algoritmo, o esquema de ponderação é aumentado através da introdução de critérios adicionais e outros parâmetros, incluindo a velocidade, a posição do veículo em relação aos segmentos possíveis e as informações de direção obtidas a partir da sequência de dados GPS ou o sistema GPS/DR integrado. Fatores de ponderação diferentes são usados para controlar a importância de cada um desses critérios para determinar o melhor procedimento map-matching [20].

3.1.3 Probabilístico

O algoritmo probabilístico requer a elipse ou o retângulo de erro, tendo como centro a posição obtida do sensor de navegação (GPS). Dentro dessa área, ao redor da posição, os segmentos existentes são avaliados em critérios de título, proximidade e conectividade. Com isso, o erro do sensor tem alta influência sobre a eficácia do algoritmo [27].

Ochieng [15] desenvolveu um algoritmo de correspondência de mapa probabilístico reforçada. Neste algoritmo, a região de erro elíptica só é construída quando o veículo se desloca através de uma junção e não há necessidade de se criar a região de erro quando o veículo se desloca ao longo de uma ligação. Isso faz com que o algoritmo fique rápido, pois não executa uma série de processos envolvidos na criação da região de erro e, por conseguinte, a identificação da ligação correta.

Este método é mais confiável, como, por exemplo, a construção de uma região de erro em cada época pode levar à identificação ligação incorreta se outras ligações estão perto do link em que o veículo está viajando. Ochieng [15] também desenvolveu um certo número de critérios baseados em estudos empíricos para detectar uma curva do veículo em uma junção, o

que ajuda a identificar, de forma eficaz, a comutação do veículo a partir de uma ligação a uma outra. Este algoritmo probabilístico aumentado também leva em conta a imprecisão do sensor de navegação, quando o veículo se desloca em baixa velocidade. Isso auxilia efetivamente o algoritmo na combinação correta dos cálculos de posição em baixa velocidade, especialmente em áreas urbanas, onde existem paradas frequentes. Além disso, uma estimativa ótima para a determinação da localização do veículo numa ligação é desenvolvida. Esta técnica leva em consideração diversas fontes de erro associadas com a sensor de navegação e a qualidade de dados da rede rodoviária.

3.1.4 Avançadas

Técnicas avançadas possuem relação com algoritmos que utilizam conceitos mais refinados, como Filtro de Kalman ou Filtro Estendido de Kalman ([12], [9], [22], [11], [14]), a Teoria de Dempster–Shafer's ([25], [6]), modelo de estado-espaco flexível e filtro de partículas ([8]), iteração de múltiplo modelo ([5]), modelo de lógica fuzzy ([10], [21], [18], [14]) e aplicação de inferência Bayesiana ([16]).

Kim desenvolveu um sistema de navegação integrado constituído de GPS, DR, e uma técnica de map-matching para suas aplicações. O estudo parte do princípio que vários serviços ITS requerem precisão de posicionamento horizontal 2-D de 5-10 m (95%). Eles tentam conseguir esta precisão através do uso eficiente dos mapas digitais de estradas.

Primeiramente, uma abordagem de correspondência ponto-a-curva simples é utilizado para identificar a ligação correta. Em seguida, uma projeção ortogonal da fixação de posição para a ligação é utilizada para obter a localização do veículo. Devido à projeção, o erro de transversais (isto é, o erro através da largura da estrada) é reduzido de forma significativa. No entanto, o erro ao longo da trajetória continua a ser uma questão fundamental. Um filtro de Kalman estendido (EKF) é então aplicado para reestimar a posição do veículo com o objetivo de minimizar o erro ao longo da trajetória.

As entradas para o EKF vêm de correções de posição GPS. O desempenho do filtro pode depender da qualidade dos dados da rede rodoviária espacial, especificamente como a curvatura da estrada está representada. Como afirmado anteriormente, o método de ponto-a-curva não é suficiente para selecionar a ligação correta especialmente em redes rodoviárias urbanas densas. Se a identificação do segmento é incorreto, então as entradas para o filtro de Kalman também serão imprecisas, o que inevitavelmente leva a mais erros de posicionamento.

O método pode ser melhorado usando uma técnica mais eficiente para a seleção da ligação correta, tendo em consideração a direção e velocidade do veículo, bem como uma análise topológica da rede de estradas [11].

Pyo [16] desenvolveu um algoritmo map-matching usando a técnica Hipótese Múltipla (MHT). A MHT, que utiliza medições a partir de uma região de validação, é reformulado como um problema único alvo para o desenvolvimento do método de map-matching. Pseudomedidas são geradas para todos os segmentos dentro da região de validação, conforme definido usando a elipse erros derivados das sensores de navegação (GPS / DR). Pseudomedidas (posição e direção) são definidas como os pontos projetados do GPS/posições DR nos segmentos.

A análise topológica da rede rodoviária (conectividade, orientação e parâmetros de projeto da estrada), juntamente com os pseudomedidas são usadas para derivar um conjunto de hipóteses e suas probabilidades para cada saída do sensor DR GPS. A principal desvantagem deste algoritmo map-matching é que ele não tem um método para o mapa de combinação inicial. O desempenho da correspondência subsequente depende em grande medida a correspondência inicial e, portanto, potenciais incompatibilidades são prováveis.

Quddus [17] desenvolveu um algoritmo map-matching de lógica fuzzy que tenta superar algumas das limitações dos algoritmos existentes. O algoritmo utiliza uma série de novas variáveis de entrada (sem custo adicional). Estes são: (1) a velocidade do veículo, (2) a conectividade entre as vias rodoviárias, (3) a qualidade da solução de posição, e (4) da posição de uma correção em relação a um segmento possível. Estas entradas são incorporadas nas regras difusas, de modo a melhorar o desempenho do algoritmo. Três conjuntos de regras difusas baseadas no conhecimento são formulados quando a solução de navegação vem do GPS autônomo ou GPS/DR.

O primeiro conjunto (seis regras) é para um processo de map-matching inicial (IMP), o segundo conjunto (treze regras) é para posterior map-matching em um link (SMP-Link), e o terceiro conjunto (quatro regras adicionais) é para posterior map-matching em uma junção (SMP-junction) [17].

Zhao [27] desenvolveu um total de oito regras, no caso de uma solução de navegação obtida a partir de um sensor de DR. Essas oito regras não representam completamente o GPS autônomo ou os cenários GPS/DR integrados [18].

3.1.5 Performance das técnicas de map-matching existentes

A tabela abaixo refere o desempenho das técnicas de map-matching que possuem método de verificação para precisão. Essa verificação foi feita através de caminhos percorridos utilizando as técnicas de map-matching e aferindo a precisão das posições de saída com a posição real.

3.2 Restrições e Limitações

Existem várias questões que impedem o aproveitamento máximo das técnicas.

3.2.1 Problemas com o início do map-matching

Algumas técnicas, para começar o processo de map-matching, selecionam uma elipse de segmentos ao redor do ponto indicado na leitura. Com isso, o erro da leitura influencia diretamente os segmentos possíveis, podendo, assim, resultar em uma elipse, a qual não possui nenhum segmento.

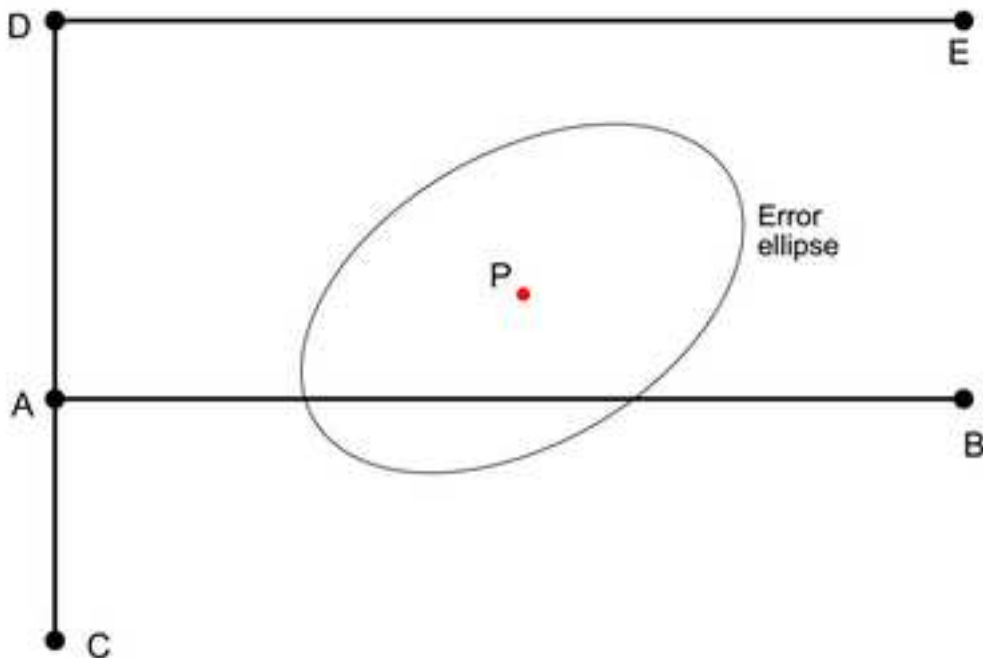


Figura 3.1: Elipse de erro [19]

Autor e Ano de Publicação	Sensor de Navegação	Ambiente de Testes	Identificação Correta de segmento (%)	Acurácia Horizontal (m)
Kim et al. (2000) [11]	GPS	Subúrbio	-	10.6(100%)
Kim e Kim (2001) [10]	GPS/DR	Urbano e Subúrbio	-	15m(100%)
White et al. (2000) [24]	GPS	Subúrbio	85.8	-
Pyo et al. (2001) [16]	GPS/DR	Urbano e Subúrbio	88.8	-
Taylor et al. (2001) [23]	GPS+Height	Subúrbio	-	11.6(95%)
Bouju et al. (2002) [3]	GPS	Subúrbio	91.7	-
Yang et al. (2003) [25]	GPS	Subúrbio	96	-
Quddus et al. (2003) [20]	GPS/DR Map Scale 1:2500	Urbana e Subúrbio	88.6	18.1(95%)
Syed e Cannon (2004) [21]	GPS/DR	Urbano e Subúrbio	92.8	-
Ochieng et al. (2004) [15]	GPS/DR Map Scale 1:2500	Urbano e Subúrbio	98.1	9.1(95%)
Quddus et al. (2006) [18]	GPS/DR Map Scale 1:2500	Urbano e Subúrbio	99.2	5.5(95%)

Tabela 3.1: Precisão das técnicas de map-matching. Tabela traduzida [19].

3.2.2 Problemas com valores limites

Os valores limites são comumente derivados empiricamente de uma série de observações de campo. E, com isso, uma abordagem mais analítica pode ser necessária para melhorar os parâmetros.

3.2.3 Problema das Y-junções

Algumas técnicas existentes podem não identificar o segmento de estrada correto perto ou em uma união Y.

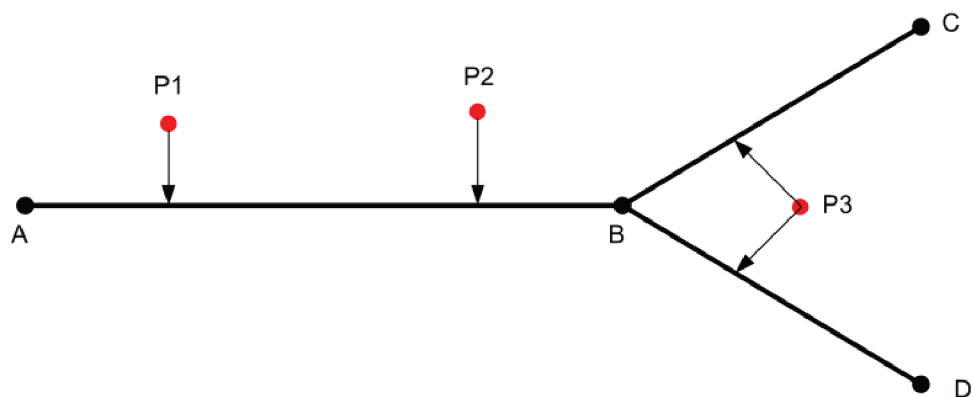


Figura 3.2: Y-junção [19]

Neste caso, o P3 pode ser atribuído a qualquer um dos segmentos e estar incorreto em relação à real posição do veículo.

3.2.4 Considerando o desenho da via nos parâmetros no map-matching

Os parâmetros do projeto da estrada, como restrições de volta em cruzamentos, classificação das estradas (uma mão, mão dupla), largura da faixa de rodagem, número de vias, viaduto e informações subterrâneas, normalmente não são incluídos como entradas para algoritmos de map-matching, contudo, a disponibilidade desses dados pode melhorar potencialmente o desempenho do algoritmo.

3.2.5 Dados da altura dos sensores de navegação

Algoritmos de map-matching não fazem uso de dados de altura, mas utilizando-se mapas 3D e contando com a precisão destes sensores, é possível identificar o segmento correto numa

estrada com múltiplos níveis.

3.2.6 Qualidade do mapa rodoviário

A qualidade do mapa usado afeta de maneira significativa a identificação correta da posição, sendo que um mapa muito preciso pode afetar o desempenho da técnica devido a quantidade de informações a serem processadas.

3.2.7 Técnicas usadas no processo de map-matching

O desempenho dos algoritmos de map-matching depende da técnica usada. Com base numa simples busca ou então numa abordagem altamente matemática, o tempo de resposta e a qualidade da resposta podem ser diretamente influenciados.

3.2.8 Problemas de validação

A validação do algoritmo de map-matching é essencial para obter estatísticas referentes ao seu desempenho, em termos de identificação do segmento correto. Uma referência precisa do veículo é necessária para a avaliação do seu desempenho. Poucos algoritmos possuem uma técnica que possibilite uma validação significativa.

3.2.9 Integridade (Nível de confiança)

A integridade de um algoritmo de map-matching reflete diretamente o nível de confiança que pode ser colocado na posição compensada do mapa. Medidas de integridade podem ser utilizadas para detectar uma falha no processo de map-matching. A capacidade de detecção pode ser utilizada para comunicar antecipadamente ao controlador que a solução de posição não deve ser usada para a navegação ou posicionamento, bem como para auxiliar os algoritmos de recuperação a partir do modo de falha. Quddus [17] descreve um método empírico simples para derivar a integridade de um algoritmo de correspondência de mapa. Yu [26] desenvolveu uma técnica para detectar soluções de map-matching errado e se recuperar de tais circunstâncias. Embora o desempenho desses métodos de integridade seja bom para a rota de teste utilizada, é essencial para investigar ainda mais o seu desempenho com outras rotas de teste, especialmente em áreas urbanas. Uma alternativa para a abordagem empírica seria considerar abordagens mais rigorosas, estatísticas semelhantes ao usado em sistemas de monitorização autónoma da integri-

dade como Receiver Autonomous Integrity Monitoring (RAIM), que baseia-se na consistência da verificação e medições de redundância (capacidade de detecção de discrepância) dentro de sensores/arquitetura de integração de dados.

4 TÉCNICA

De todas as técnicas avaliadas, foi escolhida a publicada por Quddus [20], "A general map matching algorithm for transport telematics applications", pois esta possui uma implementação genérica de multipropósito para o map-matching. Além de citar uma comparação direta com uma técnica anterior [7].

A técnica descrita no trabalho "A general map matching algorithm for transport telematics applications" [20], é uma técnica genérica topológica que utiliza cálculos de ângulo e distância entre pontos no mapa, GPS e as linhas formadas por estes para determinar os pesos para os possíveis segmentos e pontos candidatos ao map-matching. Estes pesos são utilizados como critério de decisão para a técnica e possuem diferentes importâncias dentro da técnica.

Os requisitos a nível de dados e entradas para esta técnica são: um mapa com resolução em pontos nas ruas, velocidade do veículo, heading (para onde está apontando o veículo), latitude e longitude.

Nas seções a seguir abordam apenas as equações e métodos utilizados dos existentes no trabalho desenvolvido por Quddus [20].

4.1 Selecionando o segmento correto

Definidas e carregadas as entradas, selecionam-se os segmentos (ruas) candidatos que estarão dentro do intervalo de erro, o qual é definido pelo responsável pela implementação da técnica. Escolhidos os possíveis segmentos, atribui-se pesos aos mesmos através de 4 equações.

4.1.1 Peso para o rumo do veículo e os segmentos

Essa equação determina a diferença dos ângulos entre o eixo norte e o rumo do veículo (heading) (β), além do ângulo entre os segmentos (links) e o eixo norte (β').

$$WS_H = A_h \cos(\Delta\beta')$$

Onde $\Delta\beta$:

$$\Delta\beta = \beta - \beta'$$

E $\Delta\beta'$ é calculado:

$$\Delta\beta' = \Delta\beta, \text{ para } -180 \leq \Delta\beta \leq 180$$

$$\Delta\beta' = 360 - \Delta\beta, \text{ para } \Delta\beta > 180$$

$$\Delta\beta' = 360 + \Delta\beta, \text{ para } \Delta\beta < -180$$

Quanto menor o valor para $\Delta\beta'$, maior a probabilidade de que este seja o segmento correto.

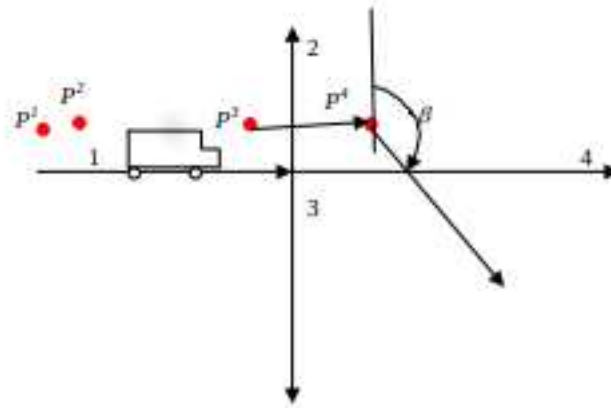


Figura 4.1: Ângulo β [20]

4.1.2 Pontuação para proximidade entre ponto e segmento

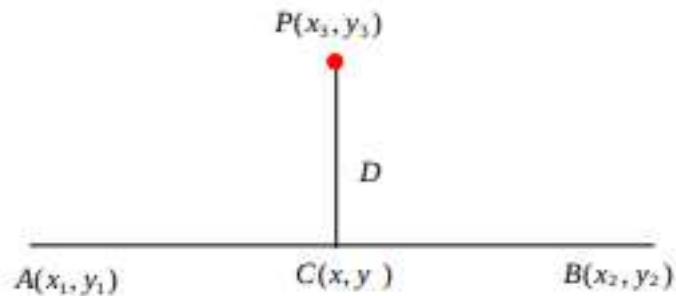


Figura 4.2: Distância perpendicular [20]

Existem duas pontuações para proximidade entre o ponto e o segmento: WS_{PD} e WS_{PI} . A primeira é baseada na distância (D) perpendicular entre o ponto e o segmento, calculada através de dois pontos (P_1, P_2), pertencentes ao segmento, e do ponto (P_3), fornecido pelo GPS, sendo que todos os pontos possuem o formato $P(x, y)$. Aplicando-se os pontos na equação da distância, temos:

$$D = \frac{x_3(y_1 - y_2) - y_3(x_1 - x_2) + (x_1y_2 - x_2y_1)}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}$$

Como a distância (D) diminui e a proximidade aumenta, a pontuação é determinada por:

$$WS_{PD} = A_p/D$$

Onde:

WS_{PD} é a pontuação da proximidade entre o ponto e o segmento.

$D (>0)$ é a distância entre o ponto e o segmento.

$A_p (>0)$ é o parâmetro que controla WS_{PD} . Qualquer valor pode ser atribuído a A_p para determinar WS_{PD} .

Quanto menor a distância, maior a pontuação. Se o ponto é próximo ao segmento, a chance de que este seja o segmento correspondente é maior.

A proximidade também pode ser determinada caso a linha formada por dois pontos consecutivos de GPS (isso é $P(x_{i-1}, y_{i-1})$ e $P(x_i, y_i)$) e o segmento se interceptam. A proximidade entre as duas linhas é medida pelo ângulo (θ) entre elas. Quando as linhas não se cruzam fisicamente, a pontuação é zero. A equação de pontuação é:

$$WS_{PI} = A_p \cos \theta$$

Onde:

WS_{PI} é a pontuação da intersecção entre as linhas, se ela existir.

A_p é previamente definido.

4.1.2.1 Pontuação para posição do ponto relativo ao segmento

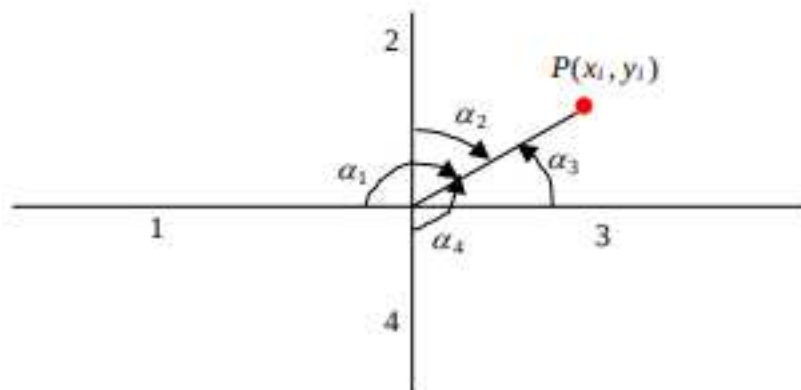


Figura 4.3: Ângulo α ponto ao segmento [20]

$P(x_i, y_i)$ é um ponto GPS, e alfa (α) é o ângulo formado entre o ponto e o segmento.

Quanto maior o ângulo, menor a probabilidade de que seja o segmento correto. A equação usada para a pontuação, utilizando alfa, é:

$$WS_{RP} = A_{RP} \cos \alpha$$

Onde:

WS_{RP} é a pontuação para a posição GPS relativa ao segmento.

α (≤ 180) é o ângulo que entre o ponto e o segmento.

A_{RP} (>0) é o peso para WS_{RP} e o valor é obtido a partir de uma equação apresentada.

4.1.3 Pontuação total de ponderação(TWS)

A pontuação total de ponderação(TWS) é obtida com a soma das demais pontuações, definindo TWS como:

$$TWS = WS_H + (WS_{PD} + WS_{PI}) + WS_{RP}$$

Ao selecionar valores para os parâmetros, o peso dos mesmos podem ser cuidadosamente controlados. Quddus [20] sugere que após suas análises, o parâmetro WS_H deve ter mais importância que WS_{RP} , e WS_{RP} deve ter mais importância que a proximidade ($WS_{PD} + WS_{PI}$). Os parâmetros são calculados usando:

$$A_H = aA_P$$

$$A_{RP} = bA_P$$

Conforme Quddus [20], a , b e A_P são valores decididos pelo responsável na implementação, onde $b > a > 1$. Após fixados a e b , o valor de A_P pode ser qualquer valor positivo, gerando bons resultados. Quddus [20], em seu estudo, sugere os valores para $a=5$, $A_P=10$ e $b=2$, não seguindo a recomendação de $b > a$. Concluiu, portanto, que o segmento com a maior pontuação(TWS) é selecionado como segmento correto.

4.1.4 Determinando a posição física do veículo no segmento selecionado

Os pontos do segmento dentro da área de erro e a velocidade(v) que o GPS fornece podem ser aplicados para calcular a posição do veículo. Utilizando o ponto anterior($P(E_i, N_i)$), calcula-se a posição do ponto atual($P(E_{i+1}, N_{i+1})$) no segmento, com a equação:

$$\Delta E_i = (v \cdot 1) \sin \theta$$

$$\Delta N_i = (v \cdot 1) \cos \theta$$

Onde θ é calculado com os vetores formados pelos pontos atual, anterior e um terceiro, que junto

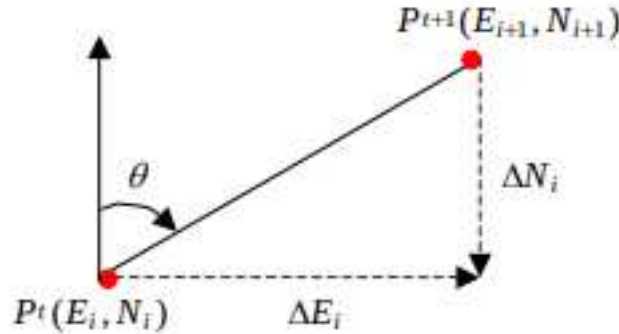


Figura 4.4: Determinando a posição do veículo. [20]

do ponto anterior, seja paralelo ao eixo norte. A posição atual do veículo pode ser calculada com a seguinte equação:

$$E_{i+1} = E_i + \Delta E_i$$

$$N_{i+1} = N_i + \Delta N_i$$

A outra forma de calcular a posição do veículo no segmento é utilizando a posição anterior, que já foi calculada, e a atual que se procura fazer o matching. Para isso é necessário encontrar os valores de PE e PN, que vão representar o ponto perpendicular ao ponto GPS no segmento.

$$PE = \frac{(x_2 - x_1)(E(x_2 - x_1) + N(y_2 - y_1)) + (y_2 - y_1)(x_1 y_2 - x_2 y_1)}{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$PN = \frac{(y_2 - y_1)(E(x_2 - x_1) + N(y_2 - y_1)) + (y_2 - y_1)(x_1 y_2 - x_2 y_1)}{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Onde E e N são as posições de longitude e latitude do ponto anterior, respectivamente.

Com essa equação encontra-se o ponto C(PE,PN) que se encontra na reta do segmento e perpendicular ao ponto do GPS. Com isso também se tem $P(E_{map}, N_{map})$ que é o ponto do mapa no segmento mais próximo ao ponto C. Para se chegar a posição do ponto GPS na reta utiliza-se a equação:

$$\hat{E} = \left(\frac{\sigma_{ES}^2}{\sigma_{ES}^2 + \sigma_{map}^2} \right) E_{map} + \left(\frac{\sigma_{ES}^2}{\sigma_{ES}^2 + \sigma_{map}^2} \right) PE$$

$$\hat{N} = \left(\frac{\sigma_{NS}^2}{\sigma_{NS}^2 + \sigma_{map}^2} \right) E_{map} + \left(\frac{\sigma_{NS}^2}{\sigma_{NS}^2 + \sigma_{map}^2} \right) PN$$

Onde:

\hat{E} é a longitude estimada do veículo

\hat{N} é a latitude estimada do veículo

σ_{map}^2 é a variância de erros associada com E_{map}

σ_{ES}^2 é a variância de erro associada com E

σ_{NS}^2 é a variância de erro associada com N

4.1.5 Examinando se o veículo está no segmento atual

Depois de atribuir a posição do veículo para os dois primeiros pontos GPS em um segmento, o algoritmo examinará se o veículo ainda está no segmento. Portanto, duas condições foram introduzidas para testar este critério:

- A diferença entre os "bearing" das duas linhas consecutivas do GPS forem maiores que 45°
- A diferença entre o heading dos dois últimos pontos GPS for maior que 45° e o alfa (α) maior que 90°

Se alguma das condições acima for verdadeira, o algoritmo assume que o veículo está se aproximando de uma interseção e inicializa o processo de map-matching. Caso contrário, o veículo ainda está no segmento anterior e o novo local é calculado, usando a equação da seção anterior.

4.2 Identificação outlier

Um outlier (ponto fora da curva), é um ponto GPS identificado se $\Delta\delta > 45^\circ$ (ou seja, se houver uma mudança repentina em dois "bearing" de pontos GPS consecutivos). Este outlier pode causar erros de matching, especialmente se $\Delta\delta$ for maior e mais próximo de outro segmento. Para evitar esses erros, um outlier é analisado e finalizado somente após os próximos pontos terem sido observados e analisados.

5 IMPLEMENTAÇÃO E RESULTADOS

Para implementação da técnica, foi realizada a coleta de traços reais de mobilidade. Por disponibilidade, utilizou-se um smartphone Android Lenovo Vibe K5, o qual possui acelerômetro, giroscópio e GPS (sensores requisitados pela técnica escolhida).

Foi desenvolvida uma aplicação própria para este trabalho, a fim de evitar que os dados colhidos fossem tratados por aplicações de terceiros e alterassem os resultados obtidos a partir da técnica de map-matching.

5.1 Implementação aplicação móvel

Na implementação, as linguagens utilizadas foram JavaScript com o framework React-native(<https://facebook.github.io/react-native/>) e Expo(<https://expo.io/>), todas disponíveis de maneira gratuita e com ampla documentação. Com isso, foi gerada uma aplicação classificada como híbrida, a qual permitiu um desenvolvimento mais rápido do que aquele que seria obtido utilizando uma aplicação nativa, além de ter facilitado o uso, uma vez que é necessário apenas publicar a aplicação via Expo. Essa publicação no site é mantida e disponibilizada pela própria Expo de forma gratuita, bastando o cadastro do usuário para acesso à publicação.

A aplicação desenvolvida se encontra no Github (<https://github.com/raffix/RealTraces>), onde é distribuída sob a licença MIT. O objetivo da aplicação é apenas a coleta dos dados do GPS do aparelho, contendo latitude, longitude, heading, e velocidade, com um intervalo pré definido pelo usuário da mesma, sendo o mínimo de 7 segundos e o máximo de 30 segundos, que, conforme revisão feita, são tempos utilizados em alguns dos trabalhos. Isso permite que o comportamento dos pontos não seja tão próximo e resultem num matching da mesma posição, nem muito distantes, onde acontecem muitos eventos entre uma coleta e outra.

Um exemplo do formato do dado obtido pela aplicação: `{"mocked":false, "timestamp":1530528756104, "coords":{"speed":0,"heading":0, "accuracy":16.809999465942383, "longitude":-52.616555, "altitude":0, "latitude":-27.0893808}}`

Após coletados, os dados foram enviados a um servidor php que os converte em um arquivo de texto, com seu conteúdo em formato JSON. Este formato é utilizado em aplicações de várias plataformas e é de fácil manipulação.



Figura 5.1: Interface aplicação desenvolvida para coleta dos dados

5.2 Validação

De acordo com a literatura, não existe uma forma para validar a saída de uma técnica de map-matching. Com isso, a solução adotada por alguns autores é fazer a validação da técnica na prática [18], ou, sabendo-se a origem e destino, calcular a distância da rota que é dada pela técnica analisada, para verificar se a mesma é plausível. Uma forma de fazer isso é através da distância de Fréchet [4]. Neste trabalho, foram utilizados dados coletados pelo autor, enquadrando-se no primeiro caso, sendo apresentado um mapa com a rota percorrida durante a coleta.

5.3 Mapa

A fonte dos mapas é o openstreetmap (<https://www.openstreetmap.org/>), uma base colaborativa e aberta, que oferece mapas de alta qualidade com várias informações, inclusive sentido da via (conforme imagem). O sistema é utilizado tanto para uso científico quanto comercial.

O mapa utilizado neste estudo foi exportado a partir do próprio navegador, selecionando a região onde as informações foram coletadas.

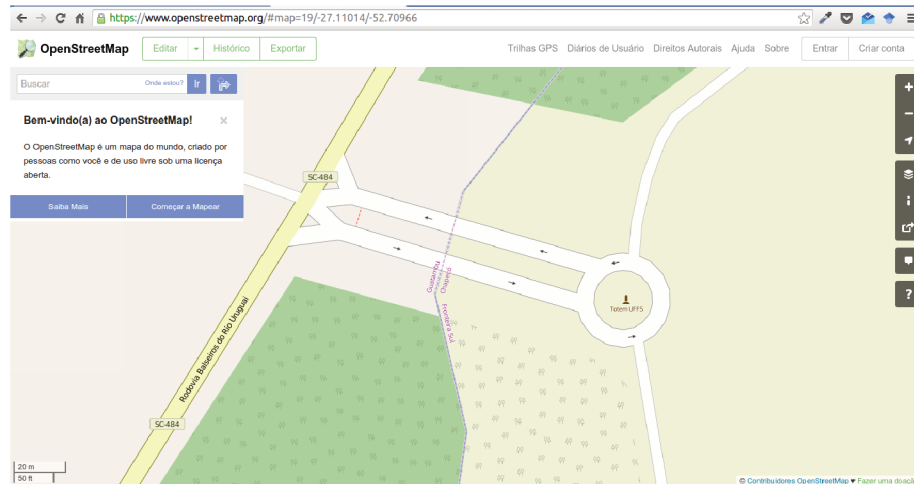


Figura 5.2: Interface OpenStreetMap [1]

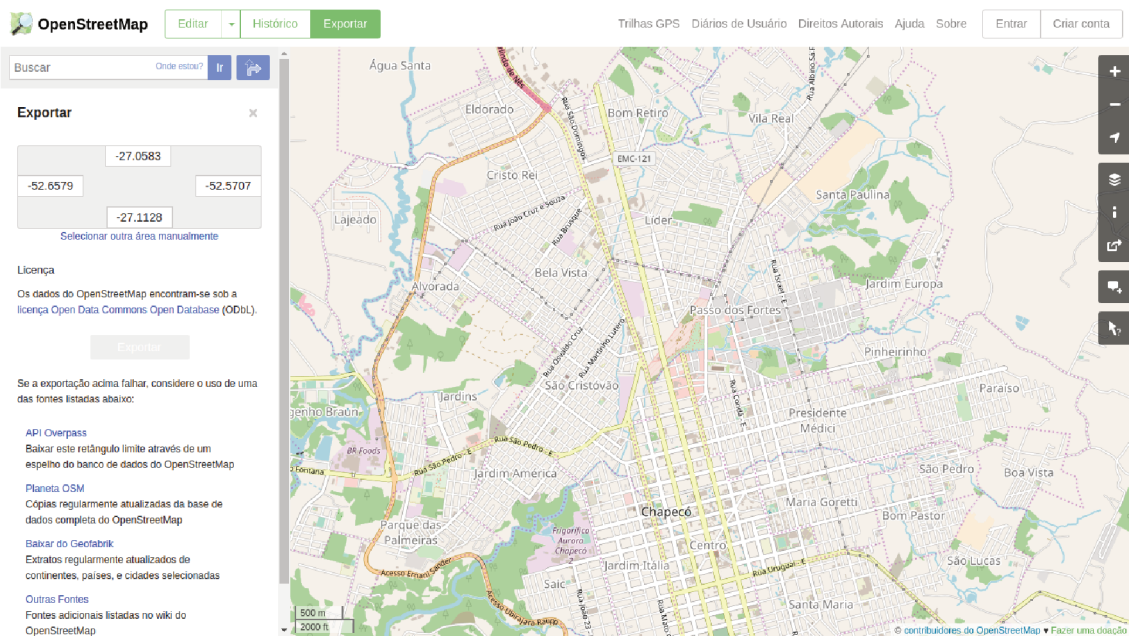


Figura 5.3: Mapa da área da coleta.

5.4 Traços Reais

Uma das entradas para o map-matching é a posição da entidade. Neste trabalho a entidade é um ciclista, que utilizando um smartphone coletou as informações com uma aplicação mobile desenvolvida para este trabalho.

5.5 Implementação

A implementação foi realizada em python3 devido a quantidade de bibliotecas disponíveis para os cálculos necessários, reduzindo assim o tempo de desenvolvimento e a complexi-

dade.

O algoritmo implementado seguiu os passos apresentados no trabalho de Quddus [20]. Inicialmente, não foi calculado o primeiro ponto do GPS, uma vez que, conforme apresentado anteriormente, é necessário calcular o outlier. Isso só é possível calculando o ponto seguinte.

Todos as implementações desenvolvidas no trabalho atual estão disponíveis em: <https://doi.org/10.5281/zenodo.1309042>

5.5.1 Parâmetros

Foi selecionada a distância de 0.001 como margem de erro na coordenada, o que representa, no mapa, a distância de aproximadamente uma quadra ou bloco. Esse valor foi aplicado na forma de um quadrado, seguindo a seguinte forma:

- $\text{minLatitude} = \text{Ponto.y} - 0.001$
- $\text{maxLatitude} = \text{Ponto.y} + 0.001$
- $\text{minLongitude} = \text{Ponto.x} - 0.001$
- $\text{maxLongitude} = \text{Ponto.x} + 0.001$

Onde ponto é a coordenada obtida do GPS. Os demais parâmetros a , b e A_p foram definidos com os valores 5, 2 e 10 respectivamente. Isso seguindo a referência feita por Quddus em seu trabalho.

5.5.2 Pontuações

Os cálculos de pesos e ângulos foram executados seguindo o que foi apresentado por Quddus [20], tendo êxito em todas as 4 equações para se chegar ao TWS (pontuação atribuída ao segmento, esta utilizada na escolha do segmento correto).

5.5.3 Seleção do ponto dentro do segmento

Quando se segue o algoritmo, com o segmento já escolhido após avaliado os segmentos possíveis, algumas equações estão incompletas. Inicialmente, quando avaliada a posição do veículo no segmento, existem duas formas de se calcular a mesma (ambas descritas acima). A primeira, que utiliza a velocidade, não obteve êxito, já que, por razões desconhecidas, a

grande maioria dos dados obtidos possui a velocidade atribuída como zero, o que inviabiliza a utilização da primeira equação. Já a segunda equação é dividida em duas partes, a primeira para encontrar as posições de latitude e longitude (apresentadas como easting e northing no trabalho de Quddus), e a segunda a fim de encontrar a posição do veículo no segmento selecionado. Esta, porém, não possui qualquer menção a alguns parâmetros: σ_{map}^2 , σ_{ES}^2 e σ_{NS}^2 . Com isso, inviabilizou o apontamento correto da posição do veículo no segmento.

Devido a esse fator, não foi possível utilizar por completo a técnica apresentada por Quddus [20], o que levou a um ajuste. Assim que se encontra o segmento com maior pontuação, atribui-se o ponto deste segmento mais próximo ao ponto do GPS como sendo a posição. Esta distância é calculada usando a distância euclidiana:

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Onde ponto GPS (x_1, y_1) e os pontos do segmento assumem as coordenadas (x_2, y_2) . Isso para todos os pontos pertencentes ao segmento.

5.5.4 Eliminação outliers

Para a eliminação de outlier não foi possível utilizar a forma apresentada por Quddus [20], já que o mesmo não demonstra como se calcula $\Delta\sigma$. Devido a este fator, nenhum modo de detecção e remoção de outlier foi implementado, já que conforme descrito pelo autor só é possível encontrar um outlier depois que ao menos um ponto tenha sua posição física no mapa indicado pelo algoritmo.

5.6 Resultados

O algoritmo teve um tempo de execução médio em 18 segundos, isso sem contar os 2 segundos de tempo de carregamento. Os testes foram executados em um notebook com processador athlonX2 1.6ghz, 3Gb de memória ram, com sistema operacional Deepin Linux 15.5 e utilizando python 3 e suas bibliotecas numpy, geopandas, shapely, math, json, sys, time e matplotlib.

Como observado, o traço real tem maior precisão que o mat-maching reproduzido. Os dados fornecidos no trabalho "A general map matching algorithm for transport telematics applications" por Quddus [20] são insuficientes para atingir os resultados apresentados por ele. Com

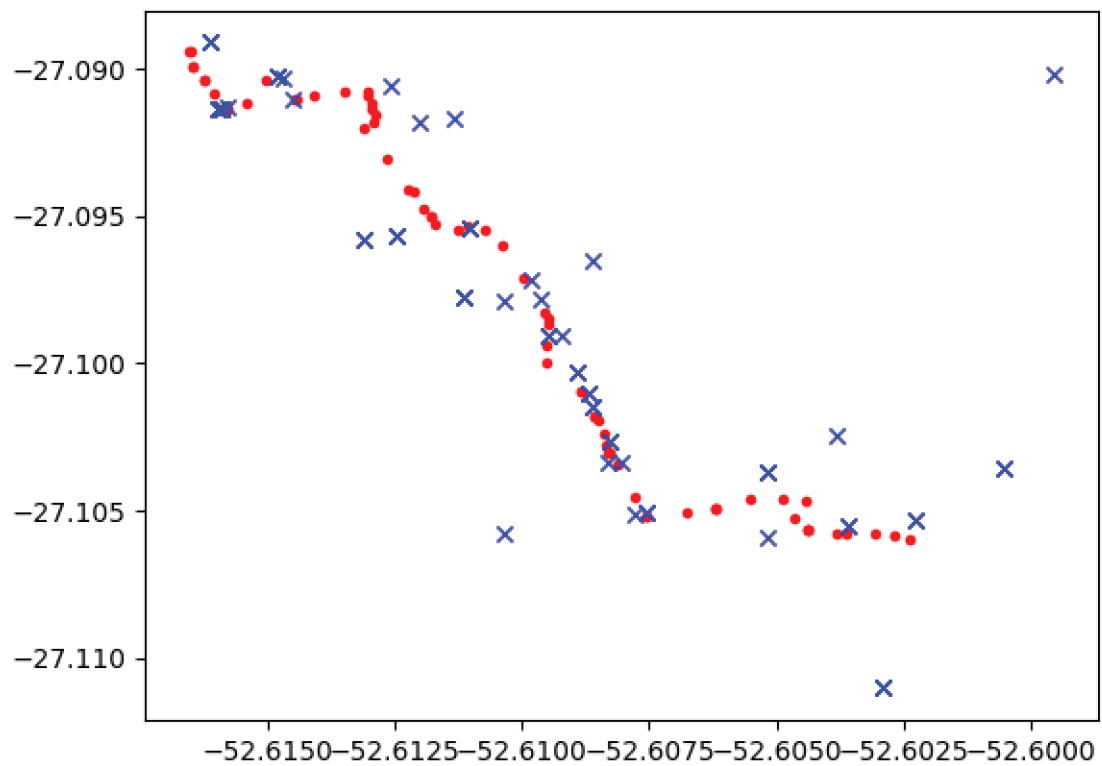


Figura 5.4: Resultado map-matching. Azul é a saída do algoritmo e vermelho os traços reais.

isso, conclui-se que a técnica, como foi possível implementar, é ineficaz, não cumprindo o seu propósito.

6 CONCLUSÃO

Este trabalho apresenta a replicação da técnica de map-matching abordada por Quddus [20] no trabalho "A general map matching algorithm for transport telematics applications", que não logrou êxito. Isso se deve pela ausência de informações relativas aos parâmetros e variáveis dos cálculos descritos para o funcionamento da técnica, concluindo que a mesma não é replicável a partir do descrito no trabalho supra citado.

REFERÊNCIAS

- [1] Open street map. www.openstreetmap.org. Accessed: 2016-06-24.
- [2] K. A. Bernstein, D. An introduction to map-matching for personal navigation assistants. <http://www.njtude.org/reports/mapmatchintro.pdf> Accessed June 19, 2002., 1996.
- [3] A. Bouju, A. Stockus, R. Bertrand, and P. Boursier. Location-based spatial data management in navigation systems. In *Intelligent Vehicle Symposium, 2002. IEEE*, volume 1, pages 172–177. IEEE, 2002.
- [4] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk. On map-matching vehicle tracking data. pages 853–864, 2005.
- [5] Y. Cui and S. S. Ge. Autonomous vehicle positioning with gps in urban canyon environments. *IEEE transactions on robotics and automation*, 19(1):15–25, 2003.
- [6] M. E. El Najjar and P. Bonnifait. A road-matching method for precise vehicle localization using belief theory and kalman filtering. *Autonomous Robots*, 19(2):173–191, 2005.
- [7] J. Greenfeld. Matching gps observations to locations on a digital map. In *proceedings of the 81st Annual Meeting of the Transportation Research Board, January, Washington D.C*, 2002.
- [8] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, 2002.
- [9] T. Itoh, S. Tsunoda, J. Tanaka, and K. Hirano. Navigation system with map-matching method. 1990.
- [10] S. Kim and J.-H. Kim. Adaptive fuzzy-network-based c-measure map-matching algorithm for car navigation system. *IEEE Transactions on industrial electronics*, 48(2):432–441, 2001.
- [11] W. Kim, G.-I. Jee, and J. Lee. Efficient use of digital road map in various positioning for its. In *Position Location and Navigation Symposium, IEEE 2000*, pages 170–176. IEEE, 2000.

- [12] E. J. Krakiwsky, C. B. Harris, and R. V. C. Wong. A kalman filter for integrating dead reckoning, map matching and gps positioning. In *Position Location and Navigation Symposium, 1988. Record. Navigation into the 21st Century. IEEE PLANS '88., IEEE*, pages 39–46, Nov 1988.
- [13] Y. Meng. Improved positioning of land vehicle in its using digital map and other accessory information. *PhD Thesis. Department of Land Surveying and Geoinformatics, Hong Kong Polytechnic University.*, 2006.
- [14] D. Obradovic, H. Lenz, and M. Schupfner. Fusion of map and sensor data in a modern car navigation system. *Journal of VLSI signal processing systems for signal, image and video technology*, 45(1-2):111–122, 2006.
- [15] Q. M. N. R. Ochieng, W.Y. Map-matching in complex urban road networks. *Brazilian Journal of Cartography (Revista Brasileira de Cartografia)*, 55(2):1–18, 2004.
- [16] J.-S. Pyo, D.-H. Shin, and T.-K. Sung. Development of a map matching method using the multiple hypothesis technique. In *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pages 23–27. IEEE, 2001.
- [17] M. A. Quddus, R. B. Noland, and W. Y. Ochieng. Effects of navigation sensors and digital map quality on performance of map-matching algorithms. In *Transportation Research Board 85th Annual Meeting*, number 06-0237, 2006.
- [18] M. A. Quddus, R. B. Noland, and W. Y. Ochieng. A high accuracy fuzzy logic based map matching algorithm for road transport. *Journal of Intelligent Transportation Systems*, 10(3):103–115, 2006.
- [19] M. A. Quddus, W. Y. Ochieng, and R. B. Noland. Current map-matching algorithms for transport applications: State-of-the art and future research directions. *Transportation Research Part C: Emerging Technologies*, 15(5):312 – 328, 2007.
- [20] M. A. Quddus, W. Y. Ochieng, L. Zhao, and R. B. Noland. A general map matching algorithm for transport telematics applications. *GPS Solutions*, 7(3):157–167, 2003.
- [21] S. Syed and M. E. Cannon. Fuzzy logic-based map matching algorithm for vehicle navigation system in urban canyons. In *ION National Technical Meeting, San Diego, CA*, volume 1, pages 26–28, 2004.

- [22] J. Takashi, M. Haseyama, and H. Kitajima. A map matching method with the innovation of the kalman filtering. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 79(11):1853–1855, 1996.
- [23] G. Taylor, G. Blewitt, D. Steup, S. Corbett, and A. Car. Road reduction filtering for gps-gis navigation. *Transactions in GIS*, 5(3):193–207, 2001.
- [24] C. E. White, D. Bernstein, and A. L. Kornhauser. Some map matching algorithms for personal navigation assistants. *Transportation research part c: emerging technologies*, 8(1):91–108, 2000.
- [25] D. Yang, B. Cai, and Y. Yuan. An improved map-matching algorithm used in vehicle navigation system. In *Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE*, volume 2, pages 1246–1250. IEEE, 2003.
- [26] M. Yu, W. Chen, Z. Li, and Y. Chen. Improvement on integrity and reliability of vehicle positioning by a new map matching method. In *Proceedings of the 17th International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS*, pages 2086–2094, 2004.
- [27] Y. Zhao. *Vehicle Location and Navigation System*. Artech House, Inc., MA., 1997.
- [28] R. Zito, G. D’Este, and M. Taylor. Global positioning systems in the time domain: How useful a tool for intelligent vehicle-highway systems? *Transportation Research Part C: Emerging Technologies*, 3(4):193 – 209, 1995.