

UNIVERSIDADE CÂNDIDO MENDES

JÔNATAS OLIVERIA LOPES SOARES
MAYCON BARRETO LOPES
WALLACE GOMES DE SOUZA

MÓDULO DE MAPEAMENTO DO TOOLKIT HÓRUS

Campos dos Goytacazes - RJ
Junho - 2009

JÔNATAS OLIVERIA LOPES SOARES
MAYCON BARRETO LOPES
WALLACE GOMES DE SOUZA

MÓDULO DE MAPEAMENTO DO TOOLKIT HÓRUS

Monografia apresentada à Universidade
Cândido Mendes como requisito obrigatório
para a obtenção do grau de Bacharel em
Ciências da Computação.

ORIENTADOR: Prof. D.Sc. Ítalo Matias

CO-ORIENTADOR: Prof. D.Sc. Dalessandro Soares

Campos dos Goytacazes-RJ

2009

JÔNATAS OLIVERIA LOPES SOARES
MAYCON BARRETO LOPES
WALLACE GOMES DE SOUZA

MÓDULO DE MAPEAMENTO DO TOOLKIT HÓRUS

Monografia apresentada à Universidade
Cândido Mendes como requisito obrigatório
para a obtenção do grau de Bacharel em
Ciências da Computação.

Aprovada em ____ de _____ de 2009.

BANCA EXAMINADORA

Prof. D.Sc. Ítalo Matias - Orientador
Doutor pela UFRJ

Prof. D.Sc. Dalessandro Soares
Doutor pela PUC-Rio

Prof. BLABLABLA
Univeridade de Londres

Dedico este trabalho a minha mãe

Jônatas

Dedico este trabalho a meus pais

Maycon

Dedico este trabalho a meus pais

Wallace

Agradecimentos

Agradecemos a Deus, pois sem Ele nada do que se fez poderia ter sido feito; a Ele que nos deu forças pra superar as dificuldades e vencer as barreiras.

Agradecimentos aos pais

Agradecimentos ao orientador e co-orientador.

Agradecemos aos demais integrantes da Banca Examinadora, os quais, pelo menos em algum momento, desde a origem até a conclusão do trabalho, deram a sua contribuição.

Agradecemos a todos os professores da Universidade Cândido Mendes do curso de ciências da computação, que nos acompanharam e nos ensinaram nessa fase única e marcante de nossas vidas, que foi nossa formação acadêmica.

Eu, Jônatas, agradeço em especial

Eu, Maycon, agradeço em especial

Eu, Wallace, agradeço em especial

Agradecemos também à Chrystiano, Leandro, Lucas e Thiago que participaram diretamente da nossa formação e, juntamente conosco, proporcionaram a conclusão deste trabalho.

E os nossos sinceros agradecimentos a todas as pessoas que, direta ou indiretamente contribuíram para que este trabalho fosse concluído.

Teste

Resumo

Palavras-chave:

Abstract

Keywords:

Sumário

1	Introdução	12
2	Inteligência Artificial	13
2.1	Agentes Inteligentes	13
2.2	Teste de Turing	14
2.3	Técnicas de IA	14
2.3.1	RNA	14
2.3.2	Algoritmos Genéticos	14
2.3.3	Lógica Fuzzy	14
3	Visão Computacional	15
4	Robótica	16
4.1	Sensores	17
4.2	Efetuadores	17
4.3	Percepção	17
4.4	Localização	17
4.5	Mapeamento	17
4.6	Planejamento	17
4.7	Movimento	17
4.8	Controle	17
5	Arquiteturas em Robótica	18
6	O Toolkit Horus	19
6.1	Objetivo	19
6.2	Arquitetura	19
6.3	Módulos do Horus	19
6.3.1	Core do Horus	19
6.3.2	Modulo de Visão	19
6.3.3	Modulo de Mapeamento	19
7	ANPR	20

8	Mapeamento	21
8.1	SLAM	22
8.1.1	Landmark Straction	22
8.1.2	Data Association	23
8.1.3	State Estimation	23
8.1.4	State Update	25
8.1.5	Landmark Update	25
9	Aplicação do ambiente virtual com um robô autônomo	26
9.1	Simulador	26
9.1.1	Requisitos para o ambiente virtual	26
9.1.2	Ambiente	27
10	Conclusões e Trabalhos Futuros	28
	Apêndices	29
A	Dependências do Tool kit	29
B	Instalações	30

Lista de Figuras

Lista de Tabelas

8.1	Algoritmo RANSAC	24
-----	----------------------------	----

Capítulo 1

Introdução

A robótica era um sonho até pouco tempo atrás, hoje em dia a sua existência é tão comum que muitas vezes nos passam despercebidos grandes avanços da área. Cada vez mais robos, ou melhor, agentes estão presentes em nosso dia-a-dia a fim de facilitar nossos afazeres a fim de nos tornar mais produtivos. A robótica móvel é um campo da robótica que estuda as vantagens da mobilidade dos agentes. Essa mobilidade os tornam capazes de avançar ainda mais em ambientes de difícil acesso ao ser humano.

Capítulo 2

Inteligência Artificial

2.1 Agentes Inteligentes

Um Agente, por definição, é todo elemento ou entidade autônoma que pode perceber seu ambiente por algum meio cognitivo ou sensorial e de agir sobre esse ambiente por intermédio de atuadores. Algumas definições do termo agente na língua portuguesa tais como "O que opera ou é capaz de operar", "O que promove negócios alheios" e "Autor". Existem definições de agentes em várias áreas do conhecimento humano.

- Sociologia
- Economia
- Comportamento Animal
- Software
- Robótica

2.2 Teste de Turing

2.3 Técnicas de IA

2.3.1 RNA

2.3.2 Algoritmos Genéticos

2.3.3 Lógica Fuzzy

Capítulo 3

Visão Computacional

Capítulo 4

Robótica

A ciência robótica é reponsável pela parte da tecnologia que tem por intuito otimizar tarefas feitas por humanos e, em alguns casos, substituí-los por motivos que vão desde a preservação da integridade do ser humano até mesmo a ocupação de seu cargo de trabalho. Alheios a um mundo de filmes e preconceitos, os robôs tornam os resultados dos serviços melhores e sua precisão é muito mais que a de um funcionário. A robótica se utiliza de vários dispositivos para emular os sentidos e as reações humanas em determinadas situações. Esses dispositivos tratam diferentemente cada um dos sentidos humanos. Algumas categorias de dispositivos são:

1. Sensores: Dispositivos diversos que "sentem" o mundo e convertem isso para dados passíves de serem analisados.
2. Efetuadores
3. Percepção
4. Localização
5. Mapeamento:
6. Planejamento:
7. Movimento: São as técnicas que podem ser utilizadas para

8. Controle:

4.1 Sensores

4.2 Efetuadores

4.3 Percepção

4.4 Localização

4.5 Mapeamento

4.6 Planejamento

4.7 Movimento

4.8 Controle

Capítulo 5

Arquiteturas em Robótica

Capítulo 6

O Toolkit Horus

6.1 Objetivo

6.2 Arquitetura

6.3 Módulos do Horus

6.3.1 Core do Horus

6.3.2 Modulo de Visão

6.3.3 Modulo de Mapeamento

Capítulo 7

ANPR

Capítulo 8

Mapeamento

O mapeamento é uma funcionalidade que é tratada de muitas formas dentro da literatura. O uso de algoritmos de mapeamento permite que um agente móvel possa identificar sua posição em um ambiente desconhecido e identificar o local em que está inserido. Com o ambiente devidamente mapeado é possível otimizar a rota uma vez que o agente já o conhece.

Algumas técnicas de mapeamento que foram estudadas:

- Técnica utilizando o algoritmo Dijkstra e Subida de Montanha.
- Método incremental convencional.
- Técnica baseada em grafos de visibilidade.
- SLAM (Simultaneous Localization and Mapping).

O método de mapeamento que será incluído no Toolkit Hórus será o SLAM (Simultaneous Localization and Mapping), tendo em vista que ele soluciona dois problemas clássicos da teoria das posições, que define a dificuldade de se localizar em um ambiente desconhecido e a dificuldade de mapear um ambiente onde não se sabe onde está.

8.1 SLAM

O Simultaneous Localization and Mapping é uma técnica utilizada em agentes autônomos para o mapeamento de ambientes desconhecidos levando em consideração a sua posição atual como a posição inicial para início do mapeamento. Os sensores que podem ser utilizados para a implementação do são diversos. Para a prova de conceito foi utilizado o odômetro, dispositivo que mensura distâncias percorridas, e o laser, dispositivo para detectar a presença de objetos no ambiente.

O SLAM é composto por vários segmentos que são independentes e tem suas comunicações muito bem estabelecidas o que os torna mais flexíveis quanto aos algoritmos utilizados em cada um dos segmentos. Cada um dos segmentos tem uma enorme gama de algoritmos que o compõe. Foram incorporadas ao Tool Kit apenas as mais otimizadas e relevantes para melhor utilização no processo.

Esses segmentos são:

1. Landmark Extraction: Segmento responsável pela extração de marcos no ambiente.
2. Data Association: Segmento que associa os dados extraídos de um mesmo marco por diferentes leituras do laser.
3. State Estimation: Segmento responsável por estimar a posição atual do robô com base em seu odômetro e nas extrações de marcos no ambiente.
4. State Update: Segmento que atualiza o estado atual do agente.
5. Landmark Update: Segmento que atualiza as posições dos marcos no ambiente em relação ao agente.

8.1.1 Landmark Straction

A forma de gestão dos marcos (objetos) e dos pontos de movimentação (áreas de movimentação do agente) foi feita através de um grafo. A escolha dessa estrutura foi baseada na sua credibilidade e largo uso na literatura.

Existem dois algoritmos que foram analisados para ser incorporados nesse segmento: o RANSAC e o SPIKE.

8.1.1.1 RANSAC

O RANSAC (Random Sampling Consensus) algoritmo que utiliza-se de uma grande quantidade de informações para estimar os dados relevantes de um modelo matemático. Em conjunto com o SLAM o RANSAC identifica linhas de acordo com os pontos passados pelo laser, através da identificação de pontos muito próximos uns dos outros pode-se concluir que ali existe linhas, ou nesse caso, paredes que impossibilitam a transposição do agente.

Através de vários marcos o RANSAC obtêm uma amostra que será analisada a fim de encontrar pontos próximos, conseqüentemente uma parede, baseado em um limiar de proximidade. Esse limiar é chamado de Consensus.

8.1.1.2 RANSAC

8.1.2 Data Association

O segmento Data Association (tradução livre: Associação de Dados) é responsável pela filtragem e associação dos dados obtidos através dos dispositivos do agente.

Uma vez que um marco seja visualizado em um passo do agente e esse mesmo marco é visto novamente em um novo passo, a sua posição mudou, em relação ao agente. O Data Association faz a análise da posição atual do marco com a sua posição imediatamente anterior, com esse paralelo certifica-se que o marco existe ou se ele foi removido da cena.

Este segmento tem como saída para o State Estimation um lista com os marcos da cena.

8.1.3 State Estimation

O segmento State Estimation (tradução livre: Estimação do Estado) tem com objetivo analisar as informações passadas pelo Data Association e estimar as posições dos marcos

while

- Houverem leituras de laser não associadas.
- **E** o número de leituras for maior que o limiar;
- **E** o número de iterações não for maior que o limite.

do

- Selecionar uma leitura de laser na lista.
- Seleciona uma quantidade S de exemplos de leitura do laser que estão associadas a uma quantidade D de graus daquele laser.
- Usando esses exemplos S e a leitura original para calcular o menor quadrado que se ajuste a linha.
- Determinar quantas leituras do laser estão dentro de X unidades de medida que melhor se ajustam a linha.

if O número de leituras do laser que ficam sobre a linha é maior que o Consensus
then

- Calcular o novo mínimo quadrado que melhor se ajuste a linha, com base em todas as leituras e a linha formada anteriormente.
- Adicionar o melhor ajuste baseado no calculo anterior.
- Remover o número de leituras que cruzam a linha do total de leituras não associadas.

end if

end while

Tabela 8.1: Algoritmo RANSAC

e do agente, com essas informações ele prepara as posições de cada elemento na cena e analisa o estado anterior já gravado.

8.1.4 State Update

O State Update (tradução livre: Atualização do Estado) faz a gravação do estado atual do agente, sua posição, em relação a posição inicial, valores relativos ao odômetro.

8.1.5 Landmark Update

Com o mesmo objetivo e com focos diferentes, o State Update (tradução livre: Atualização de marcos) faz a gravação do estado atual dos marcos, sua posição, em relação a posição do agente.

Capítulo 9

Aplicação do ambiente virtual com um robô autônomo

A fim de produzir uma prova de conceito, foi implementada uma aplicação onde foi possível demonstrar a eficiência do algoritmo de mapeamento. Foi produzido um simulador com gravidade, colisão, renderização de texturas, objetos e atores (o agente nesse caso) e, tendo esse simulador como base, foi criada a aplicação que tem um agente utilizando-se do Toolkit Hórus para mapeamento e navegação dentro do ambiente simulado.

9.1 Simulador

O simulador desenvolvido foi tratado como uma aplicação a parte, pois não estava incluído no escopo do projeto, porém tendo em vista a necessidade de um simulador customizado para a realidade de um agente móvel (e na linguagem selecionada) o simulador foi incluído ao projeto.

9.1.1 Requisitos para o ambiente virtual

Foram utilizados alguns softwares para que fosse possível a criação e renderização do ambiente. Esses softwares foram: Blender e Panda3d. Como uma das linguagens utilizadas foi

Python, buscou-se ferramentas que fossem compatíveis (na verdade ambas são em Python) com as linguagens utilizadas.

9.1.1.1 Blender

O Blender é um produto da Blender Foundation, que é open source e desenvolvido em Python para modelagem de objetos 3D que está disponível para vários sistemas operacionais sobre a licença GNU (General Public License).

9.1.1.2 Panda3D

O Panda3D é um produto da equipe de desenvolvimento da Walt Disney para renderização de jogos e ambientes virtuais em terceira dimensão. Estão sobre licença da BSD License (com algumas modificações para adequação a realidade do projeto).

9.1.2 Ambiente

Capítulo 10

Conclusões e Trabalhos Futuros

Apêndice A

Dependências do Tool kit

Apêndice B

Instalações