



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E  
DE COMPUTAÇÃO



# **Desenvolvimento de Solução para SLAM Utilizando Visão de Teto**

**Luiz Henrique Rodrigues da Silva**

Orientador: Prof. Dr. Adelardo Adelino Dantas de Medeiros

**Dissertação de Mestrado** apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e de Computação da UFRN (área de concentração: Engenharia de Computação) como parte dos requisitos para obtenção do título de Mestre em Ciências.

Número de ordem PPgEEC: M411  
Natal, RN, dezembro de 2013

UFRN / Biblioteca Central Zila Mamede  
Catalogação da Publicação na Fonte

Silva, Luiz Henrique Rodrigues da  
Desenvolvimento de solução para SLAM utilizando visão de teto. /  
Luiz Henrique Rodrigues da Silva. – Natal, RN, 2013.  
70 f.

Orientador: Prof. Dr. Aderlado Adelino Dantas de Medeiros.

Dissertação (Mestrado) – Universidade Federal do Rio Grande do  
Norte. Centro de Tecnologia. Programa de Pós-Graduação Engenharia  
Elétrica.

1. Robô móvel – Dissertação. 2. Cv-SLAM – Dissertação. 3. Filtro de  
Kalman estendido – Dissertação. 4. Manchas de luminosidade –  
Dissertação. I. Medeiros, Aderlado Adelino Dantas de. II. Universidade  
Federal do Rio Grande do Norte. III. Título.

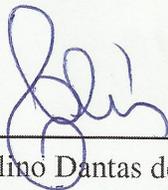
RN/UF/BCZM

CDU 621.865.8

# Desenvolvimento de Solução para SLAM Utilizando Visão de Teto

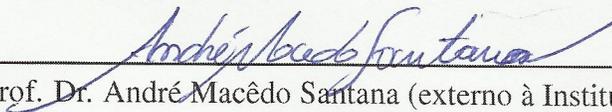
**Luiz Henrique Rodrigues da Silva**

Dissertação de Mestrado aprovada em 24 de janeiro de 2014 pela banca examinadora composta pelos seguintes membros:



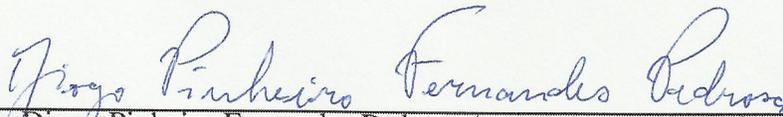
---

Prof. Dr. Adelardo Adelino Dantas de Medeiros (orientador) .... DCA/UFRN



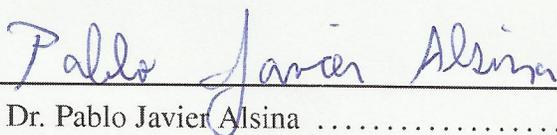
---

Prof. Dr. André Macêdo Santana (externo à Instituição) ..... UFPI



---

Prof. Dr. Diogo Pinheiro Fernandes Pedrosa (externo ao programa) .....  
DCA/UFRN



---

Prof. Dr. Pablo Javier Alsina ..... DCA/UFRN



*“A glória de Deus está nas coisas  
encobertas; mas a honra dos reis,  
está em descobri-las.” Provérbios  
25:2*



---

# Agradecimentos

---

Ao meu orientador Adelardo Adelino Dantas de Medeiros pela atenção em momentos difíceis.

Ao professor Diogo Pinheiro Fernandes Pedrosa pela imprescindível participação desde os primórdios até os derradeiros momentos deste trabalho.

Ao professor Pablo Javier Alsina que juntamente participou de várias fases do trabalho.

Aos companheiros de laboratório que estiveram mais próximos do trabalho, André Silva, Filipe Campos, Guilherme Leal, Gutemberg Santiago, João Paulo, Nicholas Bastos e Victor Menegheti.

À minha família que sempre me auxiliou durante esta jornada.

À CAPES, pelo apoio financeiro.



---

# Resumo

---

Este trabalho visa mostrar uma abordagem pouco explorada do problema de mapeamento e localização simultâneos (SLAM). Com fim de trabalhar em um ambiente fechado, uma plataforma robótica móvel deve construir um mapa do ambiente e se localizar dentro deste mapa. A plataforma robótica utilizada possui uma câmera voltada para o teto (ascendente) e odometria para as rodas. As marcas que compõem o mapa são manchas luminosas na imagem capturada pela câmera causadas pelas luminárias no teto. Este trabalho desenvolve uma solução baseada no Filtro de Kalman Estendido para o SLAM com um modelo de observação desenvolvido. Diversos testes e programas desenvolvidos para realização do SLAM são apresentados em detalhes.

**Palavras-chave:** cv-SLAM, Filtro de Kalman Estendido, manchas de luminosidade, robô móvel.



---

# Abstract

---

This work intends to show a new and few explored SLAM approach inside the simultaneous localization and mapping problem (SLAM). The purpose is to put a mobile robot to work in an indoor environment. The robot should map the environment and localize itself in the map. The robot used in the tests has an upward camera and encoders on the wheels. The landmarks in this built map are light splotches on the images of the camera caused by luminaries on the ceil. This work develops a solution based on Extended Kalman Filter to the SLAM problem using a developed observation model. Several developed tests and softwares to accomplish the SLAM experiments are shown in details.

**Keywords:** cv-SLAM, Extended Kalman Filter, light splotches, mobile robot.



---

# Sumário

---

<b>Sumário</b>	<b>i</b>
<b>Lista de Figuras</b>	<b>iii</b>
<b>Lista de Tabelas</b>	<b>v</b>
<b>Lista de Símbolos e Abreviaturas</b>	<b>vii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Histórico do SLAM . . . . .	1
1.2 Descrição do problema . . . . .	2
1.3 Motivação e Justificativa . . . . .	2
1.4 Trabalhos relacionados . . . . .	3
1.5 Organização do trabalho . . . . .	6
<b>2 Reconhecimento de Marcas no Ambiente</b>	<b>7</b>
2.1 Introdução: Possíveis abordagens . . . . .	7
2.2 Ferramentas para processamento das marcas . . . . .	8
2.3 Processamento das Marcas . . . . .	8
2.4 Definição de Propriedades das Marcas . . . . .	10
2.5 Considerações . . . . .	11
<b>3 Modelagem do SLAM</b>	<b>15</b>
3.1 O Filtro de Kalman . . . . .	15
3.2 Modelagem Cinemática . . . . .	16
3.3 Modelo de Observação . . . . .	18
3.4 Completando a definição do SLAM . . . . .	22
3.4.1 Associação de Marcas . . . . .	23
3.5 Considerações . . . . .	23
<b>4 Testes experimentais</b>	<b>25</b>
4.1 Identificação de parâmetros . . . . .	25
4.1.1 Definição de parâmetros do sistema de visão . . . . .	25
4.1.2 Calibração da câmera . . . . .	27
4.1.3 Estimação de Covariância das Marcas . . . . .	27
4.1.4 Estimação de Covariância do Deslocamento das Rodas . . . . .	27
4.2 Testes de SLAM . . . . .	28

4.2.1	Primeiro Experimento . . . . .	28
4.2.2	Segundo e Terceiro Experimentos . . . . .	30
4.2.3	Quarto Experimento . . . . .	33
4.2.4	Quinto Experimento . . . . .	37
4.3	Considerações . . . . .	38
<b>5</b>	<b>Conclusões</b>	<b>39</b>
5.1	Conclusões dos experimentos . . . . .	39
5.2	Trabalhos publicados . . . . .	40
5.3	Trabalhos futuros . . . . .	40
	<b>Referências</b>	<b>41</b>
<b>A</b>	<b>Testes com luminárias</b>	<b>45</b>
<b>B</b>	<b>Programas desenvolvidos</b>	<b>49</b>
<b>C</b>	<b>Detecção de reflexos</b>	<b>51</b>

---

# Lista de Figuras

---

1.1	Ilustração para a tarefa de SLAM . . . . .	2
1.2	Modelos de robôs de limpeza com CV-SLAM . . . . .	5
2.1	Visão sem modificações nos parâmetros de câmera. . . . .	8
2.2	Visão com mudanças nos parâmetros de câmera. . . . .	8
2.3	Esquema de processamento de imagens . . . . .	9
2.4	Aglomerção com operadores morfológicos . . . . .	9
2.5	Exemplo de processamento do sistema de visão . . . . .	10
2.6	Distribuições das coordenadas $i$ e $j$ para luminária com 4 lâmpadas . . . . .	12
2.7	Distribuições da orientação e excentricidade para luminária com 4 lâmpadas . . . . .	12
2.8	Exemplo de processamento com êxito. . . . .	13
3.1	Plataforma móvel para experimentos e sistemas de referência . . . . .	16
3.2	Plataforma de testes - visão de frente . . . . .	17
3.3	Ilustração para equação de posição do modelo de observação . . . . .	19
3.4	Esquema relacionando matrizes para a equação de posição . . . . .	20
3.5	Visão superior mostrando a relação das orientações . . . . .	20
4.1	Estimando covariância das rodas . . . . .	28
4.2	Trajeto por odometria . . . . .	29
4.3	Trajeto por filtro de Kalman . . . . .	29
4.4	Trajeto por odometria . . . . .	31
4.5	Trajeto por filtro de Kalman . . . . .	31
4.6	Posicionamento das marcas na volta 1 . . . . .	32
4.7	Posicionamento das marcas na volta 2 . . . . .	32
4.8	Posicionamento das marcas na volta 3 . . . . .	32
4.9	Posicionamento das marcas na volta 4 . . . . .	32
4.10	Convergência das marcas na última volta . . . . .	32
4.11	Covariância da segunda marca na primeira volta. . . . .	33
4.12	Covariância da segunda marca na quinta volta. . . . .	33
4.13	Covariância da terceira marca na primeira volta. . . . .	33
4.14	Covariância da terceira marca na quinta volta. . . . .	33
4.15	Resultado com odometria para quatro voltas . . . . .	34
4.16	Trajetos calculados com FKE para quatro voltas . . . . .	34
4.17	Covariância de posição do robô durante trajeto do quarto experimento . . . . .	35
4.18	Covariância de orientação do robô no quarto experimento . . . . .	35
4.19	Posicionamento do robô no quarto experimento . . . . .	36

4.20	Orientação do robô no quarto experimento . . . . .	36
4.21	Trajeto por odometria . . . . .	37
4.22	Trajeto por SLAM com FKE . . . . .	37
4.23	Trajeto com FKE com mapa pronto . . . . .	38
A.1	Luminária com ressalto e duas lâmpadas . . . . .	45
A.2	Luminária sem ressalto e duas lâmpadas . . . . .	45
A.3	Luminária sem ressalto com quatro lâmpadas . . . . .	45
A.4	Luminária com espelhamento . . . . .	45
A.5	Distribuição de $i$ da luminária com ressalto . . . . .	46
A.6	Distribuição de $j$ da luminária com ressalto . . . . .	46
A.7	Distribuição do ângulo da luminária com ressalto . . . . .	46
A.8	Distribuição da excentricidade da luminária com ressalto . . . . .	46
A.9	Distribuição de $i$ da luminária sem ressalto . . . . .	46
A.10	Distribuição de $j$ da luminária sem ressalto . . . . .	46
A.11	Distribuição do ângulo da luminária sem ressalto . . . . .	47
A.12	Distribuição da excentricidade da luminária sem ressalto . . . . .	47
A.13	Distribuição de $i$ da luminária com espelhamento . . . . .	47
A.14	Distribuição de $j$ da luminária com espelhamento . . . . .	47
A.15	Distribuição do ângulo da luminária com espelhamento . . . . .	47
A.16	Distribuição da excentricidade da luminária com espelhamento . . . . .	47
A.17	Distribuição de $i$ da luminária com 4 lâmpadas . . . . .	47
A.18	Distribuição de $j$ da luminária com 4 lâmpadas . . . . .	47
A.19	Distribuição do ângulo da luminária com 4 lâmpadas . . . . .	48
A.20	Distribuição da excentricidade da luminária com 4 lâmpadas . . . . .	48
B.1	Diagrama com os principais programas desenvolvidos . . . . .	50

---

# Lista de Tabelas

---

2.1	Definição das propriedades das marcas . . . . .	11
4.1	Parâmetros fixos do robô . . . . .	25
4.2	Variâncias de marca geradas com 800 amostras . . . . .	26



---

# Lista de Símbolos e Abreviaturas

---

- $(x_R, y_R, \theta_R)$  Coordenadas que definem a posição e orientação do robô no referencial de mundo.
- $(x_c, y_c)$  Coordenadas de algum ponto no teto tendo a câmera como referencial (metros) .
- $(x_m, y_m, \alpha_m)$  Coordenadas de uma marca no referencial de mundo.
- $A$  Matriz de homografia, relaciona dois planos.
- $C$  Vetor que relaciona o referencial de câmera com o referencial do robô.
- $G$  Jacobiano do modelo cinemático derivado em relação à pose do robô .
- $H_{ik}$  Jacobiano do modelo de observação em relação ao vetor de estado  $\mu_k$ .
- $M$  Matriz que modela as covariâncias de distâncias percorridas pelas rodas motoras do robô.
- $Q$  Matriz que modela as incertezas das marcas observadas.
- $T$  Matriz de transformação de pontos em referencial de robô para referencial de mundo e vice-versa.
- $V$  Jacobiano do modelo cinemático derivado em relação ao sinal de controle do robô  $\mathbf{u}_k$ .
- $\Sigma_t$  Matriz de covariância do vetor de estado  $\mu_k$ .
- $\bar{x}, \bar{y}$  coordenadas horizontal e vertical do centro de massa de uma dada região de uma imagem binária.
- $\mu_k$  Vetor de estado do filtro de Kalman contendo a pose do robô  $\mathbf{p}_k$  e o mapa de marcas  $\mathbf{m}$ .
- $\mathbf{m}$  Mapa de marcas que o robô já observou, são armazenadas em referencial de mundo.
- $\mathbf{p}_k$  vetor *pose* do robô formado por  $x_R, y_R$  e  $\theta_R$ .
- $\mathbf{u}_k$  vetor de controle com as distâncias a serem percorridas pelas rodas motoras em um certo instante.

- $\mathbf{z}_k$  vetor de marcas observadas pelo sensor em um determinado instante.
- $\sigma_i$  Covariância da coordenada horizontal do centro de uma marca luminosa na imagem.
- $\sigma_j$  Covariância da coordenada vertical do centro de uma marca luminosa na imagem.
- $b$  Distância entre as rodas motoras do robô, normalmente nomeada *base*.
- $g()$  Modelo cinemático definido como função matemática.
- $h()$  Modelo de observação definido como função matemática.
- $i$  coordenada vertical de uma imagem.
- $j$  coordenada horizontal de uma imagem.
- $m_{pq}$  momento de ordem  $p + q$  de uma imagem.

DCA-UFRN Departamento de Engenharia de Computação e Automação da UFRN.

FKE Filtro de Kalman Estendido.

ICP *Iterated closest point*.

PTAM *Parallel Tracking and Mapping*.

RGBD Característica de câmeras que além de cores também captam a profundidade para cada pixel - *Red Green Blue Depth*.

SLAM *Simultaneous Localization and Mapping*.

---

# Capítulo 1

## Introdução

---

### 1.1 Histórico do SLAM

O SLAM (do inglês *Simultaneous Localization and Mapping*) é um dos problemas mais estudados no campo da robótica e tem uma história recente. No início da década de 80 muitos pesquisadores estudavam métodos determinísticos para solucionar a localização de um robô assim como o mapeamento que o robô fazia do ambiente. No final dos anos 80, percebeu-se a necessidade de trabalhar com soluções probabilísticas para o mapeamento. Qualquer que fosse a aplicação do robô autônomo, a sua realização dependia de um mapeamento probabilístico consistente. Esta foi a conclusão que pesquisadores precursores da área chegaram após discussões.

Em seguida uma série de trabalhos fundamentais foram publicados. Por exemplo, Smith e Cheeseman [Smith & Cheeseman 1986] e Durrant-Whyte [Durrant-Whyte 1988] que mostraram como os marcos de um mapa estão relacionados estatisticamente. Chegou-se à conclusão que a correlação entre marcos (do mapa) aumentava à medida que fossem feitas novas observações dos marcos.

Paralelamente alguns outros pesquisadores como Raja Chatila, Oliver Faugeras, Nicholas Ayache, Crowley e Laumond trabalhavam em navegação com diferentes métodos como visão computacional (Faugeras e Ayache [Ayache & Faugeras 1988]) e sonares (Raja Chatila e Laumond [Chatila & Laumond 1985] e Crowley [Crowley 1989]).

Pouco tempo depois Smith desenvolveu um trabalho [Smith et al. 1987] mostrando que no mapeamento de um ambiente desconhecido, todos os marcos estão necessariamente correlacionados por causa do erro de localização do robô. A partir deste ponto percebeu-se importância de solucionar o problema de localização simultaneamente ao problema de mapeamento pois estão intrinsecamente relacionados.

Alguns trabalhos [Leonard & Durrant-Whyte 1992] [Rencken 1993] foram publicados na tentativa de dividir a solução em várias soluções menores devido à complexidade computacional e às condições de convergência da solução (desconhecidas naquele tempo).

Os primeiros resultados [Castellanos et al. 1998] [Deans & Hebert 2000] [Guivant et al. 2000] [Leonard & Feder 2000] só apareceram quando as variáveis de ambos os problemas eram estimadas como pertencendo à um único problema. Isto aconteceu a partir de meados da década de 90 quando alguns grupos de pesquisa se dedicaram ao problema. Neste mesmo período foi desenvolvida a teoria básica da convergência do SLAM

[Csorba 1997]. Csorba que desenvolveu a teoria e também apresentou seus resultados [Csorba et al. 1996].

## 1.2 Descrição do problema

O problema a ser tratado nesta dissertação é uma abordagem pouco usual do v-SLAM (*visual SLAM*) onde se utiliza uma câmera como sensor para detecção do ambiente. Geralmente as câmeras dos robôs são apontadas para o chão [Frese 2004] [Santana 2011] ou para a frente do robô de forma a delimitar o espaço de trabalho do robô além de evitar colisões com objetos em movimento. Na abordagem adotada neste trabalho, o robô utiliza uma câmera (apontada para o teto do ambiente) como principal sensor para o SLAM. As luminárias acesas do ambiente são utilizadas como marcos do mapa criado. O robô é posicionado dentro de um ambiente interno que é desconhecido para o robô. De acordo com as observações das marcas (luminárias) no teto do ambiente o robô deve criar um mapa com a posição de cada luminária e se situar dentro deste ambiente. A figura 1.1 ilustra o problema a ser tratado.

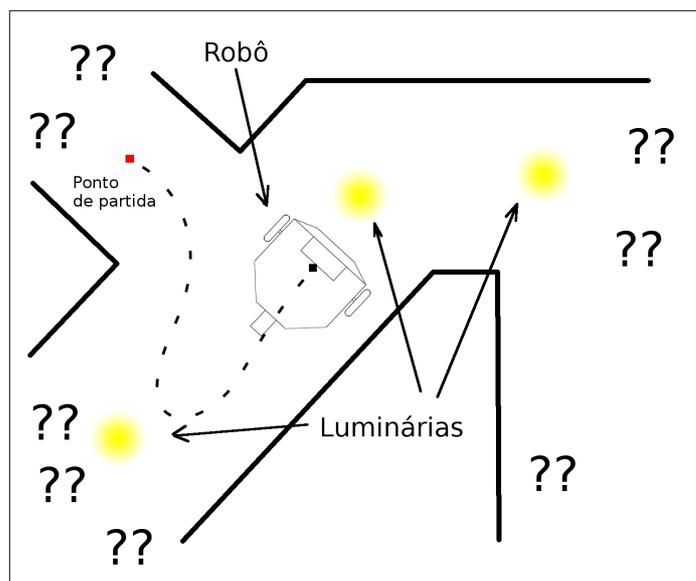


Figura 1.1: Ilustração para a tarefa de SLAM

## 1.3 Motivação e Justificativa

A área de pesquisa SLAM (*Simultaneous Localization and Mapping*) dentro da robótica cresceu muito na última década. Soluções apresentadas recentemente foram empregadas em diversos ambientes e com uma variedade de sensores. Na classe de v-SLAM houve um crescimento significativo, já na abordagem de câmeras voltadas para o teto foi

encontrado um número de trabalhos. Neste contexto está o trabalho proposto nesta dissertação: a construção de um mapa 2D através de *cv-SLAM* (*ceiling vision SLAM*) tomando como marcos do ambiente, as luminárias fixadas no teto. A motivação do uso destas marcas se deve ao fato de serem estáticas e pela fácil detecção utilizando câmeras. Existem poucas situações onde algum objeto possa obstruir a observação das marcas.

Estas vantagens da abordagem levam à construção de um mapa métrico com fácil interpretação para um ser humano. Por parte da revisão literária notou-se um escasso número de trabalhos com *cv-SLAM* para produção de um mapa com luminárias como marcas. A maioria dos trabalhos utiliza algoritmos detectores de *features* como marcas na solução do SLAM do robô. A seguir os trabalhos mais próximos da proposta são apresentados.

## 1.4 Trabalhos relacionados

A maior parte dos trabalhos relacionados utiliza o filtro de Kalman como solução para o SLAM. Alguns trabalhos [Hwang & Song 2011],[Jeong & Lee 2005] adotaram marcas menos convencionais. No artigo de Hwang & Song (2011) os autores adotam uma câmera apontada para o teto do ambiente interno com o objetivo de usar quinas de teto, aberturas de porta e lâmpadas como marcas a serem rastreadas pelo SLAM tradicional que é baseado no filtro de Kalman estendido (FKE). O assunto mais enfatizado é a detecção das marcas, onde os autores se preocuparam com os custos de complexidade de técnicas mais tradicionais, então substituídas por estratégias de natureza geométrica já implementadas ou desenvolvidas pelos autores. Casos particulares também são apresentados e tratados nestas estratégias. Como resultado possuem um algoritmo SLAM com ciclo rápido de 159 milissegundos. É o trabalho mais próximo ao apresentado nesta dissertação. As diferenças residem no tipo de luminária. Hwang & Song (2011) usam luminárias embutidas (facilita o processamento) e de formato circular. As luminárias utilizadas como marcas nesta dissertação necessitam de uma orientação mas nenhum formato específico. Além disso, durante o processamento de imagem foi desenvolvida uma rotina para identificação de reflexos (muito comuns em CV-SLAM) afim de eliminar a maioria destas marcas falsas. Já o trabalho citado não apresenta um tratamento específico, apenas considera o reflexo como uma outra marca.

Outro trabalhos com abordagens similares [Jeong & Lee 2005] de testes, buscam maior precisão na localização tendo em vista aplicações para robôs de serviços domiciliares. Com a câmera voltada para o teto, o SLAM proposto usa quinas como marcas de rastreamento. As quinas de objetos nas paredes sofrem diferentes transformações das encontradas no teto pois planos não paralelos ao plano da imagem (da câmera) são deformados de acordo com a movimentação do robô. Isto requer um tratamento diferente para identificação das quinas em paredes. Ao final do trabalho apresentam um experimento prático mostrando a execução do SLAM em tempo real com dois ciclos por segundo. Este foi o primeiro trabalho a utilizar o termo *cv-slam*.

No trabalho de Wang et al. (2012) os autores buscam calcular a localização global do robô EGO, através do sensor Kinect. A estimação do movimento utiliza o algoritmo SURF (*Speeded Up Robust Features*) nas imagens adquiridas para extração de caracte-

rísticas. Logo são situadas no espaço 3D pelo uso de mapa de profundidade dos pixels fornecido pelo sensor. Para o casamento de características em imagens subsequentes fazem uso de um método baseado em árvores de k-dimensões. O método RANSAC foi aplicado a fim de filtrar casamentos errados. Assim, restando os casamentos corretos, é possível aplicar o método de decomposição em valores singulares para obter as matrizes de rotação e de translação relativas ao movimento do robô. Posteriormente, os autores pretendem integrar dados de outros sensores do robô.

O trabalho de Rusdinar et al. (2012) experimenta marcas artificiais no teto para correção da localização de um robô móvel que além da câmera, tem a bordo dois computadores, um dedicado ao processamento de imagem e outro ao processo de localização, baseado no FKE realizando as etapas de atualização e predição. Nesta última são incluídos os dados de odometria de suas rodas.

O trabalho de Xu et al. (2009) apresenta uma fundamentação teórica para o posicionamento de robô baseado numa câmera voltada para o teto. No texto abordam a modelagem da câmera e seus parâmetros para transpor pontos de interesse de coordenadas de imagem para coordenadas de mundo, a estimação da pose inicial e posicionamento baseado nas características detectadas. Ao final apresentam uma análise sobre os erros de posição e orientação, e a influência de uma sobre a outra.

Alguns trabalhos [Scherer et al. 2012], [Choi et al. 2012] apresentam aprimoramentos no filtro de Kalman estendido. As marcas utilizadas no trabalho de Choi et al. (2012) são as linhas que formam o teto. No artigo definem o conceito de *double update* para o filtro de Kalman de forma a obter resultados melhores que a formulação tradicional. A etapa de *update* do filtro é acrescida de uma segunda atualização além de condicionar as linhas de forma que fiquem paralelas ou perpendiculares entre si. No artigo são apresentados gráficos de simulação e experimentos reais mostrando uma melhoria no resultado do SLAM. O artigo de Scherer et al. (2012) apresenta uma incorporação ao SLAM monocular de profundidade dos pixels utilizando como sensor primário o Kinect. A implementação é baseada na biblioteca PTAM (*Parallel Tracking and Mapping*) com adaptações para inclusão da informação de profundidade. Esta modificação torna a técnica de SLAM proposta por eles mais flexível, utilizando os dados de profundidade quando disponíveis. Na ausência de profundidade, a solução desenvolvida se torna similar ao SLAM monocular tradicional.

Uma forma de localização em robótica móvel em pesquisa é a odometria visual [Wang et al. 2013]. Wang et al. (2013) usa uma câmera RGBD (*Red Green Blue Depth*) apontada para o teto para realizar uma odometria visual. O teto possui um padrão de linhas que facilita a estimação do movimento entre imagens consecutivas. O cálculo do movimento é baseado no algoritmo ICP (*Iterated closest point*) e utiliza as arestas encontradas na nuvem de pontos coletada pela câmera RGBD. De forma geral, este modo de odometria que tem um custo computacional elevado, mostra uma acurácia maior em relação à odometria com *encoders*.

Entre os trabalhos de robótica móvel com câmera voltada para o teto existem aqueles de robótica cooperativa [Chen et al. 2010],[Chen et al. 2009]. No trabalho de Chen et al. (2010) três robôs são utilizados. As quinas presentes no teto são o tipo de marca utilizada. Elas são extraídas pelo algoritmo de Harris. É utilizada odometria visual em

vez de odometria por *encoders*. O filtro de Kalman estendido é utilizado para solução do SLAM. Três estratégias de SLAM são apresentadas, visto que os robôs possuem comunicação. A primeira estratégia é um robô líder fazer o cálculo do SLAM para obter sua localização. Os robôs seguidores realizam uma associação de marcas entre conjunto que observou e o conjunto que o líder observou. Com a diferença na disposição das marcas entre os dois observadores pode-se calcular a posição do seguidor em relação ao líder. Na segunda estratégia cada robô realiza o seu próprio SLAM. Dada uma incerteza na pose dos robôs seguidores é realizada um ajuste no mapa de marcas do seguidor. A terceira estratégia consiste em instituir o servidor de SLAM no robô líder que recebe as marcas observadas e dados de odometria de todos robôs e constrói um mapa único. Este servidor fica responsável por devolver a posição para cada robô. Os autores apresentam resultados experimentais obtidos de cada estratégia para uma análise comparativa.

Uma das aplicações mais diretas para o presente trabalho é a limpeza de ambientes domésticos. Uma grande parte de robôs de limpeza (baseados em sucção com ou sem dispersão de água) fazem uso do CV-SLAM para se localizar e explorar o ambiente. Atualmente existem grandes indústrias investindo no avanço tecnológico destes robôs. Alguns modelos conhecidos são da série Roomba da empresa iRobot. Outros são da linha Hom bot da empresa LG e outros mais da linha NaviBot da Samsung.



Figura 1.2: Modelos de robôs de limpeza com CV-SLAM <sup>1</sup>

Alguns trabalhos visam solucionar problemas específicos dentro do SLAM [Lee et al. 2012], [Fukutani et al. 2010]. Os autores Lee et al. (2012) mostram como resolver o problema de robô sequestrado, para um destes robôs comerciais de limpeza. O modelo utiliza uma câmera voltada para o teto para localização do robô e mapeamento do ambiente (cvSLAM), como os modelos na figura 1.2. O trabalho mostra como é possível a detecção de situações em que robô é levantado do chão. Quando o robô volta ao chão é realizada uma complexa etapa de identificação da posicionamento do robô no ambiente. A representação de possíveis posições para o robô são armazenadas em nós dentro de um grafo.

Um levantamento foi realizado por Liu & Wang (2013) mostrando os principais aspectos de pesquisa tecnológica destes robôs de limpeza doméstica através do número de

<sup>1</sup>Imagens acessadas no dia 15/11/2013 nos endereços  
<http://www.lg.com/fr/aspirateurs/lg-VR1227R>,  
<http://irobot.pl/pl/roomba/208-irobot-roomba-790-5060155406091.html> e  
<http://www.samsung.com/uk/consumer/home-appliances/vacuum-cleaner/robot/VCR8855L4B/XEU>

patentes. O estudo mostra um crescente interesse na última década (2000-2010) por parte das empresas na área P&D para estes robôs. Dentre os assuntos em estudo, a localização do robô continua recebendo muita atenção.

## **1.5 Organização do trabalho**

O segundo capítulo apresenta como é realizada a detecção das marcas através do sistema de visão. O terceiro capítulo mostra como foi modelado o filtro de Kalman e implementado. O quarto capítulo é dedicado aos resultados experimentais e sua análise. O quinto capítulo finaliza o trabalho com as conclusões extraídas e trabalhos futuros.

---

## Capítulo 2

# Reconhecimento de Marcas no Ambiente

---

### 2.1 Introdução: Possíveis abordagens

Durante o desenvolvimento inicial do processamento de imagens para extração das marcas, encontraram-se duas possibilidades. Através de ajustes nos parâmetros da câmera é possível obter uma imagem sem saturação na região da lâmpada acesa, e assim observar o formato exato da lâmpada (figura 2.2). Alguns destes parâmetros são: taxa de imagens por segundo (*fps*), tempo de exposição (*shutter*), ganho, resolução, brilho, contraste, entre outros. Isto pode propiciar uma detecção de lâmpadas queimadas desde que conhecida a quantidade de lâmpadas numa luminária. Outra vantagem desta abordagem é a eliminação de falsos positivos uma vez que toda região em torno da lâmpada fica completamente escura na imagem. Os reflexos das lâmpadas não são observados. Notou-se nesta abordagem que o ajuste dos parâmetros envolvidos influenciam consideravelmente na qualidade da marca. Desta forma ambientes com alguma diferença entre as luminárias (como marca de lâmpadas, temperatura, formato, espelhos nas luminárias...) trazem a possibilidade de detecção com ótima qualidade de algumas luminárias, enquanto outras correm risco de não serem identificadas por terem alguma diferença.

Em uma outra abordagem mais “conservadora” não existem lâmpadas como marcas, mas sim regiões saturadas pelo brilho da lâmpada. Nos experimentos de mapeamento viu-se que esta abordagem apresenta uma tolerância maior em relação às diferentes luminárias. Nesta abordagem pode-se extrair as mesmas características que a abordagem anterior pode fornecer. Esta abordagem é dita conservadora pois os parâmetros ajustados estão mais próximos de seus valores padrão. Desta forma uma luminária com quatro lâmpadas é vista como uma única marca, uma região brilhante de onde se extraem as propriedades (figura 2.1). Na abordagem anterior teríamos quatro marcas de onde se extrairiam as propriedades com valores muito similares (redundância).

Além da maior tolerância citada na segunda abordagem temos uma facilidade muito maior na etapa de casamento de marcas. Se considerarmos lâmpadas como marcas, identificar cada lâmpada em uma luminária (com quatro lâmpadas por exemplo) com propriedades muito similares será uma tarefa mais complexa que identificar uma única região brilhante daquela luminária.

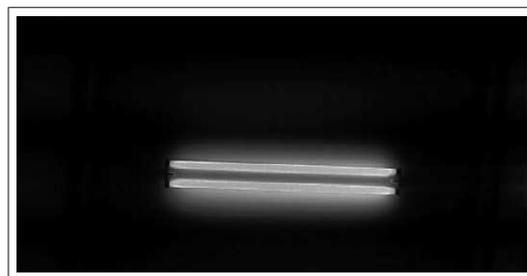
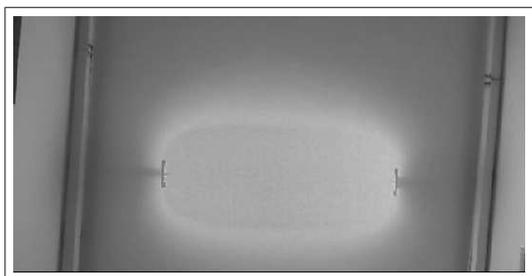


Figura 2.1: Visão sem modificações nos parâmetros de câmera.

Figura 2.2: Visão com mudanças nos parâmetros de câmera.

## 2.2 Ferramentas para processamento das marcas

Para a realização do processamento foram utilizadas operações morfológicas [Gonzalez & Woods 2000] e filtragem (convolução de imagens). Também foi desenvolvida uma rotina para identificação de reflexos [da Silva et al. 2013a].

As operações morfológicas mais difundidas são erosão e dilatação. As operações morfológicas assim como a convolução de imagens, utiliza duas imagens para produção de uma imagem resultante. As operações morfológicas são definidas para imagens binárias, mas podem ser estendidas para imagens em escala de cinza. Uma das imagens geralmente possui pequena dimensão e um formato geométrico simples. Ela é chamada *elemento estruturante*. A outra imagem (imagem de entrada) sofrerá a transformação. Em qualquer operação o elemento estruturante varre toda imagem de entrada com seus pixels sendo operados com os pixels da imagem de entrada (operações de conjunto) produzindo algum valor para a imagem transformada (ou imagem resultante). Na operação de erosão, a imagem resultante tem pixels brancos no centro do elemento estruturante nas posições em que o elemento caiba dentro das regiões brancas na imagem de entrada, isto causa uma diminuição no tamanho destas regiões, por isto o nome “erosão”. Na operação de dilatação qualquer ponto branco da imagem de entrada onde o centro do elemento passar, na imagem resultante, a mesma região será preenchida com o formato do elemento. Esta operação causa um aumento na imagem de entrada, por isso o nome “dilatação”.

A filtragem aplicada no início do processamento tem objetivo de diminuir o ruído na imagem. Isto é feito com um filtro passa-baixa chamado de *média* pois a máscara da convolução possui um mesmo valor em todos os pixels. O efeito prático da aplicação do filtro é que a imagem resultante fica borrada.

## 2.3 Processamento das Marcas

O Processamento de imagem é formado por uma série de etapas como mostrado na figura 2.3.

A imagem capturada possui dimensões de 640 pixels de largura por 480 pixels de altura. A imagem inicialmente é convertida para escala de cinza, em seguida é filtrada com um filtro média (passa-baixa). Após o filtro, a imagem é binarizada pela intensidade dos pixels. Os pixels que tiverem valor abaixo de um limiar são “pretos”, já valores de

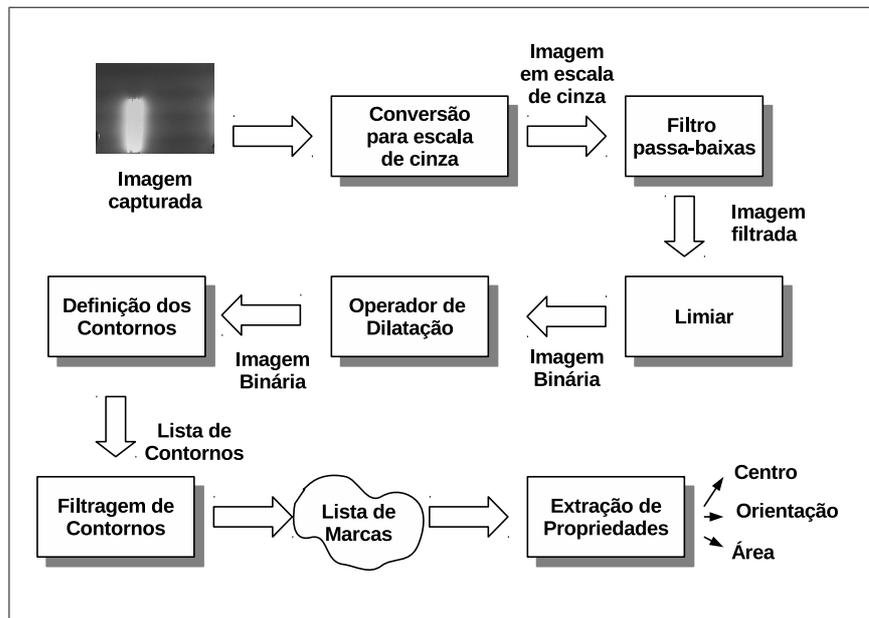


Figura 2.3: Esquema de processamento de imagens

pixels acima do limiar são considerados “brancos”. A próxima etapa é utilizar o operador morfológico de dilatação. Desta forma podemos aglomerar fragmentos de uma mesma luminária. O elemento estruturante utilizado foi um círculo.

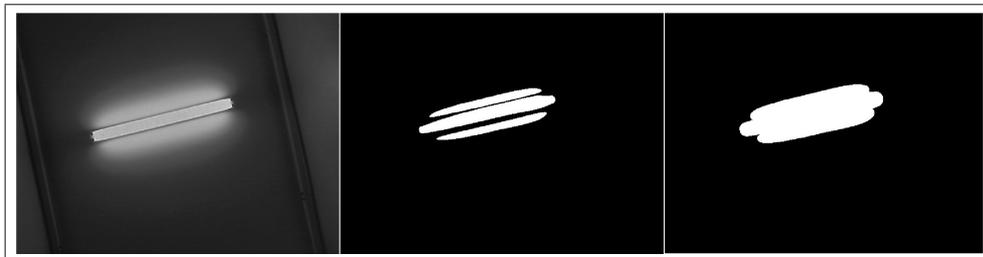


Figura 2.4: Aglomeração com operadores morfológicos

Em seguida é obtido o contorno das regiões iluminadas. Isto é feito formando uma lista de posição de todos os pixels limítrofes da região iluminada. Todos os contornos ficam armazenados em uma lista. A última tarefa no processamento é a filtragem dos contornos, para descarte de contornos que não são luminárias. Com a lista contendo apenas contornos de luminárias, pode-se avançar para a extração das propriedades de interesse, para uso no filtro de Kalman.

A filtragem dos contornos elimina primeiramente contornos internos à outros contornos. Contornos com tamanho (número de pixels do contorno) inferior a um certo limite também são descartados. Em seguida uma rotina simples de detecção de reflexos (Apêndice C) é executada para detecção de contornos que são reflexos. A rotina foi desenvolvida

para facilitar o tratamento *off-line* necessário das imagens a fim de aplicar o filtro de Kalman. A rotina ajuda na eliminação da maioria dos reflexos. Por fim os contornos que são vistos parcialmente são descartados. Esta última etapa dos descartes é discutível devido ao alto número de descartes que produz, ou seja, informações são perdidas. Um exemplo deste processamento é mostrado na figura 2.5.

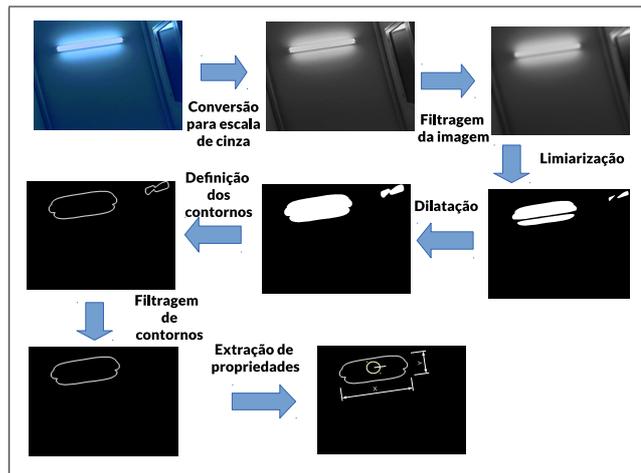


Figura 2.5: Exemplo de processamento do sistema de visão

Para cada uma das etapas apresentadas existe no mínimo um parâmetro a ser ajustado. Além destes existem os parâmetros da câmera. Notou-se então, uma necessidade de encontrar os parâmetros que influenciam na detecção de propriedades das marcas com maior constância. Foram achados três parâmetros principais: tempo de exposição (parâmetro da camera), tamanho da máscara de filtragem (segunda etapa do processamento) e o limiar de binarização da imagem (terceira etapa). Identificados estes parâmetros foi elaborada uma tabela (4.2) a fim de definir os melhores valores para estes parâmetros. Os demais parâmetros tanto de câmera como das etapas de processamento foram estipulados e fixados em testes durante o desenvolvimento do processamento de imagem.

## 2.4 Definição de Propriedades das Marcas

As propriedades extraídas da marca são ponto central, orientação, excentricidade e área. As propriedades são extraídas utilizando o conceito de *momentos*. Com uma imagem em escala de cinza definida como  $I(x, y)$ , o seu momento de ordem  $p + q$  é definido na equação 2.1.

$$m_{pq} = \sum_x \sum_y x^p y^q I(x, y) \quad (2.1)$$

Para uma imagem binária, como na etapa de extração de propriedades, a função  $I(x, y)$  retorna 0 ou 1.

Propriedades	Definição
Centro da região	$\bar{x}, \bar{y}$
Orientação	$\frac{1}{2} \arctan \left( \frac{\mu_{11}}{\mu_{20} - \mu_{02}} \right)$
Excentricidade	$\frac{\mu_{02} \cos^2(\theta) + \mu_{20} \sin^2(\theta) - \mu_{11} \sin^2(\theta)}{\mu_{02} \sin^2(\theta) + \mu_{20} \cos^2(\theta) - \mu_{11} \cos^2(\theta)}$

Tabela 2.1: Definição das propriedades das marcas

Dada a região contígua de  $n$  pixels com luz (pixels com valor 1), podemos definir algumas propriedades em função dos momentos centrais (equação 2.3) desta região partindo do centro (equação 2.2) da região em questão.

$$\bar{x} = \frac{m_{01}}{m_{00}} \quad \bar{y} = \frac{m_{10}}{m_{00}} \quad (2.2)$$

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x, y) \quad (2.3)$$

A tabela 2.4 resume as propriedades extraídas de cada marca.

Estas formulações utilizando *momentos* podem ser encontradas na literatura [Fabrizi & Saffiotti 2000].

Acerca das marcas foram realizados alguns testes para verificar se o formato da distribuição das propriedades extraídas pelo processamento são similares à distribuição normal requerida pelo filtro de Kalman. O ruído de observação presente no filtro de Kalman deve ter distribuição normal. As figuras 2.6 e 2.7 mostram a distribuição das propriedades extraídas. Deste ponto em diante as variáveis  $i$  e  $j$  serão utilizadas para representar as coordenadas vertical e horizontal respectivamente de algum ponto em particular na imagem e em vez de  $y$  e  $x$  utilizadas para formalizar uma imagem em escala de cinza. Embora a excentricidade não seja utilizada na implementação do filtro também é apresentada na figura 2.7. Foram realizados testes com outros três tipos de luminárias. Apêndice A mostra todos os gráficos de distribuições para os quatro tipos de luminárias.

Através destes gráficos e dos gráficos das outras luminárias foi possível notar uma diferença entre as distribuições das coordenadas  $i$  e  $j$ .

## 2.5 Considerações

O processamento de imagem apresentou resultados satisfatórios. Vários testes específicos foram realizados para o processamento alcançar uma boa qualidade de detecção de marcas. Mesmo com algum nível de obstrução foi possível a identificação de luminárias. A figura 2.8 mostra uma situação onde a marca foi detectada mesmo com uma canaleta obstruindo a visão além do reflexo corretamente identificado em um extintor de incêndio.

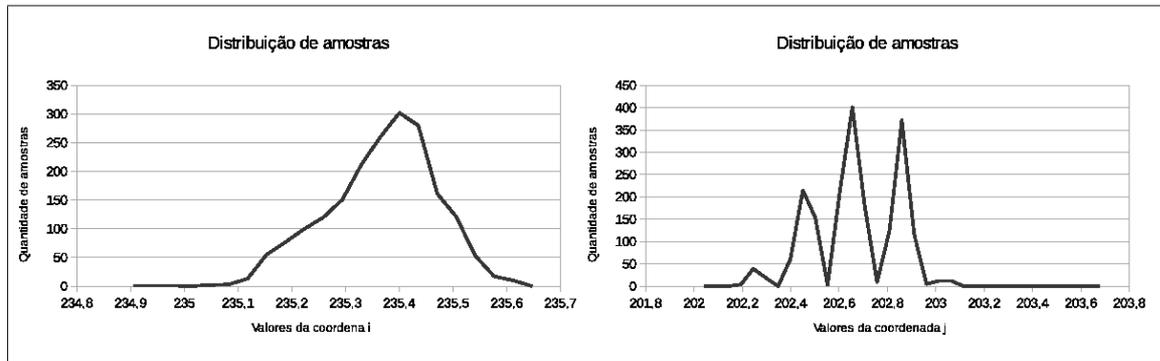


Figura 2.6: Distribuições das coordenadas i e j para luminária com 4 lâmpadas

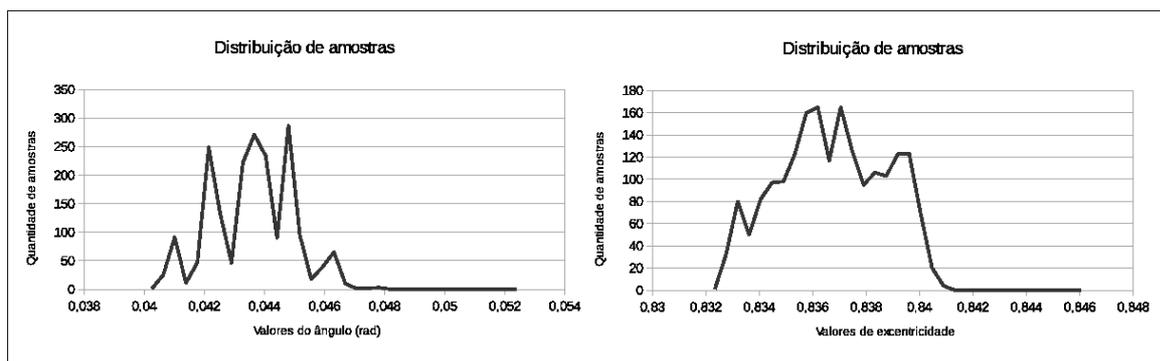


Figura 2.7: Distribuições da orientação e excentricidade para luminária com 4 lâmpadas



Figura 2.8: Exemplo de processamento com êxito.

Com as propriedades das marcas extraídas, o próximo passo é calcular a transposição destas propriedades para o referencial de mundo, isto será apresentado no capítulo 3.



---

# Capítulo 3

## Modelagem do SLAM

---

Neste capítulo o filtro de Kalman é apresentado como solução para o problema proposto. As modelagens matemáticas da observação e do processo (cinemática) são apresentadas em detalhes.

### 3.1 O Filtro de Kalman

O filtro de Kalman possui natureza estatística. Formalmente é um estimador de variáveis de um processo dinâmico linear que é perturbado por um ruído branco gaussiano, com medições linearmente relacionadas com as variáveis também corrompidas por ruído branco gaussiano. Devido à sua natureza estatística, teve uma ampla aceitação na engenharia. As equações que caracterizam o filtro de fato incorporam o tratamento de ruídos.

No desenvolvimento do trabalho foram usados alguns parâmetros de incerteza. Isto ocorre nos modelos cinemático e de observação, o que traz uma estimativa aprimorada dos parâmetros (vetor de estado).

O filtro de Kalman pode ser visto como um método recursivo, que depende do estado anterior, tanto para o cálculo do novo estado, como da sua incerteza.

O filtro de Kalman estendido possui modificações no modelo cinemático e no modelo de observação para abranger problemas onde tais modelos são não-lineares. Para isto, aplica-se uma linearização dos modelos (tanto no cinemático como de observação) utilizando a Série de Taylor afim de aplicarmos o filtro de Kalman na situação. A cada iteração temos um novo ponto de operação em torno do qual os modelos são linearizados. Isto acarreta algum custo computacional.

O algoritmo 1 sintetiza a estimação feita através do filtro de Kalman estendido. As entradas são o vetor de estado e sua covariância em um instante anterior. Os resultados do algoritmo são a atualização do vetor de estado e sua covariância com auxílio das medições realizadas (neste trabalho são marcas observadas) neste período por um sensor, denotadas pelo vetor  $\mathbf{z}$  e pelo *vetor de controle*  $\mathbf{u}$  que contém variáveis de entrada do processo (neste trabalho são as distâncias que as rodas motoras devem percorrer). A principal característica do algoritmo é que corrige a estimativa através de medições feitas com maior certeza. As matrizes sem definição presentes no algoritmo, serão apresentadas nas próximas seções.

**Algoritmo 1** FILTRO DE KALMAN**Entrada:**  $\mu_{k-1}, \Sigma_{k-1}, \mathbf{u}_k, \mathbf{z}_k$ 

- 1:  $\bar{\mu}_k = g(\mu_{k-1}, \mathbf{u}_k)$ ;
- 2:  $R_k = V \cdot M \cdot V^T$ ;
- 3:  $\bar{\Sigma}_k = G_k \cdot \Sigma_{k-1} \cdot G_k^T + R_k$ ;
- 4:  $S_{ik} = H_{ik} \cdot \bar{\Sigma}_k \cdot H_{ik}^T + Q_k$ ;
- 5:  $K_k = \bar{\Sigma}_k \cdot H_{ik}^T \cdot S_{ik}^{-1}$ ;
- 6:  $\mu_k = \bar{\mu}_k + K_k \cdot (\mathbf{z}_k - h(\bar{\mu}_k))$ ;
- 7:  $\Sigma_k = (I - K_k \cdot H_{ik}) \cdot \bar{\Sigma}_k$ ;

**Retorno:**  $\mu_k, \Sigma_k$ 

### 3.2 Modelagem Cinemática

A plataforma móvel que foi utilizada para os testes possui duas rodas motoras e um rodízio de apoio, pode ser vista nas figuras 3.1 e 3.2. O acionamento das rodas é diferencial.

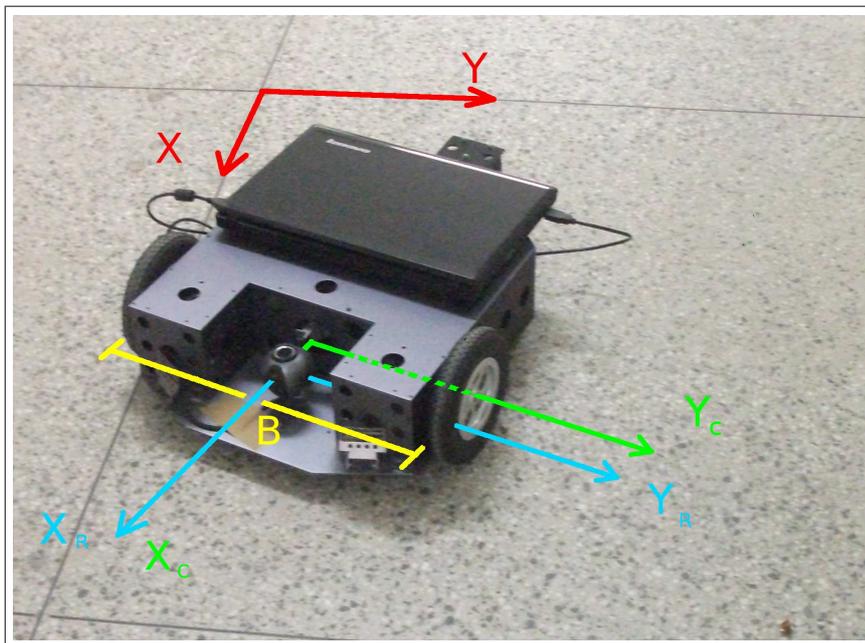


Figura 3.1: Plataforma móvel para experimentos e sistemas de referência

O modelo cinemático serve para o cálculo da pose do robô  $\mathbf{p}_{k+1}$ , conhecendo a pose anterior  $\mathbf{p}_k$  e o sinal controle  $\mathbf{u}_k$  aplicado aos atuadores de movimento do robô [Siegwart et al. 2011]. Neste trabalho em particular, o sinal de controle seria de um valor aplicado ao motor esquerdo e outro ao motor direito. Sabendo que as rodas nem sempre giram a

quantia informada no sinal de controle, optou-se por utilizar os valores dos *encoders* de cada roda para formar o vetor  $\mathbf{u}_k$ , por serem sensores mecanicamente ligados às rodas, de fato contabilizam o giro de cada roda, diferente do sinal de controle que é um valor desejado (podendo não ser alcançado). O vetor *pose* possui tradicionalmente as duas coordenadas de plano ( $x$  e  $y$ ) do robô e a sua orientação  $\theta$ . Está definido na equação 3.1.

$$\mathbf{p}_k = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} \quad (3.1)$$

Supondo que a roda esquerda percorra uma distância  $l_E$  e a roda direita uma distância  $l_D$ , fica definido o sinal dos *encoders*.

$$\mathbf{u}_k = [l_E \quad l_D]^T \quad (3.2)$$

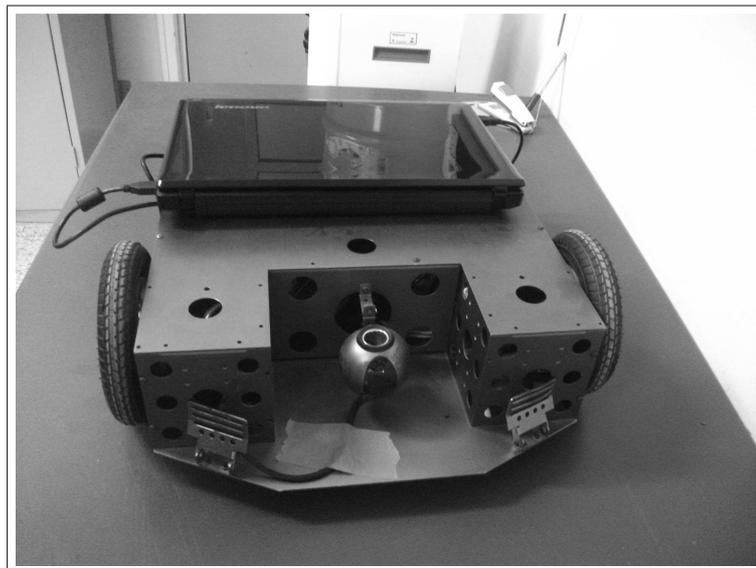


Figura 3.2: Plataforma de testes - visão de frente

Antes do modelo, serão definidas as variáveis  $\Delta s$  e  $\Delta\theta$  em 3.3 e 3.4 respectivamente. O valor  $b$  na definição 3.4 é a distância entre as rodas motoras, comumente chamada *base*.

$$\Delta\theta = \frac{l_D - l_E}{b} \quad \text{quantidade de rotação} \quad (3.3)$$

$$\Delta s = \frac{l_D + l_E}{2} \quad \text{quantidade de translação} \quad (3.4)$$

Como  $l_E$  e  $l_D$  são definidos entre o instante  $k$  e  $k+1$ , as variáveis  $\Delta\theta$  e  $\Delta s$  são definidas no mesmo escopo. Agora é possível formular o modelo de observação através de  $\mathbf{p}_{k+1}$  que é função da pose atual  $\mathbf{p}_k$  e do sinal do *encoder*  $\mathbf{u}_k$ .

$$\mathbf{p}_{k+1} = g(\mathbf{p}_k, \mathbf{u}_k) = \begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} + \begin{bmatrix} \Delta s \cos(\theta_k + \frac{\Delta\theta}{2}) \\ \Delta s \sin(\theta_k + \frac{\Delta\theta}{2}) \\ \Delta\theta \end{bmatrix} \quad (3.5)$$

Devido aos erros de sensoriamento é importante a utilização de algumas medições estatísticas. Para o modelo de posicionamento acima, pode-se desprender a covariância  $M$  na definição 3.6. Os valores  $k_r$  e  $k_l$  são constantes a serem definidas, os valores são estimados na seção 4.1.2.

$$M = covar(l_D, l_E) = \begin{bmatrix} k_r |l_D| & 0 \\ 0 & k_l |l_E| \end{bmatrix} \quad (3.6)$$

A função  $g()$  pode ser derivada em relação à pose do robô  $\mathbf{p}_k$  ou em relação ao sinal de controle  $\mathbf{u}_k$ , desta forma, defini-se o jacobiano da pose do robô por  $G_{3 \times 3}$  e a jacobiana do sinal de controle por  $V_{3 \times 2}$ . As equações 3.7 e 3.8 mostram as definições que foram implementadas.

$$\frac{\partial g}{\partial \mathbf{p}^T} = \frac{\partial g}{[\partial x \mid \partial y \mid \partial \theta]} = G = \begin{bmatrix} 1 & 0 & -\Delta s \sin(\theta + \frac{\Delta\theta}{2}) \\ 0 & 1 & \Delta s \cos(\theta + \frac{\Delta\theta}{2}) \\ 0 & 0 & 1 \end{bmatrix} \quad (3.7)$$

$$\frac{\partial g}{\partial \mathbf{u}^T} = \frac{\partial g}{[\partial l_E \mid \partial l_D]} = V = \begin{bmatrix} \frac{1}{2} \cos(\theta + \frac{\Delta\theta}{2}) - \frac{\Delta s}{2b} \sin(\theta + \frac{\Delta\theta}{2}) & \frac{1}{2} \cos(\theta + \frac{\Delta\theta}{2}) + \frac{\Delta s}{2b} \sin(\theta + \frac{\Delta\theta}{2}) \\ \frac{1}{2} \sin(\theta + \frac{\Delta\theta}{2}) + \frac{\Delta s}{2b} \cos(\theta + \frac{\Delta\theta}{2}) & \frac{1}{2} \sin(\theta + \frac{\Delta\theta}{2}) - \frac{\Delta s}{2b} \cos(\theta + \frac{\Delta\theta}{2}) \\ \frac{1}{b} & -\frac{1}{b} \end{bmatrix} \quad (3.8)$$

A matriz  $V$  é responsável por mapear as incertezas dos percursos de cada roda, também chamado *espaço de controle* para incertezas em termos da pose do robô, ou seja, em  $x$   $y$  e  $\theta$ . A matriz  $G$  é usada para calcular a influência da incerteza de todo vetor de estado na pose do robô.

### 3.3 Modelo de Observação

O modelo de observação é responsável por indicar como o sensor (no caso câmera) deve observar uma marca já presente no mapa  $\mathbf{m}$ . Especificamente para este trabalho, o modelo foi baseado em duas equações (3.10 e 3.11): uma diz respeito ao posicionamento (equação vetorial) e outra que faz relação entre a orientação da lâmpada com a orientação do robô.

A primeira equação possui três sistemas de referência para um dado ponto: sistema coordenadas de mundo, de robô e de câmera. A matriz  $T$  contém implicitamente a pose atual do robô, de forma que transforma pontos em referencial de mundo para referencial do robô e vice-versa. A matriz  $C$  relaciona o referencial do robô com o referencial de câmera, de fato é apenas uma translação. A matriz  $A$  é a homografia que relaciona um ponto no teto com um pixel na imagem (equação 3.9). As figuras 3.3 e 3.4 mostram a

relação entre estas matrizes de diferentes pontos de vista. A figura 3.5 ilustra a relação das orientações do robô e da marca de acordo com a equação 3.11.

$$\begin{bmatrix} i \\ j \\ 1 \end{bmatrix} = A_{3 \times 3} \begin{bmatrix} x_C \\ y_C \\ 1 \end{bmatrix} \quad (3.9)$$

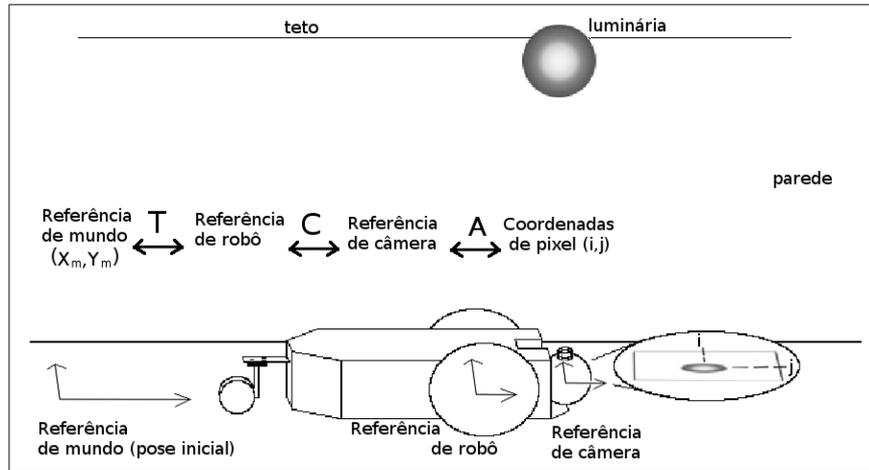


Figura 3.3: Ilustração para equação de posição do modelo de observação

A construção do modelo de observação partiu da equação de posição 3.10 que relaciona a posição no mundo de uma certa marca \$(x\_m, y\_m)\$ com as coordenadas de pixel central da marca \$(i, j)\$ na imagem capturada pela câmera (ilustração na figura 3.4) e da equação de orientação 3.11 que relaciona o ângulo observado da marca na imagem \$\alpha\$ com um ângulo da marca no mundo \$\alpha\_m\$ através da orientação do robô \$\theta\_R\$.

$$\begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = [T]_{3 \times 3} \left( [C]_{3 \times 1} + [A]_{3 \times 3}^{-1} \begin{bmatrix} i \\ j \\ 1 \end{bmatrix} \right) \quad (3.10)$$

$$\alpha_m = \alpha + \theta_R \quad (3.11)$$

O modelo tem objetivo de calcular o pixel central \$(i, j)\$ e a orientação observada da marca \$\alpha\$ a partir da sua pose no mundo \$(x\_m, y\_m\$ e \$\alpha\_m)\$ já armazenada no vetor de estado \$\mu\$ do filtro. Após manipularmos ambas as equações 3.10 e 3.11 para isolar as variáveis observadas \$(i, j, \alpha)\$, chegamos às equações do modelo de observação 3.15 e 3.16. Considerando que \$T\$ é composta por três linhas como na equação 3.12, são definidas então duas funções escalares, \$s\_i()\$ e \$s\_j()\$ nas equações 3.13 e 3.14 com objetivo de exprimir o modelo de observação de forma compacta. Os parâmetros de ambas as funções são o mapa \$\mathbf{m}\$ que contém a marca observada para formar o vetor \$[x\_m, y\_m, 1]^T\$ e a pose do robô \$\mathbf{p}\$ para formar a matriz \$T\$, as demais matrizes são constantes.

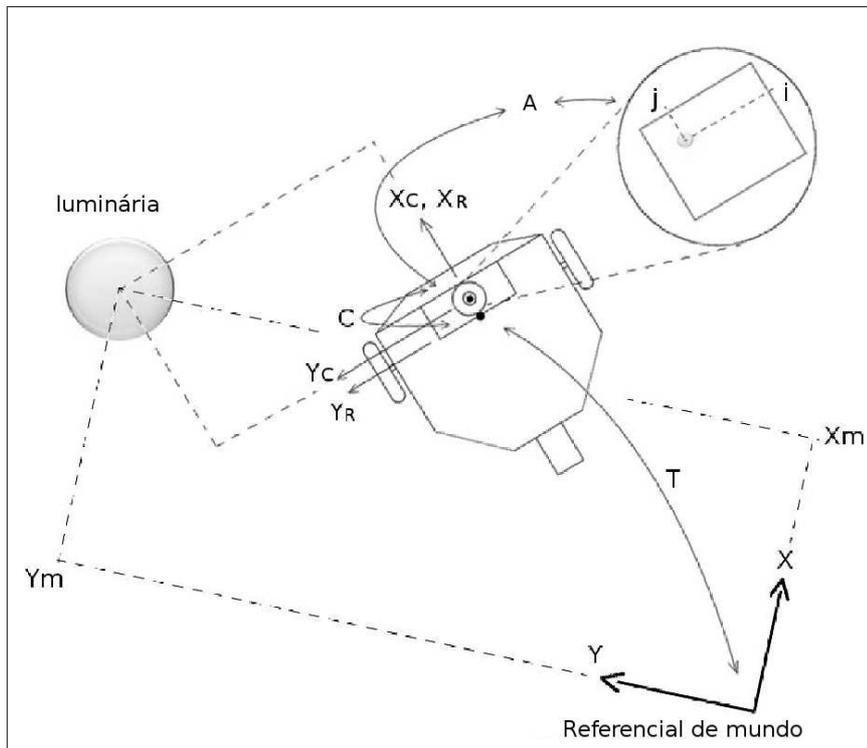


Figura 3.4: Esquema relacionando matrizes para a equação de posição

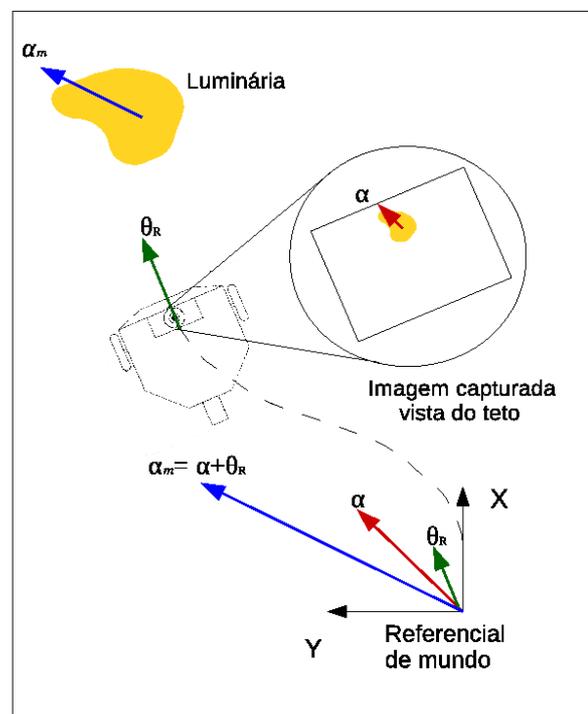


Figura 3.5: Visão superior mostrando a relação das orientações

$$T_{3 \times 3} = \begin{bmatrix} \mathbf{l}_1 \\ \mathbf{l}_2 \\ \mathbf{l}_3 \end{bmatrix} \quad (3.12)$$

$$s_i() = [ \mathbf{l}_1 ] \left( [C]_{3 \times 1} + [A]_{3 \times 3}^{-1} \begin{bmatrix} i \\ j \\ 1 \end{bmatrix} \right) \quad (3.13)$$

$$s_j() = [ \mathbf{l}_2 ] \left( [C]_{3 \times 1} + [A]_{3 \times 3}^{-1} \begin{bmatrix} i \\ j \\ 1 \end{bmatrix} \right) \quad (3.14)$$

$$\begin{bmatrix} i \\ j \\ 1 \end{bmatrix} = \begin{bmatrix} s_i(\mathbf{p}, \mathbf{m}) \\ s_j(\mathbf{p}, \mathbf{m}) \\ 1 \end{bmatrix} = [A]_{3 \times 3} \left( -[C]_{3 \times 1} + [T]_{3 \times 3}^{-1} \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} \right) \quad (3.15)$$

$$\alpha = \alpha_m - \theta_R \quad (3.16)$$

Agora pode-se juntar ambas as equações em uma única equação vetorial 3.17. A equação pode ser compreendida como uma função  $h(\mathbf{p}, \mathbf{m})$ , esta é o modelo de observação.

$$h(\mathbf{p}, \mathbf{m}) = \begin{bmatrix} i \\ j \\ \alpha \end{bmatrix} = \begin{bmatrix} s_i(\mathbf{p}, \mathbf{m}) \\ s_j(\mathbf{p}, \mathbf{m}) \\ \alpha_M - \theta_R \end{bmatrix} \quad (3.17)$$

A matriz  $Q$  define as incertezas na observação das marcas, ou seja, as covariâncias das propriedades das marcas observadas. Ela é definida na equação 3.18.

$$Q = \begin{bmatrix} \sigma_i^2 & 0 & 0 \\ 0 & \sigma_j^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix} \quad (3.18)$$

O valor de cada parâmetro da matriz  $Q$  é definido na seção 4.1.1

A matriz  $H_{ik} 3 \times 6$  é a jacobiana deste modelo. O modelo é derivado em relação à pose do robô e a pose da marca. Considerando que a matriz de homografia  $A$  possa ser expressa em 3.19

$$A_{3 \times 3} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \quad (3.19)$$

podemos realizar a derivação do modelo de observação  $h$

$$\frac{\partial h}{\partial [\mathbf{p}_k \mid \mathbf{z}_k]} = H_{ik} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} & h_{15} & 0 \\ h_{21} & h_{22} & h_{23} & h_{24} & h_{25} & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}$$

onde:

$$\begin{aligned} h_{11} &= -a \cos(\theta) + b \sin(\theta) \\ h_{12} &= -a \sin(\theta) - b \cos(\theta) \\ h_{13} &= -a \sin(\theta)x_m + a \cos(\theta)y_m + ax_R \sin(\theta) - ay_R \cos(\theta) \\ &\quad - b \cos(\theta)x_m - b \sin(\theta)y_m + by_R \sin(\theta) + bx_R \cos(\theta) \\ h_{14} &= a \cos(\theta) - b \sin(\theta) \\ h_{15} &= a \sin(\theta) + b \cos(\theta) \end{aligned}$$

$$\begin{aligned} h_{21} &= -d \cos(\theta) + e \sin(\theta) \\ h_{22} &= -d \sin(\theta) - e \cos(\theta) \\ h_{23} &= -d \sin(\theta)x_m + d \cos(\theta)y_m + dx_R \sin(\theta) - dy_R \cos(\theta) \\ &\quad - e \cos(\theta)x_m - e \sin(\theta)y_m + ey_R \sin(\theta) + ex_R \cos(\theta) \\ h_{24} &= d \cos(\theta) - e \sin(\theta) \\ h_{25} &= d \sin(\theta) + e \cos(\theta) \end{aligned}$$

(3.20)

O modelo foi simplificado de forma que as equações iniciais (3.10 e 3.11) sejam consistentes. Isto porque a equação de orientação 3.11 implicitamente considera o plano do teto paralelo ao plano da imagem, já a equação de posição 3.10 não exige este paralelismo. O ajuste foi realizado na matriz de homografia  $A$  e é mais detalhado na seção 3.5.

### 3.4 Completando a definição do SLAM

O vetor de estado definido (3.21) é formado pela pose  $\mathbf{p}$  do robô e por cada uma das marcas observadas.

$$\mu_t = \begin{bmatrix} \text{Pose atual do robô } \mathbf{p} \\ \underbrace{x_R, y_R, \theta_R}_{\text{Mapa de marcas } \mathbf{m}}, \underbrace{x_{m1}, y_{m1}, \alpha_{m1}, x_{m2}, y_{m2}, \alpha_{m2}, \dots}_{\text{Mapa de marcas } \mathbf{m}} \end{bmatrix}^T \quad (3.21)$$

Uma das principais preocupações nas implementações do filtro de Kalman é o crescimento do vetor de estado até um instante em que o custo computacional torne inviável

a aplicação do filtro. A utilização de luminárias como marcas traz uma segurança neste importante quesito.

A distribuição de luminárias em ambientes internos é bem regular na maioria dos casos, isto permite calcular parâmetros de custo-benefício e a viabilidade para a aplicação do robô de forma prática. Com o número de luminárias por metro quadrado é possível estimar o custo computacional para o FKE por metro quadrado de uma forma mais segura que em sistemas utilizando *features* como marcas do mapa. Por exemplo, cômodos ou salas de um prédio podem ter diferentes padrões de teto (forro de gesso, texturas ou padrões) o que traz uma variação no número de marcas por metro quadrado, tornando variável o rendimento da aplicação robótica. Já o número de luminárias é menos propenso à mudanças por ter maiores implicações no ambiente.

### 3.4.1 Associação de Marcas

Ao longo dos experimentos realizados, notou-se uma grande semelhança entre as marcas. Durante os primeiros experimentos, a associação de marcas foi realizada manualmente, posteriormente foi utilizada a distância euclidiana para realização da identificação das marcas. A utilização de distância euclidiana foi uma forma segura de associação para os ambientes em que o robô realizou os testes. As distâncias entre as luminárias foram suficientes para evitar equívocos na associação.

## 3.5 Considerações

A simplificação na matriz de homografia  $A$  consistiu em anular alguns elementos da matriz de forma que o plano da imagem é considerado paralelo ao plano do teto (também paralelo ao chão), além da câmera estar perfeitamente alinhada com a frente do robô, ou seja, uma linha de pixels na vertical da imagem de câmera coincide com eixo  $X$  do robô. Esta modificação não trouxe maiores preocupações, mas uma atenção durante a realização do experimentos ajustando a câmera para que estas considerações tenham fundamento. Os experimentos são o tema do próximo capítulo. A influência deste ajuste na estimação do vetor de estado foi desprezada por ser muito pequena, de acordo com teste realizado.



---

# Capítulo 4

## Testes experimentais

---

O capítulo inicia com ajustes necessários de parâmetros para a realização dos experimentos. Os experimentos são apresentados em ordem de realização, com descrição do experimento e apresentação de resultados. Alguns experimentos focam a parte de mapeamento e outros em aspectos de localização.

### 4.1 Identificação de parâmetros

Os parâmetros de construção de robô foram obtidos uma única vez através de medições manuais. Diâmetro da roda, distância entre as rodas e distância entre o centro ótico da câmera e o centro do robô foram os parâmetros necessários.

Parâmetros	Valor
Distância de base b	0,4 (metros)
Diâmetro das rodas	0,16 (metros)
Distância centro câmera - centro robô	0,064 (metros)
Resolução do Encoder	36 (posições/revolução)

Tabela 4.1: Parâmetros fixos do robô

#### 4.1.1 Definição de parâmetros do sistema de visão

A tabela 4.2 foi feita para orientar o ajuste dos principais parâmetros do processamento de imagem. A tabela mostra um teste realizado com lâmpadas fluorescentes sem caixa de luminária que direcione o fluxo de luz (figura A.3). Os parâmetros do processamento considerados mais relevantes foram manipulados. O comportamento de algumas propriedades da região brilhante foram analisadas de acordo com a manipulação. Cada linha da tabela representa uma possível combinação de valores destes parâmetros. O robô foi posto no chão logo abaixo de uma luminária, a distância do teto até o chão do ambiente é 2,95 metros. Nesta posição o robô capturou 800 imagens para cada combinação dos parâmetros. A primeira coluna mostra os valores da máscara utilizada para o filtro

	Param	$\sigma_i^2$	$\sigma_j^2$	$\sigma_\theta^2$	área	%
1	5-150-800	5,4769	1,90687	3,952258E-005	51998,9	16,9
2	5-170-800	0,3356	0,17385	2,501294E-006	41883,4	13,6
3	5-200-800	X	X	X	X	X
4	9-150-800	5,3620	0,635699	0,000015114	52604,7	17,1
5	9-170-800	0,2493	0,202125	2,154289E-006	40556	13,2
6	9-200-800	X	X	X	X	X
7	13-150-800	5,2467	0,002478	1,039336E-005	52553,3	17,1
8	13-170-800	0,31480	0,051674	0,00000228	41278,6	13,4
9	13-200-800	X	X	X	X	X
10	5-150-1200	0,15165	0,102197	0,00000551	41745,2	13,5
11	5-170-1200	E	E	E	E	E
12	5-200-1200	X	X	X	X	X
13	9-150-1200	0,10787	0,447198	7,503895E-006	41511,2	13,5
14	9-170-1200	E	E	E	E	E
15	9-200-1200	X	X	X	X	X
16	13-150-1200	0	0,034728	3,532696E-006	41698,7	13,5
17	13-170-1200	0	0	2,377779E-006	34223,2	11,1
18	13-200-1200	X	X	X	X	X
19	5-150-1400	E	E	E	E	E
20	5-170-1400	E	E	E	E	E
21	5-200-1400	X	X	X	X	X
22	9-150-1400	0,24873	0,243917	5,562473E-006	37898,4	12,3
23	9-170-1400	E	E	E	E	E
24	9-200-1400	X	X	X	X	X
25	13-150-1400	0,19698	0,0086201	4,585880E-006	38272,2	12,4
26	13-170-1400	0,24164	0,0721437	2,481770E-006	31308,9	10,1
27	13-200-1400	X	X	X	X	X

Tabela 4.2: Variâncias de marca geradas com 800 amostras

de ruídos (em pixels) seguida do limiar de binarização (intensidade de pixel) e do tempo de exposição que a câmera necessita para formação da imagem (Hertz). A segunda e a terceira coluna mostram as variâncias vertical e horizontal do ponto central da região (em pixel), a quarta coluna mostra a variância angular (radianos) da região, a quinta coluna exibe a área média da região (pixels) e a sexta coluna exibe o percentual da imagem que a marca ocupa. Os valores “X” e “E” representam situações onde ocorreu deterioração no processamento.

As linhas com “X” tiveram a região brilhante corrompida pelo alto nível do limiar de binarização, de forma que a região se fragmentou em pequenas outras regiões. As linhas com “E” tiveram a região brilhante corrompida pela baixa dimensão da máscara de suavização da imagem, caracterizada por algumas fragmentações nas extremidades da região brilhante (coincidentes com a região de extremidade das lâmpadas). Com a tabela 4.2 foi possível confirmar um comportamento diferente para  $\sigma_i^2$  e  $\sigma_j^2$ . Dependendo do ângulo de observação da luminária  $\alpha$ , um destes parâmetros tinha uma significante melhora em relação ao outro. Isto deve influenciar na formação da matriz de covariância da marcas  $Q_{3 \times 3}$  definida na equação 3.18. Os valores  $\sigma_i^2$  e  $\sigma_j^2$  devem ser função do ângulo de observação da marca. Analisando a tabela, é notável os melhores resultados na linha

17 da tabela (menores incertezas), assim, para os testes SLAM foi utilizados estes valores de tempo de exposição, dimensão da matriz do filtro e limiar de binarização presentes no sistema de visão.

### 4.1.2 Calibração da câmera

Antes de cada experimento prático foi realizada a calibração dos parâmetros extrínsecos da câmera contidos na matriz de homografia. Isto foi feito posicionando o robô em um lugar definido de forma que a câmera embarcada pudesse observar alguns pontos marcados no teto do ambiente. A posição de cada um dos pontos em relação ao robô era conhecida. Assim foram estabelecidas equações entre esta distância e o pixel do ponto no teto. A solução das equações forma a matriz de homografia, encontrada pelo método dos mínimos quadrados. Implicitamente estão presentes a altura ("pé direito") do teto ao chão e inclinações entre os planos. A distorção radial da câmera foi estimada com um valor muito baixo, ao ponto de ser descartada. Apenas a posição do pixel do centro ótico da câmera foi considerado na implementação do sistema.

### 4.1.3 Estimação de Covariância das Marcas

Já com os parâmetros do sistema de visão ajustados de acordo com os melhores parâmetros da tabela 4.2, foram estimados os valores de  $Q$ . O robô foi posicionado abaixo de uma das luminárias com maiores índices das covariâncias  $\sigma_i$  e  $\sigma_j$ , em condições similares a de teste de mapeamento para a partir de um conjunto de imagens coletadas fosse possível estimar  $\sigma_i$ ,  $\sigma_j$ ,  $\sigma_\theta$ .

Com os resultados deste teste, foi decidido a utilização do maior valor entre  $\sigma_i$  e  $\sigma_j$  para ambos parâmetros na matriz  $Q$ . O valores reais estão definidos na equação 4.1.

$$Q = \begin{bmatrix} 0.2226469183 & 0 & 0 \\ 0 & 0.2226469183 & 0 \\ 0 & 0 & 0.000024027 \end{bmatrix} \quad (4.1)$$

### 4.1.4 Estimação de Covariância do Deslocamento das Rodas

O teste idealizado para estimar os parâmetros  $k_l$  e  $k_r$  presentes na matriz  $M$  consistiu em girar o robô sobre uma de suas rodas, desta maneira uma roda não influencia na covariância da distância percorrida da outra roda. Para analisar a roda direita ( $k_r$ ) foram enviados comandos de modo que percorresse uma distância pré-determinada, em seguida, determinava-se o erro de distância. Foram gerados 30 erros sobre os quais a covariância foi calculada. Assim conseguiu-se determinar ( $k_r$ ). O teste para a roda esquerda seguiu os mesmos critérios. Para determinar o erro da distância percorrida foi necessário o suporte de outros sistemas. O robô foi posicionado em um tablado dentro de uma estrutura metálica (ambos do projeto de futebol de robôs). A estrutura possuía uma câmera no topo apontada para o robô que possuía duas marcas próximas de cada roda. Através da câmera conseguimos extrair o ângulo formado pela posição inicial e final do robô. Usando este ângulo e o parâmetro  $b$  conseguimos ter uma aproximação do valor real da distância

percorrida. A figura 4.1 mostra uma foto durante o teste da roda esquerda. O valores estimados foram  $k_l = 0.002566506$  e  $k_r = 0.002740474$ .



Figura 4.1: Estimando covariância das rodas

Existem muitos efeitos que não foram considerados, como o escorregamento da roda (parada) para fora do centro de giro, erro da câmera entre outros por envolverem muito custo sendo pouco expressíveis para o sistema como um todo.

## 4.2 Testes de SLAM

### 4.2.1 Primeiro Experimento

O primeiro teste realizado foi uma volta no corredor do DCA-UFRN onde robô guiado manualmente observou três marcas em linha. Cada movimento do robô foi enviado por um terminal de comandos. O trajeto teve um formato de “U”. Cada lâmpada estava

distante 2 metros das outras. O trajeto medido através de odometria está na figura 4.2. Já o trajeto obtido através do filtro está na figura 4.3. As figuras não apresentam a mesma escala em ambos os eixos.

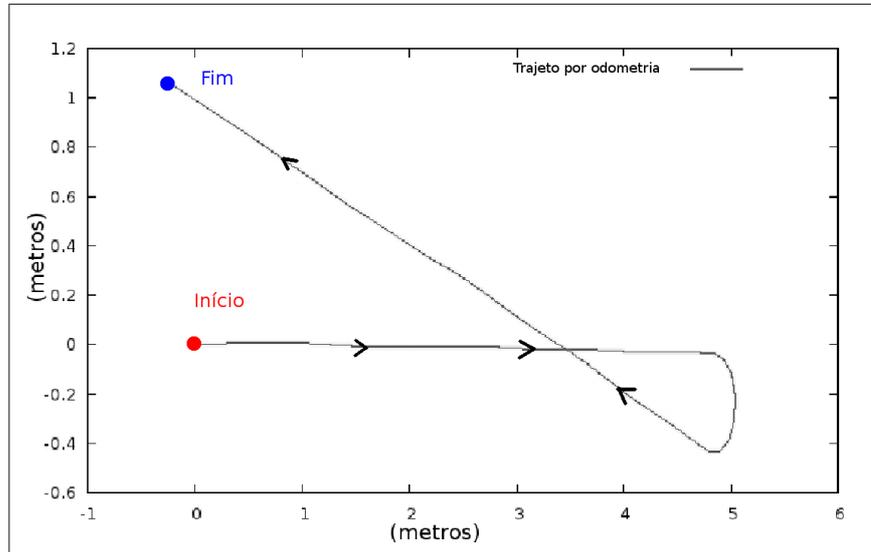


Figura 4.2: Trajeto por odometria

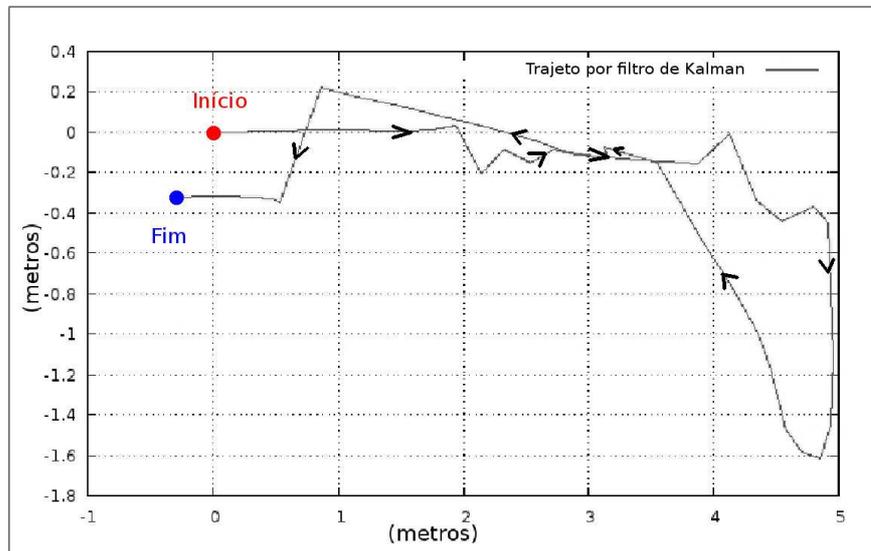


Figura 4.3: Trajeto por filtro de Kalman

Pelo gráfico da odometria notou-se a degradação da posição. Com o filtro de Kalman tem-se uma medição descontínua do trajeto devido às correções da pose quando alguma marca conhecida é avistada.

Este experimento foi guiado de uma forma diferente dos posteriores. O robô recebia como comando, as distâncias para percorrer por cada roda, após a realização do movimento era feita a observação e estimado o resultado do FKE. Este modo resultou em um número reduzido de iterações do filtro de Kalman ao longo do experimento. Foi possível notar que a variação de  $Y_R$  quando o robô observa a segunda marca pela primeira vez existem apenas 5 iterações no filtro por uma distância de pouco mais de 0,5 metro. Isto coopera para uma degradação no resultado do filtro.

### 4.2.2 Segundo e Terceiro Experimentos

Um segundo teste envolveu outras três luminárias em linha reta, distanciadas 5 metros uma das outras. O robô foi guiado por um controle manual (*joystick*). As iterações do filtro de Kalman foram desvinculadas do comando manual, a fim de ter-se mais iterações do FKE. O trajeto teve o formato de “8”. A utilização do *joystick* deixou trajeto menos retilíneo em relação ao controle via comandos de distâncias.

Neste experimento houve uma degradação de mesma magnitude na curva (em relação em primeiro experimento), no momento em que robô observa a terceira marca. A degradação ocorre pela incerteza acumulada na matriz de covariância do filtro. Ao longo da trajetória o robô permanece sem observações durante longas distâncias (maiores que no primeiro experimento). Durante estes períodos, o filtro de Kalman é atualizado utilizando apenas o modelo cinemático. Como os *encoders* possuem uma precisão muito inferior aos sensores extraceptivos de natureza ótica (câmera), há uma segurança muito maior no trajeto feito pelo robô quando está observando alguma marca. Assim ao percorrer um trajeto sem observações de marcas, cada marca nova que for observada neste trajeto não contribui para a acurácia da pose do robô mas impede o crescimento da incerteza enquanto a marca for observada. A precisão da pose do robô só é recuperada quando for observada uma das marcas próximas do ponto de partida por envolver um acúmulo de erro dos *encoders* muito menor.

Para certificar este raciocínio foi realizado um novo teste com os dados do primeiro experimento. De forma artificial descartou-se todo o trajeto percorrido pelo robô entre a primeira e segunda marca, de modo que a segunda marca foi considerada o ponto inicial do robô. Ao chegar na próxima marca onde realizou a curva de retorno, notou-se uma considerável melhoria em relação ao trajeto apresentado em 4.3.

Cientes das informações dos dois primeiros testes, realizou-se um terceiro experimento que consistiu de realizar o mesmo circuito do segundo teste mas em vez de uma única o volta, o robô percorreu cinco vezes aquele trajeto. O resultado da odometria está na figura 4.4. A figura 4.5 mostra o resultado com o filtro.

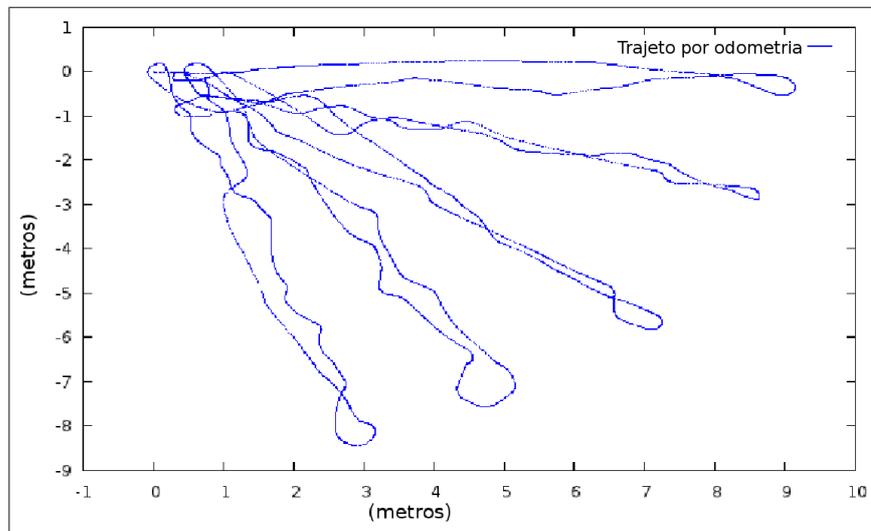


Figura 4.4: Trajeto por odometria

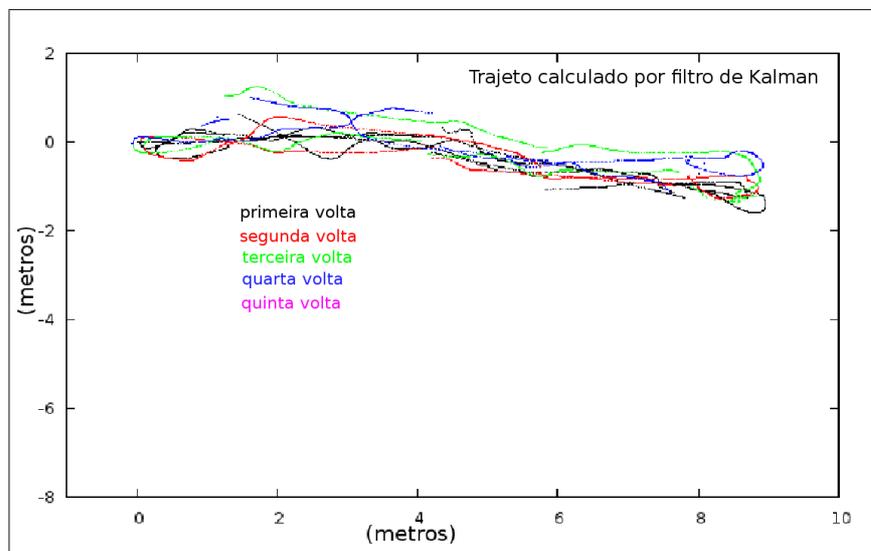


Figura 4.5: Trajeto por filtro de Kalman

Através do gráfico 4.5 é possível notar uma sensível melhoria no quinto trajeto. Este último está mais alinhado ao eixo x que o primeiro trajeto. Isto pode ser comprovado pela disposição das marcas em cada uma das voltas nas figuras 4.6, 4.7, 4.8, 4.9 e 4.10. Em cada uma destas figuras os pontos amarelos são as marcas enquanto o ponto preto é a posição do robô naquele instante indicado na figura.

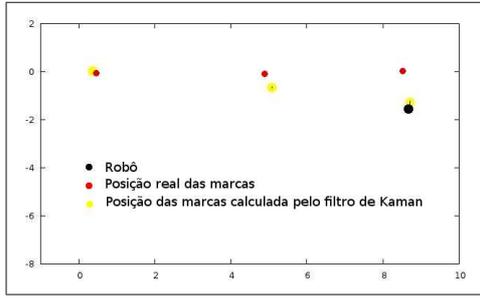


Figura 4.6: Posicionamento das marcas na volta 1

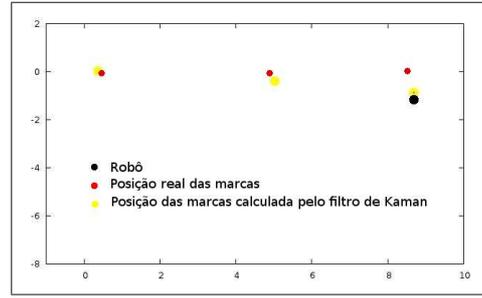


Figura 4.7: Posicionamento das marcas na volta 2

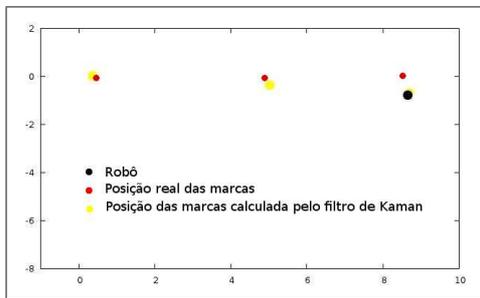


Figura 4.8: Posicionamento das marcas na volta 3

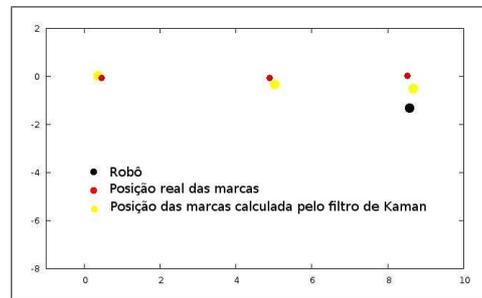


Figura 4.9: Posicionamento das marcas na volta 4

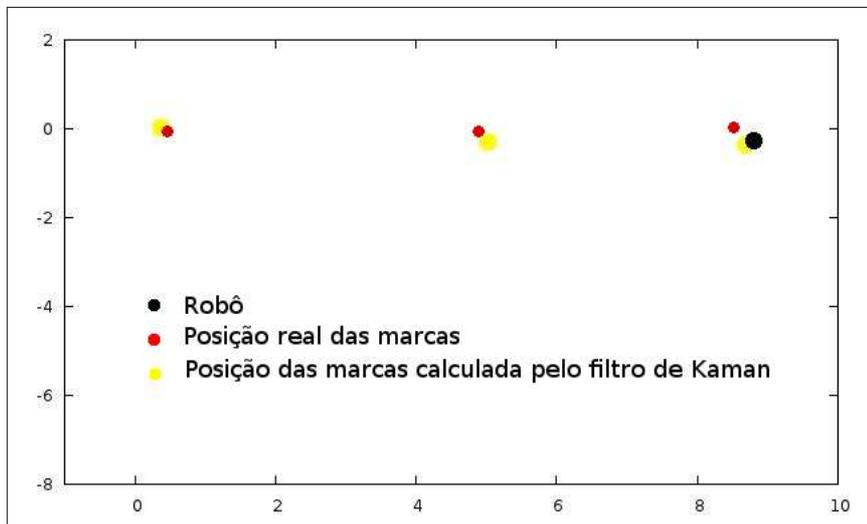


Figura 4.10: Convergência das marcas na última volta

Com os gráficos anteriores podemos afirmar que repetição do movimento contribuiu para uma melhoria do resultado do mapa. Mesmo tendo o vetor de estado do filtro sendo atualizado apenas pelos *encoders*, a observação da segunda e terceira marcas após recuperar a precisão pela primeira marca, a posição destas marcas foram aprimoradas a cada

volta, tendendo aos valores mais próximos dos reais. Para ilustrar esta idéia existem as figura 4.11 e 4.12 comparando a covariância da segunda luminária na primeira e na quinta volta. A terceira marca também possui seu posicionamento refinado entre a primeira volta (figura 4.13) e a quinta volta (figura 4.14). Deve ser ressaltado que as figuras 4.11, 4.12, 4.13 e 4.14 não possuem a mesma escala.

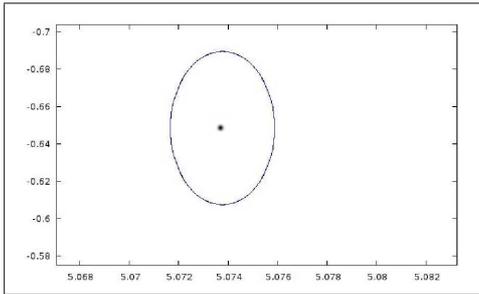


Figura 4.11: Covariância da segunda marca na primeira volta.

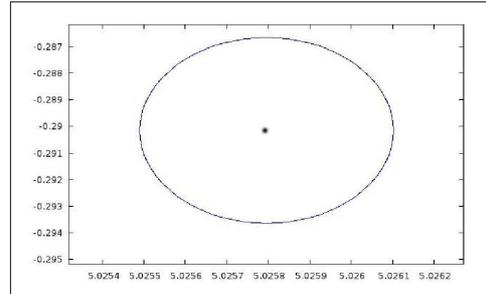


Figura 4.12: Covariância da segunda marca na quinta volta.

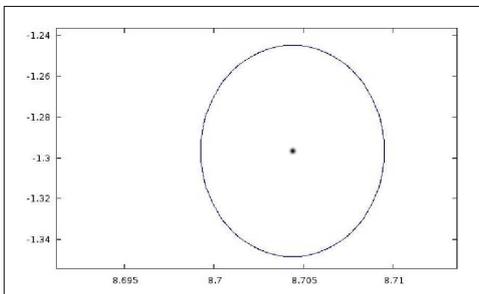


Figura 4.13: Covariância da terceira marca na primeira volta.

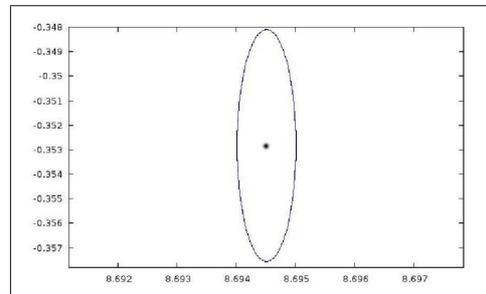


Figura 4.14: Covariância da terceira marca na quinta volta.

### 4.2.3 Quarto Experimento

Um quarto teste similar ao anterior foi realizado. O robô percorreu um trajeto observando outras três luminárias distantes cinco metros uma das outras, em linha reta. O robô foi controlado via *joystick*. No total foram 6 voltas, mas a trajetória idealizada teve um formato oval. O trajeto resultante calculado por odometria para as quatro primeiras voltas está na figura 4.15 assim como as quatro primeiras voltas calculadas pelo filtro de kalman presentes na figura 4.16. O gráfico da figura 4.17 mostra as covariâncias  $\sigma_{xx}$  e  $\sigma_{yy}$  do robô durante todo o experimento.

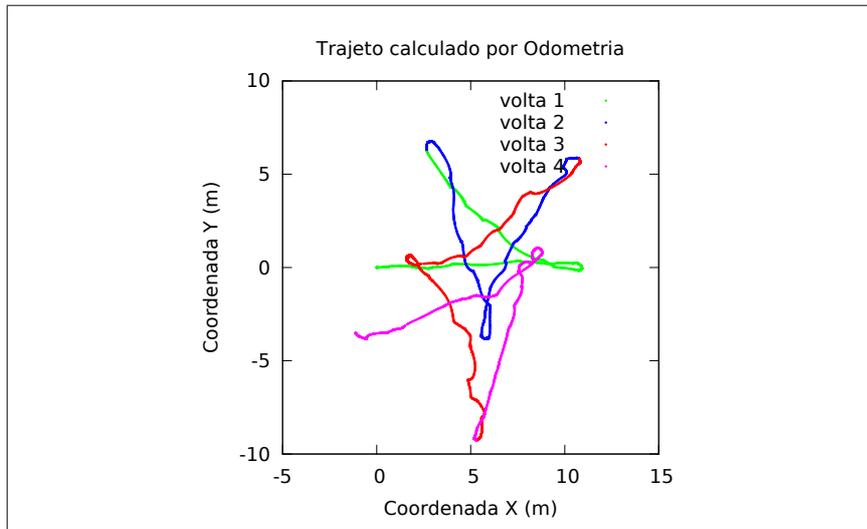


Figura 4.15: Resultado com odometria para quatro voltas

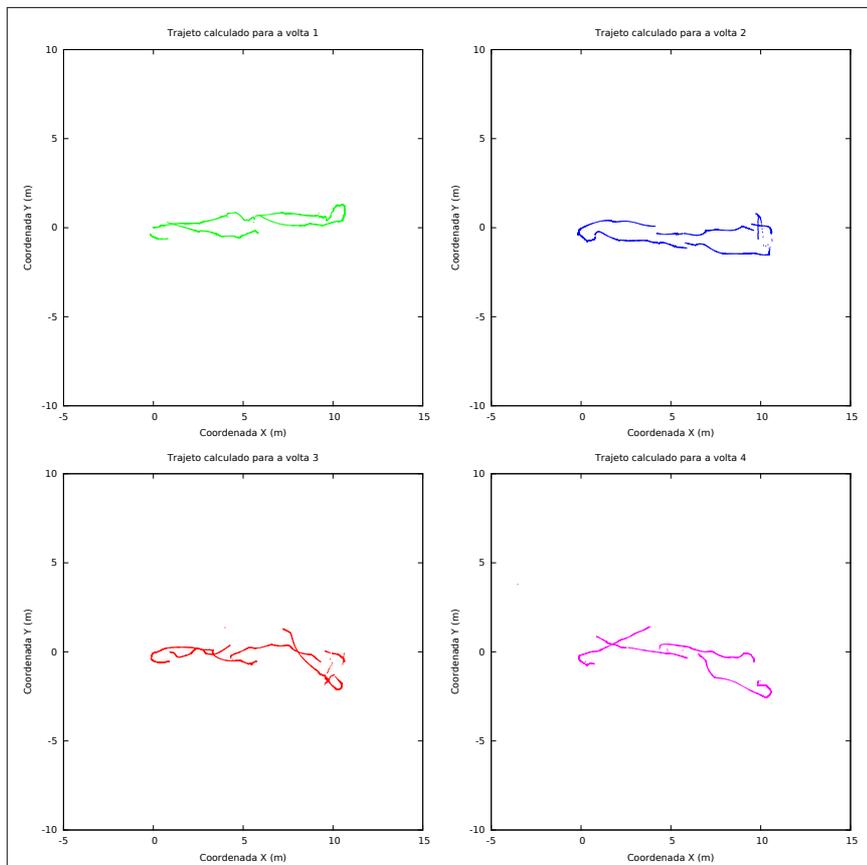


Figura 4.16: Trajetos calculados com FKE para quatro voltas

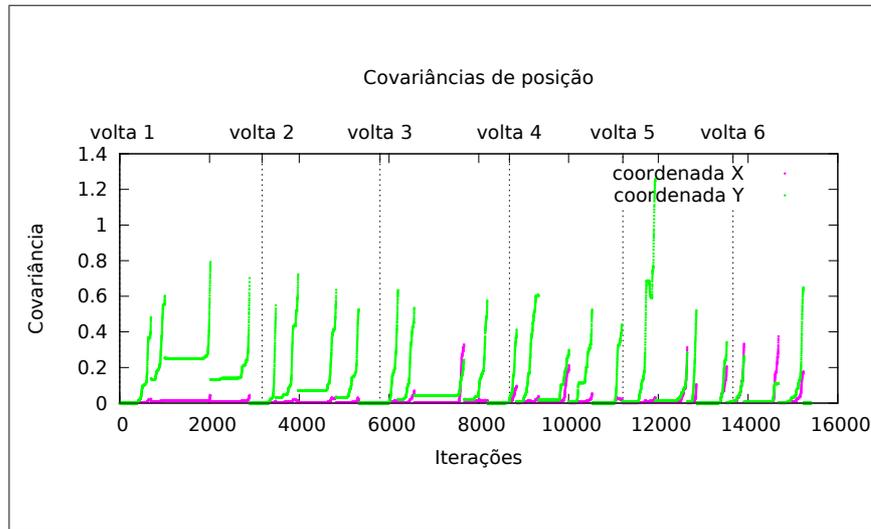


Figura 4.17: Covariância de posição do robô durante trajeto do quarto experimento

A covariância da orientação ( $\sigma_{\theta\theta}$ ) do robô ao longo de todo experimento (quarto) está na figura 4.18.

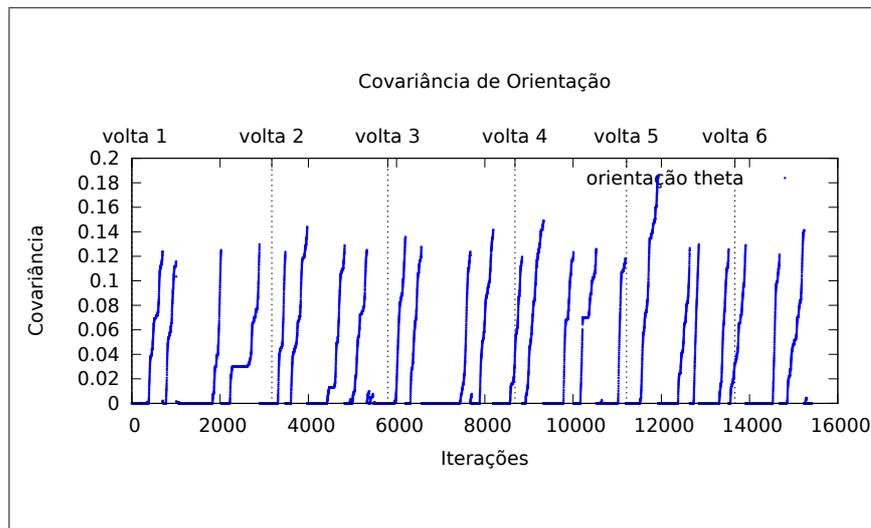


Figura 4.18: Covariância de orientação do robô no quarto experimento

Através dos gráficos 4.17 e 4.18 é notável que as covariâncias de localização são constantes no momento em que o robô observa alguma marca. Já as iterações sem observações formam as curvas ascendentes. Pelo comportamento de  $\sigma_{yy}$  no gráfico 4.17 foi identificado que o valor constante mais alto durante a primeira volta ocorre no meio da volta, justamente quando o robô observa a terceira marca. Nas voltas seguinte foi possível observar a diminuição da covariância  $\sigma_{yy}$  do robô. Os valores  $\sigma_{xx}$  são muito inferiores aos de

$\sigma_{yy}$ . Os valores de  $\sigma_{\theta\theta}$  tiveram um comportamento mais semelhante para todas as voltas embora alcançasse valores mais altos que  $\sigma_{xx}$  nos instantes sem observações.

A pose do robô ao longo deste experimento pode ser observada nos gráficos 4.19 e 4.20. O gráfico 4.19 mostra as coordenadas do robô e gráfico 4.20 mostra a orientação do robô dada em radianos.

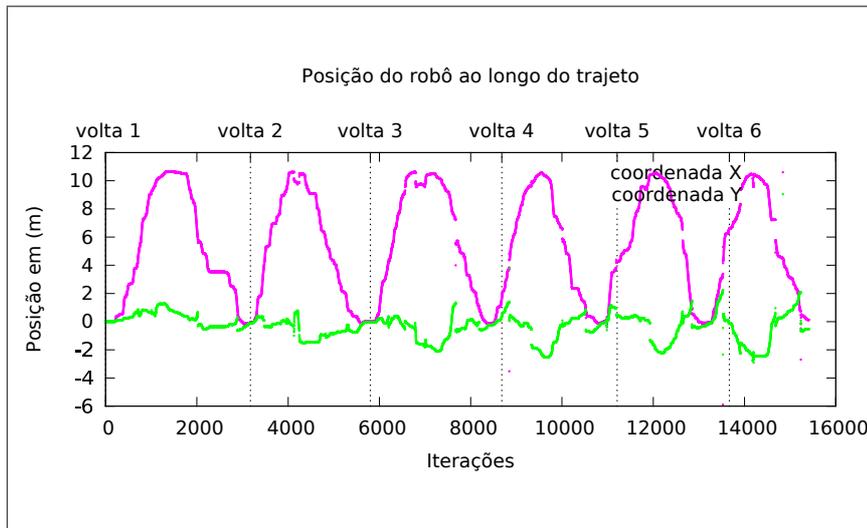


Figura 4.19: Posicionamento do robô no quarto experimento

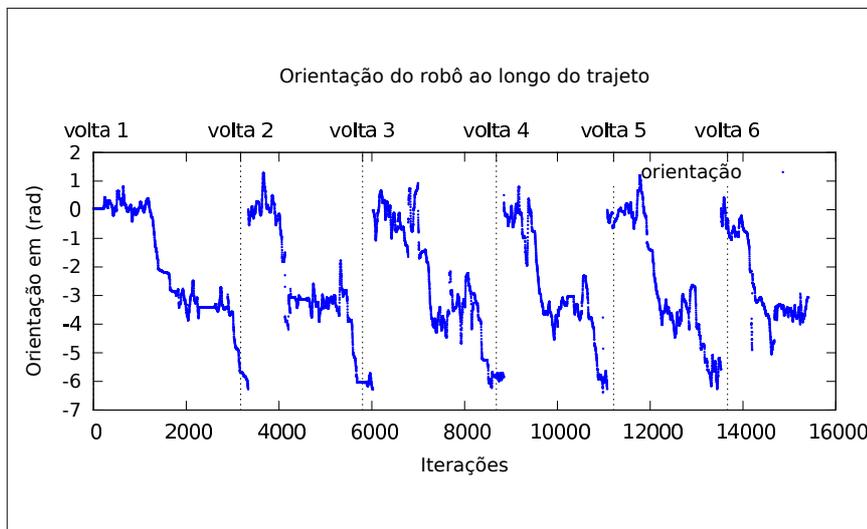


Figura 4.20: Orientação do robô no quarto experimento

Podemos observar por estes gráficos algumas descontinuidades. No gráfico 4.19 uma variação considerável de  $Y$  durante a realização da curva de volta. Já o ângulo  $\theta$  (na figura 4.20) do robô teve uma mesma variação durante todo o trajeto.

### 4.2.4 Quinto Experimento

O quinto teste foi elaborado para observar o comportamento do filtro de Kalman dado o mapa de marcas, com valores confiáveis. O robô percorreu o mesmo percurso do quarto teste mas apenas quatro voltas. A posição de cada marca (no mundo) foi mensurada. A fim de analisar a localização do robô pelo FKE, o vetor de estado foi inicializado com as três marcas. A pose do robô foi um vetor nulo de modo que o ponto inicial é a origem do sistema de mundo. A covariância foi inicializada com valores muito altos para a pose do robô, enquanto que as covariâncias de cada marca foram anuladas. O intuito desta modificação da matriz de covariância é conseguir uma melhoria na localização do robô dado um momento em que as marcas possuem um posicionamento consistente no mapa. Os gráficos 4.21, 4.22, 4.23 fazem o comparativo entre odometria, a solução do SLAM e a localização do robô dado o posicionamento das marcas. A solução do SLAM é iniciada com um mapa sem marcas. Já a solução de localização contém o mapa de marcas bem definido.

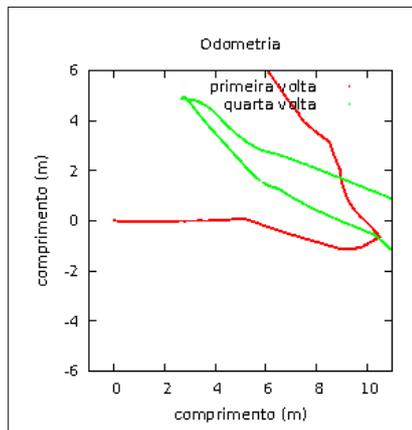


Figura 4.21: Trajeto por odometria

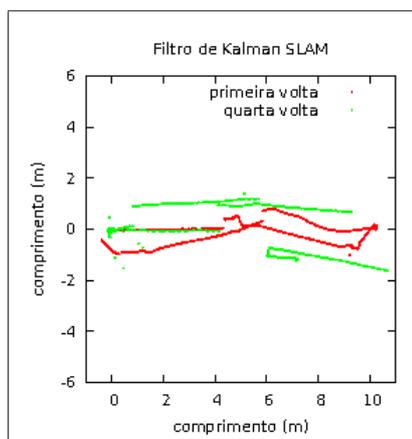


Figura 4.22: Trajeto por SLAM com FKE

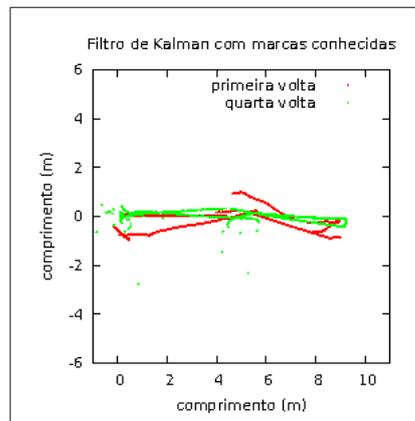


Figura 4.23: Trajeto com FKE com mapa pronto

Foi obtido um melhor resultado sobre o trajeto mostrado na figura 4.23.

### 4.3 Considerações

Os experimentos tiveram uma postura conservadora já que foram utilizados parâmetros de incerteza nas piores condições encontradas. Mesmo em ambientes com luminárias que produzissem melhores covariâncias, foram utilizados valores de covariâncias de luminárias que produziram piores valores.

---

# Capítulo 5

## Conclusões

---

### 5.1 Conclusões dos experimentos

Através dos experimentos realizados verificou-se que é viável a utilização do filtro de Kalman quando são consideradas luminárias como marcos no mundo. Pelos resultados, foi obtida uma convergência do mapa, embora requisitasse algum tempo de movimentação pelo ambiente de trabalho. Para robôs de serviços isto não seria um agravante já que o robô deve deslocar-se pelo ambiente para execução da sua tarefa.

O trabalho desenvolvido, além do foco principal (estudo do FKE), teve vários desafios de natureza prática:

- Elaboração e verificação de circuitos eletrônicos
- Contorno de limitações técnicas de sistemas (*software e hardware*)
- Identificação de defeitos em componente eletrônicos e mecânicos
- Testes de movimentação
- Testes de câmera
- Desenvolvimento de programas auxiliares

Também foi estudada a inclusão de um segundo sensor, uma câmera RGBD orientada para frente do robô com objetivo principal de evitar colisões e objetivo secundário de incrementar o filtro de Kalman.

Em relação ao SLAM desenvolvido por Hwang & Song (2011) temos uma flexibilidade na detecção de luminárias de diversos formatos. A utilização de conceitos matemáticos no processamento de marcas (*momento*) traz esta versatilidade. No entanto só será aproveitada pelo sistema robótico, com a inclusão da *excentricidade* como uma das propriedades das marca no filtro de Kalman. Isto é mais detalhado na seção de Trabalhos futuros.

A utilização de luminárias como marcas também traz como benefício uma prática e consistente avaliação do desempenho do sistema em locais estruturados, sejam hospitais, prédios, galpões, escritórios ou residências.

## 5.2 Trabalhos publicados

Foram realizadas duas publicações baseadas neste trabalho. A primeira foi realizada no BRASERO 2013 (*Brazilian Workshop on Service Robotics*) que aconteceu na cidade de Santa Maria-RS. O trabalho tratava especificamente do processo de detecção de reflexos [da Silva et al. 2013a]. A segunda publicação foi realizada no SBAI 2013 (Simpósio Brasileiro de Automação Inteligente) que ocorreu na cidade de Fortaleza-CE. O trabalho publicado mostrava a realização da tarefa de SLAM com um enfoque maior no sistema de observação do robô [da Silva et al. 2013b].

## 5.3 Trabalhos futuros

Existem diversas maneiras de aprimorar os resultados obtidos neste documento. A cerca do filtro de Kalman podem-se realizar melhorias nas equações do modelo de observação. É possível melhorar a modelagem da matriz de covariância das marcas  $Q$  deixando alguns parâmetros em função do ângulo da marca observada.

Outra melhoria importante seria aproveitar marcas parcialmente observadas para as correções feitas pelo filtro de Kalman. Esta melhoria poderia diminuir o intervalo de espaço em que o robô permanece sem observações. Embora demande grandes mudanças no filtro de Kalman, teria-se um maior aproveitamento de informações produzidas pelo sistema de visão. Uma consequência direta desta melhoria seria uma convergência mais rápida do mapa.

Outro ponto interessante seria a utilização da câmera para realização da odometria. Isto é chamado de odometria visual. A literatura mostra um erro muito menor de posicionamento quando é usada esta forma de odometria em vez da tradicional utilização de *encoders*. A cada movimento que o robô realizasse, seria utilizado algum algoritmo de extração de *features* para a identificação de pontos para o cálculo do movimento do robô. Este pontos são úteis quando são encontrados em imagens consecutivas para o cálculo de translação e rotação do robô. Isto exige mínimos recursos de memória, mas uma relevante capacidade de processamento.

A inclusão da excentricidade trará mais versatilidade ao sistema robótico pois permitirá que marcas sem orientação sejam incluídas dentro do mapa do FKE. A excentricidade sendo um coeficiente variando entre os valores 0 e 1 dá um grau de arredondamento de uma determinada marca. Desta forma forma lâmpadas circulares (ou quadradas) sem orientação poderiam de acordo com este coeficiente receber um tratamento no FKE diferente das marcas com orientação.

É necessário para uma aplicação prática, a inclusão de um segundo sensor para a detecção de obstáculos. Entre os possíveis sensores para desempenhar esta função, foi estudado a viabilidade do uso do Kinect (câmera RGBD). Além do incremento no FKE, o sensor serviria como uma interface com um usuário do robô a fim de deliberar ou suspender tarefas do robô.

---

# Referências Bibliográficas

---

- Ayache, Nicholas & Olivier Faugeras (1988), 'Building, registrating and fusing noisy visual maps', *International Journal of Robotics Research* **7**(6), 45–65.
- Castellanos, José A., José M. Martínez, José Neira & Juan D. Tardós (1998), Experiments in multisensor mobile robot localization and map building, *em* 'IN 3RD IFAC SYMPOSIUM ON INTELLIGENT AUTONOMOUS VEHICLES, MACH'ID', pp. 173–178.
- Chatila, R. & J. Laumond (1985), Position referencing and consistent world modeling for mobile robots, *em* 'Robotics and Automation. Proceedings. 1985 IEEE International Conference on', Vol. 2, pp. 138–145.
- Chen, Haoyao, Dong Sun & Jie Yang (2009), 'Global localization of multirobot formations using ceiling vision {SLAM} strategy', *Mechatronics* **19**(5), 617 – 628.
- Chen, Haoyao, Dong Sun, Jie Yang & Jian Chen (2010), 'Localization for multirobot formations in indoor environment', *Mechatronics, IEEE/ASME Transactions on* **15**(4), 561–574.
- Choi, Hyukdoo, Dong Yeop Kim, Jae Pil Hwang, Chang-Woo Park & Euntai Kim (2012), 'Efficient simultaneous localization and mapping based on ceiling-view: Ceiling boundary feature map approach', *Advanced Robotics* **26**(5-6), 653–671.
- Crowley, J.L. (1989), World modeling and position estimation for a mobile robot using ultrasonic ranging, *em* 'Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on', pp. 674–680 vol.2.
- Csorba, Michael (1997), Simultaneous Localisation and Map Building, Tese de doutorado, University of Oxford.
- Csorba, Michael, Jeffrey K. Uhlmann & Hugh F. Durrant-Whyte (1996), New approach to simultaneous localization and dynamic map building, Vol. 2738, pp. 26–36.
- da Silva, Luiz H. R., Diogo P. F. Pedrosa & Adelardo A. D. de Medeiros (2013a), A simple heuristic to detect lamp reflexes in cv-slam, *em* 'Brazilian Workshop on Service Robotics'.
- da Silva, Luiz H. R., Diogo P. F. Pedrosa & Adelardo A. D. de Medeiros (2013b), Slam em ambientes internos baseado no reconhecimento visual de luzes no teto, *em* 'Simpósio Brasileiro de Automação Inteligente'.

- Deans, Matthew & Martial Hebert (2000), 'Experimental comparison of techniques for localization and mapping using a bearing-only sensor'.
- Durrant-Whyte, H.F. (1988), 'Uncertain geometry in robotics', *Robotics and Automation, IEEE Journal of* **4**(1), 23–31.
- Fabrizi, Elisabetta & Alessandro Saffiotti (2000), Behavioral navigation on topology-based maps, *em* 'Proc. of the 8th International Symposium on Robotics and Applications', Citeseer.
- Frese, Udo (2004), An  $O(\log n)$  Algorithm for Simultaneous Localization and Mapping of Mobile Robots in Indoor Environments, Tese de doutorado, Universität Erlangen-Nürnberg.
- Fukutani, Y., T. Takahashi, M. Iwahashi, T. Kimura, S.S. Salbiah & N.B. Mokhtar (2010), Robot vision network based on ceiling map sharing, *em* 'Advanced Motion Control, 2010 11th IEEE International Workshop on', pp. 164 –169.
- Gonzalez, R.C. & R.E. Woods (2000), *Processamento de imagens digitais*, Edgard Blucher.
- Guivant, Jose, Eduardo Nebot & Stephan Baiker (2000), 'Localization and map building using laser range sensors in outdoor applications', *Journal of Robotic Systems* **17**(10), 565–583.
- Hwang, Seo-Yeon & Jae-Bok Song (2011), 'Monocular vision-based slam in indoor environment using corner, lamp, and door features from upward-looking camera', *Industrial Electronics, IEEE Transactions on* **58**(10), 4804 –4812.
- Jeong, WooYeon & Kyoung Mu Lee (2005), Cv-slam: a new ceiling vision-based slam technique, *em* 'Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on', pp. 3195 – 3200.
- Lee, Seongsoo, Sukhan Lee & Seungmin Baek (2012), 'Vision-based kidnap recovery with slam for home cleaning robots', *Journal of Intelligent & Robotic Systems* **67**(1), 7–24.
- Leonard, John J. & Hans Jacob S. Feder (2000), 'A computationally efficient method for large-scale concurrent mapping and localization'.
- Leonard, John J & Hugh F Durrant-Whyte (1992), *Directed sonar sensing for mobile robot navigation*, Vol. 448, Kluwer Academic Publishers Dordrecht.
- Liu, Kuotsan & Chulun Wang (2013), 'A technical analysis of autonomous floor cleaning robots based on us granted patents', *European International Journal of Science and Technology* **2**(7), 199–216.

- Rencken, W. D. (1993), Concurrent localisation and map building for mobile robots using ultrasonic sensors, *em* 'Intelligent Robots and Systems '93, IROS '93. Proceedings of the 1993 IEEE/RSJ International Conference on', Vol. 3, pp. 2192–2197 vol.3.
- Rusdinar, Angga, Jungmin Kim, Junha Lee & Sungshin Kim (2012), 'Implementation of real-time positioning system using extended kalman filter and artificial landmark on ceiling', *Journal of Mechanical Science and Technology* **26**, 949–958. 10.1007/s12206-011-1251-9.
- Santana, André M. (2011), Localização e Mapeamento Simultâneos de Ambientes Planos Usando Visão Monocular e Representação Híbrida do Ambiente, Tese de doutorado, UFRN.
- Scherer, S.A., D. Dube & A. Zell (2012), Using depth in visual simultaneous localisation and mapping, *em* 'Robotics and Automation (ICRA), 2012 IEEE International Conference on', pp. 5216 –5221.
- Siegwart, R., I.R. Nourbakhsh & D. Scaramuzza (2011), *Introduction to Autonomous Mobile Robots*, Intelligent Robotics and Autonomous Agents, Mit Press.
- Smith, R., M. Self & P. Cheeseman (1987), Estimating uncertain spatial relationships in robotics, *em* 'Robotics and Automation. Proceedings. 1987 IEEE International Conference on', Vol. 4, pp. 850–850.
- Smith, Randall C. & Peter Cheeseman (1986), 'On the representation and estimation of spatial uncertainty', *Int. J. Rob. Res.* **5**(4), 56–68.
- Wang, Han, Wei Mou, Gerald Seet, Mao-Hai Li, M.W.S. Lau & Dan-Wei Wang (2013), 'Real-time visual odometry estimation based on principal direction detection on ceiling vision', *International Journal of Automation and Computing* **10**(5), 397–404.
- Wang, Han, Wei Mou, Minh Hiep Ly, M.W.S. Lau, G. Seet & Danwei Wang (2012), Mobile robot ego motion estimation using ransac-based ceiling vision, *em* 'Control and Decision Conference (CCDC), 2012 24th Chinese', pp. 1939 –1943.
- Xu, De, Liwei Han, Min Tan & You Fu Li (2009), 'Ceiling-based visual positioning for an indoor mobile robot with monocular vision', *Industrial Electronics, IEEE Transactions on* **56**(5), 1617 –1628.



---

# Apêndice A

## Testes com luminárias

---

Os testes de covariância das propriedades extraídas das marcas foram realizados com quatro tipos de luminárias como pode ser visto nas imagens A.1, A.2, A.4 e A.3.



Figura A.1: Luminária com ressaltos e duas lâmpadas

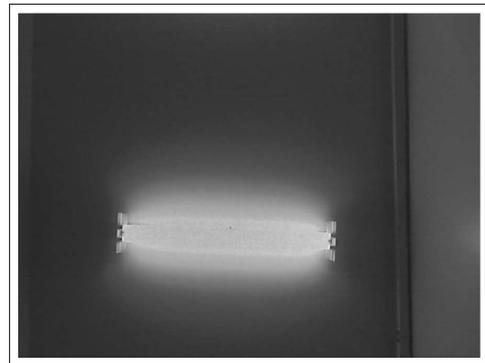


Figura A.2: Luminária sem ressaltos e duas lâmpadas

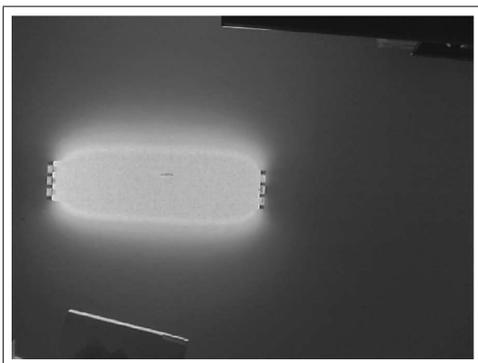


Figura A.3: Luminária sem ressaltos com quatro lâmpadas



Figura A.4: Luminária com espelhamento

Para cada tipo de luminária foram retiradas cerca de 2000 imagens para levantamento das estatísticas. Os valores  $i$  e  $j$  apresentados nas legendas são a coordenada do vertical do centro da marca e a coordenada horizontal respectivamente. Cada gráfico foi pro-

duzido utilizando 20 intervalos de valores igualmente espaçados entre o maior valor da propriedade e menor valor.



Figura A.5: Distribuição de  $i$  da luminária com ressalto



Figura A.6: Distribuição de  $j$  da luminária com ressalto



Figura A.7: Distribuição do ângulo da luminária com ressalto

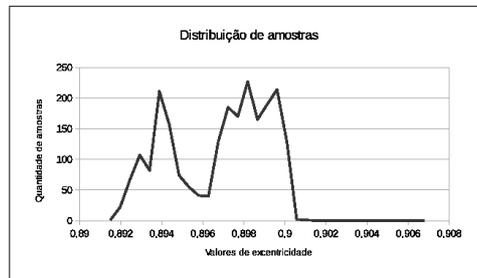


Figura A.8: Distribuição da excentricidade da luminária com ressalto



Figura A.9: Distribuição de  $i$  da luminária sem ressalto

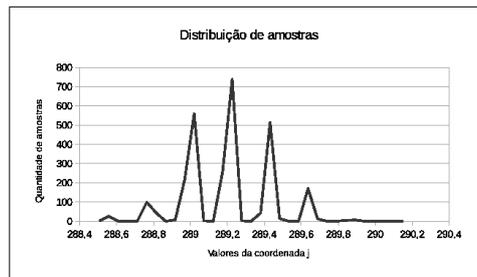


Figura A.10: Distribuição de  $j$  da luminária sem ressalto

Os gráficos mostram curvas semelhantes se analisarmos as mesmas propriedades para os quatro tipos de luminárias.



Figura A.11: Distribuição do ângulo da luminária sem ressalto

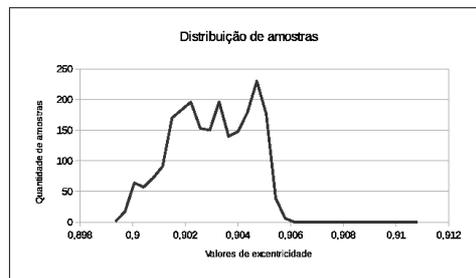


Figura A.12: Distribuição da excentricidade da luminária sem ressalto



Figura A.13: Distribuição de  $i$  da luminária com espelhamento

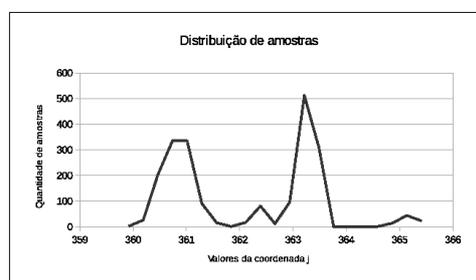


Figura A.14: Distribuição de  $j$  da luminária com espelhamento

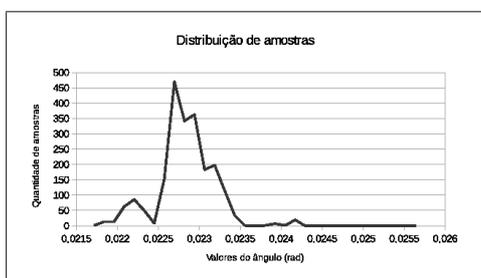


Figura A.15: Distribuição do ângulo da luminária com espelhamento



Figura A.16: Distribuição da excentricidade da luminária com espelhamento



Figura A.17: Distribuição de  $i$  da luminária com 4 lâmpadas

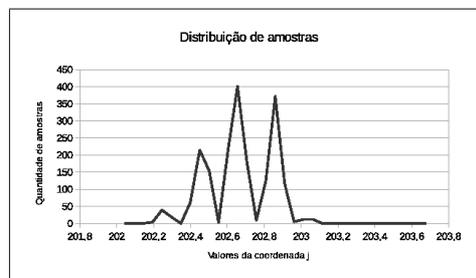


Figura A.18: Distribuição de  $j$  da luminária com 4 lâmpadas

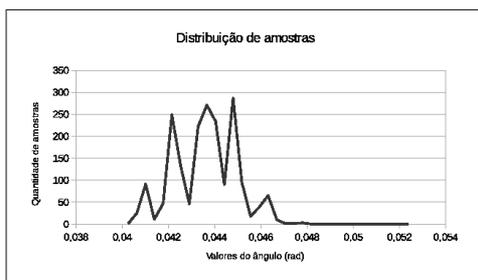


Figura A.19: Distribuição do ângulo da luminária com 4 lâmpadas



Figura A.20: Distribuição da excentricidade da luminária com 4 lâmpadas

---

## Apêndice B

### Programas desenvolvidos

---

O primeiro software desenvolvido foi para a visão computacional. Uma biblioteca foi criada para facilitar o trabalho com *webcams*. Esta foi desenvolvida utilizando primitivas do “v4linux”. Foi incluída uma interface para “OpenCV” que teve suas rotinas matemáticas utilizadas na implementação do filtro de Kalman. Desta forma as imagens capturadas são transformadas em estruturas próprias do “OpenCV”, prontas para extensos processamentos experimentais. A biblioteca criada também usa recursos da biblioteca “SDL” com objetivo de criar uma interface visual para o usuário. Após a consolidação da biblioteca, foram implementadas as rotinas de acionamento das rodas robô. A etapa seguinte foi a implementação do filtro de Kalman, onde foram despendidos muitos testes e atenção. Ainda assim foi necessário mais algum tempo para depuração de erros entre códigos e matemática. Devido aos diversos testes realizados, ficou notável a necessidade de criação de rotinas que facilitassem a realização dos testes, que tornassem mais rápido o cálculo dos resultados. Ao longo do desenvolvimento algumas rotinas foram criadas, outras descartadas de modo que atualmente pode-se mostrar a figura B.1 com os programas utilizados para realização dos experimentos.

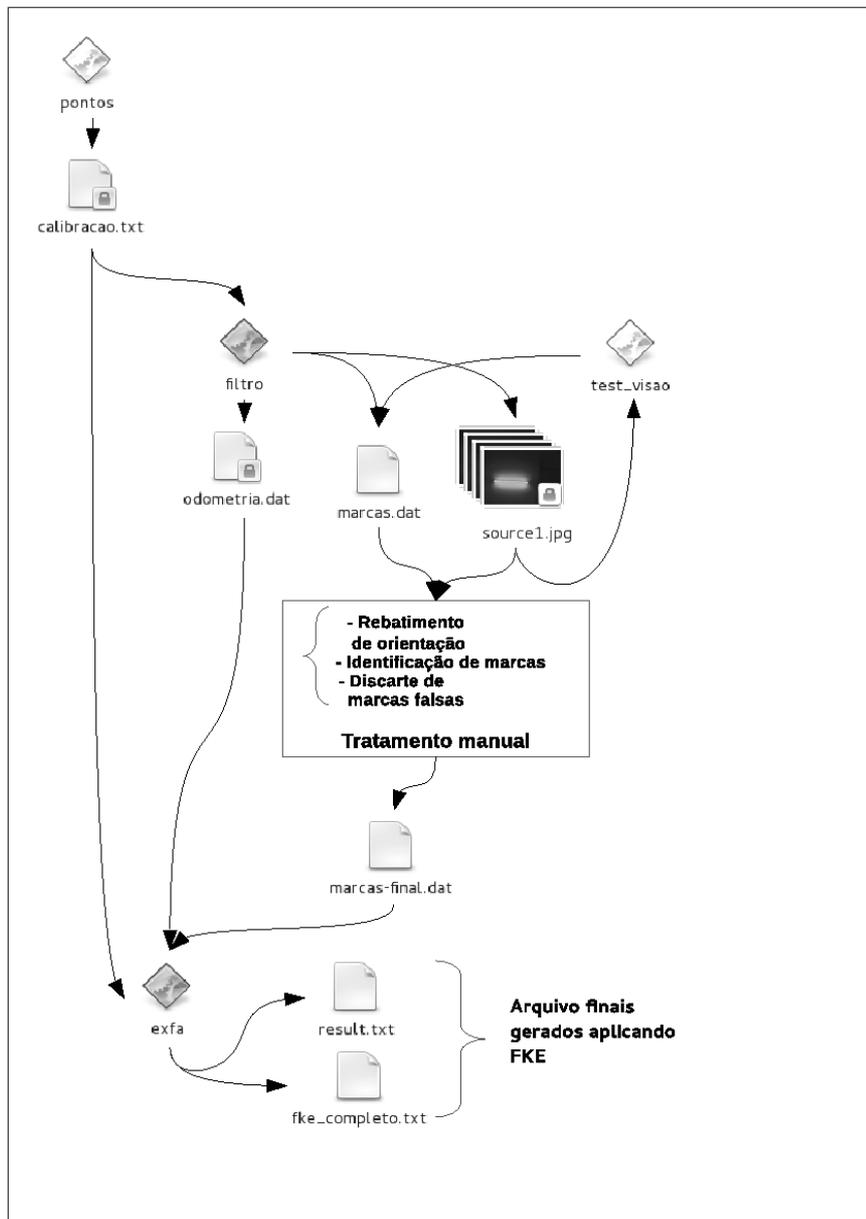


Figura B.1: Diagrama com os principais programas desenvolvidos

---

# Apêndice C

## Detecção de reflexos

---

O algoritmo 2 é uma forma determinística de definir quais contornos são reflexos e quais são genuínos. Durante os testes iniciais de SLAM, notou-se que a abordagem é suscetível à detecção de reflexos, diante disso, foi feita uma análise das situações onde os reflexos ocorriam. O reflexo ocorre quando o robô está próximo à uma luminária e existe algum objeto de material reflexivo (metal, vidro entre outros) acima do robô (geralmente na parede). Estes reflexos quando observados na imagem possuem duas características: são contornos menores que as luminárias reais e estão nas extremidades da imagem, nas bordas da imagem. A partir destas características levantadas foi possível desenvolver o algoritmo. O algoritmo é aplicado dentro da etapa de filtragem do sistema de visão. A entrada é a lista de contornos com  $n$  contornos e a saída é um vetor com a classificação dos contornos.

---

### Algoritmo 2 DETECÇÃO DE REFLEXOS

---

```
1: Calcular o centro de massa de cada contorno.
2: Calcular o centro de massa da imagem.
3: Ordenar os contornos pela distância até o centro de massa da imagem no vetor cont.
4: O contorno mais próximo é definido como marca genuína (cont[0]).
5: for  $i = 1$  to  $n$  do
6:     Definir uma linha do centro de massa da imagem até o centro de massa de cont[ $i$ ].
7:     Fazer  $j = i - 1$ 
8:     while  $j \geq 0$  and cont[ $j$ ] não intersecta a linha do
9:         decrementar  $j$ 
10:    end while
11:    if  $j \geq 0$  and cont[ $j$ ] intersecta a linha then
12:        if cont[ $j$ ] é reflexo or  $2 \times \text{area}[\mathit{cont}[\mathit{i}]] < \text{area}[\mathit{cont}[\mathit{j}]]$  then
13:            cont[ $i$ ] é um reflexo
14:        else
15:            cont[ $i$ ] é uma marca genuína
16:        end if
17:    else
18:        cont[ $i$ ] é uma marca genuína
19:    end if
20: end for
```

---

Uma situação onde o algoritmo falharia seria em uma imagem com diversas luminárias de pequeno tamanho ao redor de uma luminária de tamanho maior.