



Miguel Ángel Cárdenas Rueda

Localização Híbrida para Um Veículo Autônomo em Escala Usando Fusão de Sensores

34/2014

CAMPINAS
2013



**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA**

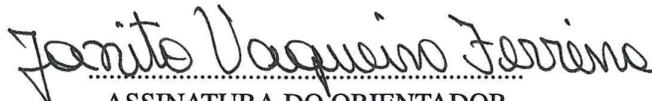
Miguel Ángel Cárdenas Rueda

**Localização híbrida
para um veículo autônomo em escala
usando fusão de sensores**

Orientador: Janito Vaqueiro Ferreira

Dissertação de Mestrado apresentada à Faculdade de Engenharia Mecânica da Universidade Estadual de Campinas, para a obtenção do título de Mestre(a) em Engenharia Mecânica, na Área de Mecânica dos Sólidos e Projeto Mecânico.

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL
DA DISSERTAÇÃO DEFENDIDA PELO ALUNO Miguel
Ángel Cárdenas Rueda, E ORIENTADO PELO PROF. DR
Janito Vaqueiro Ferreira


ASSINATURA DO ORIENTADOR

CAMPINAS
2013

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

C178L Cárdenas Rueda, Miguel Ángel
Localização híbrida para um veículo autônomo em escala usando fusão de sensores \ Miguel Ángel Cárdenas Rueda. – Campinas, SP: [s.n.], 2013.

Orientador: Janito Vaqueiro Ferreira.
Dissertação de Mestrado - Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Kalman, Filtragem de. 2. Sistema de Posicionamento Global. 3. Robôs móveis. 4. Visão por computador. 5. Veículos autônomos. I. Ferreira, Janito Vaqueiro, 1961-. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Localização híbrida para um veículo autônomo em escala usando fusão de sensores.

Título em Inglês: Hybrid Localization for an scale R/C car using sensor fusion
Palavras-chave em Inglês: Kalman filtering, Global Positioning System, Mobile robots, Computer Vision, autonomous vehicles.
Área de concentração: Mecânica dos Sólidos e Projeto Mecânico
Titulação: Mestre em Engenharia Mecânica
Banca Examinadora: Cairo Lucio Nascimento Junior, Ely Carneiro de Paiva
Data da defesa: 28-01-2013
Programa de Pós Graduação: Engenharia Mecânica

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE MECÂNICA COMPUTACIONAL

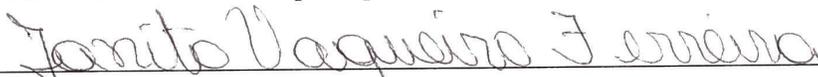
DISSERTAÇÃO DE MESTRADO ACADÊMICO

Localização híbrida para um veículo autônomo em escala usando fusão de sensores

Autor: Miguel Ángel Cárdenas Rueda

Orientador: Prof. Dr. Janito Vaqueiro Ferreira

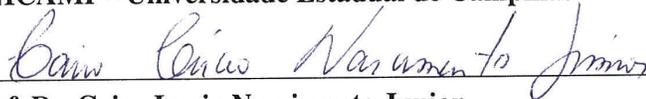
A Banca Examinadora composta pelos membros abaixo aprovou esta Dissertação:



Prof. Dr. Janito Vaqueiro Ferreira, Presidente
UNICAMP - Universidade Estadual de Campinas



Prof. Dr. Ely Carneiro de Paiva
UNICAMP - Universidade Estadual de Campinas



Prof. Dr. Cairo Lucio Nascimento Junior
ITA - Instituto Tecnológico de Aeronáutica

Campinas, 28 de Janeiro de 2013.

Dedicatória

Dedico este trabalho aos que de alguma forma acreditam em mim, Luis Miguel, Hilda Isabel e Diana Marcela.

Agradecimentos

Ao meu orientador pela confiança e paciência.

Aos Professores que ministraram as disciplinas que permitiram o desenvolvimento deste trabalho.

Aos meus colegas e amigos pelo apoio.

À minha família pelo amor e compreensão.

À Andre Linarth pela ajuda nos aspectos conceituais sobre odometria visual.

À Alejandro Weinstein pela ajuda na formulação do filtro de kalman no problema de localização.

À Giovani Bernardes pela ajuda no desenvolvimento do simulador de GPS.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo apoio financeiro.

Acreditar em nossa própria mentira é o primeiro passo para o estabelecimento de uma nova verdade.

Carlos Drummond de Andrade

Resumo

CARDENAS RUEDA, Miguel Angel. Localização híbrida para um veículo em escala usando fusão de sensores. 2012. Dissertação (Mestrado). Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas.

A localização tem sido definido como um dos problemas-chave da navegação autônoma. Este problema levanta a questão em relação ao fato de determinar a posição e orientação em qualquer instante durante uma trajetória.

Um dos métodos mais usados para determinar a localização de um robô movel é a odometria; embora não garanta informação precisa da posição e orientação, esse método é a base dos métodos de localização relativa. Também existem métodos de localização absoluta, que em relação a um referencial fixo provêm informação mais precisa sobre o estado de um robô, como, por exemplo, o GPS. Mas, quando erros de diferentes naturezas ocorrem, e como consequência introduzem ruído em qualquer sistema de localização, é necessário ter informação redundante para estabelecer uma informação mais precisa.

Esta dissertação de mestrado propõe um sistema de localização para um veículo em escala baseado em sistemas de odometria visual e odometria clássica a fim de fornecer uma estimativa robusta de posição e orientação através de um sistema de posicionamento global simulado.

Palavras-chave: Filtragem de Kalman, Sistema de Posicionamento Global, Robôs moveis, Visão por computador, Veículos autônomos.

Abstract

CARDENAS RUEDA, Miguel Angel. Hybrid Localization for an scale R/C car using sensor fusion. Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas.

Localization has been defined as one of the key problems of autonomous navigation. This issue rises the question about the fact of determining the position and orientation at any time while tracking a trajectory.

One of the most used methods to determine the localization of a mobile robot is odometry; however, despite the fact that can't provide accurate results for getting the position and orientation, this method is the basis of the relative localization methods. Also, there are methods for absolute localization which provide information more accurately about the state of the robot, such as the commonly used GPS. But when errors of different nature occur, and as a consequence, introduce noise at any localization system, it is necessary to have redundant information to establish a more accurate information.

This master thesis proposes a tracking system for a scale indoor vehicle based on classic and visual odometry in order to provide a robust estimation of position and orientation relative through a simulated global positioning system.

Keywords: Kalman filtering, Global Positioning System, Mobile robots, Computer Vision, autonomous vehicles.

Lista de Ilustrações

1.1	Aspirador de pó Roomba	4
1.2	Projeto Mars Rover	4
1.3	Descrição geral da navegação autônoma	5
1.4	Arquitetura da proposta	7
1.5	Plataforma: Veículo em escala	8
2.1	Veículo autônomo da UniBW	12
2.2	Resumo do processamento usado no projeto PROMETHEUS.	13
2.3	Projeto ARGO.	14
2.4	Veículo autônomo <i>Stanley</i>	15
2.5	Imagens da CarOLO-Cup	16
2.6	Principais etapas no processo de localização	17
2.7	Diagrama em blocos do V-SLAM	20
3.1	Modelo de Ackermann para veículo de 4 rodas	24
3.2	Modelo cinemático da bicicleta	25
3.3	Deslocamento entre duas amostras $k \rightarrow k + 1$	28
3.4	Deslocamento do veículo numa manobra de esterçamento	30
3.5	Evolução de incerteza para o modelo de veículo	35
3.6	Sensores embarcados no veículo	36
3.7	Imagem das Réguas	37
3.8	Canais de Onda	38
3.9	Mecanismo de esterçamento	38
3.10	Câmera RGB-D: Kinect	39
3.11	Modelo Pinhole	40
3.12	Modelo Geometrico da câmera de RGB-D	42
4.1	Esquema geral da odometria visual	46
4.2	Calibração de profundidade	48
4.3	Padrão de calibração	49
4.4	Imagens geradas com o <i>Imagens</i>	49
4.5	RANSAC para ajustar uma linha	51
4.6	Corrêspndencias com SURF	52

4.7	Correspondências Robustas com Ransac	53
4.8	Odometria de veículo em escala	59
4.9	Odometria <i>Dead Reckoning</i>	62
5.1	Modelo da solução de localização	67
5.2	Bloco de odometria convencional	68
5.3	Trajatória de referência odometria convencional	69
5.4	Resultado odometria convencional: primeira experiencia.	70
5.5	Erro e limite de confiança $\pm 3\sigma$	71
5.6	Erro acumulativo odometria convencional: primeira experiencia	72
5.7	Trajatória de referência odometria convencional	73
5.8	Resultado odometria convencional: segunda experiencia.	74
5.9	Erro e limite de confiança $\pm 3\sigma$	75
5.10	Erro acumulativo odometria convencional: segunda experiencia	75
5.11	Bloco de odometria visual	76
5.12	Posição aproximada da câmera na trajetória de teste	77
5.13	Trajatória observada pelo simulador do GPS.	77
5.14	Sequência de imagens da câmera no percurso.	78
5.15	Sequência de imagens da câmera no percurso.	79
5.16	Esquema da odometria visual	80
5.17	Resposta odometria visual.	82
5.18	Erro acumulativo da odometria visual	83
5.19	Modelo de localização	84
5.20	Simulação de estimação para localização.	86
5.21	Erro e limite de confiança $\pm 3\sigma$	87
5.22	Bloco de fusão: GPS + D.R	88
5.23	Trajatória de teste.	88
5.24	Resultado odometria convencional: experimetno final.	89
5.25	Estimação de localização.	91
5.26	Erro e limite de confiança $\pm 3\sigma$	92
5.27	Bloco de fusão: GPS / D.R / V.O	93
5.28	Resposta odometria visual.	94
5.29	Resposta do sistema de localização final	97
5.30	Erro e limite de confiança $\pm 3\sigma$	97
A.1	Veículo em escala <i>Tamiya</i> modificado.	107
B.1	Arquitetura de <i>hardware</i> do veículo em escala	109

B.2	Imagens geradas com o <i>EAGLE PCB Software</i>	111
B.3	Esquema elétrico Encoder e Circuito Impresso	112
C.1	Resposta Encoder	113
C.2	Faixa em frequência do encoder	114
C.3	Resposta Encoder	114
C.4	Resposta da calibração do ENCODER nas rodas	115
C.5	Potenciômetro Rotativo	116
C.6	Histerese Potenciômetro	117
C.7	Curva de calibração sensor esterçamento	117
C.8	Resposta sensor esterçamento	118
D.1	Marcador de posicionamento local	119
D.2	Campo de visão da câmera	120
D.3	Esquema de calibração do simulador	120
D.4	Percurso observado pela câmera	122
D.5	Observações do sistema posicionamento local para odometria visual	123

Lista de Tabelas

3.1	Tabela Modelo Observação odometria	32
3.2	Tabela matrizes de covariância	34
4.1	Coeficiente função profundidade	48
4.2	Estados do Modelo de odometria	60
4.3	Estados do Modelo Localização	64
5.1	Caraterização do erro da odometria visual	96
A.1	Componentes e descrição do veículo em escala	108

Lista de Algoritmos

4.1.1 Algoritmo determinação do movimento	56
4.1.2 Algoritmo determinação do movimento robusto com RANSAC	57
4.2.1 Algoritmo Filtro de Kalman Fusão Encoder	62
4.3.1 Algoritmo Filtro de Kalman GPS	65
5.2.1 Algoritmo de reconstrução da trajetória usando Odometria Visual	81
5.3.1 Algoritmo de fusão Odometria visual + GPS + Odometria convencional	95
D.2.1 Algoritmo Localização Local usando visão computacional	121

Lista de Abreviaturas e Siglas

Matrizes e Vetores

- \dot{x} - Vetor de estados
- (x, y) - Sistema de referência
- $[R_q | t]$ - Matriz de transformação
- R_q - Matriz de rotação
- t - Vetor de translação
- P, Q, R - Matrizes de covariância

Letras Latinas

- D - Diâmetro exterior
- d - Diâmetro interior
- L - Distância/Comprimento entre os eixos
- W - Largura entre as rodas

Letras Gregas

- ω - Ângulo de esterçamento
- σ - Covariância
- ϵ - Erro

Siglas

- FEM** - Faculdade de Engenharia Mecânica
- DOF** - *Degrees of Freedom* (Graus de Liberdade)
- LMA** - Laboratório de Mobilidade Autônoma
- RANSAC** - *RANdom SAmples Consensus*
- EKF** - *Extended Kalman filter*

SUMÁRIO

Lista de Ilustrações	xvii
Lista de Tabelas	xxi
Lista de Algoritmos	xxiii
Lista de Abreviaturas e Siglas	xxv
SUMÁRIO	xxvii
1 INTRODUÇÃO	3
1.1 Motivação e Contextualização	3
1.2 Proposta do trabalho	6
1.2.1 Objetivos	6
1.2.2 Arquitetura	7
1.2.3 Plataforma	8
1.3 Organização da dissertação	9
2 REVISÃO BIBLIOGRÁFICA	11
2.1 Histórico evolutivo dos veículos autônomos	11
2.2 Estado da arte	15
2.2.1 Localização a partir de observações relativas (<i>dead-reckoning</i>)	18
Odometria Convencional	18
Odometria Visual	18
2.2.2 Localização a partir observações absolutas	20
Sistema de posicionamento global	20
2.2.3 Localização robusta	21
Filtro de Kalman	21
3 FUNDAMENTOS TEÓRICOS	23
3.1 Modelo do veículo	23
3.1.1 Modelo geométrico do veículo	23

3.1.2	Modelo de odometria do veículo	28
3.1.3	Modelo probabilístico	33
3.2	Sensores	36
3.2.1	Encoder óptico	36
3.2.2	Sensor ângulo de esterçamento	38
3.2.3	Câmera RGB-D: Kinect	39
3.3	Formulações matemáticas em Odometria Visual	40
3.3.1	Modelo matemático da câmera RGB-D	40
3.3.2	Extração de pontos de interesse	43
3.3.3	Estimação dos parâmetros de movimento	44
4	MÉTODOS DE LOCALIZAÇÃO	45
4.1	Odometria Visual	45
4.1.1	Calibração da Câmera RGB-D	46
	Função de profundidade	47
4.1.2	Correspondência Robusta de imagens usando RANSAC	51
4.1.3	Estimação de movimento	54
4.2	Odometria (<i>Dead-Reckoning</i>)	59
4.3	Localização	63
5	EXPERIMENTOS E RESULTADOS	67
5.1	Aquisição	67
5.2	Odometria	68
5.2.1	Odometria convencional (<i>Dead-Reckoning</i>)	68
5.2.2	Odometria convencional: primeira experiência	69
5.2.3	Odometria convencional: segunda experiência	72
5.2.4	Odometria Visual	76
	Detecção de características	78
	Correspondência de características	79
	Determinar transformação rígida	80
5.3	Localização	83
5.3.1	Modelo	83
5.3.2	Fusão de sensores: odometria convencional / emulador de GPS	87
	<i>Dead-reckoning</i>	88
	Localização	90

5.3.3	Fusão de sensores: odometria convencional / GPS(emulador) / odometria visual	92
	Localização: Odometria visual	93
	Localização: odometria convencional / GPS(emulador) / odometria visual	95
6	DISCUSSÃO	99
6.1	Conclusão	99
6.2	Perspectivas Futuras	99
	REFERÊNCIAS	101
	APÊNDICES	106
A	Plataforma: Veículo em escala	107
B	Plataforma: Hardware(<i>Arduino</i>)	109
B.1	Fonte Reguladora	109
B.2	Circuito Auxiliar Encoder	110
C	Plataforma: Sensores de odometria convencional (<i>Dead-Reckoning</i>)	113
C.1	Encoder óptico	113
C.2	Sensor do ângulo de esterçamento(<i>Steering</i>)	116
D	Plataforma: Sistema de posicionamento global (Emulador)	119
D.1	Calibração	119
D.2	Algoritmo	121

1 INTRODUÇÃO

A navegação autônoma é um dos principais paradigmas da robótica móvel e é definida como deslocamento entre dois lugares independente de intervenção humana. Levar à prática esse conceito é o objetivo do laboratório de mobilidade autônoma (LMA) da Faculdade de Engenharia Mecânica (FEM-UNICAMP), assim, este trabalho busca contribuir na implementação de uma solução. A navegação autônoma com sucesso envolve essencialmente a resolução de três problemas: o controle de movimento, a percepção e a localização.

Determinar a posição e orientação de um veículo num ambiente desconhecido é a hipótese de um dos problemas da navegação autônoma definido como localização. Esse problema tem sido uma área de pesquisa muito ativa na última década e ainda está em andamento. Apesar dos muitos anos de pesquisa, é considerado um problema "resolvido" ao nível teórico e conceitual como é detalhado no trabalho de (DURRANT-WHYTE E BAILEY, 2006). Devido à complexidade e robustez requerida atualmente, o problema de localização encontra-se num nível de descrição probabilístico, em que as abordagens mais comuns são feitas através do filtro estendido de Kalman ou através do filtro de partículas.

1.1 Motivação e Contextualização

A pergunta que se faz após décadas de constante evolução na área da robótica móvel é: como pode um veículo se mover sem supervisão e ainda cumprir certo tipo de tarefas em ambientes do mundo real? (SIEGWART E NOURBAKHS, 2004). A resposta para essa pergunta desencadeou um grande número de trabalhos afins como Desouza e Kak (2002), Souissi *e outros* (2011), Dai *e outros* (2007) e Uhlin e Johansson (1996). Estes trabalhos apresentaram avanços significativos em relação à pesquisa, ao desenvolvimento de novas tecnologias, a conceitos e a arquiteturas aplicáveis ao problema.

A contribuição da aplicação da robótica móvel nos últimos anos é refletida na melhora da qualidade, produtividade e competitividade em muitos campos (DUDEK E JENKIN, 2010). Atualmente o uso de robôs móveis não está somente voltado para a pesquisa científica, mas também à indústria, os escritórios e o uso doméstico. Em linhas gerais, o projeto desse tipo de robôs está

baseado em realizar tarefas que demandam um esforço muito grande ou que substitua o ser humano em situações de perigo. Atualmente tem-se robôs móveis autônomos em muitos espaços, que vão desde o uso doméstico, como é caso do aspirador de pó Roomba (figura 1.1) até o uso em pesquisa científica como os robôs exploradores de Marte (figura 1.2).

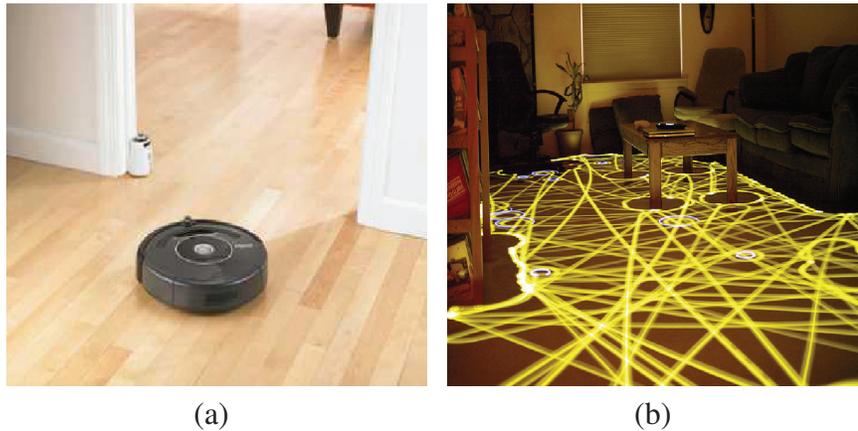


Figura 1.1: Aspirador de pó Roomba: (a) robô móvel; e (b) percurso do robô realçado.

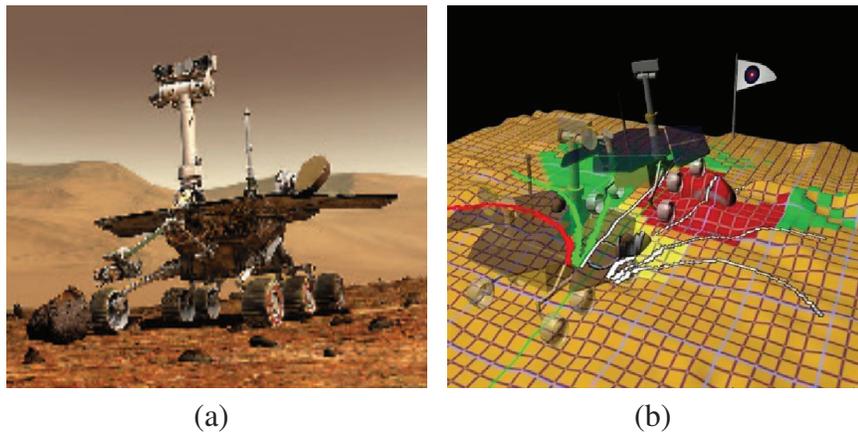


Figura 1.2: Robôs exploradores de Marte: (a) Veículo; e (b) Navegação.

A área de interesse desse trabalho encontra-se em veículos terrestres, principalmente influenciado por um dos maiores desafios já realizado, voltado para navegação autônoma de automóveis: o DARPA (*Defense Advanced Research Projects Agency*) challenge (BUEHLER e outros, 2007). Esse desafio ocorreu no deserto de Mojave entre os anos 2004 e 2007 e trouxe consigo muitos avanços em relação à solução do problema de localização como foram sumarizados na publicação de Rouff e Hinchey (2011). Esse trabalho permitiu, mesmo que não exista um padrão, estabelecer uma relação entre as diferentes camadas que fazem parte da navegação autônoma, do qual faz parte

a localização. Na figura 1.3 tem-se uma descrição geral da navegação autônoma distribuída em 3 camadas:

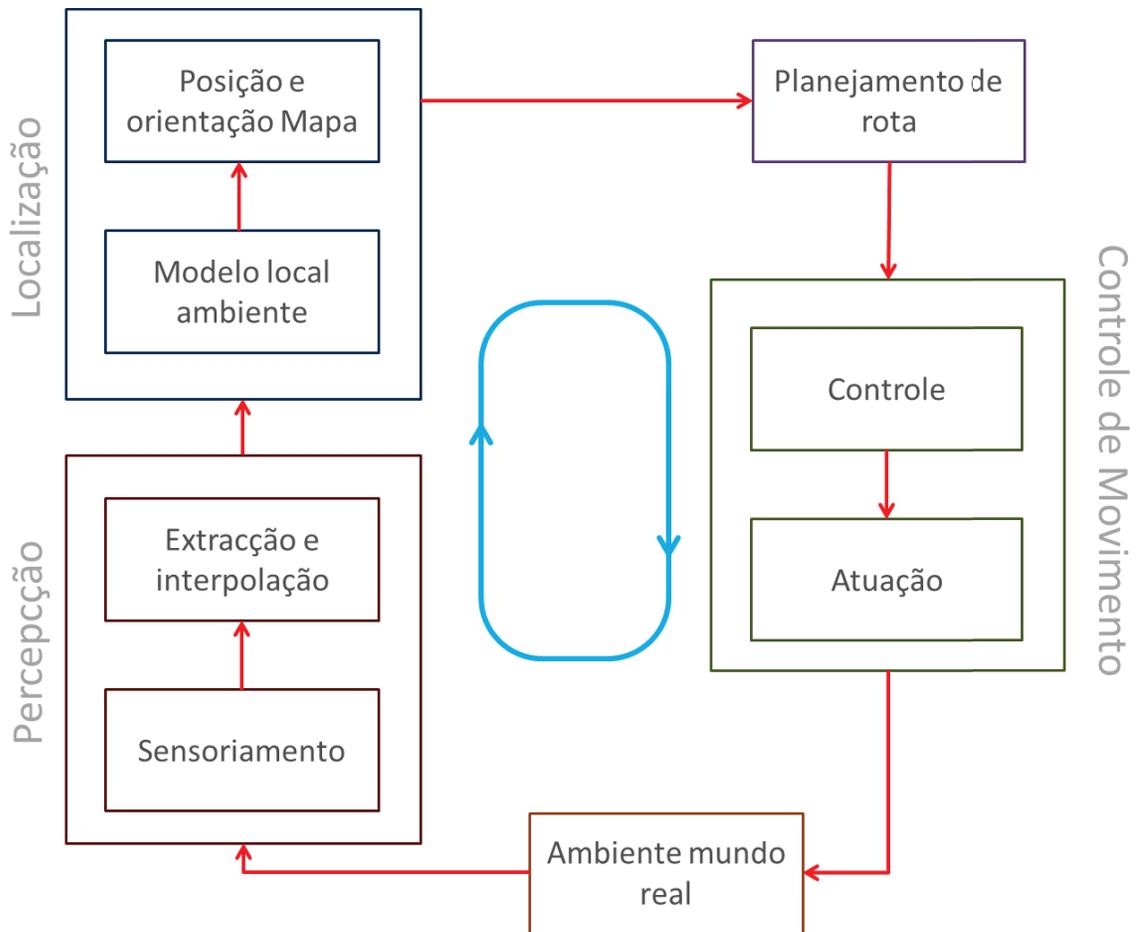


Figura 1.3: Descrição geral da navegação autônoma

- Controle de movimento: Nessa camada encontram-se os desafios mecânicos em relação aos mecanismos de locomoção, cinemática e dinâmica associados ao controle do robô.
- Percepção: Com o objetivo de obter de forma robusta a informação do estado do ambiente e do robô, é necessário o domínio sobre o processamento de sinais como um sistema geral, com subáreas como visão computacional.
- Localização: é uma das áreas mais exploradas hoje em dia, que possui o conjunto de três áreas computacionais que são: algoritmos, inteligência artificial e teoria de probabilidade.

No esquema apresentado na figura 1.3 pode-se observar que esse trabalho envolve tanto a camada de localização como da percepção.

1.2 Proposta do trabalho

A navegação autônoma com sucesso está baseada tanto no planejamento de trajetória, como na localização, assim, um investimento em função da robustez dos sistemas de medição é necessário a fim de projetar um veículo autônomo. A ideia que está por trás de um sistema de localização robusta é o uso de diferentes sensores. Juntos, esses sensores funcionam como um sistema de apoio caso algum dos sensores apresente um erro muito grande ou, no pior dos casos, falhe.

1.2.1 Objetivos

Para resolver o problema de localização foram levantados alguns objetivos.

Como objetivo principal tem-se:

- Desenvolver um sistema de localização para um veículo em escala.

Como objetivos secundários tem-se:

- Desenvolvimento de um sistema de odometria convencional usando informação redundante do deslocamento das rodas e do ângulo de esterçamento.
- Desenvolvimento de um sistema de odometria visual usando uma câmera RGB-D de baixo custo.
- Desenvolvimento de um sistema de localização global para um ambiente interior (*indoor*).
- Integrar os sistemas de odometria e localização usando fusão sensorial.

1.2.2 Arquitetura

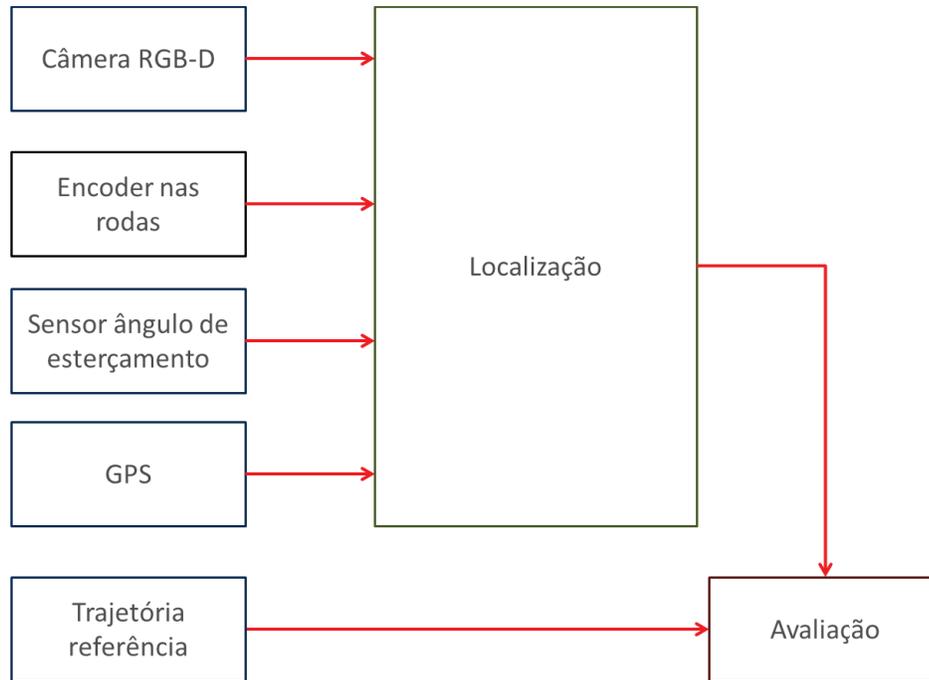


Figura 1.4: Arquitetura da proposta

Na figura 1.4 tem-se a arquitetura proposta, nela estão contidos um conjunto de sensores envolvidos na solução do problema de localização. Uma forma de localizar um veículo é usando a odometria convencional ou *dead-reckoning*; esse método busca localizar o veículo com a informação do deslocamento das rodas e do ângulo de esterçamento. Outro método bastante usado para localizar é o sistema de posicionamento global (GPS), que usa informação de satélites para esse propósito.

Um paradigma recentemente introduzido no problema de localização é a possibilidade de extrair os parâmetros de movimento de um veículo usando um sistema de visão embarcado, através de um algoritmo de estimação de pose conhecido como odometria visual (NISTER *e outros*, 2006). Essa nova técnica junto com a odometria convencional (CHOI *e outros*, 2011) configuram um sistema redundante de observação.

Com o objetivo de reduzir o erro de posição dos diferentes sensores, esse trabalho pretende fusionar a informação fornecida pela odometria convencional, odometria visual e o simulador de

posicionamento global. Além da implementação, que é considerada por Botelho *e outros* (2009) como uma série de subprocessos em que cada um depende dos resultados do antecessor, existem questões a respeito do desempenho e características *on-line* que não serão abordadas neste enfoque, assim, todo processamento neste trabalho é feito *off-line*.

1.2.3 Plataforma

No desenvolvimento deste trabalho utiliza-se um veículo de rádio controle da marca TAMIYA onde nele foram embarcados diferentes sensores.

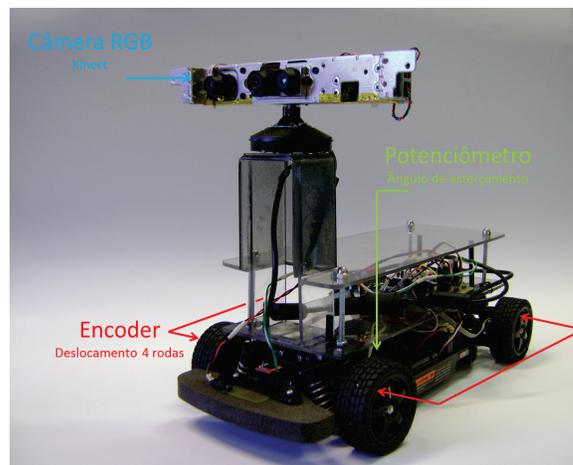


Figura 1.5: Plataforma: Veículo em escala

Os sensores embarcados no veículo são:

- Câmera RGB-D (Câmera de cor e de profundidade).
- Encoder ótico (Deslocamento das rodas).
- Potenciômetro de esterçamento (Ângulo de esterçamento).

Uma descrição detalhada da plataforma pode ser encontrada no apêndice A.

1.3 Organização da dissertação

A dissertação encontra-se dividida da seguinte forma.

No capítulo 2 é apresentada a revisão da literatura que fundamenta este trabalho, tendo como principais temas: Histórico evolutivo dos veículos autônomos, Odometria visual, Odometria *dead-reckoning* e Localização.

Logo a seguir, no capítulo 3, são apresentados os fundamentos teóricos e a modelagem matemática dos métodos de localização empregados.

Adiante, no capítulo 4, são abordadas as técnicas e algoritmos que foram desenvolvidos para a localização.

O capítulo 5 apresenta os resultados obtidos usando diferentes técnicas de localização a partir da utilização de algumas trajetórias de teste.

No capítulo 6 são apresentadas as conclusões deste trabalho, assim como também algumas propostas de trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

A navegação autônoma é definida como o deslocamento entre dois lugares independente de intervenção humana. Considera-se, então, que a navegação com sucesso envolve essencialmente a resolução de três problemas:

- **Localização:** A localização define-se como o processo em que um veículo autônomo constrói um mapa consistente e ao mesmo tempo usa-lhe para calcular a posição.
- **Estimação:** É o foco probabilístico, em que busca-se estimar, baseado em estados anteriores, a futura posição do veículo, afim de reduzir a incerteza no percurso.
- **Planejamento de trajetórias:** É a determinação de um caminho entre uma posição inicial do veículo e uma posição final, afim de atingir um alvo ou meta, de forma que o veículo não colida com nenhum obstáculo.

Considerando as diferentes abordagens que encontram-se em desenvolvimento, foi feita uma revisão bibliográfica baseada no contexto histórico das abordagens usando sensores de localização, para logo se reunirem trabalhos atuais no estado da arte.

2.1 Histórico evolutivo dos veículos autônomos

Há cerca de 30 anos vem-se desenvolvendo experimentos na robótica móvel, particularmente em veículos terrestres. A história de desenvolvimento tem início no final dos anos 1970, quando, em 1977 é fabricado no laboratório nacional de engenharia mecânica na cidade de Tsukuba, no Japão, o primeiro veículo autônomo. O objetivo desse experimento foi o de seguir linhas brancas na estrada com velocidades de até 30 km/h. Já em 1980 Ernst Dickmanns e seu grupo de pesquisa da Universität der Bundeswehr München (UniBW), na Alemanha, implantaram um veículo autônomo (Figura 2.1) usando um sistema de visão conhecido como *saccadic vision*. Esse sistema possuía estimadores tipo filtro de Kalman com um enfoque de paralelismo computacional, permitindo assim, alcançar velocidades de até 96 km/h (DICKMANNS, 1998).



Figura 2.1: Carro usado no projeto de Ernst Dickmanns (SICILIANO E KHATIB, 2008)

Durante os anos seguintes (1987 – 1995) foi criado o projeto pan-europeu chamado de PROMETHEUS (PROgramme for a European Traffic of Highest Efficiency and Unprecedented Safety). Esse projeto foi patrocinado pela Comissão Europeia e teve um orçamento de quase 1 bilhão de dólares. No estágio final do projeto conseguiu-se dirigir 1000 *km* sob condições de trânsito pesado com velocidades de até 130 *km/h*. Nesse experimento todos os comandos de controle do veículo (velocidade, direção e frenagem) estavam baseados no processamento, em tempo real, da sequência de imagens. Na figura 2.2 tem-se um resumo dos algoritmos usados no processamento das imagens (HOFFLINGER *e outros*, 1990).

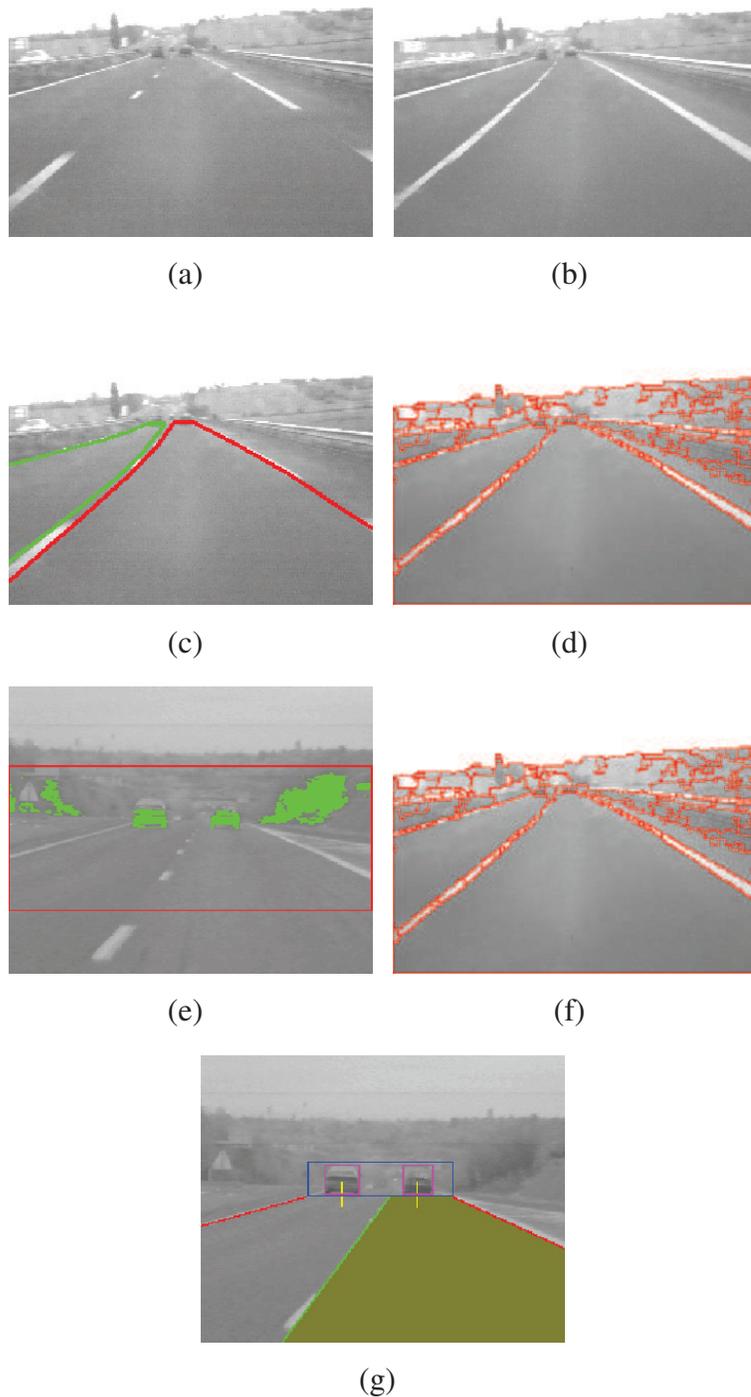


Figura 2.2: Processamento de imagens no Projeto PROMETHEUS: (a) original; (b) filtro morfológico dinâmico (TDF); (c) transformada watershed; (d) segmentação da estrada; (e) obstáculos simétricos; (f) obstáculos detectados; e, (g) sistema completo.

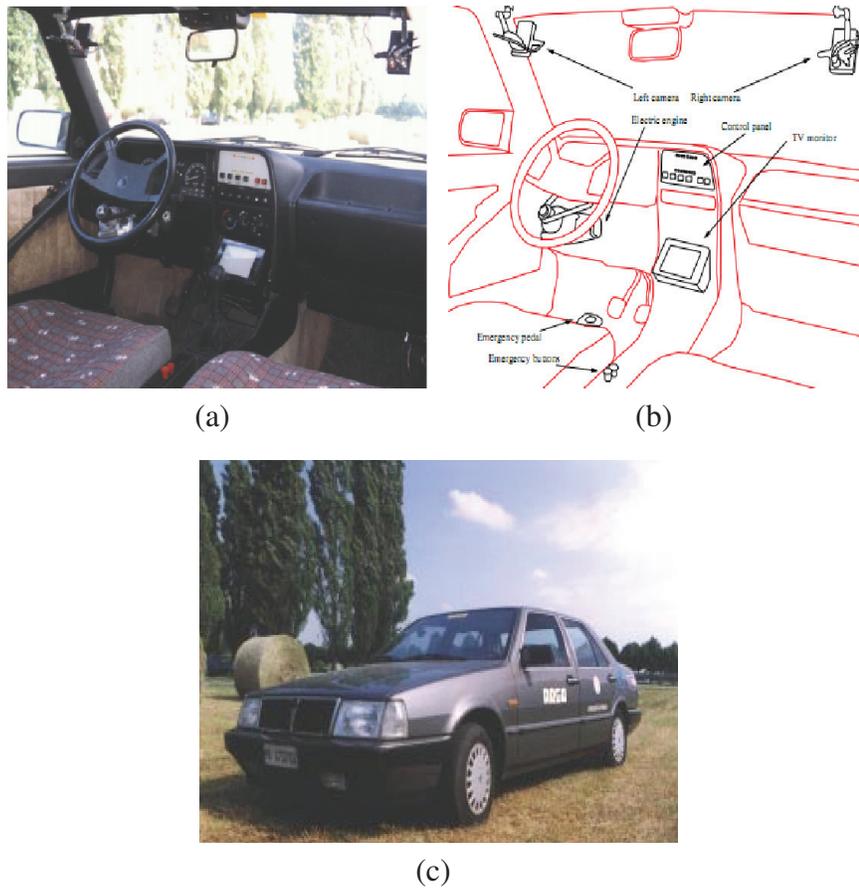


Figura 2.3: Projeto ARGO: (a) interior do veículo; (b) detalhe: Equipamento Interior; e, (c) veículo Lancia usado no ARGO.

Logo depois, em 1995, uma parceria entre a Mercedes-Benz e a UniBW conseguiu que o veículo de *dickmann* (DICKMANN *e outros*, 1993) dirigisse sozinho 95 % do percurso desde a cidade de München até Copenhague, completando assim, esse caminho de forma autônoma com velocidades de até 177 *km/h*.

Entre os anos 1996 e 2001, foi criado, na universidade de Parma (Itália), o projeto *ARGO* (BERTOZZI *e outros*, 1998). O objetivo desse projeto foi o de seguir linhas brancas na estrada usando visão estéreo (figura 2.3) com câmeras em preto e branco. O projeto foi concluído com sucesso, ao completar-se um percurso de 2000 Km conhecido como "MilleMiglia in Automatico" com velocidade média de 90 *km/h*.

Entre os anos 2004 e 2005 foi realizado o desafio "Grand Challenge", patrocinado pelo

DARPA (Defense Advanced Research Projects Agency). Este desafio consistia em percorrer um percurso de 240 km. Essa rota tinha vários obstáculos e caminhos não triviais. O ganhador do desafio foi o equipe *Stanford Racing Team* da Universidade de Stanford (THRUN e outros, 2006) ao conseguir completar a rota no menor tempo. Na figura 2.4 é apresentado o veículo autônomo *Stanley*, ganhador do desafio. Essa experiência fez ressurgir a pesquisa em veículos autônomos.

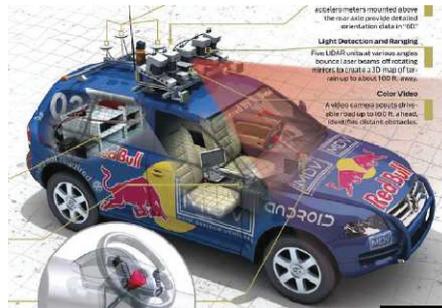


Figura 2.4: veículo autônomo *Stanley* ganhador do desafio DARPA Grand Challenge 2005

De forma similar em 2007 o DARPA patrocinou outro desafio baseado em veículos autônomos. O novo desafio foi direcionado para ambientes urbanos; o objetivo foi dirigir 96 km num tempo menor que 6 horas, sabendo de antemão que tinham-se outros veículos dirigidos por humanos ao longo do percurso, ou seja, sob condições de tráfego normal. Diferente da primeira versão, nessa vez foi-lhes permitido a participação de equipes que viessem de fora dos Estados Unidos. Duas equipes estrangeiras que tiveram uma participação importante foram o AnnieWay Team (Universität Karlsruhe) e o CarOLO (Technische Universität Braunschweig). O ganhador da versão urbana do desafio foi a equipe "Tartan Racing Team"; equipe que nasceu de uma parceria entre a Universidade Carnegie Mellon e General Motors Corporation.

Nos últimos anos (desde 2008) a iniciativa de CarOLO (figura 2.5), uma das equipes alemãs que participaram no desafio urbano do DARPA, vem fazendo competições anuais de veículos autônomos em escala. Esse tipo de aplicações têm um investimento relativamente baixo.

2.2 Estado da arte

Como dito anteriormente, resolver o problema da localização é a base da locomoção autônoma. Esse problema consiste em responder à pergunta "Onde eu estou?", amplamente abordado pelo trabalho de Borenstein e Koren (1995). A solução não é trivial, aliás, é considerada por alguns

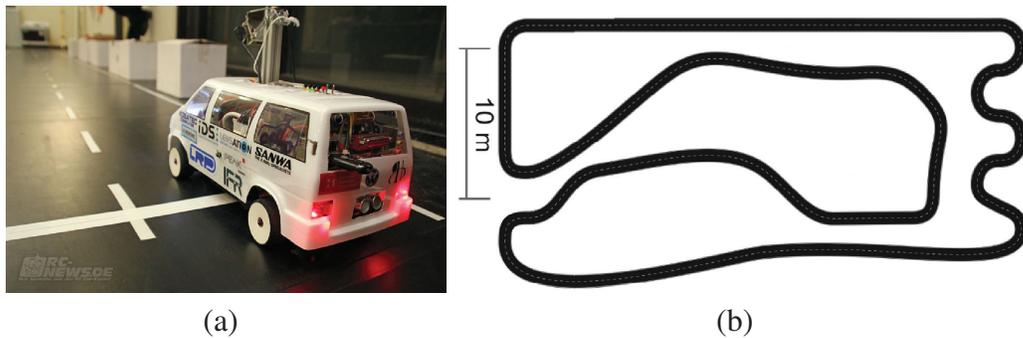


Figura 2.5: CarOLO-Cup: (a) Veículo em escala e (b) Percurso de Competição

autores da robótica móvel como o caminho verdadeiro na construção de um veículo autônomo.

A percepção de variáveis do ponto de vista do veículo tanto do ambiente como do estado do veículo, é chamada de observações. Elas provêm medidas de localização absolutas ou relativas. A Localização é fornecida a partir de observações relativas por sensores proprioceptivos, ou seja, que têm a informação do estado do veículo; sendo sumarizadas em:

- Odometria convencional (Encoders).
- Navegação Inercial (IMU).
- Odometria Visual (Câmeras).

A localização a partir de medidas absolutas é feita baseada em sensores exteroceptivos, que têm informação da localização em relação ao ambiente em qualquer momento; geralmente são:

- Detecção de características artificiais. (Landmarks)
- Detecção de características naturais. (GPS)

O processo típico de localização (Figura 2.6), como mostrado no trabalho de Blanco *e outros* (2007), está dividido em quatro passos. No primeiro passo (a) é feita a observação inicial, (b) com a locomoção do veículo é estimada a nova posição, (c) já na nova posição é feita uma observação, e (d) finalmente é feita a integração de previsões e observações.

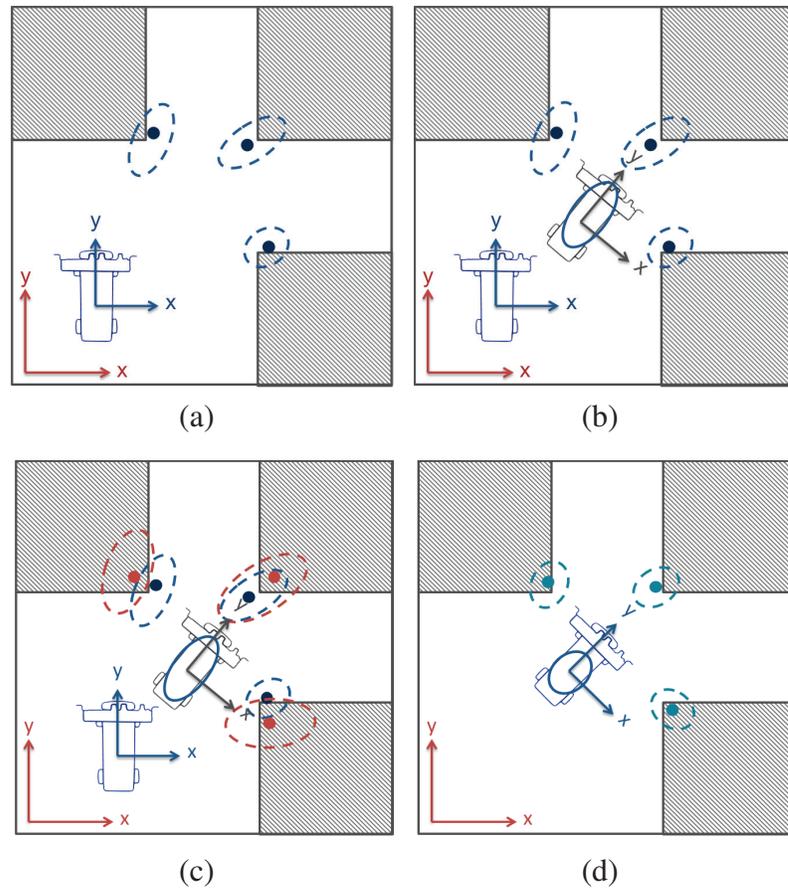


Figura 2.6: Etapas principais no processo de localização: (a) primeira observação; (b) estimação da posição; (c) novas observações; e, (d) fusão para obter pose do veículo.

2.2.1 Localização a partir de observações relativas (*dead-reckoning*)

O objetivo da odometria é determinar, com certa precisão, a posição e orientação do veículo baseadas na informação da odometria e dos sensores inerciais. Para um veículo autônomo, a odometria é um dos principais meios para medir o deslocamento. No entanto, eles experimentam erros, tanto sistemáticos como não sistemáticos, devidos à integração ao longo do tempo.

Odometria Convencional

Esse tipo de técnica refere-se à navegação baseada na medição de estados do veículo que precisam do conhecimento de um estado anterior desse. Geralmente, as observações são feitas por medições de velocidade ou posição e ângulo de esterçamento do veículo. A abordagem mais comum, como no trabalho de (SHEN *e outros*, 2011), é buscar uma solução através da fusão de diferentes sensores. Ali, por exemplo, foram usados um sistema de medição inercial de baixo custo e encoder nas rodas, onde chegou-se à conclusão de que nenhum dos dois tipos de sensores por si só conseguem localizar com precisão o veículo. No trabalho de (AZIZI E HOUSHANGI, 2004), por exemplo, a informação dos encoders é integrada com sistemas inerciais usando estimadores Bayesianos. A localização relativa a partir da odometria precisa de um modelo de erro, a fim de gerar um procedimento para estimar os parâmetros derivados da covariância dos sensores do robô. Um modelo de odometria convencional foi implementado por Doh *e outros* (2003) com as características mencionadas anteriormente, no entanto, é válido somente para veículos holonômicos.

Odometria Visual

A localização por câmera baseada em imagens é uma ideia que vem amadurecendo desde o trabalho de (HARTLEY E ZISSERMAN, 2004). Esse trabalho está baseado em modelos e algoritmos derivados da geometria projetiva, e geralmente é chamado de *Structure from Motion*. Com o objetivo de extrair a pose relativa usando imagens, geralmente usam-se câmeras calibradas e um processamento sequencial (SCARAMUZZA E SIEGWART, 2008). A pose relativa permite obter posição e orientação, e, de forma implícita, uma reconstrução no espaço euclidiano da cena enxergada (POLLEFEYS *e outros*, 2004). Essa abordagem baseada em visão tem-se tornado comum na navega-

ção autônoma; a revisão apresentada por Bonin-Font *e outros* (2008) mostra como esta abordagem representa atualmente uma solução econômica, desafiante e promissora em veículos autônomos. As estatísticas mostram um incremento na utilização de sistemas de visão na robótica móvel. O processo típico de localização baseado em visão, de acordo com Lacroix et. al (2008), está dividido em três passos:

1. *Detecção de características*: Nessa etapa, são determinados um conjunto de marcadores em que os dados lidos pela câmera, como por exemplo características do ambiente, são facilmente observáveis, e que permitam medir a posição relativa do robô.
2. *Correlação de Características*: As observações dos marcadores são uteis para calcular a pose do robô somente se forem feitas em diferentes posições. Porém, precisa-se de um algoritmo de correspondência robusto.
3. *Estimação de movimento*: Através das predições e observações, são estimadas as posições dos marcadores e do robô em relação ao sistema global de referência.

Apesar da odometria visual não resolver o problema de localização, ela faz parte da solução que é fornecida pela integração da observação de outros sensores associados com a localização relativa. Um dos trabalhos mais importantes na implementação de um sistema de localização e mapeamento foi apresentando por Karlsson *e outros* (2005); de forma geral, ele exibe as seguintes características:

1. Baixo custo tanto no sistema de visão como no sistema de odometria.
2. Os marcadores visuais são criados e reconhecidos no *front-end*.
3. Esquema de atualização para orientação e localização está baseado no filtro de kalman e filtro de partículas, permitindo uma implementação em tempo real.
4. A distribuição por camadas (figura 2.7) faz com que seja melhorada a recuperação da localização de eventos que geram perturbação como por exemplo a obstrução dos satélites que provêm a posição global do veículo, também conhecido como *kidnapping*.

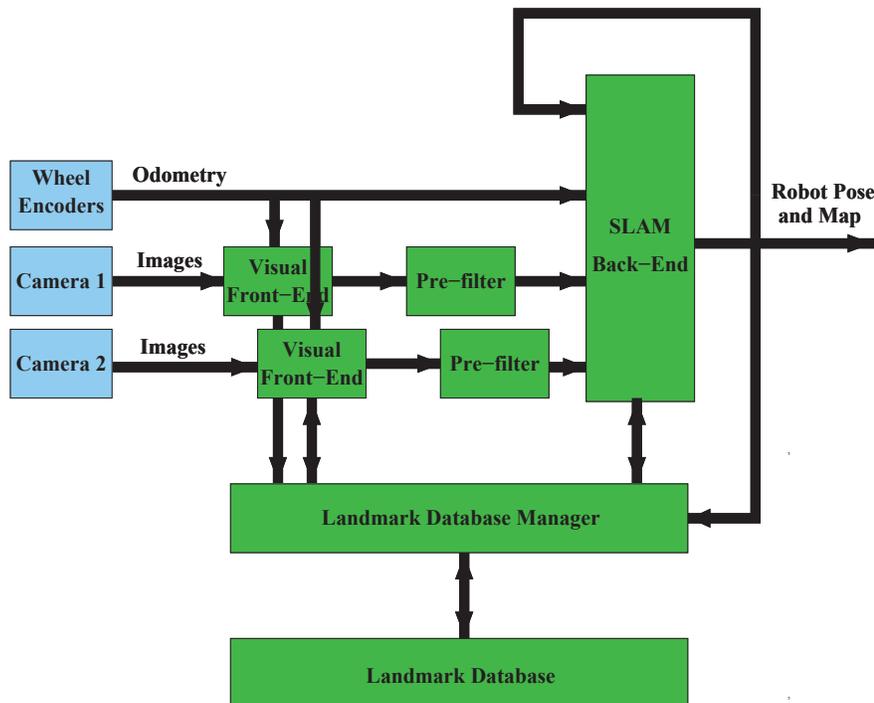


Figura 2.7: Diagrama em blocos do V-SLAM

2.2.2 Localização a partir observações absolutas

Sistema de posicionamento global

O sistema de posicionamento global (GPS, do inglês Global Positioning System) é um sistema de navegação baseado em satélites. O seu funcionamento compõe-se de um sistema de emissão e recepção de sinais de radiofrequência, os quais permitem o cálculo da posição e velocidade do veículo. Esse cálculo é feito usando a técnica de triangulação a fim de achar as coordenadas globais, ou seja, latitude, longitude e altitude. A triangulação é possível se tiver no mínimo três satélites. Mesmo que seja um sistema de localização absoluto bastante usado, existem fenômenos naturais que alteram a precisão da medição entregue, como a reflexão e refração das ondas de rádio. Além das fontes de ruído mencionadas anteriormente, tem-se outros tipos de fatores associados com parâmetros intrínsecos do sensor, como o viés (Grewal,2007). O viés é um ruído que pode ser modelado como uma caminhada aleatória, em que o efeito enxergado no sistema é um aparente movimento mesmo que esteja parado.

2.2.3 Localização robusta

Como dito anteriormente, tanto os métodos de localização robusta quanto os métodos de localização relativa apresentam certos erros próprios que fazem com que a região de incerteza de uma observação seja muito grande, porém qualquer dessas soluções por si só não garante robustez. Para corrigir essas medições, a combinação de sensores é muito usada (DASARATHY, 1997). A fusão de sensores busca com que seja possível combinar a informação de diferentes sensores a fim de aumentar a confiabilidade de cada um deles. Normalmente, esses sensores possuem diferentes modelos de erro e taxas de amostragem. Na revisão feita por Singhal (1997) apresentam-se as duas principais abordagens usadas na fusão sensorial: as técnicas estatísticas e probabilísticas. A abordagem estatística está baseada na estimação de mínimos quadrados, enquanto a abordagem probabilística está baseado no raciocínio de Bayes. Um método bastante usado na fusão de sensores, que possuem uma distribuição uni-modal, é o filtro de kalman que, aliás, pertence às técnicas estatísticas.

Filtro de Kalman

O filtro de kalman é um método matemático utilizado para estimar resultados que tendem a se aproximar aos valores reais das grandezas físicas medidas ao longo do tempo. Essas medições normalmente estão contaminadas com ruído que geram uma incerteza em relação ao valor real. A estimação feita pelo filtro de kalman é recursiva, ou seja, está baseada em estimações e medições de um estado anterior para calcular um estado atual.

O filtro de kalman é um boa ferramenta para sistemas lineares, mas na prática sabe-se que a maioria dos sistemas são não lineares. Neste casos se emprega uma versão para sistemas não lineares chamada de filtro de kalman estendido (EKF), no qual o sistema é linearizado em torno de um estado do sistema. Esse estado é uma estimação a priori ao estado atual. Usando o EKF, Weinstein e Moore (2010) implementaram um sistema de localização robusta. A fusão sensorial foi feita usando um GPS de baixo custo e uma bússola como sensores absolutos, enquanto encoder nas rodas e no ângulo de viragem foram usados como sensores relativos. O objetivo foi, então, estimar a pose do veículo e o viés do GPS, usando como entrada a velocidade média das rodas (medidas no eixo traseiro do carro) e o ângulo de esterçamento. Apesar de obter bons resultados, não foi achado um modelo de odometria válido para um automóvel que garanta maior precisão na estimação do

deslocamento ou velocidade.

Um modelo consistente com as medições do deslocamento das rodas e o ângulo de esterçamento foi apresentado por Bonnifait *e outros* (2001), em que foram concatenados os filtros de odometria e localização. Essa odometria foi implementada com os freios ABS do veículo e um encoder absoluto para a direção de esterçamento. Uma versão modificada desse trabalho foi apresentado por (SURRECIO *e outros*, 2005); ali foram integrados encoders em cada uma das quatro rodas e marcadores magnéticos fixos no chão, onde foi identificado que o erro sistemático de maior grandeza era devido aos encoders; eles desenvolveram um sistema de medição de odometria usando informação das 4 rodas e do sensor de esterçamento (direção do veículo), garantindo assim uma boa precisão comparado ao sistema que utiliza somente a informação das rodas traseiras que é uma abordagem muito comum.

3 FUNDAMENTOS TEÓRICOS

A fim de localizar um veículo num ambiente desconhecido baseado em observações tanto proprioceptivas como exteroceptivas, há um conjunto de parâmetros, mostrados na equação equação 3.1, necessários para definir a posição e orientação em relação a um referencial global (X, Y) . Neste capítulo, são apresentados os modelos matemáticos utilizados na localização e reconstrução da trajetória do veículo, além dos sensores empregados.

$$x = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (3.1)$$

3.1 Modelo do veículo

O modelo de um sistema é uma aproximação matemática usando determinadas características. Com o objetivo de representar a evolução temporal do sistema utiliza-se de um modelo cinemático. Esse modelo é uma composição entre o modelo geométrico e o modelo de odometria que serão descritos a seguir. Além do modelo cinemático, na abordagem estatística, necessita-se de um modelo que contenha a incerteza das observações e para isso também apresenta-se o modelo probabilístico.

3.1.1 Modelo geométrico do veículo

O modelo geométrico utilizado neste trabalho é o modelo de *Ackermann* (SNIDER, 2009), esse modelo pode ser estendido para ser usado no veículo em escala. Na figura 3.1 tem-se a representação do veículo, no qual as rodas traseiras são fixas e as rodas dianteiras são atuadas por meio do volante. Como parâmetros desse modelo tem-se o ângulo de esterçamento (ω), a distância entre os eixos (L) e o ângulo de orientação (θ) em relação ao sistema de referência (x, y) . Observa-se que o referencial do veículo denota-se por (x_k, y_k) e que o modelo é válido para baixas velocidades.

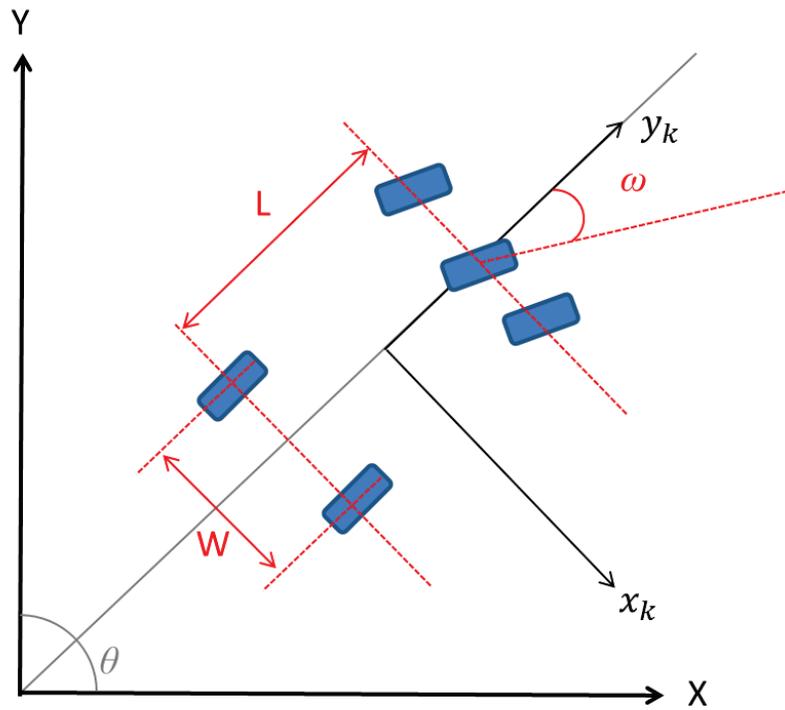


Figura 3.1: Modelo de Ackermann para veículo de 4 rodas

Para a implementação do modelo cinemático são feitas duas simplificações na geometria de *Ackermann*. A primeira delas é que modelo pode-se reduzir ao modelo da bicicleta (figura 3.2), em que nessa redução considera-se que tanto as rodas dianteiras quanto as rodas traseiras são representadas por um modelo de duas rodas. A segunda simplificação deve-se ao fato que o veículo está restrito a se deslocar somente no plano. As equações para o modelo cinemático da bicicleta estão de forma detalhada no trabalho de Campbell (2009).

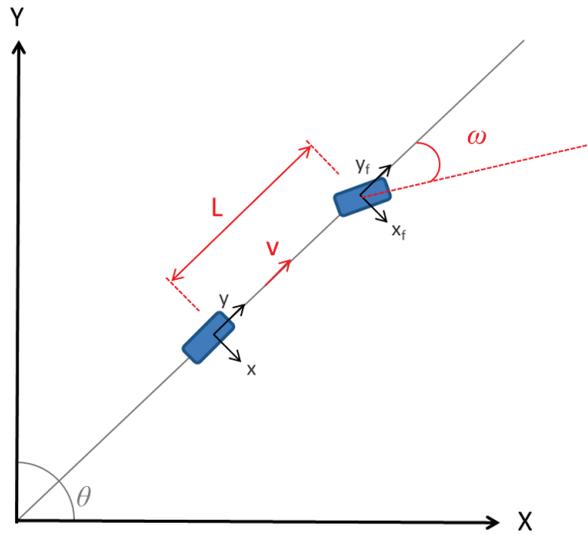


Figura 3.2: Modelo cinemático da bicicleta

Baseado nas simplificações anteriores, as restrições não-holonômicos para as rodas traseiras e dianteiras são:

$$\dot{x}_f \sin(\theta + \omega) - \dot{y}_f \cos(\theta + \omega) = 0 \quad (3.2)$$

$$\dot{x} \sin(\theta) - \dot{y} \cos(\theta) = 0 \quad (3.3)$$

Nesse caso (x, y) é o sistema global de coordenadas da roda traseira, (x_f, y_f) é o sistema de coordenadas da roda dianteira, θ é o ângulo de orientação do veículo em relação ao sistema global de coordenadas e ω é o ângulo de esterçamento da bicicleta, em que (x_f, y_f) podem ser expressas como:

$$\begin{aligned} x_f &= x + L \cos(\theta) \\ y_f &= y + L \sin(\theta) \end{aligned} \quad (3.4)$$

Substituindo (x_f, y_f) da equação (3.4) na equação (3.2) temos:

$$\frac{d(x + L \cos(\theta))}{dt} \sin(\theta + \omega) - \frac{d(y + L \sin(\theta))}{dt} \cos(\theta + \omega) = 0 \quad (3.5)$$

ou ainda na forma,

$$\left(\dot{x} - \dot{\theta} L \sin(\theta) \right) \sin(\theta + \omega) - \left(\dot{y} + \dot{\theta} L \cos(\theta) \right) \cos(\theta + \omega) = 0 \quad (3.6)$$

Simplificando a equação (3.6) tem-se:

$$\dot{x} \sin(\theta + \omega) - \dot{y} \cos(\theta + \omega) - \dot{\theta} L (\sin^2(\theta) + \cos^2(\theta)) \cos(\omega) = 0 \quad (3.7)$$

ou ainda na forma,

$$\dot{x} \sin(\theta + \omega) - \dot{y} \cos(\theta + \omega) + \dot{\theta} L \cos(\omega) = 0 \quad (3.8)$$

A restrição não-holonômica da roda traseira, equação 3.3, é satisfeita por $\dot{x} = v \cos(\theta)$, e $\dot{y} = v \sin(\theta)$ e qualquer escalar. Esse escalar corresponde a velocidade longitudinal v , então tem-se:

$$\begin{aligned} \dot{x} &= v \cos(\theta) \\ \dot{y} &= v \sin(\theta) \end{aligned} \quad (3.9)$$

Isolando $\dot{\theta}$ da equação (3.8) tem-se:

$$\dot{\theta} = \frac{\dot{x} \sin(\theta + \omega) - \dot{y} \cos(\theta + \omega)}{L \cos(\omega)} \quad (3.10)$$

Substituindo a equação (3.9) na equação (3.10) temos:

$$\dot{\theta} = \frac{v \cos(\theta) (\sin(\theta) \cos(\omega) + \cos(\theta) \sin(\omega))}{L \cos(\omega)} - \frac{v \sin(\theta) (\cos(\theta) \cos(\omega) + \sin(\theta) \sin(\omega))}{L \cos(\omega)} \quad (3.11)$$

ou ainda na forma simplificada:

$$\dot{\theta} = \frac{v (\cos^2(\theta) + \sin^2(\theta)) \sin(\omega)}{L \cos(\omega)} \quad (3.12)$$

resultando na equação da solução de $\dot{\theta}$.

$$\dot{\theta} = \frac{v}{L} \cdot \tan(\omega) \quad (3.13)$$

O raio instantâneo de rotação R do veículo é determinado a partir de v e $\dot{\theta}$ assim:

$$R = \frac{v}{\dot{\theta}} \quad (3.14)$$

Substituindo a equação 3.13 na equação 3.14 temos:

$$\frac{v \tan(\omega)}{L} = \frac{v}{R} \quad (3.15)$$

ou ainda na forma simplificada:

$$\tan(\omega) = \frac{L}{R} \quad (3.16)$$

O modelo discretizado por Euler, que representa o modelo do veículo no espaço de estado, é

mostrado na equação 3.17, onde dt é o período de amostragem.

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} x_k + dtv_k \cos(\theta) \\ y_k + dtv_k \sin(\theta) \\ \theta_k + L^{-1}dtv_k \tan(\omega) \end{bmatrix} \quad (3.17)$$

3.1.2 Modelo de odometria do veículo

O propósito da odometria é criar um modelo de integração usando medições das rotações das rodas e, no caso particular dos veículos tipo automóvel, do ângulo de esterçamento. As relações geométricas usadas nesta seção foram desenvolvidas no trabalho de Bonnifait *e outros* (2001). A figura 3.3 representa o modelo geométrico do veículo numa manobra de esterçamento.

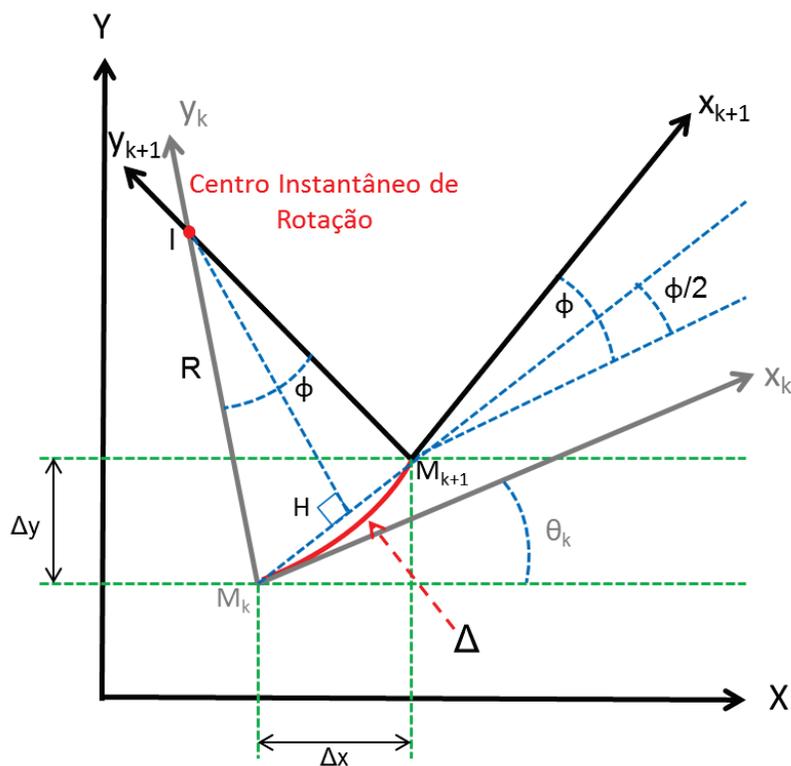


Figura 3.3: Deslocamento entre duas amostras $k \rightarrow k + 1$

Considere-se o modelo de Ackermann (figura 3.1). O quadro de referência móvel M encontra-

se no meio do eixo traseiro. O eixo x está alinhado com o eixo longitudinal do carro. Para um tempo t_k a posição do veículo é representada pelo sistema de coordenadas cartesiano (x_k, y_k) . Então duas posições sucessivas são definidas como M_k e M_{k+1} . Supondo-se que a superfície de movimento seja planar e o movimento seja circular, tem-se que:

$$\Delta = R\phi \quad (3.18)$$

onde Δ é o comprimento do arco(movimento circular), ϕ é a rotação do eixo e R é o raio de curvatura para um centro instantâneo de rotação I . Supondo-se que o carro desloque-se para frente, a variação da posição fica expressa por:

$$\Delta_x = |M_k M_{k+1}| \cdot \cos(\theta_k + \phi/2)$$

$$\Delta_y = |M_k M_{k+1}| \cdot \sin(\theta_k + \phi/2)$$

Da geometria euclidiana tem-se que $\Delta = |M_k M_{k+1}|$, então o processo de integração pode ser escrito como:

$$x_{k+1} = x_k + \Delta_x = x_k + \Delta \cos\left(\theta_k + \frac{\phi}{2}\right) \quad (3.19)$$

$$y_{k+1} = y_k + \Delta_y = y_k + \Delta \sin\left(\theta_k + \frac{\phi}{2}\right) \quad (3.20)$$

$$\theta_{k+1} = \theta_k + \phi \quad (3.21)$$

onde Δ e ϕ podem ser calculados usando o ângulo de esterçamento ω e a distância percorrida por cada uma das rodas, Δ_{RR} e Δ_{RL} indicam a distância percorrida pelas rodas traseira direita e esquerda respectivamente, Δ_{FR} e Δ_{FL} indicam a distância percorrida pelas rodas dianteiras direita e esquerda respectivamente, e a medição de ω prevê uma relação entre Δ e ω . Reescrevendo a equação 3.16 e 3.18 tem-se:

Na técnica de odometria diferencial (ANTONELLI *e outros*, 2010), Δ e ω são calculados usando só as medições das rodas traseiras (Δ_{RR} e Δ_{RL}). Assumindo que não há derrapagem, isto é,

$$\begin{aligned}\Delta &= \frac{\Delta_{RR} + \Delta_{RL}}{2} \\ \omega &= \frac{\Delta_{RR} - \Delta_{RL}}{2W}\end{aligned}\tag{3.25}$$

onde W é distância entre as rodas traseiras ou o comprimento do eixo traseiro.

Encontrar o deslocamento das rodas dianteiras é menos trivial dado que a sua orientação em relação ao chassis não é constante. Para isto, é feita uma simplificação usando uma roda virtual no centro do eixo dianteiro do veículo (figura 3.4). Para o comprimento da roda virtual, tem-se

$$\Delta_F = D\phi\tag{3.26}$$

Usando a figura 3.4 pode-se escrever também que:

$$D = \frac{R}{\cos(\omega)}\tag{3.27}$$

Multiplicando-se cada lado por ϕ e utilizando as equações 3.26 e 3.18, tem-se:

$$\Delta_F \cos(\omega) = \Delta\tag{3.28}$$

Aplicando o mesmo procedimento para cada uma das rodas dianteiras, obtém-se:

$$\begin{aligned}\tan(\omega_L) &= \frac{L}{R - \frac{W}{2}} \\ \tan(\omega_R) &= \frac{L}{R + \frac{W}{2}}\end{aligned}\tag{3.29}$$

A variável R pode ser eliminada da equação 3.29 usando a relação da equação 3.16, isto é,

$$\omega_L = \arctan \left(\frac{\tan(\omega) \cdot L}{L - \frac{W}{2} \cdot \tan(\omega)} \right) \quad (3.30)$$

$$\omega_R = \arctan \left(\frac{\tan(\omega) \cdot L}{L + \frac{W}{2} \cdot \tan(\omega)} \right) \quad (3.31)$$

Finalmente, adaptando a equação 3.28 para cada roda dianteira, tem-se

$$\Delta_{FL} \cos(\omega_L) = \Delta - \phi \frac{W}{2} \quad (3.32)$$

$$\Delta_{FR} \cos(\omega_R) = \Delta + \phi \frac{W}{2} \quad (3.33)$$

Um conjunto de equações podem ser estabelecidas baseados nas observações dos diferentes sensores de odometria das rodas e do ângulo de esterçamento. No trabalho (BONNIFAIT *e outros*, 2001) foi definido o sistema redundante e não-linear que relaciona tanto as grandezas físicas do vetor de entrada u e as observações $(\Delta_{RR}, \Delta_{RL}, \Delta_{FR}, \Delta_{FL}, \omega)$. Essa relação, definida pelas equações 3.22, 3.23, 3.24, 3.32 e 3.33, é apresentada na tabela 4.3.

$\tan(\omega)$	$L \frac{\phi}{\Delta}$
Δ_{RL}	$\Delta - \phi \frac{W}{2}$
Δ_{RR}	$\Delta + \phi \frac{W}{2}$
$\Delta_{FL} \cos(\omega_L)$	$\Delta - \phi \frac{W}{2}$
$\Delta_{FR} \cos(\omega_R)$	$\Delta + \phi \frac{W}{2}$

Tabela 3.1: Equações de observação odometria

3.1.3 Modelo probabilístico

A incerteza inicial sobre a pose do veículo, representado pelo desvio padrão das variáveis do sistema, aumenta com o tempo quando o veículo se desloca com um sinal de controle u . Mas, conforme o veículo for se deslocando, percebe-se que uma forma de diminuir essa incerteza é usar a predição estatística. Como no modelo controlável discutido na seção do modelo cinemático, o passo da predição é usado na forma não-linear. Esse modelo controlável pode ser representado pela equação 3.34, em que $u(k)$ é o vetor de controle.

$$x(k+1) = f(x(k), u(k)); \quad u(k) = \begin{bmatrix} v(k) \\ \omega(k) \end{bmatrix} \quad (3.34)$$

Para um sinal de controle u , num tempo $k+1$, pode-se definir, baseado no filtro de kalman (KALMAN, 1960), um modelo de estimação de estado usando uma boa medição de um estado anterior k na forma

$$x(k | k+1) = f(x(k-1 | k-1), u(k), k) \quad (3.35)$$

Além da predição, há associado uma covariância do sistema P , isto é,

$$P(k | k-1) = \nabla F_x P(k-1 | k-1) \nabla F_x^T + \nabla F_u Q \nabla F_u^T \quad (3.36)$$

Um aspecto importante na predição estatística são as matrizes de covariância, assim, o correto ajuste dos valores garante a convergência do filtro de Kalman. Um resumo das matrizes usadas no filtro estatístico e uma breve definição são mostradas na tabela 3.2.

Assim, na equação 3.36 associa-se a matriz P e matriz Q . A matriz P , que representa o desvio padrão das variáveis de estado do sistema e cujo vetor de estado é dado por $X = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$, pode ser escrita na forma:

Parâmetro	Descrição
P	Matriz referente ao ruído do sistema associados à imperfeição do modelo.
Q	Matriz referente ao ruído associado à entrada do sistema.
R	Matriz referente ao ruído presentes nas observações ou medições.

Tabela 3.2: Tabela matrizes de covariância

$$P = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix} \quad (3.37)$$

A matriz Q , que é a covariância do sinal de controle, ou seja, o desvio padrão da velocidade σ_v e o desvio padrão do ângulo do volante da direção σ_ω , pode ser escrita como:

$$Q = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix} \quad (3.38)$$

O último parâmetro R refere-se ao ruído, inerente ao equipamento de medição, porém, difícil de identificar. Essas características que geram erro de medição são muito bem identificadas pela filtragem de Kalman adaptativa, que usa por exemplo redes neurais. Neste caso, o ruído é modelado como um ruído branco com um desvio padrão de valor 0.1.

$$R = \begin{bmatrix} \varepsilon_v^2 & 0 \\ 0 & \varepsilon_\omega^2 \end{bmatrix} \quad (3.39)$$

Note-se que na equação 3.36 é calculada a probabilidade usando uma versão linearizada do modelo cinemático do veículo. Para isso é calculado o Jacobiano da matriz do sistema 3.40 e da matriz de controle 3.41:

$$\nabla F_x = \frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial x}{\partial x} & \frac{\partial x}{\partial y} & \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial x} & \frac{\partial y}{\partial y} & \frac{\partial y}{\partial \theta} \\ \frac{\partial \theta}{\partial x} & \frac{\partial \theta}{\partial y} & \frac{\partial \theta}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -v \cdot dt \cdot \sin(\theta) \\ 0 & 1 & v \cdot dt \cdot \cos(\theta) \\ 0 & 0 & 1 \end{bmatrix} \quad (3.40)$$

$$\nabla F_u = \frac{\partial f}{\partial u} = \begin{bmatrix} \frac{\partial x}{\partial v} & \frac{\partial x}{\partial \omega} \\ \frac{\partial y}{\partial v} & \frac{\partial y}{\partial \omega} \\ \frac{\partial \theta}{\partial v} & \frac{\partial \theta}{\partial \omega} \end{bmatrix} = \begin{bmatrix} dt \cdot \cos(\theta) & 0 \\ dt \cdot \sin(\theta) & 0 \\ \frac{\tan(\omega)}{L} & \frac{v \cdot dt \cdot \sec^2(\omega)}{L} \end{bmatrix} \quad (3.41)$$

Na figura 3.5 apresenta-se uma simulação que mostra o aumento da incerteza da posição e da orientação do veículo para uma entrada de controle com velocidade constante e um ângulo de esterçamento sinusoidal, em que a covariância é representada por uma elipse (Gaussiana bidimensional) para σ_x e σ_y .

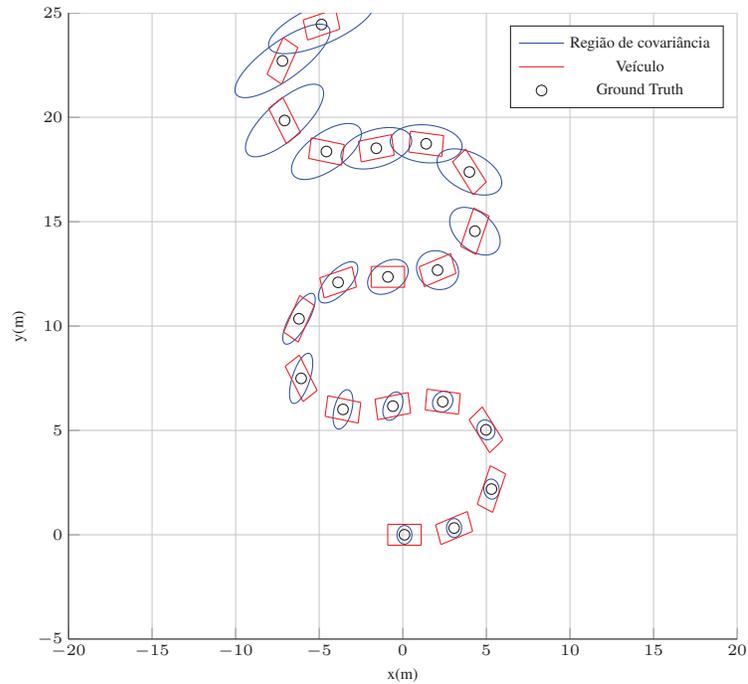


Figura 3.5: Evolução de incerteza para o modelo de veículo

3.2 Sensores

Calcular a posição de um veículo para se localizar através de diferentes observações ao longo do percurso requer informação redundante lida de diferentes sensores. Assim, diferentes medições das rodas enquanto houver deslocamento ou do ângulo de esterçamento se tiver manobra de viragem podem ser utilizadas para determinar a posição, a velocidade e a aceleração.

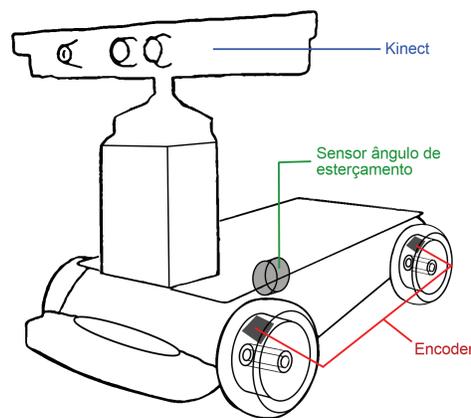


Figura 3.6: Sensores embarcados no veículo

3.2.1 Encoder óptico

Funcionalmente, existem dois tipos de encoder de posição: incremental e absoluto. A principal característica que faz diferença entre os dois tipos de encoder é que enquanto o encoder absoluto, mesmo que haja uma queda de tensão, não perde a posição real, o encoder incremental fornece a posição pela contagem de pulsos a partir de uma referência. O Encoder absoluto garante uma leitura única para cada posição. A diferença também pode ser quantizada em termos de preço, que, dependendo da aplicação, permite ter uma escolha apropriada. Nesse experimento utilizou-se o princípio de encoder óptico incremental, porém, a seguir introduz-se uma breve descrição desse sensor em sistemas rotativos baseado no livro de Pallás-areny and Webster(2001).

O encoder incremental caracteriza-se por possuir uma região de medição que tenha uma propriedade que permita diferenciá-los, essa região deve ter um padrão repetitivo(figura C.1). Essa

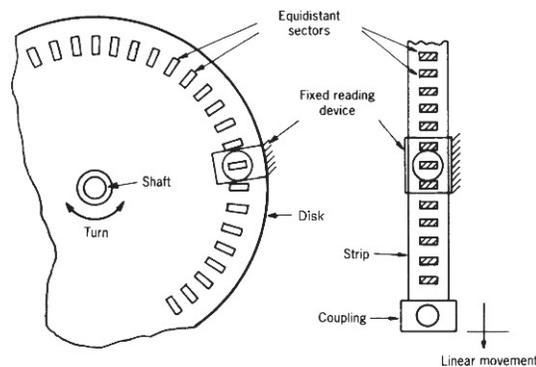


Figura 3.7: Princípio de encoder incremental

propriedade pode ser de distintos tipos de grandeza física, não obstante, os mais comuns são de tipo magnético e óptico. Quanto à construção física dos encoder acoplados a elementos rotativos, usam-se discos que possuem uma inércia muito baixa, que é desejável para não interferir no comportamento do sistema. A leitura (sensor fixo) da variação das propriedades antes citadas é obtida como um trem de pulsos elétricos (onda quadrada) com ciclo de trabalho (duty cycle) de 50%. O número de pulsos segue uma proporção na geometria da circunferência equação (3.42).

$$m = \frac{\pi d}{2p} \quad (3.42)$$

em que se vão ter m pulsos a cada volta, dado que a saída muda quando aumentar a posição em duas vezes o passo p , para um disco de diâmetro d .

Se for usar esse tipo de encoder para determinar a direção de deslocamento, requer-se de uma configuração adicional que geralmente é implementada com outro dispositivo de leitura, juntando-se para fornecer dois sinais Figura (3.8). Essa eletrônica coloca dois pulsos (Canal A e Canal B na Figura 3.8) com uma defasagem de 90; esta configuração recebe o nome de quadratura.

Os encoders ópticos estão baseados em regiões refletivas e não refletivas. O sensor é composto de uma fonte de luz (LED infravermelho) e de um fotodetector (foto transistor). Um dos principais problemas que há com esse tipo de sensor é o atraso na resposta dos componentes eletrônicos (frequência de operação) e vibrações no elementos de medição (réguas).

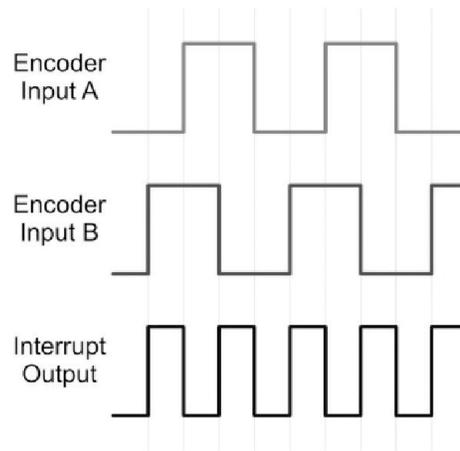


Figura 3.8: Canais de Onda

3.2.2 Sensor ângulo de esterçamento

Uma variável importante na cinemática de um automóvel é o ângulo de esterçamento. Esse ângulo é normalmente medido como a posição absoluta do eixo do mecanismo da direção do veículo (figura 3.9), que está geralmente acoplado ao volante da direção.

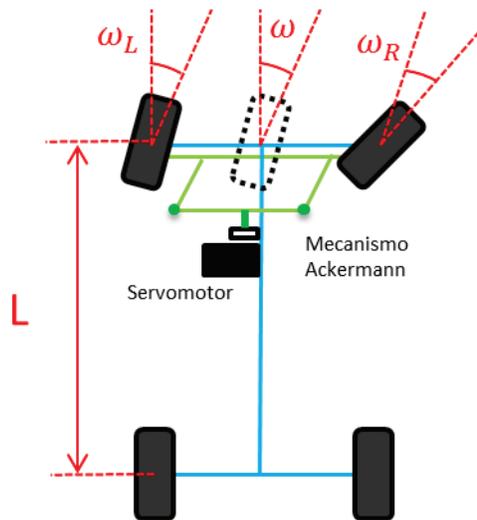


Figura 3.9: Mecanismo de esterçamento

A informação desse sensor, na maioria de veículos comerciais, está disponibilizada na rede

CAN ou redes tais como *I2C* em veículos em escala. Usualmente encontra-se implementado com sensores absolutos como o encoder ótico (descrito na seção anterior). Nesse trabalho, devido a restrições de mecânicas, foi utilizado um potenciômetro linear. Para mais informação em relação à calibração deste sensor, ver Anexo C.

3.2.3 Câmera RGB-D: Kinect



Figura 3.10: Câmera RGB-D: *Kinect* da *microsoft*

Sensores de mapeamento de baixo custo oferecem uma boa alternativa aos caros escâner a laser(*SICK* ou *LIDAR*), voltados para o mapeamento e localização em ambientes interiores(*indoor*). Uma alternativa para estes sensores são as câmeras RGB-D. As câmeras RGB-D capturam uma imagem colorida RGB aumentadas com uma estimativa per-pixel da profundidade. Neste trabalho será usada uma câmera RGB-D comercial fabricada pela *Microsoft* chamada *Kinect*(Figura 3.10), que se compõe dos seguintes sensores ópticos:

- Câmera RGB: câmera com uma resolução de 640 x 480
- Sensor de Profundidade: câmera com um filtro infravermelho e um projetor de texturas.

O *KINECT* oferece capacidades potenciais na robótica móvel, porém, foi lançado para o setor de diversão e entretenimento, assim, tem-se pouca informação disponibilizada sobre o uso para

outras aplicações que sejam diferentes. Muitos sensores similares foram explorados em trabalhos anteriores, por exemplo o SwissRanger[MESA imaging] e PDM[CamlCube 3.0], mas são de alto custo. O *Kinect* está baseado numa patente que explora o processo de triangulação [Depth mapping using projected patterns - Patente emitida: US8150142]. Na próxima seção será desvendando o funcionamento do *KINECT* a fim de formular a metodologia na reconstrução do percurso.

3.3 Formulações matemáticas em Odometria Visual

3.3.1 Modelo matemático da câmera RGB-D

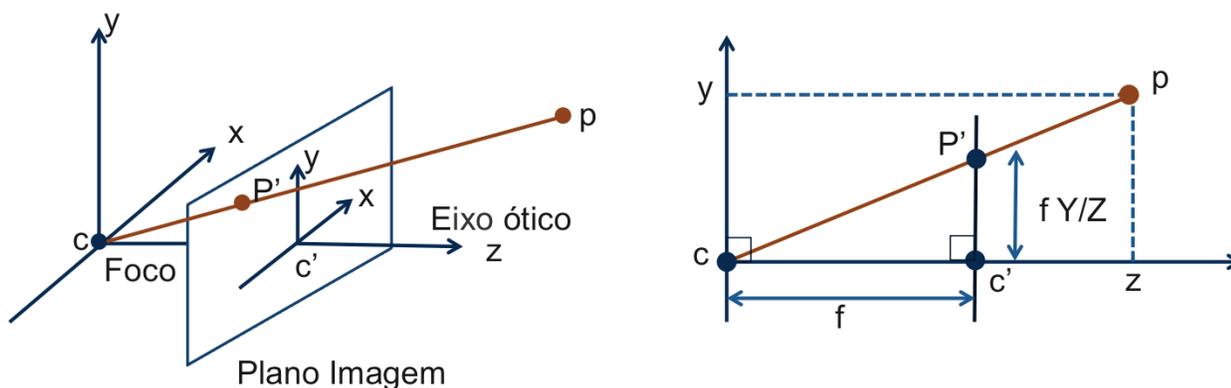


Figura 3.11: Modelo Câmera Pontual

Um modelo simples, mas útil, para representar o conceito de percepção de visão através da visão computacional é o modelo de câmera pontual ou *pinhole*. Nesse modelo, os raios passam por um único ponto para logo serem projetados numa superfície plana (figura 3.11). Esse modelo está baseado no trabalho feito por Hartley e Zisserman (2004), no qual foram definidos os parâmetros que descrevem a geometria projetiva. Esses parâmetros estão definidos como o mapeamento entre dois espaços projetivos $K : \mathbb{R}^3 \rightarrow \mathbb{R}^2$, ou seja, do espaço imagem para o espaço dos objetos(contidos na cena enxergada pela câmera). Essa relação entre dois pontos é dada pela equação 3.43, onde $q \in \mathbb{R}^2$ é um ponto no espaço imagem e $Q \in \mathbb{R}^3$ é um ponto no espaço do objeto.

$$q = KQ \tag{3.43}$$

A fim de expressar as relações entre a câmera e objetos de uma cena é necessário definir dois tipos de parâmetros. Há um conjunto de parâmetros que dependem da própria câmera, os quais são conhecidos como parâmetros intrínsecos. Esses parâmetros associam as coordenadas de *pixel* de um ponto na imagem com as coordenadas no referencial de câmera (equação 3.44).

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (3.44)$$

onde f é a distância entre o plano da imagem e o centro de projeção. A linha perpendicular ao plano da imagem que passa pelo centro de projeção é chamada de eixo óptico. O ponto de interseção entre o eixo óptico e o plano da imagem é chamado de ponto principal ou centro da imagem, com coordenadas c_x e c_y .

Do outro lado, tem-se os parâmetros extrínsecos. Ditos parâmetros determinam a posição t e orientação da câmera em relação aos objetos na cena. Com os parâmetros antes mencionados, tem-se a matriz de câmera P (equação 3.45). Essa matriz define a relação entre os elementos de um referencial global e os elementos da imagem, usando os parâmetros de posição t e orientação R .

$$P = K [R_q | t] \quad (3.45)$$

As câmeras RGB-D capturam uma imagem colorida RGB aumentada com uma estimativa *per-pixel* da profundidade. Essas câmeras caracterizam-se pela capacidade para estimar um valor de profundidade em regiões de baixa textura, graças ao padrão de luz projetado, porém limitado pelos projetores de textura, aliás baseados em tecnologia laser. Essa técnica foi desenvolvida pela desenvolvedora *Primesense* (Disponível em: <http://www.primesense.com/>), apesar de não se ter informação disponível em relação ao funcionamento desse sensor foi desvendado no trabalho apresentado por Khoshelham e Elberink (2012) uma análise detalhada das capacidades do *Kinect*. O esquema do (*Kinect*) é mostrado na figura 3.12. A seguir, apresenta-se um breve resumo do processo usado para determinar a profundidade.

O laser, que por causa da difração, produz uma projeção padrão uniforme sobre a superfície dos objetos na cena enxergados. Esse padrão é percebido pela câmera infravermelha, em que é feita

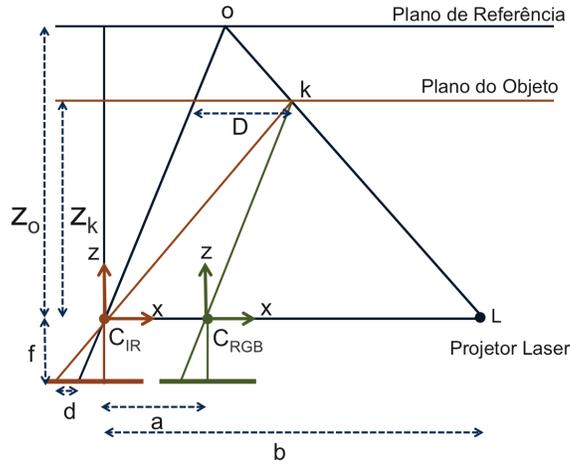


Figura 3.12: Modelo Geométrico da Câmera RGB-D

a correlação entre o padrão de referência e o ponto projetado. Quando um ponto bater num objeto numa distância maior ou menor do que o padrão de referência, produz um deslocamento na imagem da câmera infravermelha. Esses deslocamentos de cada ponto de projeção são medidos para gerar a imagem de disparidade, feita como uma operação de correlação.

$$\frac{D}{d} = \frac{Z_o - Z_k}{Z_o} \quad (3.46)$$

$$\frac{d}{f} = \frac{D}{Z_k} \quad (3.47)$$

onde Z_k é a distância (Profundidade) do ponto k no espaço de objeto, b é o comprimento da base, f é a distância focal da câmera IR (Infravermelha), D é o deslocamento do ponto no espaço de objeto, e d é a disparidade observável no espaço imagem. Agora, se substituir D da equação 3.47 na equação 3.46 tem-se:

$$Z_k = \frac{Z_o}{1 + \frac{Z_o}{fb}d} \quad (3.48)$$

Então fica definido que a escala do k -ésimo ponto pela coordenada Z_k é a distância focal f .

As coordenadas de objeto restantes podem ser calculadas a partir das coordenadas de imagem e da escala como apresentado nas equações 3.49 e 3.50, onde x_k e y_k são as coordenadas de imagem, x_o e y_o são as coordenadas da intersecção do eixo óptico com o plano de imagem e δx e δy são as constantes de correção de distorção.

$$X_k = -\frac{Z_k}{f} (x_k - x_o + \delta x) \quad (3.49)$$

$$Y_k = -\frac{Z_k}{f} (y_k - y_o + \delta y) \quad (3.50)$$

3.3.2 Extração de pontos de interesse

Na área de visão computacional é amplamente usado o conceito de pontos-chave ou *key-points*; esse tipo de características podem ser empregadas na solução de problemas como calibração de câmeras, registro de imagens, reconstrução 3D, e muitos outros. Na odometria visual, os *key-points* representam características do ambiente que logo são usadas como marcas, além disso, elas devem ser estáveis e observáveis desde diferentes pontos de vista. Isso faz com que seja necessário usar um método de extração que seja invariante à escala e a rotações.

Um dos métodos mais citados na literatura é o SURF (*Speeded Up Robust Feature*), um detector de características (invariante à escala e à rotação) robusto e apresentado por Bay e outros (2006). Esse método possui um custo computacional baixo em relação à outros descritores de alto desempenho (LOWE, 2004).

O segredo por trás das características invariantes à escala encontra-se no filtro laplaciano (equação 3.51), que, ao se aplicar em duas imagens de um mesmo objeto, estando em diferentes escalas, produz uma variância máxima σ que corresponde à taxa da escala. Assim, as características são obtidas como máximos locais no espaço imagem.

$$H(x, y) = \begin{bmatrix} \frac{\delta^2 x}{\delta x^2} & \frac{\delta^2 I}{\delta x \delta y} \\ \frac{\delta^2 I}{\delta x \delta y} & \frac{\delta^2 x}{\delta y^2} \end{bmatrix} \quad (3.51)$$

3.3.3 Estimação dos parâmetros de movimento

Sejam $X_i = [X_w \ Y_w \ Z_w \ 1]^T$ e $X_{i-1} = [X_w \ Y_w \ Z_w \ 1]^T$, um conjunto de pontos-chave no espaço $3D$, representando um conjunto de características comuns para uma cena enxergada por dois pontos de vista. No caso ideal(sem ruído), existe uma solução única (3.52) para serem achados os parâmetros de movimento. Esses parâmetros fazem referencia à matriz de transformação de corpo rígido $[R_q|t] \in R^{3 \times 4}$, onde R é a matriz de rotação e t é o vetor de translação.

$$X_i = R_q X_{i-1} + t \quad (3.52)$$

Logo requer-se encontrar o mapeamento $\hat{T} = [R_q | t]$ tal que $X_i = \hat{T} X_{i-1}$. Essa transformação pode ser rígida (translação e rotação) ou deformável (variação de escala ou outras deformações). Essa transformação pode ser definida de forma geral pela equação 3.53

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.53)$$

A solução deste sistema será abordada na seção de odometria visual(Capitulo 4, seção 4.1.3).

4 MÉTODOS DE LOCALIZAÇÃO

Com o objetivo de navegar de forma autônoma é necessário conhecer a posição e orientação (*heading*) do veículo. As diferentes técnicas que têm sido desenvolvidas para resolver esse problema podem ser classificadas em duas categorias:

- Localização relativa: Avaliar a posição e orientação através da integração das observações fornecidas por diferentes sensores. A integração começa numa posição inicial e ao longo do tempo é continuamente atualizada.
- Localização absoluta: Essa técnica permite determinar a posição no domínio da evolução do movimento. Esse método geralmente conta com sistemas como balizas, *Landmarks* (ativos ou passivos), correspondência de mapas ou informação global (GPS).

Neste capítulo são apresentados os métodos de localização usados neste trabalho. Assim, são explorados métodos considerados tanto relativos como absolutos.

4.1 Odometria Visual

Neste trabalho é apresentada uma técnica baseada em visão para localizar o veículo. Esta técnica, conhecida como Odometria Visual (NISTER *e outros*, 2006), utiliza características encontradas em uma sequência de imagens para encontrar a transformação rígida (figura 4.1) que há entre cada par de imagens (posição e orientação). O sistema de visão utilizado neste trabalho é baseado no *pipeline* proposto no trabalho Henry *e outros* (2010).

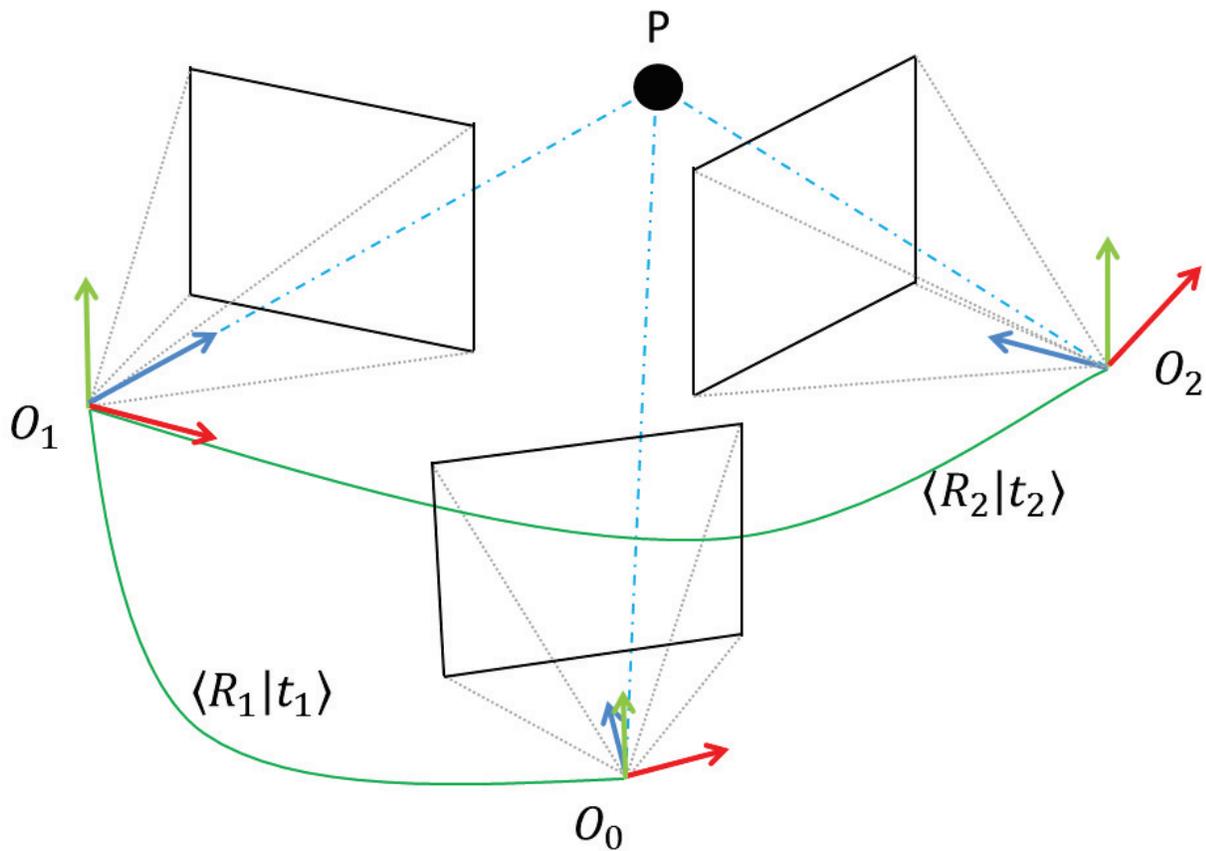


Figura 4.1: Esquema geral da odometria visual

4.1.1 Calibração da Câmera RGB-D

Para uma câmera que possui informação tanto da cor como da profundidade dos objetos apresentados na imagem, a calibração define-se como o processo para estimar os parâmetros intrínsecos e os parâmetros extrínsecos da câmera: os parâmetros intrínsecos fornecem características ópticas e geométricas internas da câmera, assim, torna-se possível calcular a posição de um ponto numa cena em termos da sua coordenada em pixels. Os parâmetros extrínsecos contêm a informação da posição e orientação da câmera em relação a um referencial global.

Neste trabalho foram implementados dois estágios voltados à obtenção da calibração da câmera *RGB-D*. O primeiro estágio aborda a calibração da profundidade, onde é modelada a resposta como uma função do valor quantizado pelo sensor (algoritmo *Light Coding Algorithm* desenvolvido

pelo fabricante). O segundo estágio baseia-se na utilização de um modelo linear entre as câmeras, assumindo-se como insignificantes os efeitos da distorção radial das lentes e como nulo o coeficiente de cisalhamento.

Função de profundidade

Para utilizar o sensor de profundidade é preciso calibrar a distância quantizada no processo de triangulação mencionado no capítulo 3 (seção 3.3.1). Sabe-se que o sensor tem uma resolução de 11 bits, ou seja, possui 2048 níveis de profundidade, mas que, devido à ausência de informação em relação à calibração (uso privativo do fabricante), não é possível achar um comportamento para a função de distância do Kinect. Mesmo assim, a função de profundidade que é usada no *framework* da Primsense ¹ é modelada como na equação (4.1), onde k_1 , k_2 e k_3 são constantes e $Z(u, v)$ é função do valor de quantização da profundidade em coordenadas do pixel.

$$depth = k_1 * \tan \left(\frac{Z(u, v)}{k_2 + k_3} \right) \quad (4.1)$$

Embora a função 4.1 seja muito usada, outros trabalhos têm sido desenvolvidos a fim de estabelecer diferentes modelos matemáticos para modelar a profundidade. O trabalho feito por Daniel Herrera e outros (2012), formulou um modelo de profundidade (equação 4.2) com sua respectiva calibração; aliás, os autores desse trabalho disponibilizaram um *toolkit* para usar na calibração do Kinect baseado no matlab, de forma que as constantes (tabela 4.1) α e β possam ser achadas.

$$depth = \frac{1}{\alpha Z(u, v) + \beta} \quad (4.2)$$

Na figura 4.2 faz-se uma comparação entre as duas curvas de calibração do sensor de profundidade.

¹OpenNI website <http://75.98.78.94/>

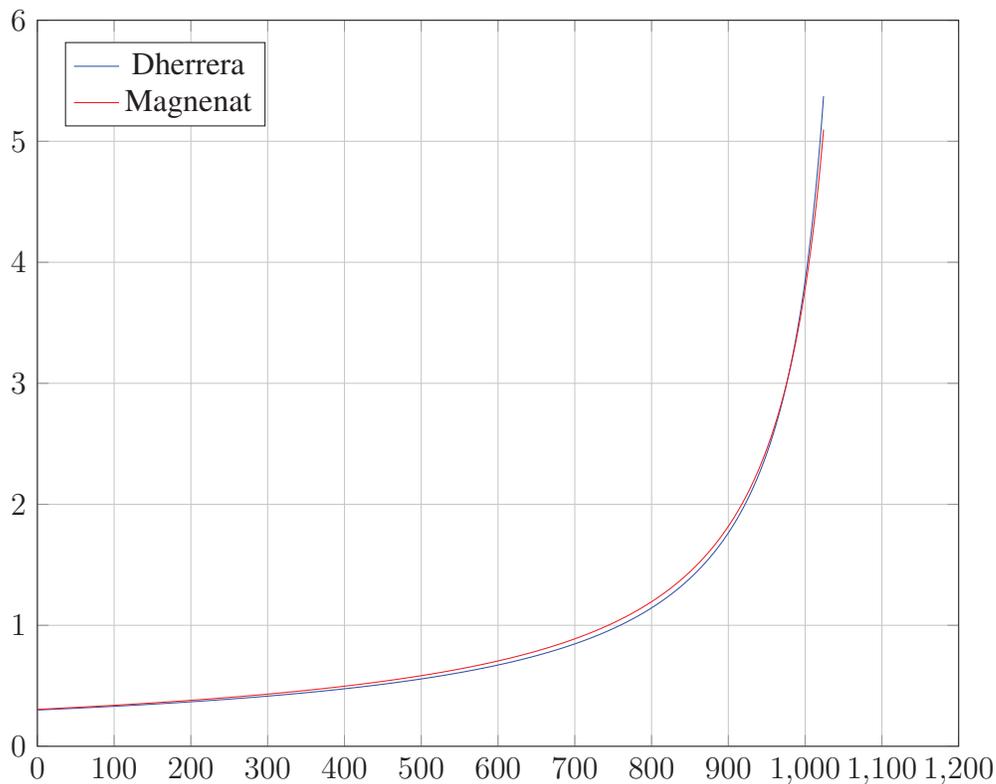


Figura 4.2: Aproximação da resposta em profundidade do Kinect

P_n	Coefficiente
α	-0.0030711016
β	3.3309495161

Tabela 4.1: Coeficiente função profundidade.

Os parâmetros intrínsecos de uma câmera no modelo *pinhole* são representados pela distância focal f e pelo ponto principal $\begin{bmatrix} C_u & C_v & 0 \end{bmatrix}^T$, que define-se como a posição em pixels da projeção ortogonal do centro óptico na imagem (esse ponto não é necessariamente o centro da imagem devido à diferença na precisão das câmeras). Outros fatores como o cisalhamento γ e a distorção das lentes, não foram considerados nesse trabalho.

Apesar de atualmente serem utilizados métodos inversos para mapear um ponto comum entre câmeras (RGB e *Depth*), como o usado no trabalho de Coleman *e outros* (2006), neste trabalho optou-se por um método direto. O mapeamento direto entre câmeras permite relacionar pontos-

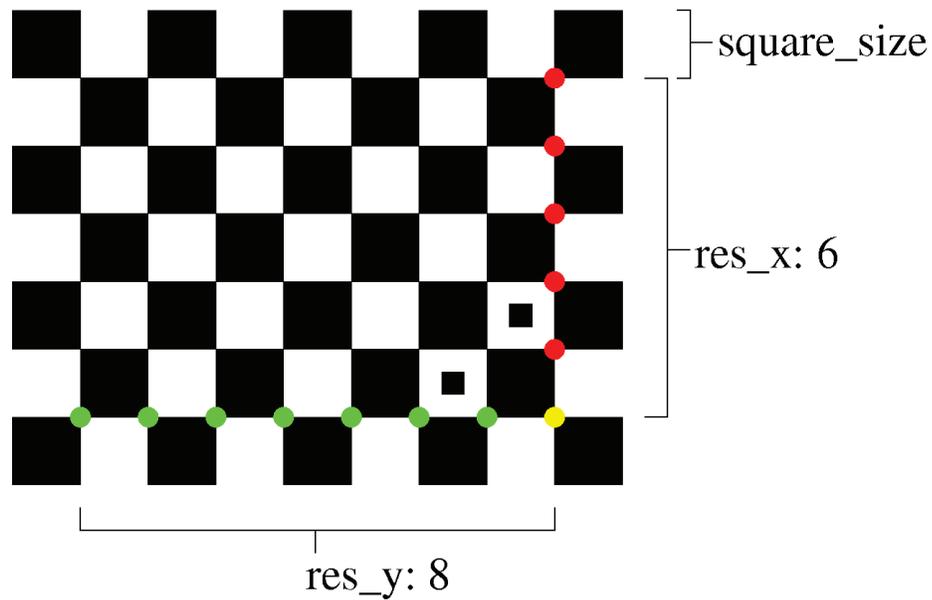


Figura 4.3: Padrão de calibração: Tabuleiro de xadrez

chave, e para isso é necessário conhecer os parâmetros intrínsecos da câmera infravermelha, para logo serem mapeados na câmera de cor (figura 4.4). Diferentes trabalhos em relação a esse mapeamento tem sido apresentados, como os trabalhos de Herrera *C e outros* (2012), Zhang e Zhang (2011) e Khoshelham e Elberink (2012). Esses trabalhos mostram que o processo de calibração não é trivial devido ao ruído na imagem de profundidade e precisa ser feito para cada câmera.

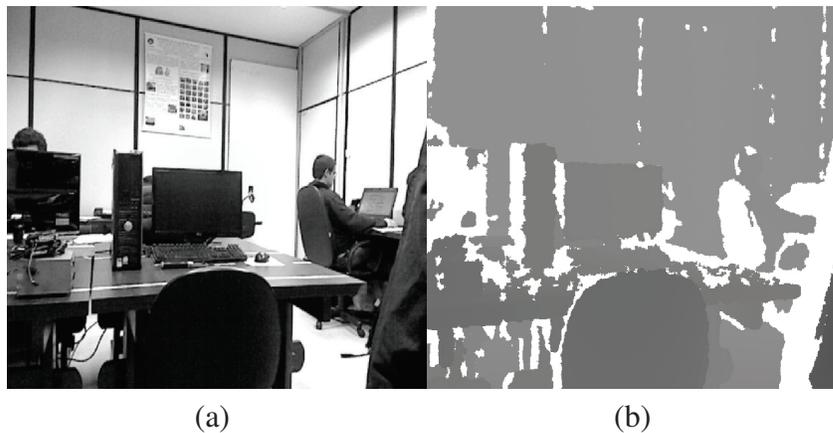


Figura 4.4: Cena gerada pelo Kinect.(a) Imagem tom de cinza do espectro visível. (b) Imagem profundidade(escalada para 8bits) do espectro infravermelho

Para a realização da calibração da câmera, utilizou-se o *Framework* desenvolvido por Burrus²,

²Burrus website <http://labs.manctl.com/rgbdemo>

amplamente documentado em Kramer *e outros* (2012). Esse *framework* está baseado no método de calibração apresentado por Zhang e Zhang (1998), que requer que a câmera observe um padrão (figura 4.3) planar em pelo menos duas orientações ou posições diferentes. Assim, consegue-se obter uma sequência de imagens apropriadas para modelar a distorção das lentes e obter um conjunto de parâmetros aceitáveis. Desse processo de calibração obtêm-se as matrizes de parâmetros intrínsecos das duas câmeras, k_{RGB} (4.3) e k_{Depth} (4.4).

$$k_{RGB} = \begin{bmatrix} 520.33 & 0 & 324.07 \\ 0 & 517.47 & 244.39 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.3)$$

$$k_{Depth} = \begin{bmatrix} 594.21 & 0 & 339.30 \\ 0 & 591.04 & 242.73 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

Esse método de mapeamento propõe uma transformação linear (equação 4.5) entre câmeras computando numa forma fechada as correspondências (UMEYAMA, 1991) dado um conjunto de pontos $\{x_i\}$ e $\{x'_i\}$ $i = 1, 2, \dots, n$, sob os seguintes pressupostos:

- Não existe nenhuma rotação entre câmeras.
- Não se tem fator de cisalhamento (elemento k_{12} da matriz de parâmetros intrínsecos).
- Os efeitos da disparidade são descartados.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \begin{bmatrix} u' \\ v' \end{bmatrix} + \begin{bmatrix} C \\ D \end{bmatrix} \quad (4.5)$$

Essas constantes são obtidas através do método dos mínimos quadrados para um mínimo de 8 pontos correspondentes. Esse método de mapeamento entre a nuvem de pontos e a imagem RGB é avaliado fazendo uma análise do erro de reprojeção, como apresentado no trabalho de Fiala e Ufkes (2011), em que o erro é pequeno em relação ao mapa de disparidade do Kinect.

4.1.2 Correspondência Robusta de imagens usando RANSAC

O RANSAC (FISCHLER E BOLLES, 1981) que vem das siglas em inglês *RAN*dOm *SA*mple *CO*nsensus, é um método iterativo para encontrar um conjunto de parâmetros de um modelo matemático. O algoritmo seleciona de forma randômica um conjunto de pontos para estimar os parâmetros do modelo, depois é avaliado o parâmetro do modelo fazendo a medição da distância de todos os pontos e o modelo atual (figura 4.5)

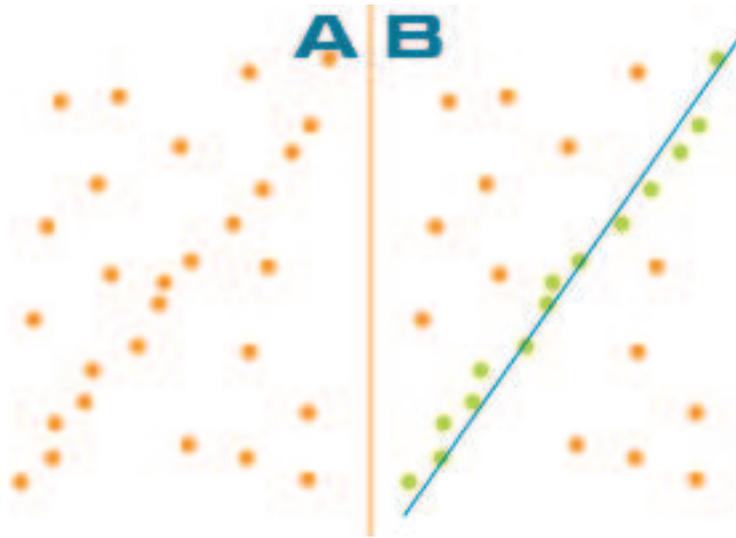


Figura 4.5: RANSAC para ajustar uma linha

A distância é avaliada dentro de um limiar, se estiver abaixo do limiar é classificado como um *inlier* ou valor que faz parte do modelo requerido, caso estiver fora do limite é rejeitado e classificado como *outlier*. Assim que o algoritmo for avaliado várias vezes, então é escolhido aquele que tiver o maior número de *inliers*. Após a escolha faz-se o cálculo dos parâmetros do modelo baseando-se no conjunto de pontos válidos. Esse número de vezes N que é avaliada uma amostra randômica, é dado pela seguinte equação,

$$N = \frac{\ln(1 - p)}{\ln(1 - (1 - \epsilon)^s)} \quad (4.6)$$

onde ϵ é a probabilidade de escolher um *outlier* e s é o número de pontos mínimos necessários para calcular os parâmetros do modelo.

Tendo um conjunto de correspondências $\{x_i \longleftrightarrow x'_i\}$, pode-se considerar que a única fonte

de erro é dada pela medição da posição de um ponto (Distribuição Gaussiana), mas algumas vezes pode acontecer que o conjunto definido não seja o correto. Esses pontos são chamados de *outliers*. A faixa restrita para os pares válidos está determinada pelos pontos que estão contidos na linha epipolar. Para isso, é necessário obter a matriz fundamental, mas precisa-se de um conjunto de características boas (*inliers*) para ser calculada.

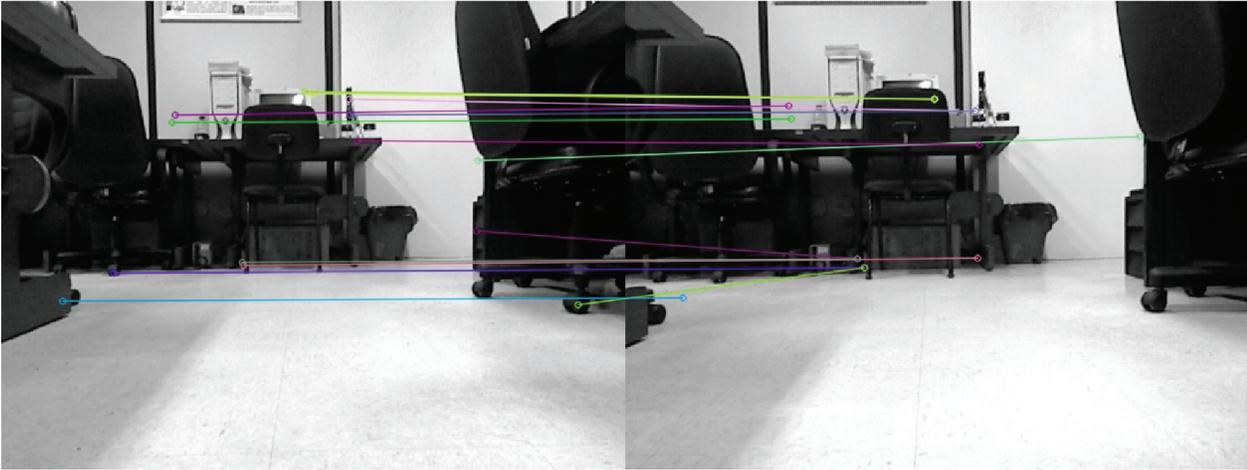


Figura 4.6: Corrêspndências com SURF

O objetivo é determinar um conjunto de *inliers* das correspondências através da restrição epipolar. Na geometria epipolar relaciona-se um par de correspondências mediante a matriz fundamental, para isso tem-se $\tilde{m} = \begin{bmatrix} x & y & 1 \end{bmatrix}^T$, que é um ponto sobre o plano imagem I e $\tilde{m}' = \begin{bmatrix} x' & y' & 1 \end{bmatrix}^T$ que é um ponto sobre o plano imagem I' , então a matriz define-se como:

$$\tilde{m}^T F \tilde{m}' = 0 \quad (4.7)$$

O algoritmo faz a busca de um conjunto de pontos para a estimação linear da matriz fundamental, calculando o número de verdadeiras correlações para cada matriz e escolhe a matriz que maximiza esse número.

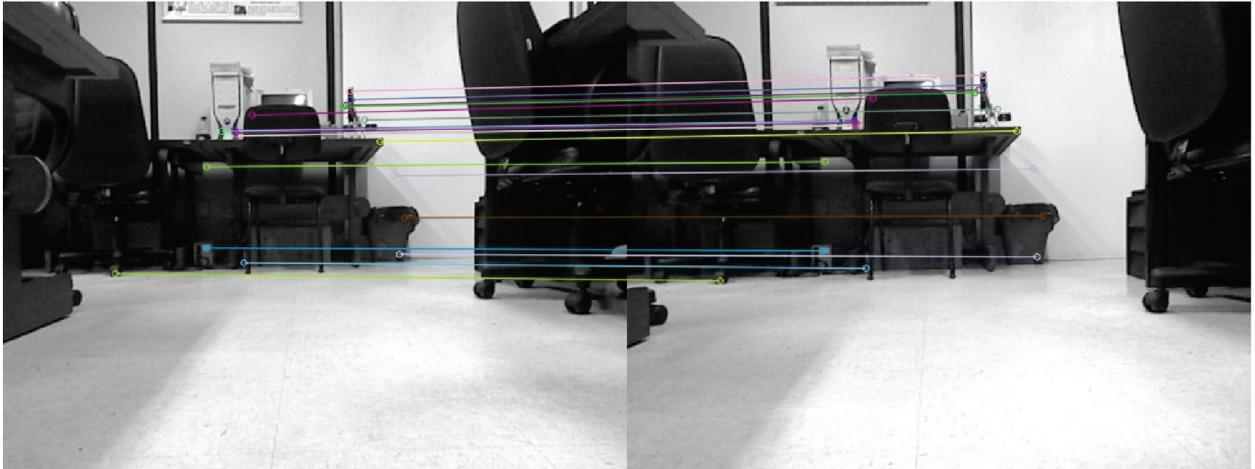


Figura 4.7: Correspondências Robustas com Ransac

O algoritmo é composto por 5 etapas, resumidas a seguir:

1. **Detecção de pontos de interesse e descritores:** Nessa etapa são extraídos e descritos os pontos-chaves das duas imagens usando o método SURF.
2. **Encontrar correspondências:** Para cada ponto de interesse achado na etapa anterior são achados possíveis pontos de correspondência de uma imagem para outra. O método utilizado nessa etapa é o classificador k-Nearest Neighbor, com um valor de $k = 2$, isso faz com que se tenham para um ponto-chave no Frame t dois possíveis pontos no frame $t + 1$.
3. **Filtragem de correspondências pelo Limiar:** Não obstante, existe a probabilidade com que os dois pontos-chave possuam uma distância muito próxima, para isto é feito um teste baseado no limiar, dessa forma conseguem-se eliminar a ambiguidade que há entre pontos.
4. **Remoção de correspondências assimétricas:** Como na etapa anterior foram filtrados os pontos baseados no limiar e a relação de distâncias em cada frame, agora é aplicado um teste para garantir que cada par de pontos, sobreviventes da etapa anterior, sejam mutuamente correspondente.
5. **Validação com o RANSAC:** Agora que se tem um conjunto de pontos robusto, faz-se o cálculo da matriz fundamental, com o objetivo de rejeitar correspondências que não estão sujeitas à restrição epipolar.

4.1.3 Estimação de movimento

O maior problema que existe na odometria visual é determinar a transformação rígida que relaciona dois conjuntos (nuvens) de pontos; lembrando que esses conjuntos de pontos são o resultado das correspondências que são observadas em frames subsequentes ao longo do percurso. Com base no pressuposto de que há um conjunto de pontos correto que relaciona dois estados diferentes de uma mesma cena, pode-se estabelecer uma transformação, equação 4.8.

$$X_B = R_q X_A + t \quad (4.8)$$

O algoritmo (KRAMER *e outros*, 2012) para encontrar a transformação ótima é composto por três etapas.

1. Calcular os centroides das nuvens de pontos X_{mean} e Y_{mean} .
2. Levar o conjunto de pontos para a origem de coordenadas a fim de achar a matriz de rotação R .
3. Fazer uma combinação linear com matriz de rotação R_q e o centroides X_{mean} e Y_{mean} .

O primeiro passo é achar a matriz de rotação R_q . Para isso, são calculados os centroides da nuvem de pontos, equações 4.9 e 4.10, que é a média aritmética de cada um dos conjuntos,

$$\bar{X}_A = \frac{1}{N} \sum_{i=1}^N X_A^i \quad (4.9)$$

$$\bar{X}_B = \frac{1}{N} \sum_{i=1}^N X_B^i \quad (4.10)$$

onde $X_t = \begin{bmatrix} x & y & z \end{bmatrix}^T$, estão em coordenadas não-homogêneas. Após calculados os centroides são levados as nuvens de pontos para a origem de coordenadas. Uma vez tendo computado esses

valores pode-se determinar a matriz de Rotação R , equação 4.12, que baseia-se na decomposição SVD da matriz H , equação 4.11.

$$H = \sum_{i=1}^N (X_A^i - \bar{X}_A) (X_B^i - \bar{X}_B)^T \quad (4.11)$$

$$R_q = VU^T \quad (4.12)$$

Assim, pode-se estipular o par $(R_q|t)$, que, para efeitos de simplicidade, será chamado de \hat{T} ; Na equação 4.13, tem-se a transformação que produz o alinhamento de duas nuvens de pontos subsequentes.

$$\hat{T} = \begin{bmatrix} I & -\bar{X}_A \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I & \bar{X}_B \\ 0 & 1 \end{bmatrix} \quad (4.13)$$

Esse método, apresentado no algoritmo 4.1.1, é simples para fazer a primeira aproximação da transformação que existe entre dois conjuntos de dados. Essa transformação pode ser calculada com um mínimo de três (3) pontos, baseando-se na definição do plano. No entanto, esse método apresenta uma solução teórica, onde é pressuposto que X_B é uma versão perfeitamente transformada (rotação e translação) de X_A . Essa suposição não é muito aceita, até porque tem-se fontes de

ruído, como perturbações elétricas do sensor ou correspondências erradas entre frames.

Algorithm 4.1.1: Algoritmo determinação do movimento

Entrada: Nuvem de Pontos X_A e X_B

Saída : Matriz de transformação rígida $[R \mid t]$

for $i \leftarrow 1$ **to** N **do**

$$\left[\begin{array}{l} \bar{X}_A \leftarrow \bar{X}_A + X_A^i \\ \bar{X}_B \leftarrow \bar{X}_B + X_B^i \end{array} \right.$$

for $i \leftarrow 1$ **to** N **do**

$$\left[\begin{array}{l} X_A^i \leftarrow X_A^i - \bar{X}_A \\ X_B^i \leftarrow X_B^i - \bar{X}_B \end{array} \right.$$

$$H \leftarrow X_B^T \times X_A$$

$$[U, S, V] \leftarrow \text{SVD}(H)$$

$$R_q \leftarrow V \times U^T$$

$$t \leftarrow \bar{X}_A - R_q \times \bar{X}_B$$

O Ransac é um método iterativo para determinar os parâmetros de um modelo matemático a partir de um conjunto de observações que contém ruído. Nesse caso, tem-se a matriz de transformação rígida \hat{T} estimada em k iterações. A classificação (*inliers/outliers*) é feita em cada iteração, em que são considerados como inliers aquelas correspondências 3D $X_A \longleftrightarrow X_B$ que estão sujeitas ao limiar entre a distância de X_B e $\hat{T}X_A$, como mostrado no algoritmo 4.1.2. O objetivo é maximizar a probabilidade que uma das soluções calculadas seja uma solução real, para isso são usados um número mínimo de pontos no cálculo de cada uma delas. Nesse caso são necessárias três correspondências, portanto, em cada iteração é calculada uma transformação a partir de três

correspondências aleatórias. Fora do limiar são considerados como outliers.

Algorithm 4.1.2: Algoritmo determinação do movimento robusto com RANSAC

Entrada: Nuvem de Pontos X_A e X_B , Número de iterações k e Limiar de erro para inliers

limiar

Saída : Matriz de transformação rígida ótima $[R_q | t]$

Aux : Correspondências verdadeiras P^k e coeficientes de erro cf

for $i \leftarrow 1$ **to** k **do**

for $j \leftarrow 1$ **to** 3 **do**

$id \leftarrow \text{RNG}(N)$

$\hat{X}_A^k \leftarrow X_A^{id}$

$\hat{X}_B^k \leftarrow X_B^{id}$

 Pose-Estimation ($\hat{X}_A^k, \hat{X}_B^k, R^k, t^k$)

for $n \leftarrow 1$ **to** N **do**

$p_2 \leftarrow X_B^n$

$p_1 \leftarrow R^k p_2 + t^k$

$cf \leftarrow p_1 - X_A^n$

$error \leftarrow \text{sqrt}(cf^2)$

if $error \leq \text{limiar}$ **then**

$P^k \leftarrow \text{Inliers}$

for $m \leftarrow 1$ **to** Inliers **do**

$index \leftarrow \text{Inliers}[m]$

$\hat{X}_A^m \leftarrow X_A^{id}$

$\hat{X}_B^m \leftarrow X_B^{id}$

 Pose-Estimation ($\hat{X}_A^k, \hat{X}_B^k, R_q, t$)

Agora, determina-se a matriz de transformação homogênea (equação 4.14) entre o sistema de coordenadas da câmera numa observação k e o sistema de coordenadas de câmera numa observação $k + 1$.

$$T = \begin{bmatrix} R_q & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) & t_x \\ 0 & 1 & 0 & 0 \\ \sin(\theta) & 0 & \cos(\theta) & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.14)$$

A posição e orientação atual, ou instante k , do veículo é definida pela equação 5.5 a partir da referência do sistema global de coordenadas.

$$X_{k+1} = \prod_{k=0}^N R_{qk} \cdot X_k + t_k \quad (4.15)$$

Por outro lado, considerando a posição e orientação do veículo baseada na odometria visual, que é um modelo incremental, precisa-se inicializar o sistema de coordenadas da câmera na posição $X_{gps}^0 = [x_{gps}^0 \quad y_{gps}^0 \quad \theta_{gps}^0]^T$, que neste caso corresponde à primeira observação do sistema global de coordenadas ou posição onde é inserido o veículo dado pelo sistema de posicionamento global. A matriz 4.16 apresenta a primeira transformação do sistema de odometria visual para representar o modelo consistente de integração desde a posição.

$$T_o^0 = \begin{bmatrix} \cos(\theta_{gps}^0) & 0 & -\sin(\theta_{gps}^0) & x_{gps}^0 \\ 0 & 1 & 0 & 0 \\ \sin(\theta_{gps}^0) & 0 & \cos(\theta_{gps}^0) & z_{gps}^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.16)$$

4.2 Odometria (*Dead-Reckoning*)

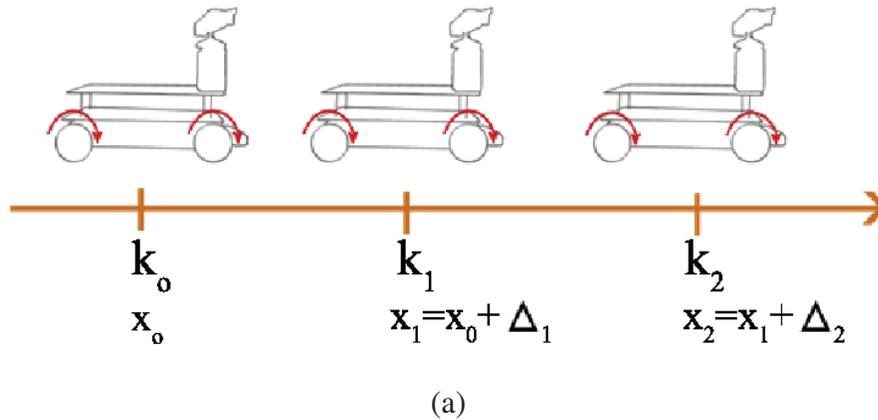


Figura 4.8: Odometria do veículo (*dead-reckoning*)

O modelo de odometria consiste em encontrar a representação matemática da integração (figura 4.8) dos deslocamentos. Na navegação autônoma, o termo *dead-reckoning* refere-se à estimação da posição (e orientação) através da observação do movimento das rodas e o ângulo de direção em certo intervalo de tempo. Como foi definido no capítulo 3, e baseado no trabalho de Bonnifait e outros (2001), a odometria convencional está definida pela medição de 5 sensores embarcados no veículo. As informações representadas por esses sensores representam o deslocamento de cada uma das rodas do veículo e o ângulo medido do mecanismo de esterçamento. Lembrando que o conjunto de equações que relacionam as variáveis medidas (Capítulo 3, 3.1.2) tem-se:

$$\begin{aligned}
 \tan(\omega) &= L \frac{\phi}{\Delta} \\
 \Delta_{RL} &= \Delta - \phi \frac{W}{2} \\
 \Delta_{RR} &= \Delta + \phi \frac{W}{2} \\
 \Delta_{FL} \cos(\omega_L) &= \Delta - \phi \frac{W}{2} \\
 \Delta_{FR} \cos(\omega_R) &= \Delta + \phi \frac{W}{2}
 \end{aligned} \tag{4.17}$$

No modelo de odometria antes mencionado pode-se estabelecer um vetor de medição ou observação baseado nos sensores disponíveis. Então, da equação 4.17 e arranjando-os num vetor de medição, tem-se:

$$Z = \left[\tan(\omega) \quad \Delta_{RL} \quad \Delta_{RR} \quad \Delta_{FL} \cdot \cos(\omega_L) \quad \Delta_{FR} \cdot \cos(\omega_R) \right] \quad (4.18)$$

Com objetivo de criar um modelo consistente que funcione sob situações além de ideais, o vetor $x = \left[\Delta \quad \omega \right]^T$ é calculado usando todas as medições. A proposta neste trabalho é estimar o vetor x usando o filtro de *Kalman*. Devido à não linearidade da equação de observação (4.18), é necessário usar a versão estendida do filtro.

Um resumo das variáveis do modelo *dead-reckonig* adotado para à odometria estão mostrados na tabela 4.2.

Entradas	ω	ângulo de esterçamento
	Δ_{RL}	Δ roda traseira esquerda
	Δ_{RR}	Δ roda traseira direita
	Δ_{FL}	Δ roda dianteira esquerda
	Δ_{FR}	Δ roda dianteira direita
Saídas	Δ	Δ total do veículo
	$\arctan\left(L \cdot \frac{\phi}{\Delta}\right)$	ângulo de esterçamento
Estados	$\Delta + \gamma$	Δ
	$\arctan\left(L \cdot \frac{\phi}{\Delta}\right) + \gamma$	ângulo de esterçamento

Tabela 4.2: Descrição das variáveis do modelo de odometria

Neste estudo de caso, no qual somente é usado o modelo odométrico, a variação do vetor de estado pode ser modelado como um processo estocástico (4.19) cuja entrada é um ruído gaussiano com média zero γ_k .

$$x_{k+1} = x_k + \gamma_k \quad (4.19)$$

Além do estado, associa-se ao sistema, a covariância da estimação do erro P_k :

$$P_{k+1} = P_k + \gamma_k \quad (4.20)$$

Além disso, é necessário definir o modelo de observação em função do sistema. Como o modelo de medição (equação 4.17) é não linear, a matriz H (Jacobiano - equação 4.21) precisa ser calculada a fim de linearizá-lo. É importante notar que esse modelo só é usado se tiver um deslocamento maior que zero $\Delta > 0$, ou uma velocidade suficientemente alta.

$$H = \begin{bmatrix} \frac{\delta Z}{\delta x} \end{bmatrix} = \begin{bmatrix} -L\frac{\omega}{\Delta} & \frac{L}{\Delta} \\ 1 & -\frac{W}{2} \\ 1 & +\frac{W}{2} \\ 1 & -\frac{W}{2} \\ 1 & +\frac{W}{2} \end{bmatrix} \quad (4.21)$$

Como as medições relacionam-se com o vetor de estado de forma não-linear, onde o filtro apropriado é o filtro estendido de kalman (*EKF*), como no trabalho apresentado por (BENTO e outros, 2005); requer de um modelo expresso da seguinte forma:

$$\begin{aligned} x_{k+1} &= \Phi_k x_k + \Gamma_k u_k \\ y_k &= H_k x_k + D u_k \end{aligned} \quad (4.22)$$

Assim, teria-se o algoritmo 4.2.1, cuja função é a de estimar tanto o deslocamento total do veículo como o ângulo de esterçamento de forma consistente, baseado no modelo cinemático.

A figura 4.9 apresenta a reconstrução de um percurso usando odometria convencional, em

Algorithm 4.2.1: Algoritmo Filtro de Kalman Fusão Encoder

Entrada: Observações de sensores $\Delta_{RL}, \Delta_{RR}, \Delta_{FL}, \Delta_{FR}$ e Ψ

Saída : Vetor de odometria $x = [\Delta \ \omega]^T$

for $k \leftarrow 3$ **to** $nMeasurements$ **do**

$$\begin{cases} \bar{x}_{k+1} = x_k + \gamma_k \\ \bar{P}_{k+1} = P_k + \gamma_k \\ y_k = Z_k - H_k x_k \\ S_k = H_k P_k H_k^T + R \\ K_k = P_k H_k^T S_k^{-1} \\ x_k = \bar{x}_k + K_k y_k \\ P_k = (I - K_k H_k) P_k \end{cases}$$

que o *ground truth* é representado pela linha azul e em cima dessa uma caixa de cor preta mostra o estado real do veículo enquanto o percurso reconstruído usando o método *dead-reckoning* é mostrado como a caixa de cor vermelha.

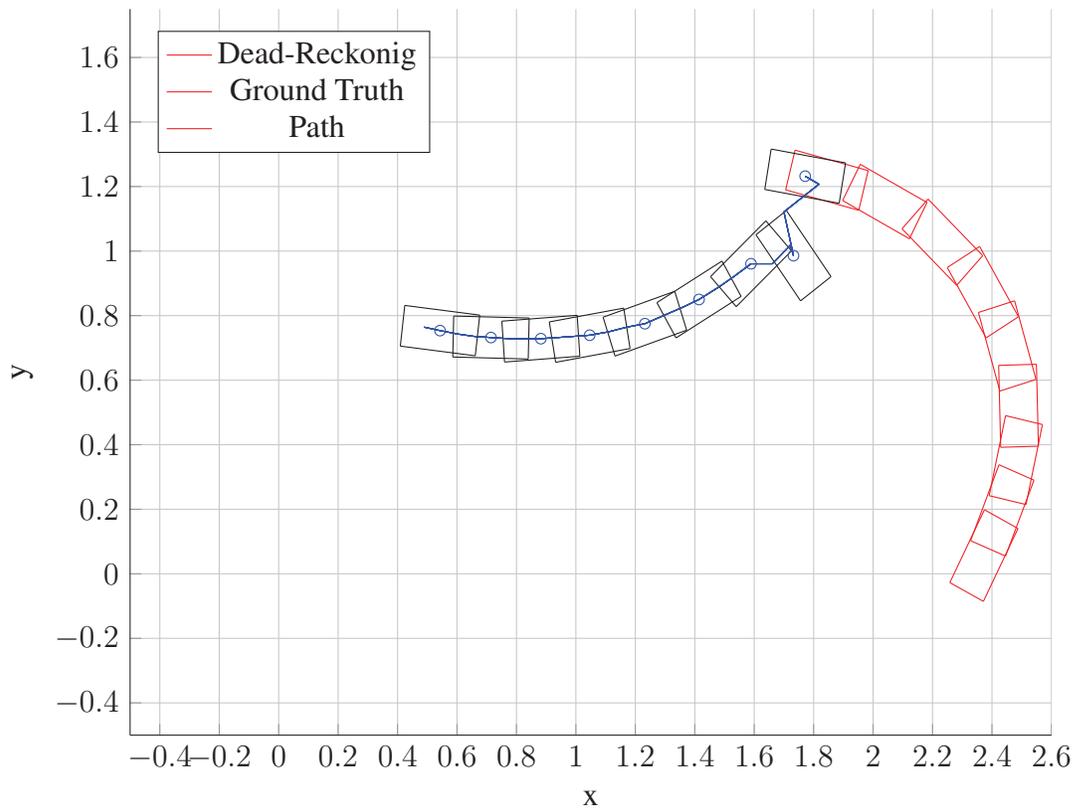


Figura 4.9: Odometria do veículo Deslocamento Incremental

O modelo de odometria de *dead-reckoning* é o processo para calcular a posição atual do veículo usando como referência uma posição anterior conhecida, porém está sujeita à acumulação de erro. Então, faz-se necessária uma correção da posição baseado num método global de observação.

4.3 Localização

O problema de localização é definido como a estimação do estado do veículo $x_k = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$, usando informação dos sensores tanto proprioceptivos quanto exteroceptivo. Como este trabalho foi desenvolvido em ambientes *indoor*, criou-se um sistema de posicionamento local com objetivo de fornecer dados similares aos entregues por um GPS real. Informação do funcionamento do simulador de GPS encontra-se no apêndice D.

Como os métodos de odometria discutidos antes possuem características diferentes, uma observação global parece razoável afim de corrigir e obter uma melhor estimação do estado do veículo. Baseado no modelo probabilísticos visto no capítulo 3 e usando o filtro de Kalman estendido pode ser feita uma fusão de sensores como encoder nas rodas, sensor do ângulo de esterçamento e GPS. Dado que este é um trabalho que visa à observação do veículo autônomo, o vetor de entrada desta fase provêm do estágio anterior, cujo vetor de controle é a velocidade $v = \frac{d\Delta}{dt}$ e o ângulo de esterçamento ω estimado.

O modelo discretizado do veículo (Capitulo 3) pode-se estender para ser usado com outras variáveis importantes do GPS como o *bias*. A tabela 4.3 abaixo sumariza os estados do filtro de localização.

Assim, teriam-se as matrizes

$$\Phi = \begin{bmatrix} 1 & 0 & -dt * \frac{d\Delta}{dt} * \sin(\theta) & 0 & 0 & 0 \\ 0 & 1 & dt * \frac{d\Delta}{dt} * \cos(\theta) & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.23)$$

Entradas	ω Δ	ângulo de esterçamento deslocamento do veículo
Saídas	x y θ	componente X da posição componente Y da posição orientação θ
Estados	x y θ x_B y_B θ_B	componente X da posição componente Y da posição orientação θ componente X do bias componente Y do bias componente θ do bias

Tabela 4.3: Descrição das variáveis do modelo de localização

$$\Gamma = \begin{bmatrix} dt * \cos(\theta) & 0 \\ dt * \sin(\theta) & 0 \\ dt * \frac{\tan(\omega)}{L} & dt * \frac{d\Delta}{dt} * \sec^2(\omega) \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (4.24)$$

A matriz de medição H_k fica então:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \quad (4.25)$$

O algoritmo como proposto por Weinstein e Moore (2010) pode ser visto no algoritmo 4.3.1.

Algorithm 4.3.1: Algoritmo Filtro de Kalman GPS

for $k \leftarrow 1$ **to** $nSteps$ **do**

$$\bar{x}_{k+1} = \Phi_k x_k + \Gamma_k \mu_k$$

$$\bar{P}_{k+1} = \Phi_k P_k \Phi_k^T + \Gamma_k Q \Gamma_k^T$$

$$y_k = Z_k - H_k x_k$$

$$S_k = H_k P_k H_k^T + R$$

$$K_k = P_k H_k^T S_k^{-1}$$

$$x_k = \bar{x}_k + K_k y_k$$

$$P_k = (I - K_k H_k) \bar{P}_k$$

5 EXPERIMENTOS E RESULTADOS

Os experimentos que foram conduzidos, a fim de estudar uma solução ao problema de localização, são apresentados neste capítulo. A arquitetura usada na solução do problema é a mencionada no capítulo 1, seção 1.2.2; seguindo esse esquema, a trajetória de referência é obtida por um sistema de visão computacional, que também serve como base do emulador de GPS (detalhes do emulador de GPS no anexo D). Devido a ausência de módulo de controle e planejador de trajetória, fez-se o percurso de forma manual usando o controle remoto do veículo em escala.

O modelo da solução de localização mostra-se na figura B.1 em que, tem-se uma camada de aquisição, uma de odometria e, finalmente, uma de localização. Isso, é possível, usando a concatenação de diferentes filtros de fusão de sensores.

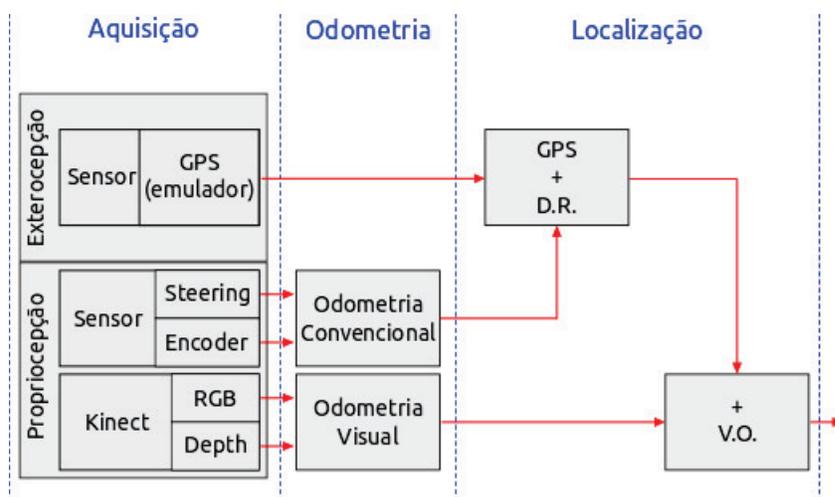


Figura 5.1: Modelo da solução de localização

5.1 Aquisição

O sistema de aquisição é formado por um conjunto de sensores (exteroceptivos e proprioceptivos), e uma placa de aquisição e medição de dados, que interage com um computador embarcado no próprio veículo. O *hardware*, neste caso o Arduino, coleta amostras periódicas dos sensores do veículo em uma taxa de amostragem fixa de $300ms$.

5.2 Odometria

5.2.1 Odometria convencional (*Dead-Reckoning*)

Como foi mencionado no capítulo 4, neste trabalho usou-se o modelo de odometria desenvolvido por Bonnifait *e outros* (2001). O bloco que representa o módulo de odometria convencional (*dead-reckoning*) apresenta-se na figura 5.2



Figura 5.2: Bloco de odometria convencional

O objetivo é usar a medição redundante, tanto do deslocamento da rodas, quanto do ângulo de esterçamento, para criar um modelo matemático consistente; assim o estado do sistema de odometria convencional está definido por:

$$x = \begin{bmatrix} \Delta \\ \omega \end{bmatrix} \quad (5.1)$$

Da seção 2.2 infere-se que a única matriz de covariância a ser encontrada pertence à covariância dos sensores; isso deve-se ao fato que o sistema seja modelado como um processo estocástico; até agora percebe-se que a fusão é obtida usando mínimos quadrados ponderados. O ajuste da matriz R foi realizado baseado na definição de covariância para um conjunto de medições, da definição formal, isto é,

$$\sigma^2 = \frac{1}{M} \sum_{i=1}^M (Z_i - \bar{Z}) \cdot (Z_i - \bar{Z})^T \quad (5.2)$$

De forma experimental, fazendo um mesmo percurso de forma repetitiva, obtêm-se um con-

junto medições que representam de forma estatística a covariância de cada sensor. Assim, aplicando a equação 5.2 no desvio padrão para esse conjunto de medições (Z_k) obtém-se a matriz R da equação 3.39,

$$R = \begin{bmatrix} \sigma_{\omega} & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\Delta_{RR}} & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\Delta_{RL}} & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\Delta_{FR}} & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\Delta_{FL}} \end{bmatrix} = \begin{bmatrix} 0.002 & 0 & 0 & 0 & 0 \\ 0 & 0.004 & 0 & 0 & 0 \\ 0 & 0 & 0.0035 & 0 & 0 \\ 0 & 0 & 0 & 0.0032 & 0 \\ 0 & 0 & 0 & 0 & 0.0037 \end{bmatrix} \quad (5.3)$$

5.2.2 Odometria convencional: primeira experiência

Para avaliar o filtro de odometria foi realizado um percurso curvilíneo, que está apresentado na figura 5.3.

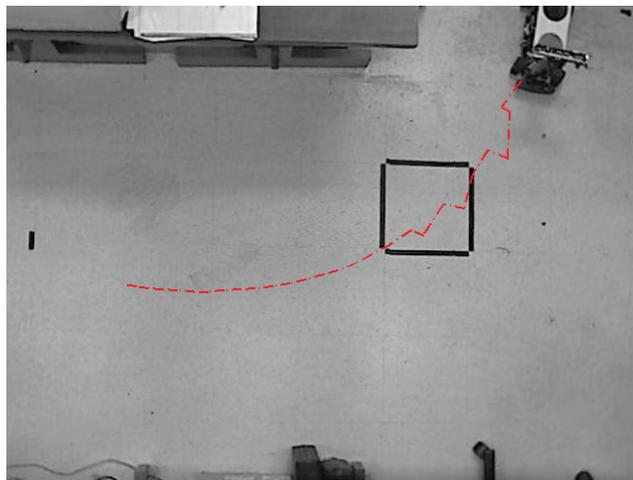


Figura 5.3: Trajetória de referência odometria convencional

Usando o algoritmo 4.2.1, obtém-se o resultado da odometria convencional mostrado na figura 5.4.

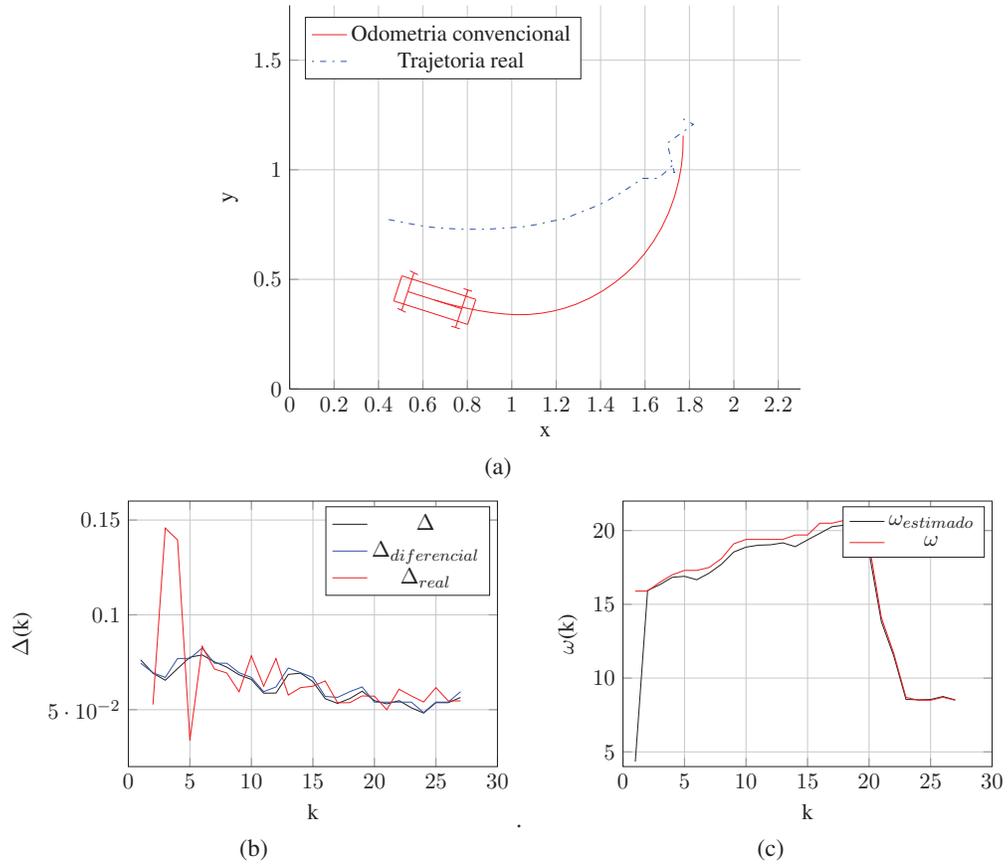


Figura 5.4: Resultado odometria convencional: (a) deslocamento incremental; (b) estimativa do deslocamento total; e, (c) estimativa do ângulo de esterçamento.

Na figura (5.4(a)) pode-se observar que o resultado da localização para o veículo, usando só a odometria, não é satisfatório porque não atinge a trajetória real. A posição inicial usada pertence à primeira observação do emulador de GPS; é nessa posição inicial que começa o processo de integração da odometria convencional.

Os gráficos (figuras 5.4(b)-(c)) mostram a resposta da estimação de cada uma das variáveis de odometria; essa resposta está dentro da expectativa do filtro, já que a função do filtro de kalman, usado na odometria convencional, é a de ser um método de mínimos quadrados ponderado para o deslocamento e o ângulo de esterçamento. A figura 5.5 mostra o resultado da estimação dentro dos limites de confiança, a envoltória de valor $\pm 3\sigma$.

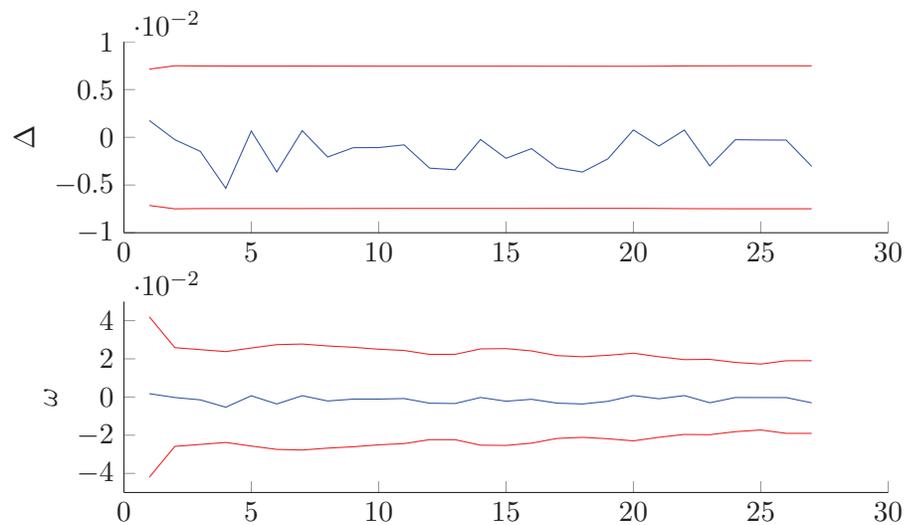


Figura 5.5: Comparação do erro com o limite de confiança $\pm 3\sigma$

A inevitável acumulação de erro produz uma incerteza na posição e orientação como é mostrado na figura 5.4. No gráfico 5.6 apresenta-se, de forma quantitativa o erro acumulado durante o percurso, em que k é o número de iterações.

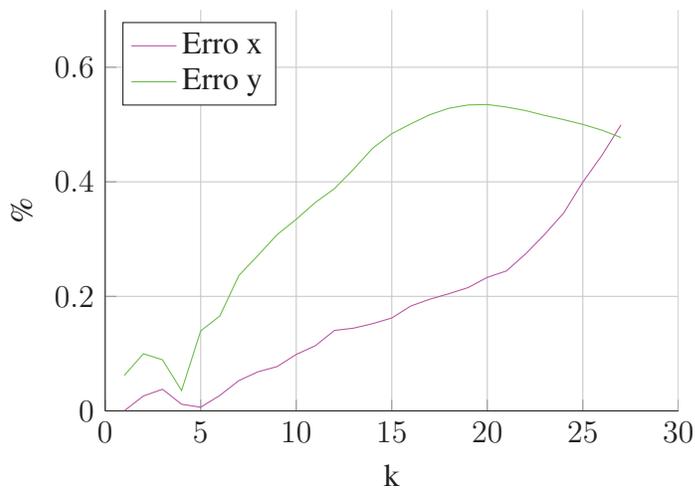


Figura 5.6: Erro acumulado odometria convencional

5.2.3 Odometria convencional: segunda experiência

Um segundo experimento foi realizado com a finalidade de observar a divergência da resposta da odometria convencional com outro percurso, lembrando que a posição inicial é a partir da primeira observação do GPS, neste caso, a primeira observação do emulador de GPS. O novo percurso é apresentado na figura 5.7.

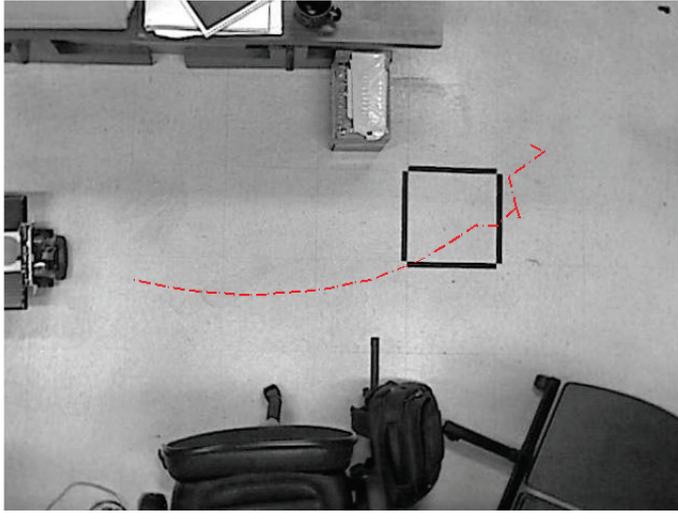
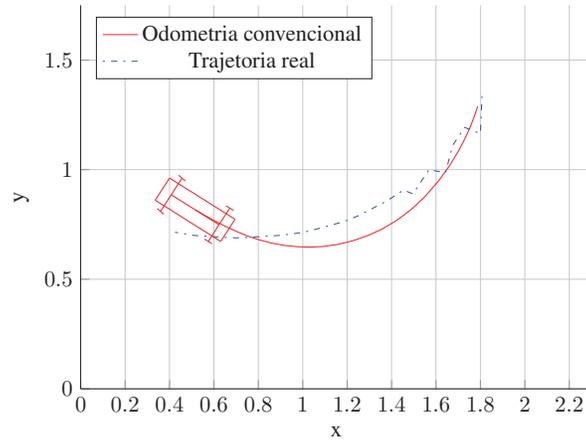
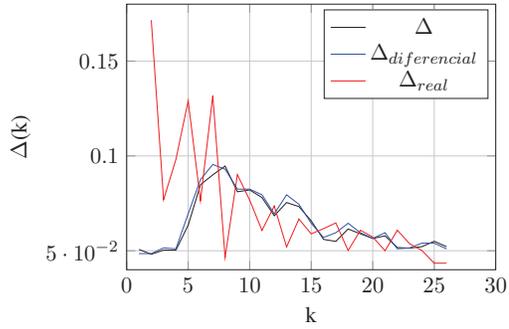


Figura 5.7: Trajetória de referência odometria convencional

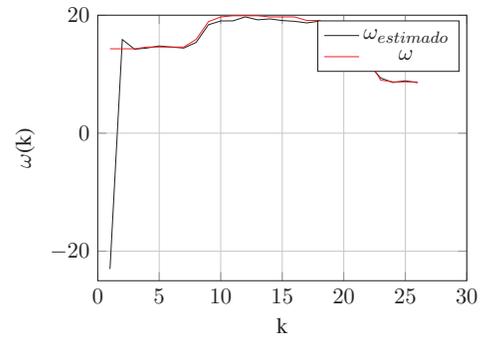
A convergência do método depende da quantidade de medições feitas, que está relacionado com a taxa de amostragem, que neste caso é particularmente baixa, mesmo sendo para este modelo de baixa velocidade. Como consequência, o modelo de integração contemplado na odometria, é suscetível à acumulação de erro. No gráfico (figura 5.24) observa-se a estimativa feita para o percurso, usando odometria convencional. A estimação dos parâmetros apresantem-se nas figuras 5.8(b)-(c).



(a)



(b)



(c)

Figura 5.8: Resultado odometria convencional: (a) deslocamento incremental; (b) estimativa do deslocamento total; e, (c) estimativa do ângulo de esterçamento.

A figura 5.9 mostra o resultado da estimação dentro dos limites de confiança, a envolvente de valor $\pm 3\sigma$.

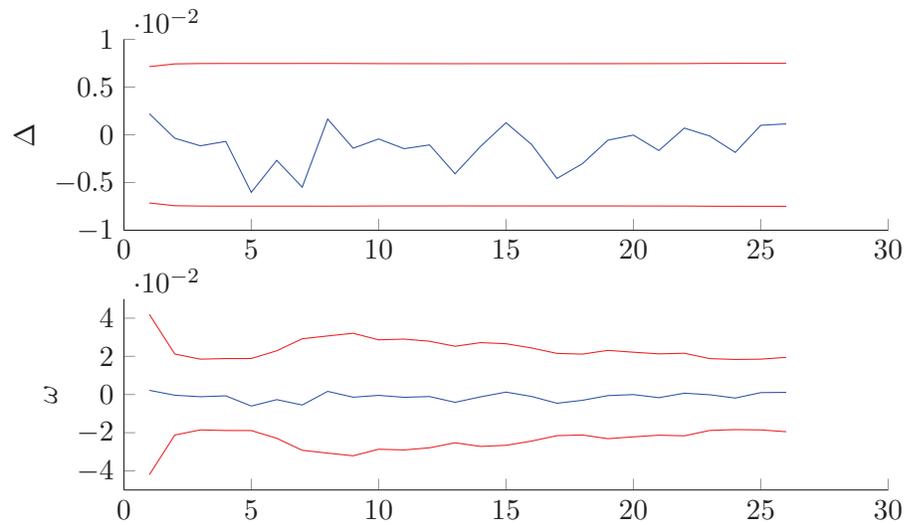


Figura 5.9: Comparação do erro com o limite de confiança $\pm 3\sigma$

A acumulação de erro é mostrada na figura 5.10.

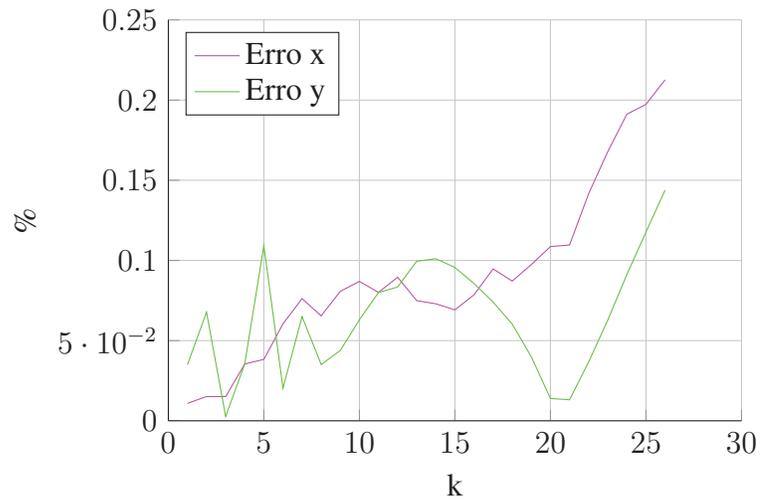


Figura 5.10: Erro acumulativo odometria convencional

5.2.4 Odometria Visual

A única condição necessária para que se possa calcular a pose de um veículo, baseado em visão, é se encontrar em um ambiente que contenha diversos objetos (riqueza de características); dessa forma, garante-se um número apropriado de marcos visuais. A arquitetura projetada e implementada para se conseguir odometria visual é mostrada na figura 5.11.

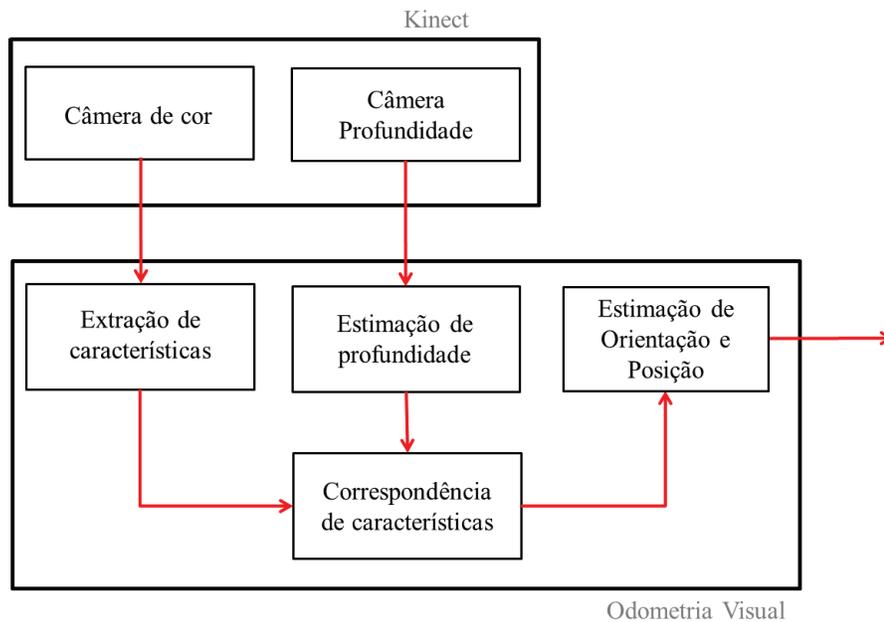


Figura 5.11: Bloco de odometria visual

Para avaliar a odometria visual como método de localização foi criado um percurso curvilíneo. O objetivo é comparar esse método com a trajetória real, amostrada neste caso pelo sensor emulador de *GPS*. Para garantir um deslocamento uniforme entre as amostras, usa-se velocidade constante durante todo o percurso; assim, foi realizado um percurso com nove medições ao longo desse. O percurso é mostrado na figura 5.12. Devido ao tempo que leva a captura da informação visual, pelo *kinect*, a taxa de amostragem escolhida foi de $800ms$.

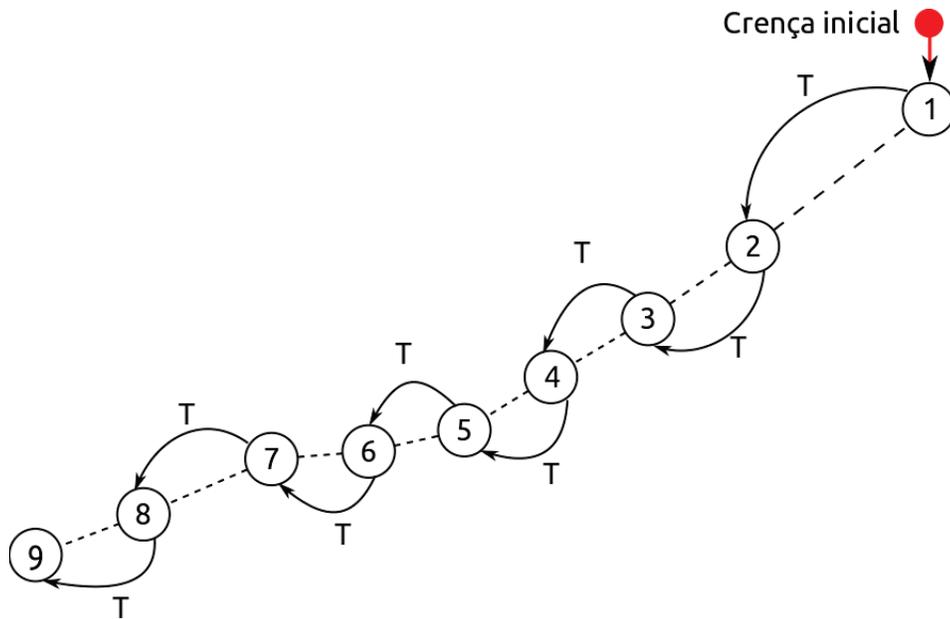


Figura 5.12: Posição aproximada da câmera na trajetória de teste

A trajetória de teste é apresentada na figura 5.13. A figura 5.13(a) mostra o percurso realizado no mapa real. A figura 5.13(b) mostra a trajetória em relação ao referencial global (x, y) .

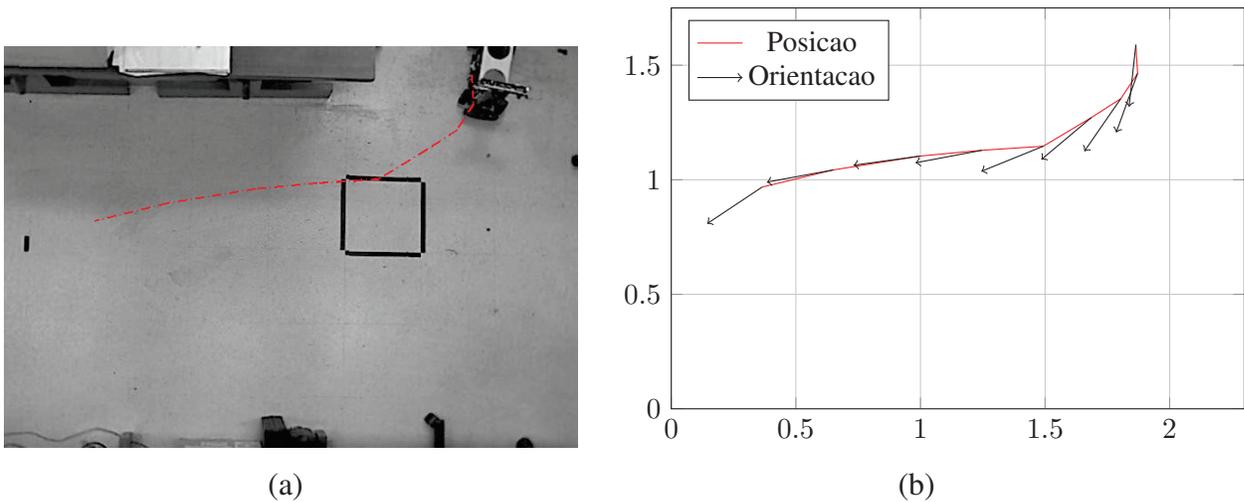


Figura 5.13: Observação do simulador de GPS: (a) trajetória no mapa; e, (b) coordenadas da trajetória.

Deteção de características

A extração de características, usando o método apresentado no capítulo 4, faz-se para cada uma das imagens capturadas no percurso. Na figura 5.14 são apresentadas algumas imagens que fazem parte da trajetória.

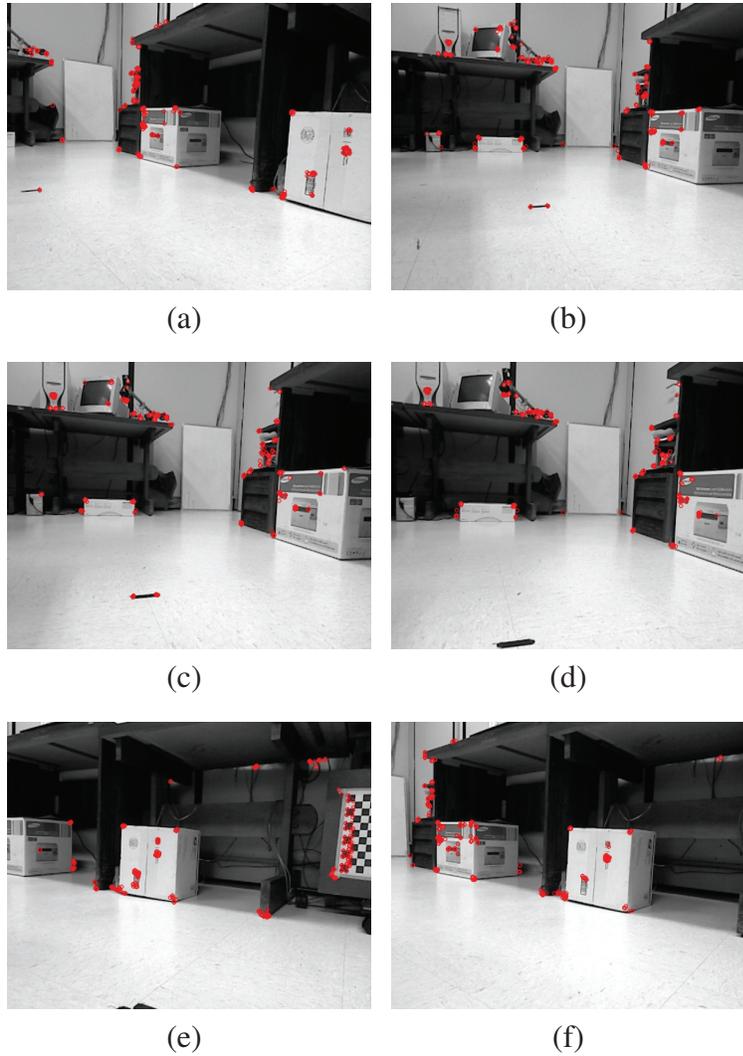
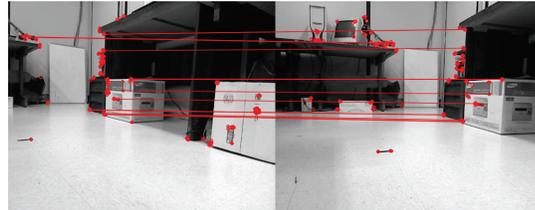


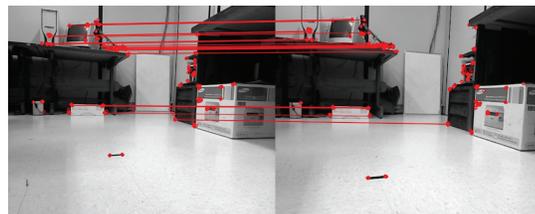
Figura 5.14: Sequência de imagens da câmera no percurso. A ordem de deslocamento é da esquerda para direita e de cima para baixo: (a) 1; (b) 2; (c) 3; (d) 4; (e) 5; e, (f) 6.

Correspondência de características

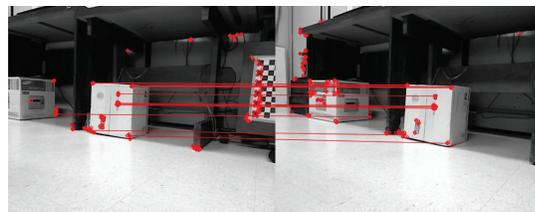
A correspondência entre características, usando o método apresentado no capítulo 4, visuais entre cada par de imagens (figura 5.14) é apresentada na figura 5.15.



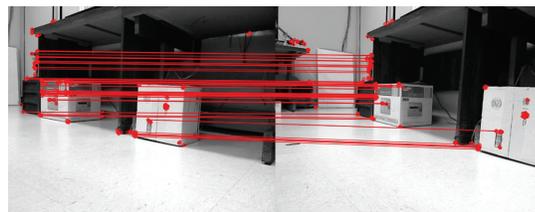
(a)



(b)



(c)



(d)

Figura 5.15: Sequência de imagens da câmera no percurso. A ordem de deslocamento é da esquerda para direita e de cima para baixo: (a) 1-2; (b) 2-3; (c) 3-4; (d) 4-5.

Determinar transformação rígida

Determinar a transformação rígida, ou seja encontrar a transformação T , é o objetivo da odometria visual. O esquemático da figura 5.16 representa essa transformação, onde estão indicados os eixos do sistema de coordenadas da câmera em relação ao referencial fixo.

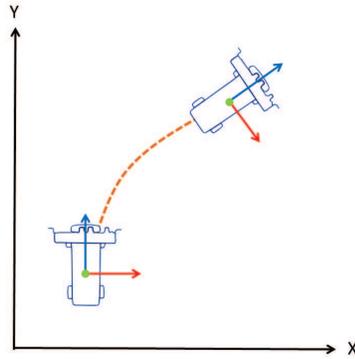


Figura 5.16: Esquema da odometria visual

Com o conjunto de transformações rígidas é feita a reconstrução da trajetória. A transformação rígida é representada pela seguinte equação:

$$T = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) & \Delta_x \\ 0 & 1 & 0 & 0 \\ \sin(\theta) & 0 & \cos(\theta) & \Delta_y \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.4)$$

Para reconstruir o percurso usa-se a regra cinemática mostrada na seguinte equação:

$$X_{k+1} = \prod_{k=0}^N R_k \cdot X_k + t_k, \quad (5.5)$$

onde R_k é a rotação relativa, e t_k é a translação relativa, para o vetor X_k .

A odometria visual, cujo objetivo é determinar a localização em um instante k com base

na acumulação das transformações anteriores, é resumizada pelo algoritmo 5.2.1. A primeira observação é colocada na posição da primeira observação do emulador de GPS, isto é, ${}^oX_{gps} = \begin{bmatrix} {}^o x_{gps} & {}^o y_{gps} & {}^o \theta_{gps} \end{bmatrix}^T$, que também é chamado de crença inicial.

Algorithm 5.2.1: Algoritmo de reconstrução da trajetória usando Odometria Visual

Entrada: Vetor de transformações $H_k \forall k \in N$

Saída : Vetor de estados (localização) $\begin{bmatrix} {}^k X_{visualod} & {}^k Y_{visualod} & {}^k \theta_{visualod} \end{bmatrix}^T$

Aux : matriz $\begin{bmatrix} R & | & t \end{bmatrix}$

$T = {}^o T_{gps}$

for $k \leftarrow 0$ **to** $nObservations - 1$ **do**

$\begin{bmatrix} R & | & t \end{bmatrix} = H_k$
 $rotation - vector = Rodrigues(R)$
 $T = T \cdot \begin{bmatrix} R & | & t \end{bmatrix}$
 $\dot{X}_k = \begin{bmatrix} t_x & t_z & R_y \end{bmatrix}$

Ao aplicar-se o algoritmo anterior, como mostrado na figura 5.17, obtém-se uma reconstrução do percurso, em que, tem-se uma leve divergência.

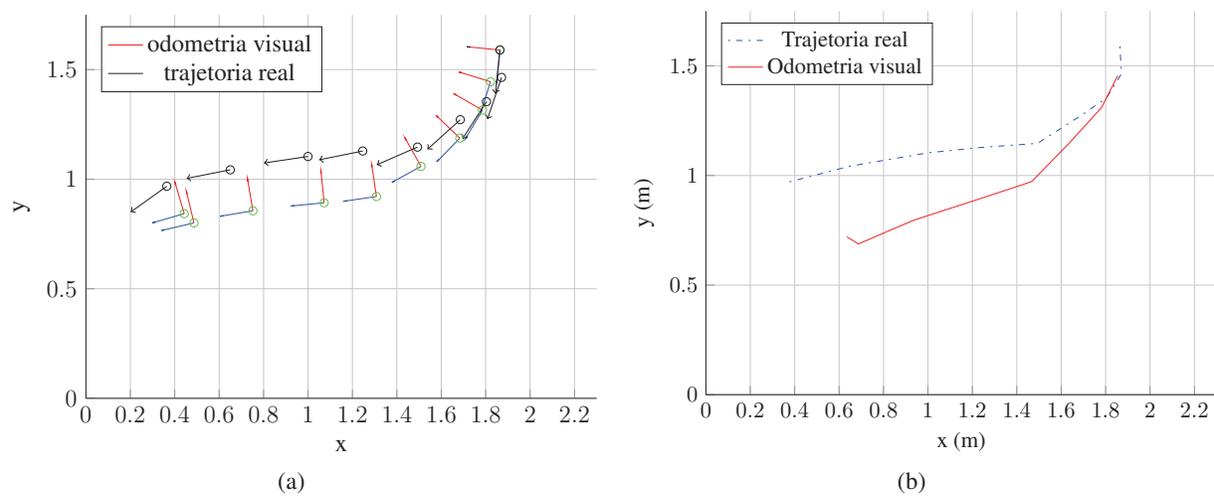


Figura 5.17: Resposta odometria visual: (a) sistema de coordenadas de câmara; e, (b) trajetória reconstruída.

A figura 5.17(a) representa o sistema de coordenadas da câmera em cada iteração em relação ao vetor de localização do simulador de GPS (seta e linha pretas); enquanto, na figura 5.17(b) pode-se observar que a trajetória reconstruída tem um erro acumulativo, que conforme for aumentando o número de iterações, vai aumentando a incerteza da posição e orientação; isto é mostrado em comparação à trajetória real. O erro é apresentado no gráfico da figura 5.18.

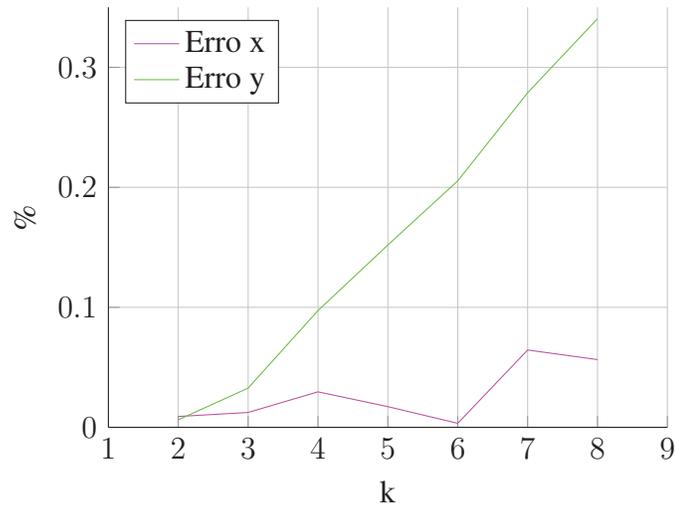


Figura 5.18: Erro acumulativo da odometria visual

5.3 Localização

O procedimento proposto para localizar o veículo de forma robusta, consiste em concatenar dois filtros de fusão. O primeiro deles pertence à fusão da odometria convencional com as observações do emulador do GPS. O segundo filtro, faz a fusão do vetor resultado do filtro anterior com o resultado da odometria visual. A seguir, apresenta-se a simulação do modelo de localização.

5.3.1 Modelo

A localização robusta define-se como a fusão de um método de localização relativa ou local, com um método de localização global. O modelo usado para fazer a fusão da odometria *dead-reckoning* usando as observações do GPS é mostrado na figura 5.19.

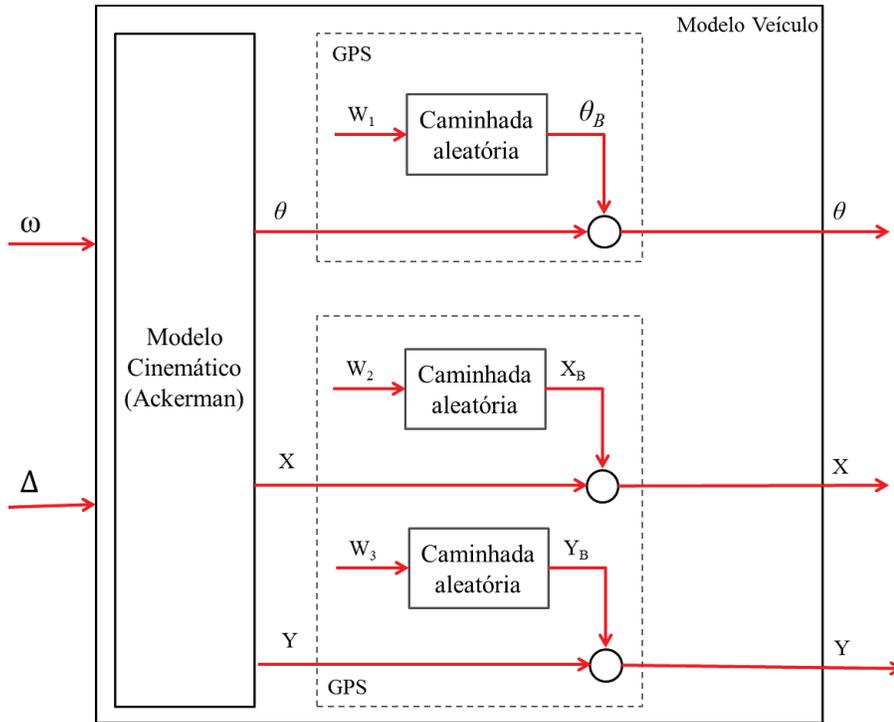


Figura 5.19: Modelo de localização

A simulação apresentada aqui envolve uma entrada de controle definida como,

$$u_k = \left[0.17 \quad \frac{\pi}{8} \sin \left(4 \frac{k\pi}{N} \right) \right]^T$$

em que, a velocidade é constante, e o um ângulo de esterçamento segue um comportamento senoidal. Usou-se um ruído de controle com desvio padrão de 0.1m/s para o valor da velocidade, e uma incerteza de 4° para a variável direção. Assim, a matriz de covariância (equação 5.6) do sistema é definida pelo desvio padrão de cada uma das variáveis do sistema. Isso, interpreta-se como o próprio erro do sistema, e neste caso, é um valor pequeno.

$$P = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix} = \begin{bmatrix} 0.1^2 & 0 & 0 \\ 0 & 0.1^2 & 0 \\ 0 & 0 & 0.1^2 \end{bmatrix} \quad (5.6)$$

Da mesma maneira, a equação 5.7 representa a covariância da entrada de controle com o ruído associado.

$$Q = \begin{bmatrix} \sigma_{\Delta}^2 & 0 \\ 0 & \sigma_{\omega}^2 \end{bmatrix} = \begin{bmatrix} 0.1^2 & 0 \\ 0 & 4\frac{\pi}{180}^2 \end{bmatrix} \quad (5.7)$$

Uma incerteza de medida, com desvio padrão, de $\sigma = 0.1m/s$ foi utilizado para as variáveis de posição, x e y e uma incerteza de $\sigma = 4^\circ$ para a orientação θ , sumarizados pela equação 5.8. Esta matriz é melhor caracterizada pelo apropriado modelo de GPS.

$$R = \begin{bmatrix} Z_x^2 & 0 & 0 \\ 0 & Z_y^2 & 0 \\ 0 & 0 & Z_\theta^2 \end{bmatrix} = \begin{bmatrix} 0.1^2 & 0 & 0 \\ 0 & 0.1^2 & 0 \\ 0 & 0 & 4^2 \end{bmatrix} \quad (5.8)$$

A figura 5.20 mostra o resultado da fusão.

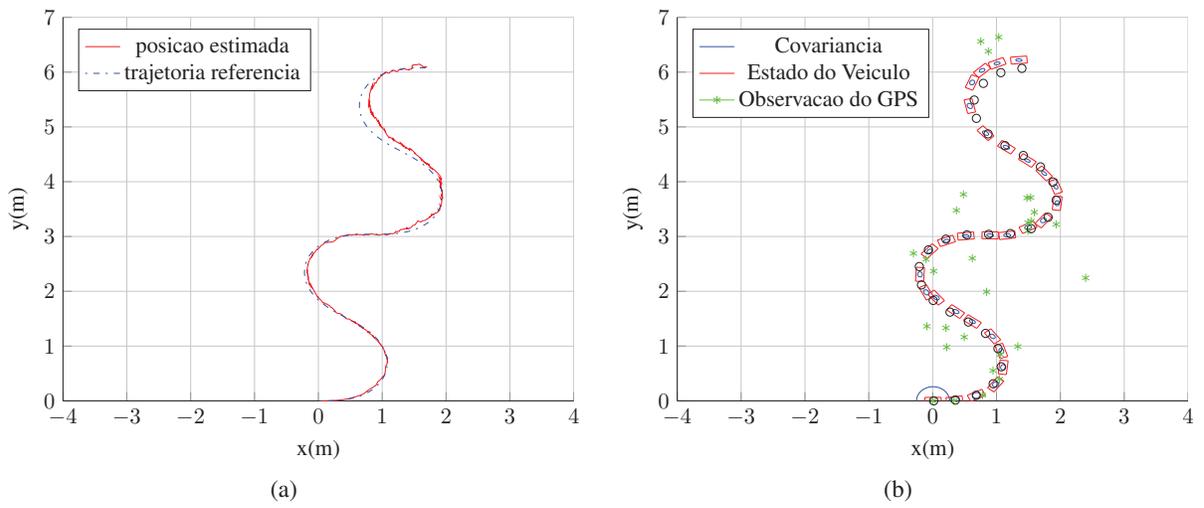


Figura 5.20: Simulação de estimação para localização: (a) resposta do filtro de fusão; e, (b) representação da resposta da estimação.

A análise de desempenho da estimação foi baseada no critério 3σ . Isto é, se o erro estiver dentro dos limites da covariância pode-se afirmar que a resposta tem um comportamento similar ao valor verdadeiro. O resultado para cada uma das variáveis do veículo é mostrado na figura 5.21.

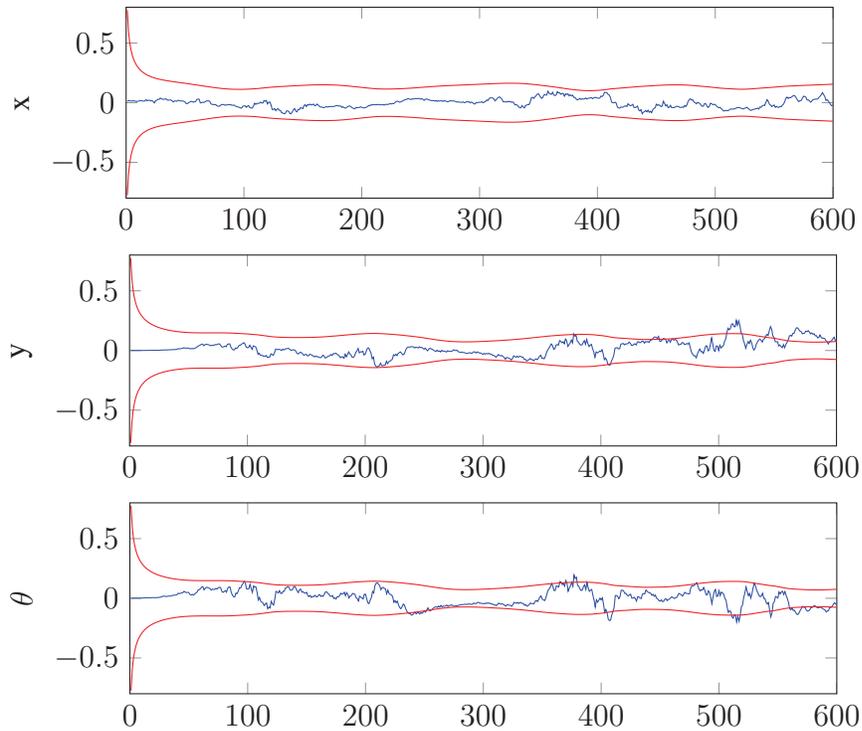


Figura 5.21: Comparação do erro com o limite de confiança $\pm 3\sigma$

5.3.2 Fusão de sensores: odometria convencional / emulador de GPS

No primeiro filtro de localização, é usado o filtro de kalman (desenvolvido na seção 4.2, do capítulo 4), para obter o estado do sistema baseado na dometria convecional (*dead-reckonig*) e o GPS. No gráfico 5.22 apresenta-se o bloco desta etapa.

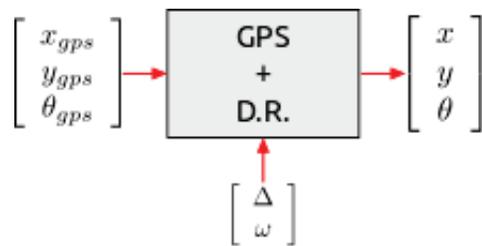
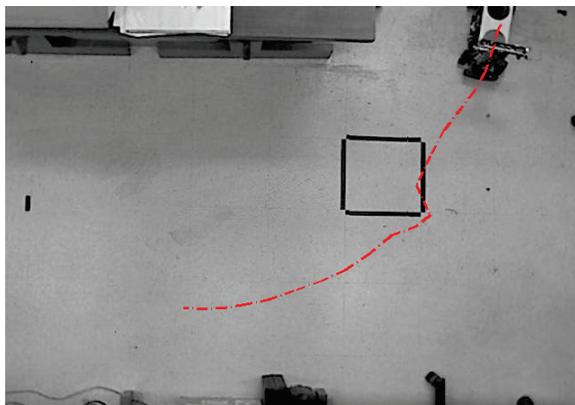
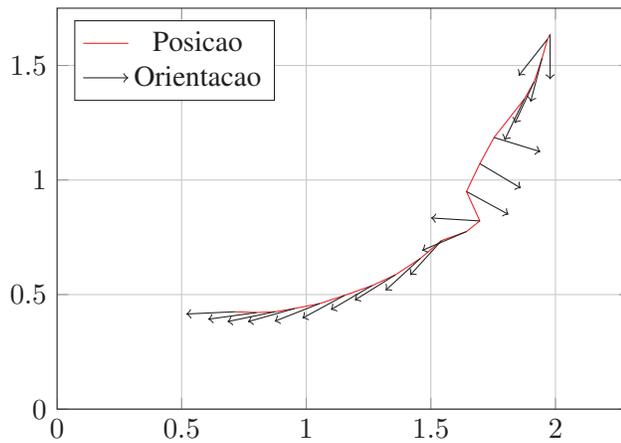


Figura 5.22: Bloco de fusão: GPS + D.R

Para realizar a avaliação experimental do filtro de fusão navegou-se em uma trajetória como mostrada na figura 5.23. Nessa figura tem-se o mapa e a trajetória definida e a informação proveniente do simulador de posicionamento global.



(a)

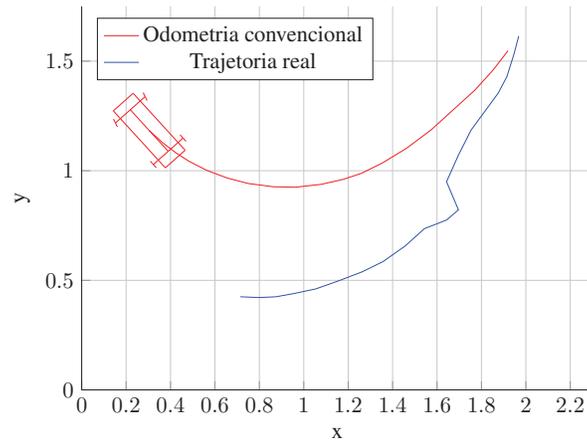


(b)

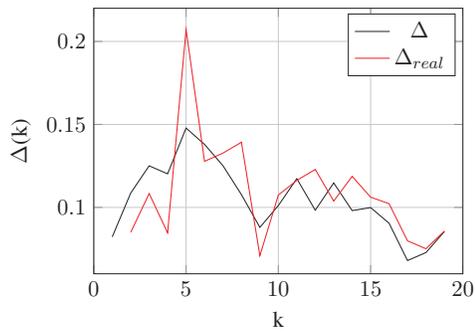
Figura 5.23: Trajetória de teste: (a) percurso no mapa ; e, (b) observação do GPS.

Dead-reckoning

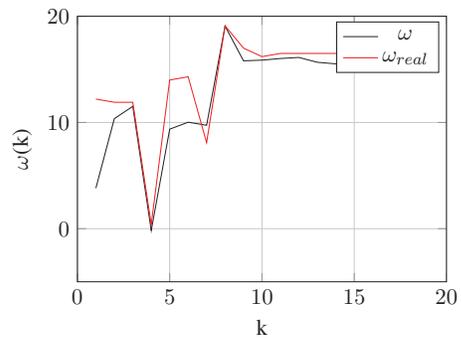
Para o percurso apresentado, é necessário estimar o vetor de estado de odometria como mostrado na seção 5.2.3. A figura 5.24 mostra o resultado da odometria convencional na trajetória percorrida, lembrando que os parâmetros de covariância continuam iguais.



(a)



(b)



(c)

Figura 5.24: Resultado odometria convencional: (a) deslocamento incremental; (b) estimativa do deslocamento total; e, (c) estimativa do ângulo de esterçamento.

Localização

Agora, aplicando-se o modelo de localização discutido antes, corrige-se a posição e orientação do veículo por meio do filtro de fusão. A figura 5.25 mostra a trajetória corrigida usando as medições do emulador de GPS; Na figura 5.25(b) é apresentado o resultado da estimação em comparação com a trajetória real.

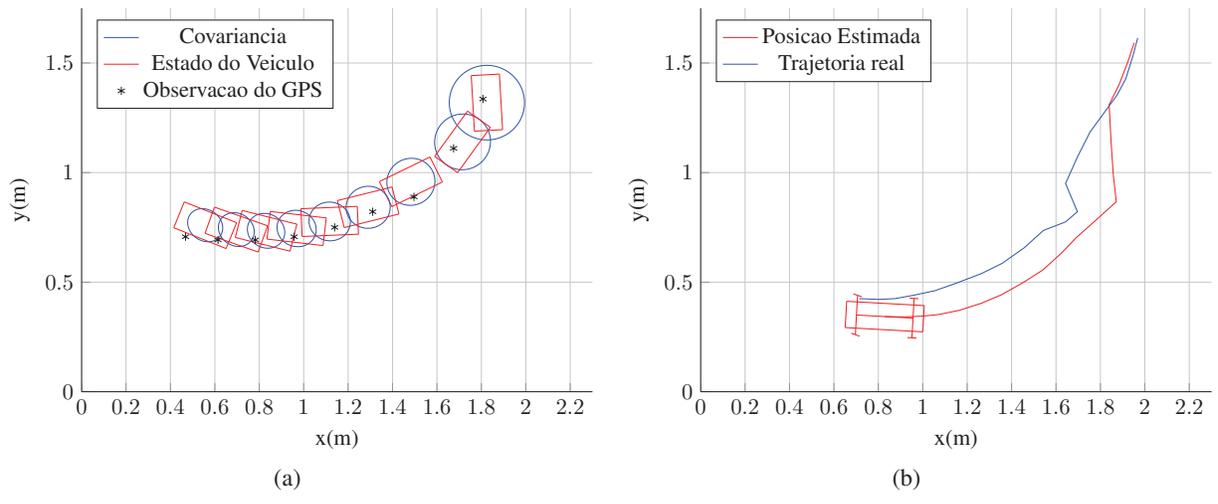


Figura 5.25: Estimação de localização: (a) evolução da incerteza; e, (b) resposta da localização.

Nota-se aqui que a posição e orientação têm uma tendência a convergir com o valor verdadeiro. Como na seção anterior, é apresentado o gráfico (figura 5.26) dos limites do valor do erro para cada uma das variáveis de estado do veículo dentro do limite confiança da estimação $\pm 3\sigma$.

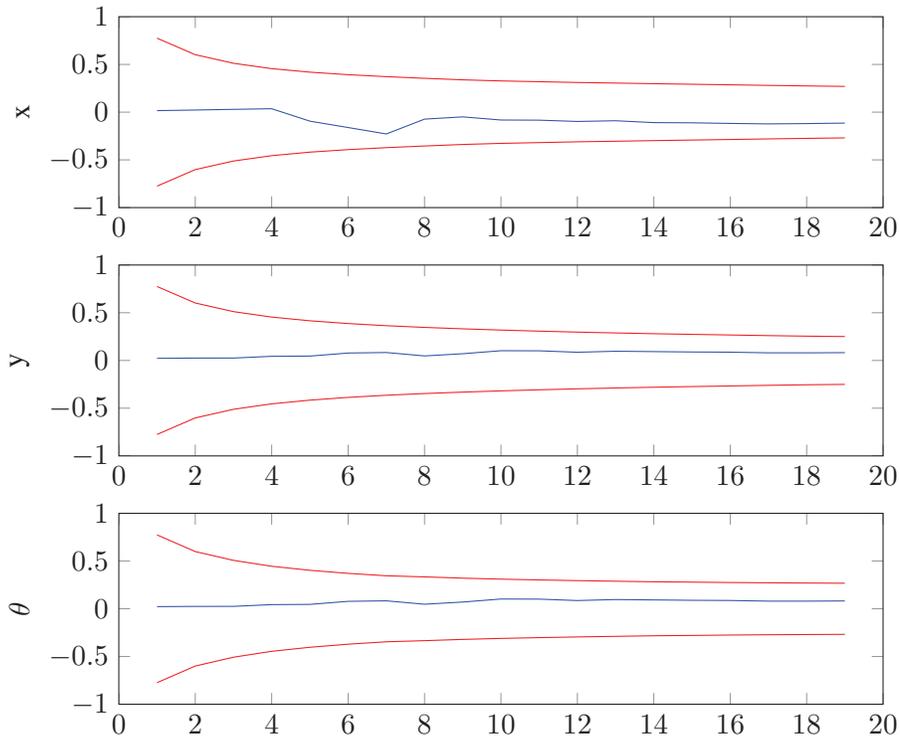


Figura 5.26: Comparação do erro com o limite de confiança $\pm 3\sigma$

5.3.3 Fusão de sensores: odometria convencional / GPS(emulador) / odometria visual

Nesta seção aborda-se o fusão final, em que, usa-se a odometria visual para corrigir a posição e orientação do veículo. A figura 5.27 mostra a última etapa da fusão.

Tem-se como entrada, o vetor de estados da odometria visual $\begin{bmatrix} x_{vo} & y_{vo} & \theta_{vo} \end{bmatrix}$ e as medições resultado da fusão, apresentada na seção anterior $\begin{bmatrix} x & y & \theta \end{bmatrix}$.

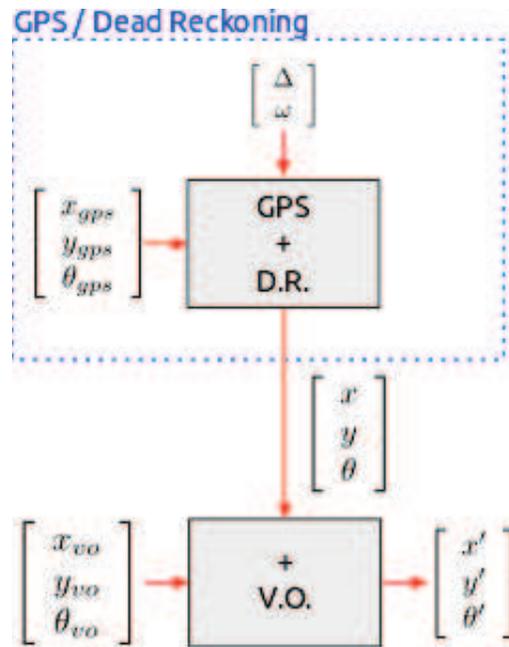


Figura 5.27: Bloco de fusão: GPS / D.R / V.O

Localização: Odometria visual

Para o percurso proposto, figura 5.23, apresenta-se o resultado obtido na Figura 5.28 usando somente a odometria visual como método de localização. Essa solução baseia-se no algoritmo introduzido na seção 4.1.

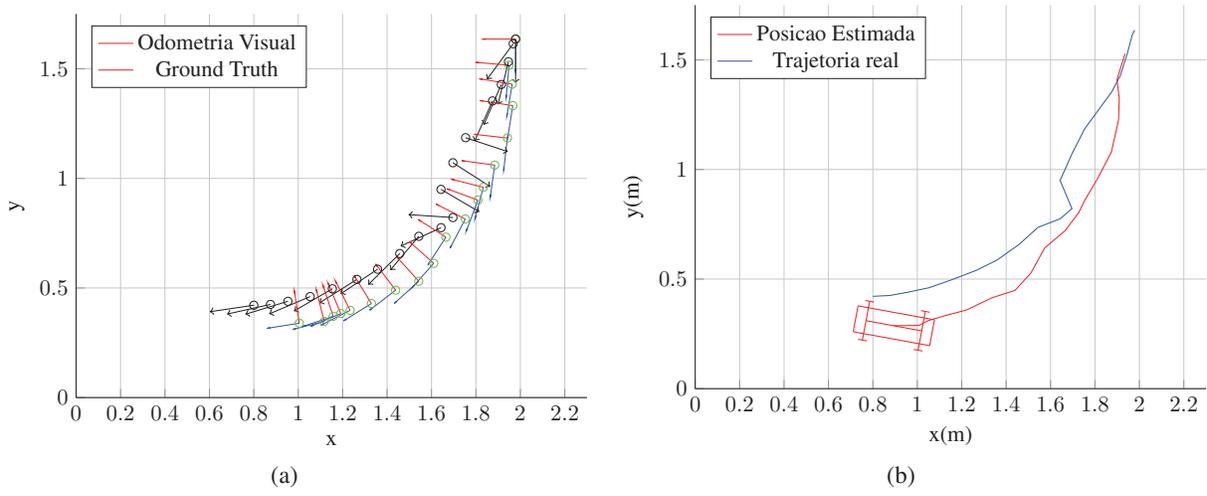


Figura 5.28: Resposta odometria visual: (a) sistema de coordenadas de câmara; e, (b) trajetória reconstruída.

Localização: odometria convencional / GPS(emulador) / odometria visual

Com o objetivo de diminuir o erro, ainda mais, entre a localização do veículo e a trajetória real, foi proposto um modelo linear para usar o resultado da odometria visual ${}^k x_{visual-odometry}$ e o resultado do filtro da fusão entre o GPS e a odometria convencional ${}^k x_{gps+dr}$ conforme figura 5.27. Na equação 5.9 apresenta-se o modelo matemático da integração, aonde o sub-índice vo representa a mudança de estado como resultado da odometria visual.

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} + \begin{bmatrix} \Delta x_{vo} \\ \Delta y_{vo} \\ \Delta \theta_{vo} \end{bmatrix} \quad (5.9)$$

Este modelo é uma representação linear do sistema, e traduzindo-o na representação matricial do filtro de Kalman fica: $A = I$, $B = I$ e $C = I$, sendo I a matriz identidade. Dentro do esquema de localização proposto, o filtro de kalman pode-se definir como um filtro da média ponderada, entre a Odometria Visual e o filtro de GPS + D.R. O calculo é feito usando duas medidas redundantes de posição e orientação (equação 5.9). A descrição da fusão encontra-se no algoritmo 5.3.1, onde mostra-se a iteração através de todas as observações.

Algorithm 5.3.1: Algoritmo de fusão Odometria visual + GPS + odometria convencional

Entrada: Observações do sistema de visão $[{}^k X_{vo} \quad {}^k Y_{vo} \quad {}^k \theta_{vo}]^T$

Saída : Vetor de localização $x = [X' \quad Y' \quad \theta']^T$

Aux : Observações do GPS + D.R. $Z = [{}^k X_{gps+dr} \quad {}^k Y_{gps+dr} \quad {}^k \theta_{gps+dr}]^T$

for $k \leftarrow 1$ **to** $nMeasurements$ **do**

$$\left[\begin{array}{l} \bar{x}_{k+1} = A \cdot x_k + B \cdot u_k \\ \bar{P}_{k+1} = A \cdot P_k \cdot A^T + A \cdot Q_k \cdot B^T \\ y_k = Z_k - C_k x_k \\ S_k = H_k P_k H_k^T + R \\ K_k = P_k H_k^T S_k^{-1} \\ x_k = \bar{x}_k + K_k y_k \\ P_k = (I - K_k H_k) P_k \end{array} \right.$$

A estimação de posição e orientação através da fusão de dados da odometria visual, a odometria convencional e o simulador de GPS via o filtro de Kalman foi implementada com as matrizes

de covariância da equação 5.10. Utilizou-se um período de amostragem de 260 *ms*. A matriz P representa o ruído cujo valor foi configurado como a matriz diagonal $\begin{bmatrix} 0.1m & 0.1m & 0.1^\circ \end{bmatrix}$. De igual forma foi configurada a matriz do ruído da medição do GPS, a fim de dar uma incerteza à essa observação.

$$P = \begin{bmatrix} 0.1^2 & 0 & 0 \\ 0 & 0.1^2 & 0 \\ 0 & 0 & 4^2 \end{bmatrix} \quad Q = \begin{bmatrix} 0.04085 & 0 & 0 \\ 0 & 0.0209 & 0 \\ 0 & 0 & 0.05055 \end{bmatrix} \quad (5.10)$$

A matriz Q , que representa o desvio padrão da odometria visual, provém de um conjunto de medições feitas para caracterizar o erro de cada grau de liberdade possível para a câmera embarcada no veículo. Esta medições foram obtidas de um teste de caracterização onde os resultados obtidos de erro relativo (tabela 5.1) foram introduzidos na matriz. De outra forma, a matriz de incerteza da medição, R foi atualizada em cada iteração da etapa de fusão de odometria convencional e do emulador de GPS.

X	Y	θ
3.9	2.7	4.1
2.7	6.5	3.5
2.8	3.8	6.9
0.4	6.2	5.9
2.1	10.4	2.9

Tabela 5.1: Caraterização do erro da odometria visual

O Resultado, na figura 5.29, mostra a estimação da posição e orientação através da fusão dos dados da odometria visual, odometria convencional e as medições do emulador de GPS. De forma geral, pode-se afirmar que usando medições redundantes é possível estimar os parâmetros do veículo que melhoram a convergência com a trajetória real.

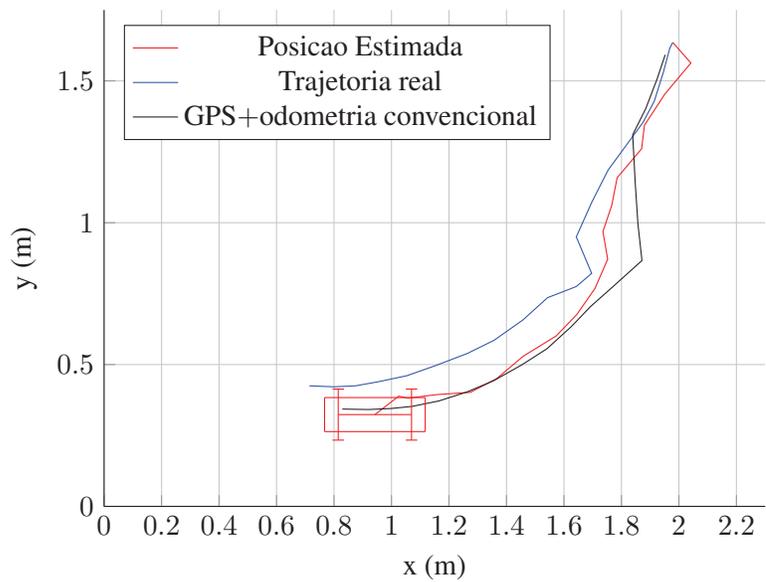


Figura 5.29: Resposta do sistema de localização final

Como critério de avaliação, a figura 5.30 mostra a evolução do erro ao longo do percurso, contido pela envolvente do desvio padrão, para cada uma das variáveis de estado do sistema.

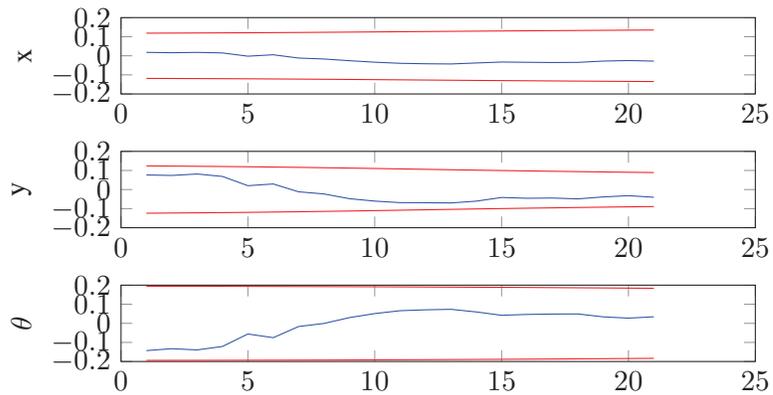


Figura 5.30: Comparação do erro com o limite de confiança $\pm 3\sigma$

6 DISCUSSÃO

6.1 Conclusão

De forma geral, o desenvolvimento de uma solução para a localização e mapeamento simultâneos de um veículo autônomo, baseado em um conjunto de sensores redundantes, como encoders, sensor de esterçamento e câmera RGB-D, foi concluído.

Essa solução de localização e mapeamento simultâneos envolve diferentes modelos e métodos para estimação e orientação para um veículo terrestre tipo automóvel. Além de ter apresentado essas implementações em um veículo em escala, esses modelos e métodos representam o ponto de partida do objetivo de conseguir navegação autônoma.

Este trabalho apresentou alguns métodos de localização relativa e absoluta, mostrando algumas características de cada um deles. Porém, não pode-se concluir que algum dos métodos seja suficiente por si só para garantir uma localização robusta.

A principal contribuição deste trabalho é mostrar uma estratégia de localização para veículos autônomos utilizando o filtro de Kalman. Tal filtro é utilizado como uma ferramenta para estimar a posição e orientação do veículo. A função do filtro foi a de combinar informação disponível a fim de estabelecer uma estimação robusta.

O desenvolvimento tanto teórico como experimental da localização nesse trabalho constitui uma aproximação ao objetivo de conseguir navegação autônoma, objetivo do Laboratório de Mobilidade Autônoma (LMA), do qual faz parte esta pesquisa.

6.2 Perspectivas Futuras

Como perspectiva futura, pretende-se modificar o sistema de localização desenvolvido para que possa funcionar em tempo real, já que neste trabalho foram utilizadas taxas de amostragem e processamento baixas, só com o objetivo de avaliar os métodos propostos. Assim, um próximo estágio envolveria uma mudança na implementação computacional dos algoritmos apresentados

neste trabalho.

Uma das principais restrições da odometria visual neste projeto deve-se ao fato de usar um sensor cujo comprimento de onda e potência não permite realizar a sua utilização em ambientes exteriores. Isto, por exemplo, pode-se ser resolvido com o uso de câmeras RGB apropriadas para ambientes externos.

No que se refere a odometria convencional, caberia fazer testes em terrenos que envolvam derrapagem nas rodas ou condições difíceis como lugares onde há terrenos acidentados e caminhos de terra. Em relação a testes em ambientes mais difíceis, também poderia se aproveitar o uso de um modelo mais consistente do que o modelo cinemático do veículo, como pode ser o modelo dinâmico a fim de melhorar a estimação.

Já um aperfeiçoamento do sistema de localização consiste no uso de um GPS ou DGPS real no esquema de localização proposto para garantir seu funcionamento em ambientes exteriores.

Outro problema apresentado foi o processamento em tempo real, que não era objetivo deste trabalho, mas envolve o uso do veículo numa situação real; assim, uma próxima etapa estaria relacionada com a integração em tempo real e determinação de taxas de amostragem para os diferentes tipos de sensores.

REFERÊNCIAS

- ANTONELLI, G.; CACCAVALE, F.; GROSSI, F. e MARINO, A. Simultaneous calibration of odometry and camera for a differential drive mobile robot. In **Robotics and Automation (ICRA), 2010 IEEE International Conference on**, pp. 5417 –5422. may 2010.
- AZIZI, F. e HOUSHANGI, N. Mobile robot position determination using data from gyro and odometry. In **Electrical and Computer Engineering, 2004. Canadian Conference on**, v. 2, pp. 719 – 722 Vol.2. May 2004.
- BAY, H.; TUYTELAARS, T. e GOOL, L.V. Surf: Speeded up robust features. In **European Conference on Computer Vision**, pp. 404–417. 2006.
- BENTO, L.; NUNES, U.; MOITA, F. e SURRECIO, A. Sensor fusion for precise autonomous vehicle navigation in outdoor semi-structured environments. In **Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE**, pp. 245 – 250. sept. 2005.
- BERTOZZI, M.; BROGGI, A.; CONTE, G. e FASCIOLI, A. The experience of the argo autonomous vehicle. In **IN PROCS. SPIE'98 - AEROSENSE CONF**, pp. 218–229. 1998.
- BLANCO, J.L.; FERNANDEZ-MADRIGAL, J.A. e GONZALEZ, J. A new approach for large-scale localization and mapping: Hybrid metric-topological slam. In **ICRA**, pp. 2061–2067. 2007.
- BONIN-FONT, F.; ORTIZ, A. e OLIVER, G. Visual navigation for mobile robots: A survey. **J. Intell. Robotics Syst.**, v. 53, n. 3, 263–296, novembro 2008.
- BONNIFAIT, P.; BOURON, P.; CRUBILLE, P. e MEIZEL, D. Data fusion of four abs sensors and gps for an enhanced localization of car-like vehicles. In **Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on**, v. 2, pp. 1597 – 1602 vol.2. 2001.

BORENSTEIN, J. e KOREN, Y. Motion control analysis of a mobile robot. 1995.

BOTELHO, S.S.D.C.; DREWS JUNIOR, P.L.J.; FIGUEIREDO, M.D.S.; ROCHA, C.H.D. e OLIVEIRA, G.L. Appearance-based odometry and mapping with feature descriptors for underwater robots. **Journal of the Brazilian Computer Society**, v. 15, 47 – 54, 09 2009.

BOUGUET, J.Y. Camera calibration toolbox for Matlab. 2008.

BUEHLER, M.; IAGNEMMA, K. e SINGH, S. **The 2005 DARPA Grand Challenge: The Great Robot Race**. Springer Publishing Company, Incorporated, 1st ed., 2007. ISBN 3540734287, 9783540734284.

CAMPBELL, S.F. Steering control of an autonomous ground vehicle with application to the darpa urban challenge. 2009.

CHOI, S.; JOUNG, J.H.; YU, W. e CHO, J.I. What does ground tell us? monocular visual odometry under planar motion constraint. In **Control, Automation and Systems (ICCAS), 2011 11th International Conference on**, pp. 1480 –1485. oct. 2011.

COLEMAN, T.P.; LEE, A.H.; MEMBER, S.; MEDARD, M.; EFFROS, M.; MEMBER, S. e MEMBER, S. Low-complexity approaches to slepian-wolf near-lossless distributed data compression. **IEEE Trans. Inform. Theory**, v. 52, 3546–3561, 2006.

DAI, X.; ZHANG, H. e SHI, Y. Autonomous navigation for wheeled mobile robots-a survey. In **Innovative Computing, Information and Control, 2007. ICICIC '07. Second International Conference on**, p. 551. sept. 2007.

DANIEL HERRERA, C.; KANNALA, J. e HEIKKILA, J. Joint depth and color camera calibration with distortion correction. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 99, 2012.

DASARATHY, B. Sensor fusion potential exploitation-innovative architectures and illustrative

applications. **Proceedings of the IEEE**, v. 85, n. 1, 24–38, jan 1997.

DESOUZA, G. e KAK, A. Vision for mobile robot navigation: a survey. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, v. 24, n. 2, 237–267, feb 2002.

DICKMANN, E.; BEHRINGER, R.; BRUDIGAM, C.; DICKMANN, D.; THOMANEK, F. e VAN HOLT, V. An all-transputer visual autobahn-autopilot/copilot. In **Computer Vision, 1993. Proceedings., Fourth International Conference on**, pp. 608–615. 1993.

DICKMANN, E.D. Vehicles capable of dynamic vision: a new breed of technical beings? **Artificial Intelligence**, v. 103, n. 1 - 2, 49 – 76, 1998. <ce:title>Artificial Intelligence 40 years later</ce:title>.

URL: <http://www.sciencedirect.com/science/article/pii/S000437029800071X>

DOH, N.; CHOSET, H. e CHUNG, W.K. Accurate relative localization using odometry. In **Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on**, v. 2, pp. 1606 – 1612 vol.2. sept. 2003.

DUDEK, G. e JENKIN, M. **Computational Principles of Mobile Robotics**. Cambridge University Press, New York, NY, USA, 2nd ed., 2010. ISBN 0521871573, 9780521871570.

DURRANT-WHITE, H. e BAILEY, T. Simultaneous localization and mapping: part i. **Robotics Automation Magazine, IEEE**, v. 13, n. 2, 99–110, june 2006.

FIALA, M. e UFKES, A. Visual odometry using 3-dimensional video input. In **Computer and Robot Vision (CRV), 2011 Canadian Conference on**, pp. 86–93. may 2011.

FISCHLER, M.A. e BOLLES, R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. **Commun. ACM**, v. 24, n. 6, 381–395, junho 1981.

HARTLEY, R.I. e ZISSERMAN, A. **Multiple View Geometry in Computer Vision**. Cambridge

University Press, ISBN: 0521540518, second ed., 2004.

HENRY, P.; KRAININ, M.; HERBST, E.; REN, X. e FOX, D. Rgb-d mapping: Using depth cameras for dense 3d modeling of indoor environments. In **RGB-D: Advanced Reasoning with Depth Cameras Workshop in conjunction with RSS**. 2010.

HERRERA C, D.; KANNALA, J. e HEIKKILA, J. Joint depth and color camera calibration with distortion correction. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, v. PP, n. 99, 1, 2012.

HOFFLINGER, B.; CONTE, G.; ESTEVE, D. e WEISGLAS, P. Integrated electronics for automotive applications in the eureka program prometheus. In **Solid-State Circuits Conference, 1990. ESSCIRC '90. Sixteenth European**, v. 2, pp. 13–17. 1990.

KALMAN, R.E. A new approach to linear filtering and prediction problems. **Transactions of the ASME–Journal of Basic Engineering**, v. 82, n. Series D, 35–45, 1960.

KARLSSON, N.; DI BERNARDO, E.; OSTROWSKI, J.; GONCALVES, L.; PIRJANIAN, P. e MUNICH, M. The vslam algorithm for robust localization and mapping. In **Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on**, pp. 24 – 29. april 2005.

KHOSHELHAM, K. e ELBERINK, S. Accuracy and resolution of kinect depth data for indoor mapping applications. **Sensors (Basel)**, v. 12, n. 2, 1437–54, 2012.

KRAMER, J.; BURRUS, N.; C., D.H.; ECHTLER, F. e PARKER, M. **Hacking the Kinect**. Apress, Berkely, CA, USA, 1st ed., 2012. ISBN 1430238674, 9781430238676.

LOWE, D.G. Distinctive image features from scale-invariant keypoints. **Int. J. Comput. Vision**, v. 60, n. 2, 91–110, novembro 2004.

NISTER, D.; NARODITSKY, O. e BERGEN, J. Visual odometry for ground vehicle applications.

Journal of Field Robotics, v. 23, 2006, 2006.

POLLEFEYS, M.; VAN GOOL, L.; VERGAUWEN, M.; VERBIEST, F.; CORNELIS, K.; TOPS, J. e KOCH, R. Visual modeling with a hand-held camera. **Int. J. Comput. Vision**, v. 59, n. 3, 207–232, setembro 2004.

ROUFF, C. e HINCHEY, M. **Experience from the Darpa Urban Challenge**. Springer, 2011. ISBN 9780857297716.

URL: <http://books.google.com.br/books?id=98Xdcw-5mBsC>

SCARAMUZZA, D. e SIEGWART, R. Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. **Robotics, IEEE Transactions on**, v. 24, n. 5, 1015–1026, oct. 2008.

SHEN, J.; TICK, D. e GANS, N. Localization through fusion of discrete and continuous epipolar geometry with wheel and imu odometry. In **American Control Conference (ACC), 2011**, pp. 1292–1298. 29 2011-july 1 2011.

SICILIANO, B. e KHATIB, O., editores. **Springer Handbook of Robotics**. Springer, 2008. ISBN 978-3-540-23957-4.

SIEGWART, R. e NOURBAKHSI, I.R. **Introduction to Autonomous Mobile Robots**. Bradford Company, Scituate, MA, USA, 2004. ISBN 026219502X.

SINGHAL, A. Issues in autonomous mobile robot navigation. Relatório técnico, None, 1997.

SNIDER, J.M. Automatic steering methods for autonomous automobile path tracking. Relatório Técnico CMU-RI-TR-09-08, Robotics Institute, Pittsburgh, PA, February 2009.

SOUISSI, S.; IZUMI, T. e WADA, K. Distributed algorithms for cooperative mobile robots: A survey. In **Networking and Computing (ICNC), 2011 Second International Conference on**, pp. 364–371. 30 2011-dec. 2 2011.

SURRECIO, A.; NUNES, U. e ARAUJO, R. Fusion of odometry with magnetic sensors using kalman filters and augmented system models for mobile robot navigation. In **Industrial Electronics, 2005. ISIE 2005. Proceedings of the IEEE International Symposium on**, v. 4, pp. 1551 – 1556. 20-23, 2005.

THRUN, S.; MONTEMERLO, M.; DAHLKAMP, H.; STAVENS, D.; ARON, A.; DIEBEL, J.; FONG, P.; GALE, J.; HALPENNY, M.; HOFFMANN, G.; LAU, K.; OAKLEY, C.; PALATUCCI, M.; PRATT, V.; STANG, P.; STROHBAND, S.; DUPONT, C.; JENDROSSEK, L.E.; KOELEN, C.; MARKEY, C.; RUMMEL, C.; VAN NIEKERK, J.; JENSEN, E.; ALESSANDRINI, P.; BRADSKI, G.; DAVIES, B.; ETTINGER, S.; KAEHLER, A.; NEFIAN, A. e MAHONEY, P. Winning the darpa grand challenge. **Journal of Field Robotics**, 2006. Accepted for publication.

UHLIN, T. e JOHANSSON, K. Autonomous mobile systems: A study of current research. 1996.

UMEYAMA, S. Least-squares estimation of transformation parameters between two point patterns. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, v. 13, n. 4, 376 –380, apr 1991.

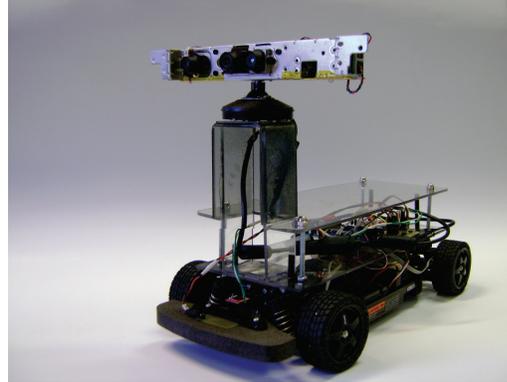
WEINSTEIN, A. e MOORE, K. Pose estimation of ackerman steering vehicles for outdoors autonomous navigation. In **Industrial Technology (ICIT), 2010 IEEE International Conference on**, pp. 579 –584. march 2010.

ZHANG, C. e ZHANG, Z. Calibration between depth and color sensors for commodity depth cameras. In **Multimedia and Expo (ICME), 2011 IEEE International Conference on**, pp. 1 –6. july 2011.

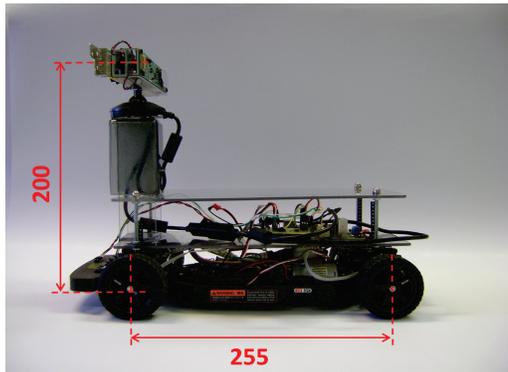
ZHANG, Z. e ZHANG, Z. A flexible new technique for camera calibration. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 22, 1330–1334, 1998.

A Plataforma: Veículo em escala

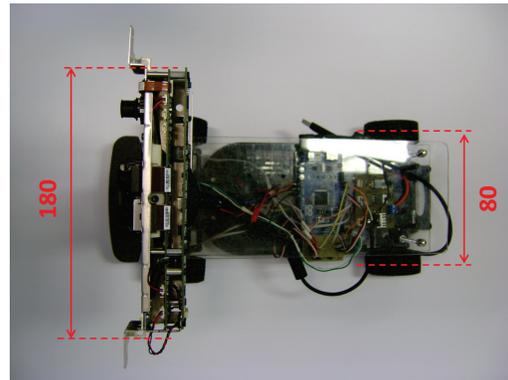
Para realizar a implementação deste trabalho foi modificado um veículo em escala da marca *TAMIYA* (figura A.1). Este veículo foi levemente modificado para embarcar o equipamento necessário para o controle e medição.



(a)



(b)



(c)

Figura A.1: Veículo em escala *Tamiya* modificado: (a) vista ISO; (b) vista lateral; e, (c) vista superior.

Uma descrição dos componentes nele embarcados está contido na tabela A.1.

Componente	Descrição
<i>tablet</i>	Computador Embarcado
Microcontrolador	Arduino Mega 2560
Encoder Rodas	Encoder ótico Pololu
Sensor Esterçamento	Potenciômetro Linear
Câmera RGB-D	Microsoft Kinect
GPS	Marcador para o GPS baseado em visão

Tabela A.1: Componentes e descrição do veículo em escala

O computador embarcado é um *tablet* PC, cujas principais características são mostradas a seguir:

- Processador: Intel ATOM N450
- Memória: 2GB DDR2
- HD: 320 GB

B Plataforma: Hardware(*Arduino*)

A prototipagem eletrônica realizada neste projeto é baseada na plataforma de software e hardware livre *Arduino*. Esta permite controlar alguns atuadores e comunicar-se com outros dispositivos. Na figura B.1 mostram-se as diferentes camadas de *hardware* implementadas usando o *arduino* como controlador. O veículo em escala usado tem um mecanismo de locomoção diferencial igual aos automóveis reais baseado em um motor DC simples com caixa de redução. O controle do motor de locomoção é feito usando modulação PWM(*Pulse-Width Modulation*). O veículo também tem um servo motor embarcado para atuar sobre o mecanismo de esterçamento. Estes atuadores fazem parte do veículo original(Fabricado pela *Tamiya*) e são alimentados por uma bateria tipo *NiMH*.

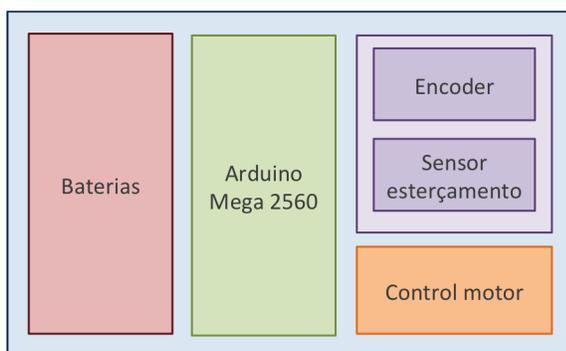


Figura B.1: Arquitetura de *hardware* do veículo em escala

Além dos atuadores acima mencionados, implementaram-se alguns tipos de sensores para medir algumas variáveis do estado do veículo. Todos esses componentes tem um consumo de energia que deve ser fornecido por um conjunto de baterias, que demandaram o projeto de placas auxiliares em relação à administração da energia. Nas seguintes sessões apresenta-se uma breve descrição das placas projetadas para à aquisição dos sensores acompanhadas dos esquemas usando o software *eagle*¹.

B.1 Fonte Reguladora

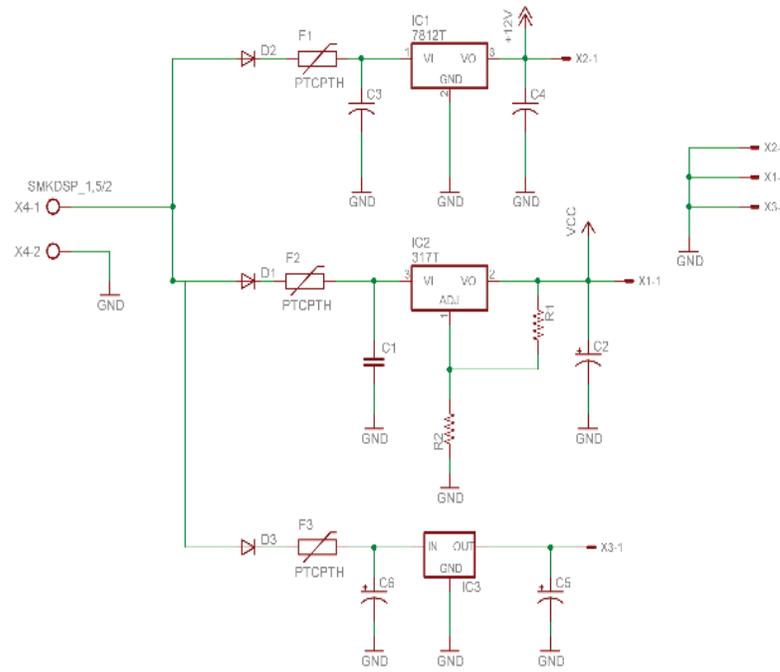
O consumo tanto dos atuadores como dos sensores e da placa *Arduino* esta limitado pelas baterias embarcadas no veículo. Para este proposito foi projetada uma placa auxiliar para cada

¹Cadsoft website <http://www.cadsoftusa.com/eagle-pcb-design-software>

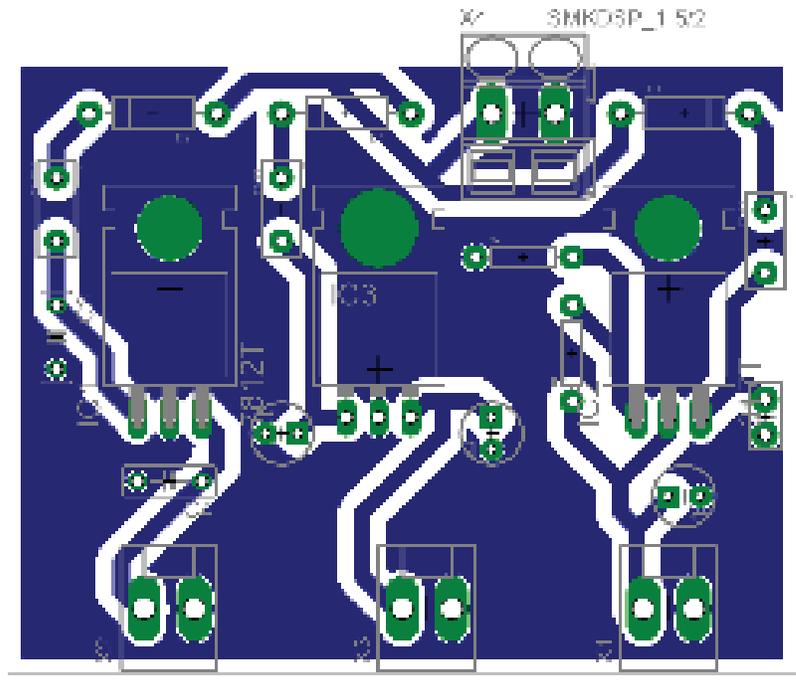
bateria. O circuito eletrônico destinado a fornecer 3 diferentes valores de tensão contínua para uma entrada à bateria tipo *LiPo* de $11.1v$ que possui uma capacidade de $1500mAh$ esta baseado em uma conversão DC-DC mas mantendo a corrente o mais estável possível. Esta solução é providenciada usando reguladores à transistor que são estáveis na corrente fornecida, porém tem uma queda de tensão que é pequena em relação à entrada. A placa projetada Figura B.2 tem 3 saídas de tensão diferentes para satisfazer as diferentes placas da plataforma, assim, tem-se disponíveis tensões de $12v$, $5v$ e $7v$ à $1200mA$.

B.2 Circuito Auxiliar Encoder

Como apresentado no capítulo 3, foram implementados nas rodas do veículo, um tipo de encoder óptico de baixo custo. Para adaptar estes à plataforma de controle, foi projetada uma placa (Figura B.3). A placa compõe-se de um regulador de tensão à transistor de $3,3v$ para alimentar cada sensor em cada roda.

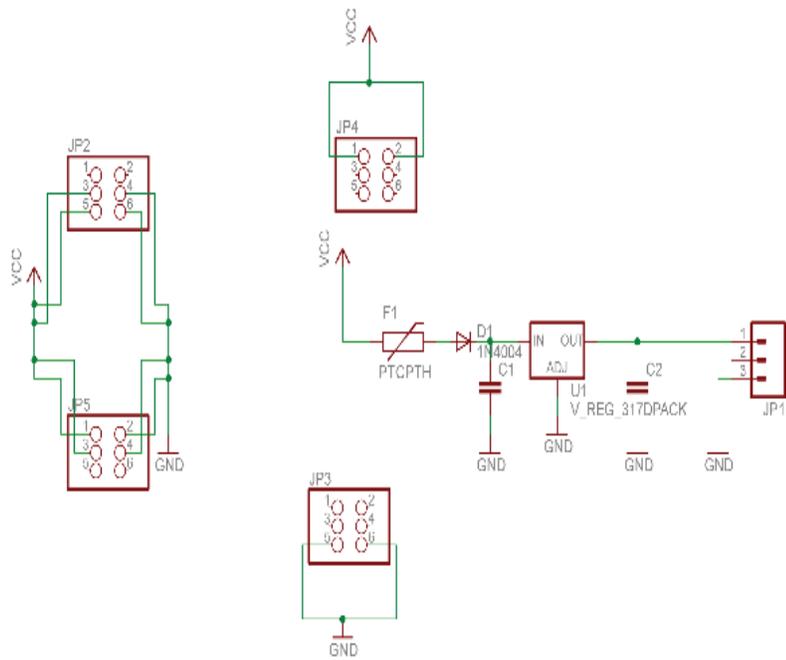


(a)

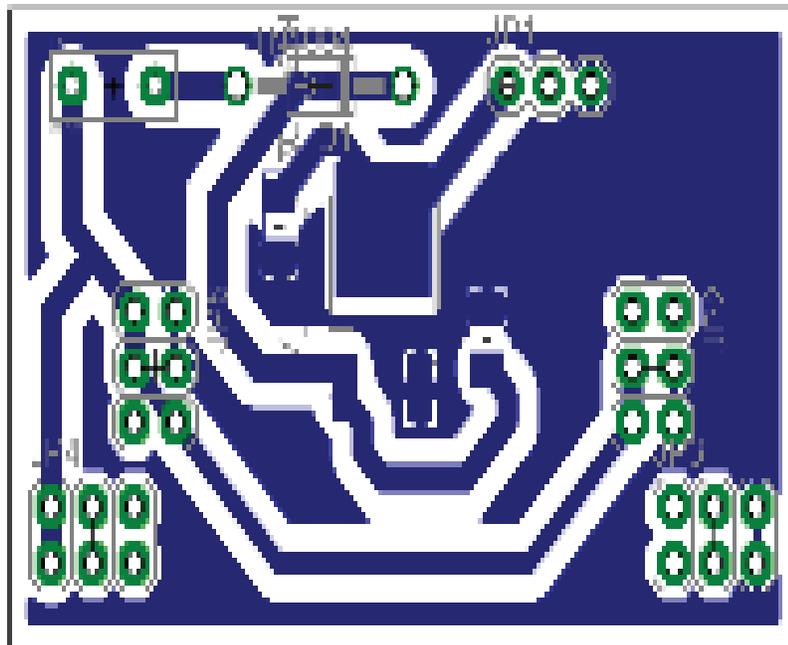


(b)

Figura B.2: Placa da fonte regulada (a) Esquema elétrico. (b) Placa PCB(*Printed Circuit Board*)



(a)



(b)

Figura B.3: Placa Encoder: (a)Esquema elétrico e (b)Placa PCB

C Plataforma: Sensores de odometria convencional (*Dead-Reckoning*)

Para determinar a orientação e posição através da observação e integração do veículo usando a técnica de *dead-reckoning* foram implementados encoders em cada roda do veículo e um potenciômetro no mecanismo de direção(volante) para medir o ângulo de esterçamento. A caracterização é necessária para estabelecer a função matemática que relaciona mudanças no estado do veículo, de forma precisa.

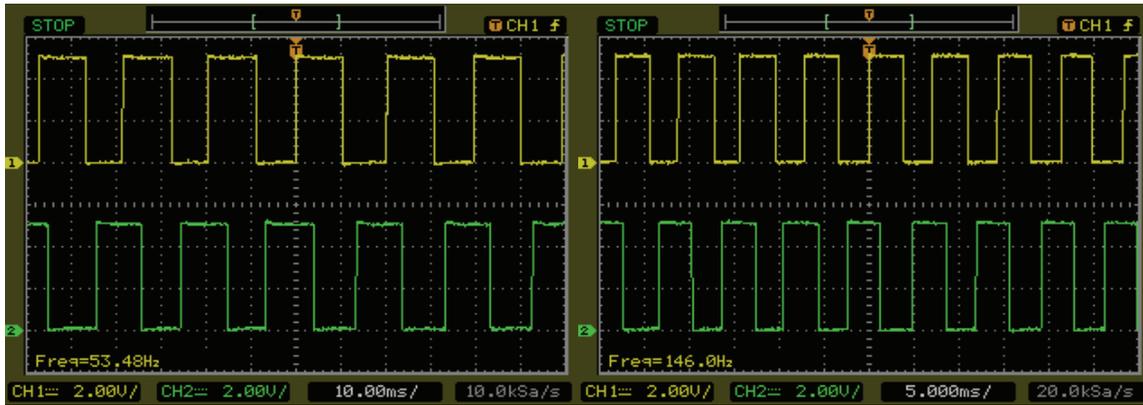
C.1 Encoder óptico



Figura C.1: Resposta do encoder incremental nas 4 rodas

Fixo à estrutura do veículo(Figura C.1), o encoder óptico faz a conta dos pulsos obtidos por volta. Um número de pulsos N_k é obtido depois de certo numero de rotações. O primeiro critério em ser achado é a frequência de operação. A faixa útil encontrada é de $[53, 146]$ Hz. A figura C.4 mostra os intervalos de operação. Essa faixa depende da frequência do sensor, que como foi mencionando anteriormente é óptico. A frequência depende só do padrão de pulsos instalado no interior do pneu.

Para converter esse número de pulsos em distância percorrida é necessário criar uma função de deslocamento. A correspondente função consiste em contar-se a quantidade de pulsos obtidos num intervalo de tempo, assumindo que não há escorregamento. A figura C.3 apresenta uma análise da rotação das rodas dianteiras do veículo sem escorregamento.



(a)

(b)

Figura C.2: Faixa em frequência do encoder: (a)53Hz (b)146Hz

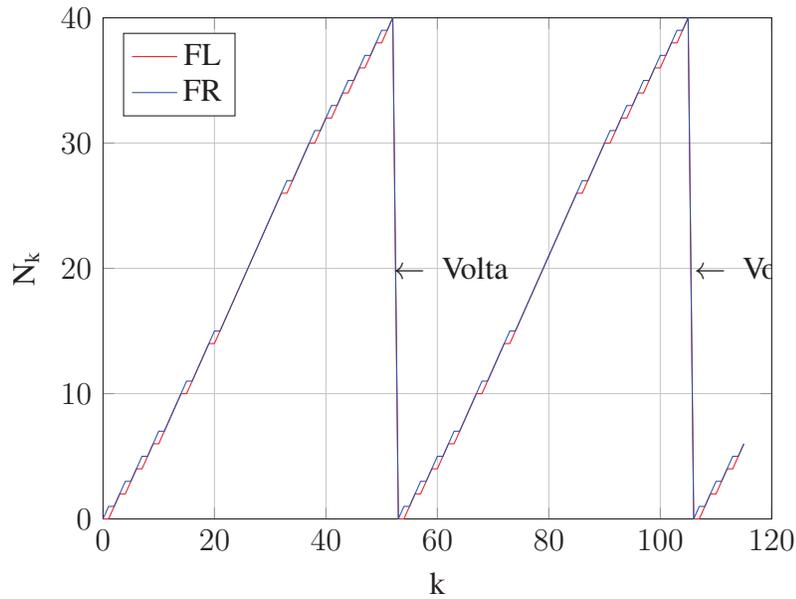


Figura C.3: Resposta do encoder incremental nas rodas dianteiras FR(direita) e FL(esquerda)

Para determinar o deslocamento num intervalo de tempo é determinada a resolução λ . A resolução do encoder é baseada na dimensões da roda no número de pulsos instalado nesta dimensão da roda, resultando um $\lambda = 9 \frac{mm}{pulso}$. Assim pode-se determinar a distância percorrida por cada roda Δ_k usando a equação C.1 tendo como única entrada o número de pulsos N_k .

$$\Delta_k = \frac{N_k d \pi}{360} \lambda \quad (\text{C.1})$$

Onde, d é o diâmetro da roda, que assume-se constante, isto assumindo que o pneu é um objeto rígido e não-deformável.

A figura C.4 ilustra a resposta gerada pela função numa trajetória "linha reta" sobre uma superfície livre de escorregamento para cada uma das 4 rodas do veículo sob condições de velocidade constante.

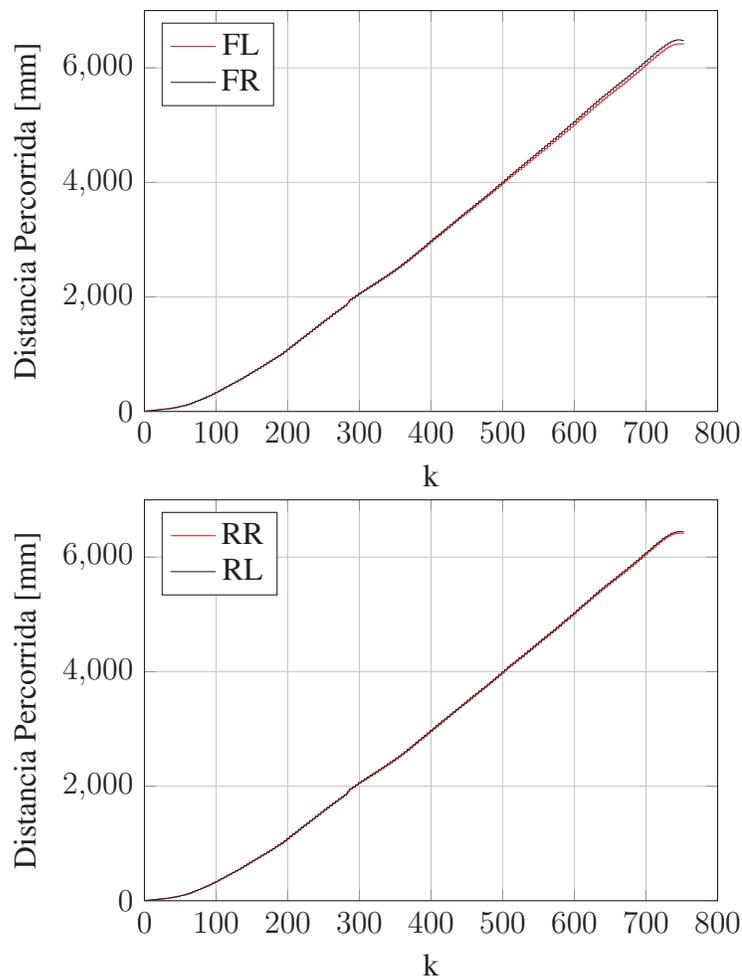


Figura C.4: Resposta da calibração do ENCODER nas rodas: (a)Rodas dianteiras(b)Rodas traseiras

C.2 Sensor do ângulo de esterçamento(Steering)

Para embarcar um sensor que permitisse medir o ângulo de esterçamento, dadas as restrições espaciais existentes no veículo, foi utilizado um potenciômetro (figura C.5). Este sensor além de ser baixo custo tinha um tamanho apropriado para ser colocado no mecanismo da direção do veículo.

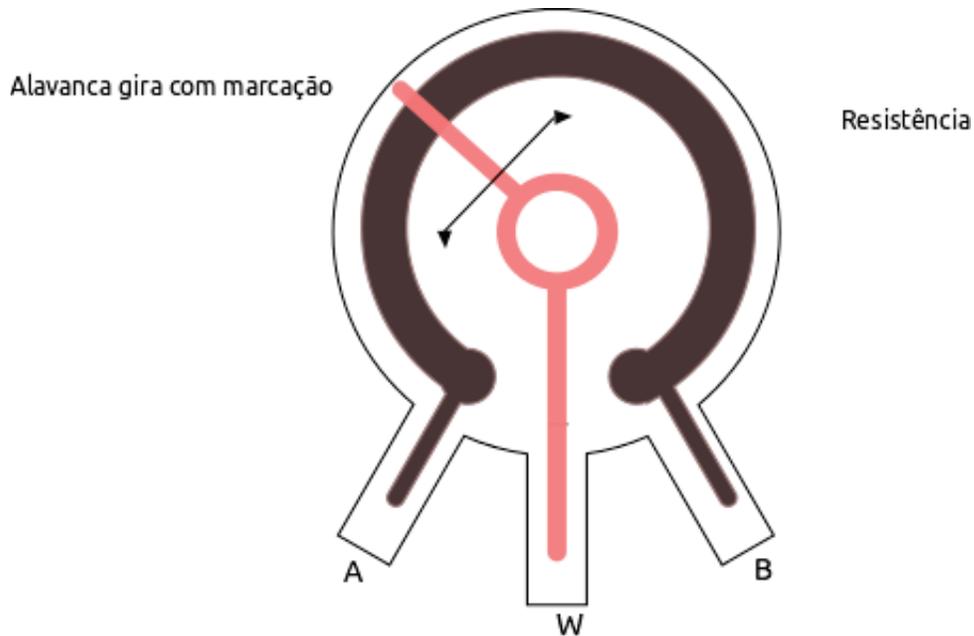


Figura C.5: Potenciômetro Rotativo

Fixo no eixo de rotação do atuador do mecanismo da direção(*Ackerman*), onde um único valor corresponde a posição absoluta. No entanto, uma primeira experiência mostrou que a incerteza pela histerese é muito grande. A figura C.6 mostra a curva de histerese do potenciômetro e uma primeira aproximação de calibração.

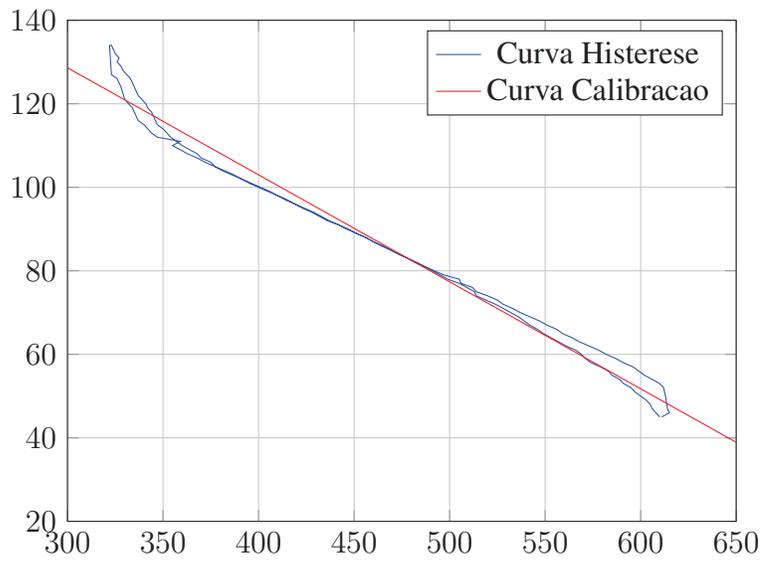


Figura C.6: Histerese Potenciômetro

Como esperado, foi necessário estabelecer um método de calibração mais preciso. Tendo em conta que a região de trabalho é de somente 90° achou-se uma região apropriada para representar estes valores. A figura C.7 mostra a resposta do sensor num intervalo e sua respectiva curva de calibração.

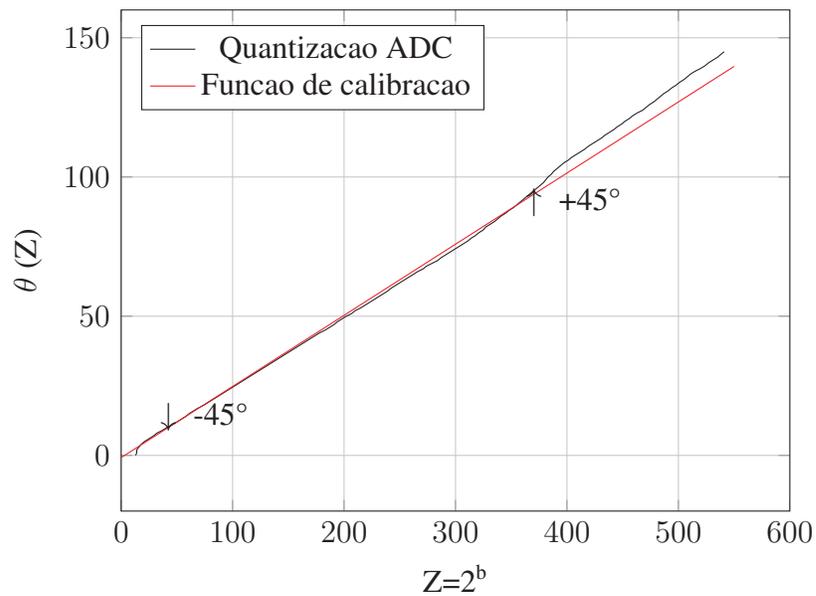


Figura C.7: Curva de calibração sensor esterçamento

A curva de calibração, mostrada acima, foi encontrada por um método de regressão linear. Esta relaciona os valores convertidos ou quantizados pelo ADC e um ângulo do potenciômetro. Dita função tem como único parâmetro o valor de quantização do ADC (Conversor Analógico-Digital) e é apresentada na equação C.2.

$$\theta(n) = 0.2502n + 5.2696 \quad (C.2)$$

O potenciômetro foi posicionado de tal forma que pudesse trabalhar na região linear encontrada experimentalmente. A sua resposta no curso total do mecanismo é apresentada na figura C.8.

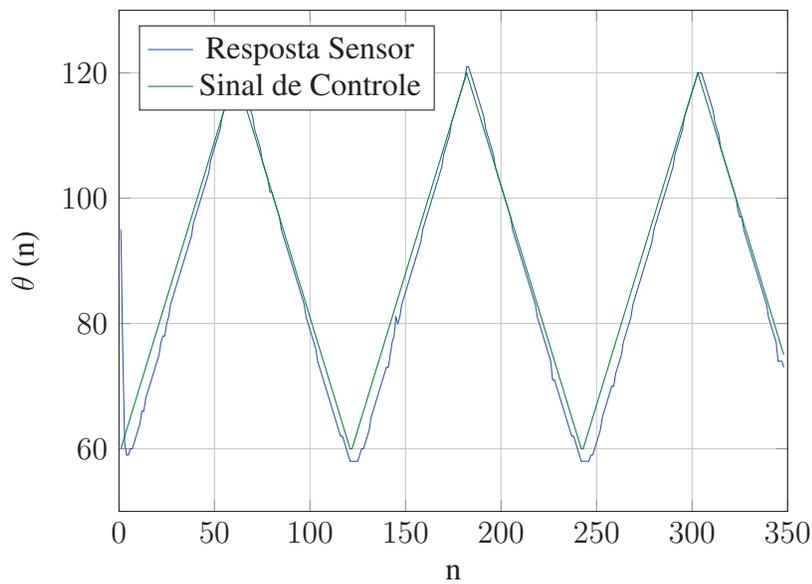


Figura C.8: Resposta do potenciômetro como sensor do ângulo de esterçamento

D Plataforma: Sistema de posicionamento global (Emulador)

Neste apêndice apresenta-se um simulador para um sistema de posicionamento local para qualquer robô móvel em ambientes *indoor* usando uma câmera de teto. A vantagem que apresenta o uso de uma câmera como observador do *ground-truth* é a resolução de pixel que esta tem, isto permite obter linhas, pontos ou arestas bem definidos, assim um algoritmo pode determinar de forma confiável a posição de um objeto. O veículo em escala tem embarcado um marcador D.1 composto de duas bolas de duas cores diferentes a fim de encontrar tanto a posição como a orientação. Onde a bola de cor laranja representa a parte dianteira do veículo e a bola azul representa a parte traseira do veículo. Este sistema de posicionamento funciona bem sobre um campo com dimensões máximas de 2.28 x 1.71 m.

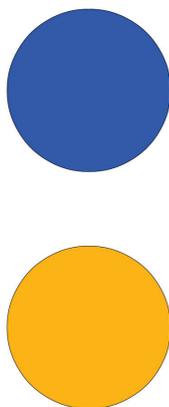


Figura D.1: Marcador de posicionamento local

D.1 Calibração

A calibração da câmera foi feita usando o *toolkit* desenvolvido por Bouguet (2008). A correspondente calibração produz os parâmetros Intrínsecos da câmera, os quais permitem obter uma imagem sem a distorção radial própria da lente. Na figura D.2 tem-se a imagem original (a) e a imagem retificada (b).

Tendo a imagem sem distorção e sabendo que o veículo (marcador em cima) se desloca num

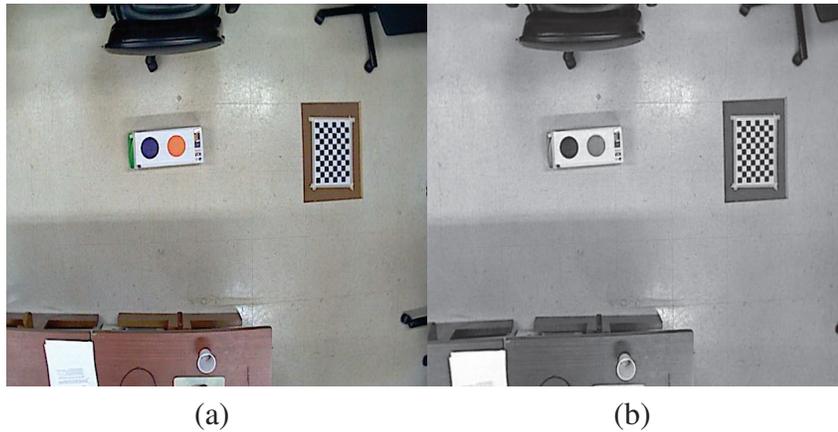


Figura D.2: Campo de visão da câmera: (a)Imagem original (b)Imagem retificada

plano cuja distância é conhecida, pode-se encontrar uma relação entre o sistema de coordenadas da imagem e o sistema de coordenadas da cena $x' \rightarrow X'$. Na imagem D.3 mostram-se as dimensões da área enxergada pela câmera e o padrão de referência(Quadro preto).

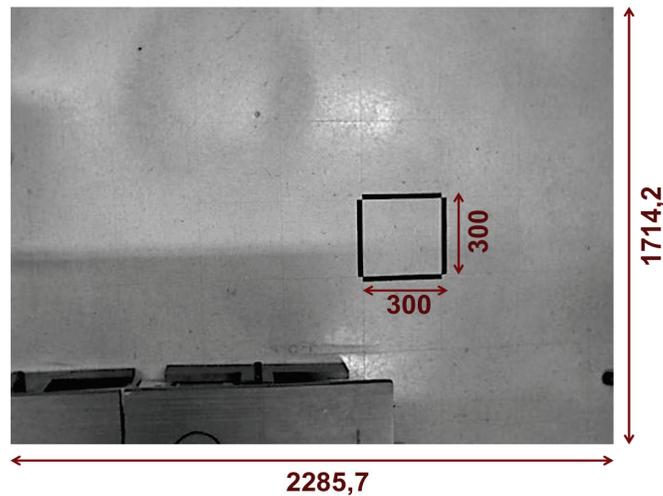


Figura D.3: Esquema de calibração do simulador

Cada observação da câmera possui uma resolução de 640×480 , enquanto o espaço de observação corresponde à uma área de $2.28 \times 1.71 \text{ m}$. Assim é encontrada uma relação linear para transformar do espaço de imagem para espaço real. A equação D.1 mostra o valor constante que relaciona os pixels com as coordenadas em metros.

$$\begin{bmatrix} X \\ Y \end{bmatrix} = 0.003571429 \cdot \begin{bmatrix} x \\ y \end{bmatrix} \quad (\text{D.1})$$

D.2 Algoritmo

O simulador foi implementado para realizar o cálculo das variáveis de localização $X_{gps}^k = \begin{bmatrix} x_{gps}^k & y_{gps}^k & \theta_{gps}^k \end{bmatrix}^T$ de forma *off-line*. O algoritmo desenvolvido busca extrair a posição e orientação do marcador embarcado no veículo. Este é composto por 5 etapas:

1. Converter para espaço de cor HSV a imagem capturada.
2. Encontrar separadamente cada bola de cor do marcador.
3. Tendo encontrado cada bola, achar o centroide.
4. Calcular o centroide do marcador a partir do centroide de cada bola gerando assim a posição do marcador.
5. Calcular o ângulo de orientação do marcador.

O algoritmo D.2.1 sumariza as funções criadas usando a biblioteca *OpenCV*.

Algorithm D.2.1: Algoritmo Localização Local usando visão computacional

Entrada: Imagem I_k

Saída : Vetor de estados de Localização $\begin{bmatrix} x_{gps}^k & y_{gps}^k & \theta_{gps}^k \end{bmatrix}^T$

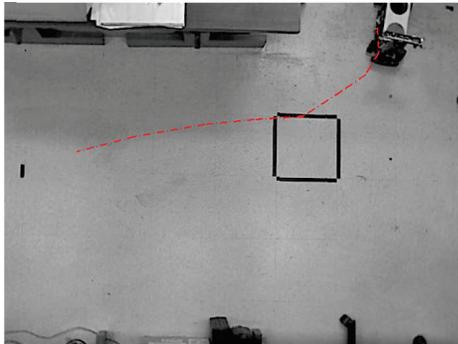
for $k \leftarrow 1$ **to** $nObservations$ **do**

```

     $I_{hsv} = \text{RGB2HSV}(I)$ 
     $I_{orange} = \text{DetectOrange}(I_{hsv})$ 
     $C_{orange} = \text{Centroid}(I_{orange})$ 
     $I_{blue} = \text{DetectBlue}(I_{hsv})$ 
     $C_{blue} = \text{Centroid}(I_{blue})$ 
     $\text{MassCenter}(C_{blue}, C_{orange})$ 
     $\text{HeadingAngle}(C_{blue}, C_{orange})$ 

```

Assim e usando os parâmetros da calibração são obtidas as coordenadas do veículo dada uma observação do sistema de posicionamento. A figura D.4 mostra um percurso gerado pelo simulador do GPS para o deslocamento do veículo baseado em 9 observações. A figura D.5 mostra a sequência de observações feitas para gerar o percurso.



(a)

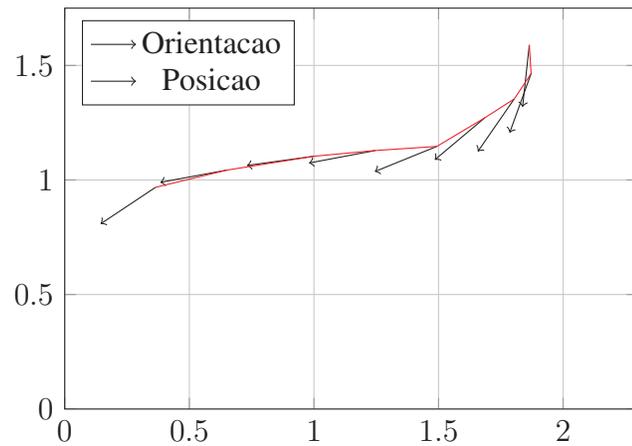


Figura D.4: Observação do GPS: (a) Percurso (b)Localização

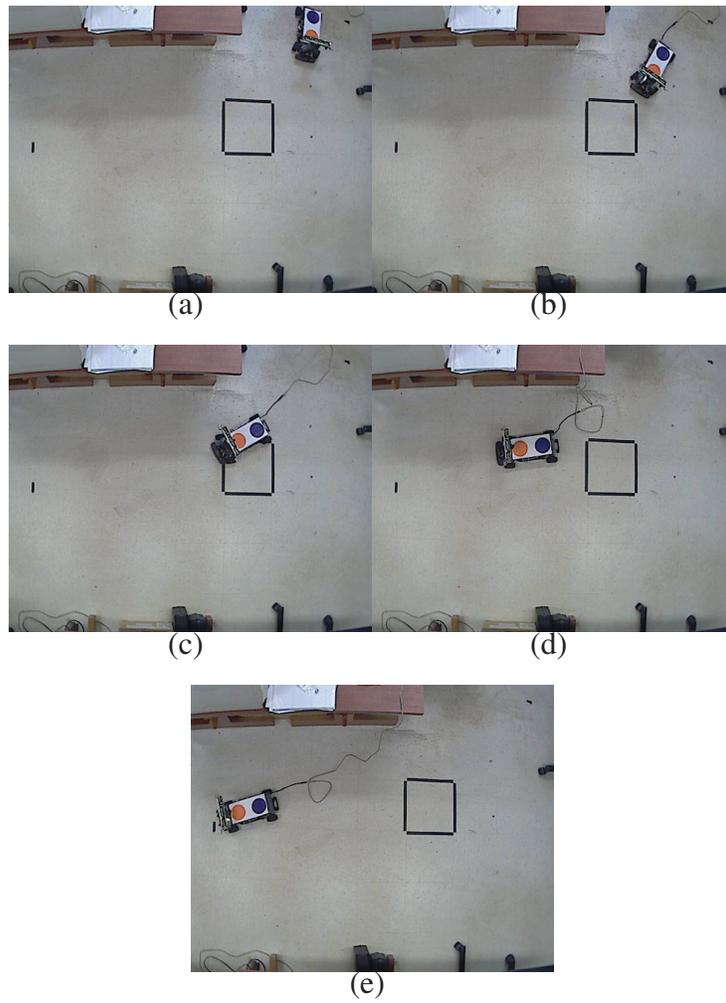


Figura D.5: Observações do sistema posicionamento local para odometria visual: (a), (b), (c), (d), (e), (f) (g).

