

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS ARARANGUÁ**

Joildo Schueroff

**DESENVOLVIMENTO DE UM SISTEMA DE VISÃO
ARTIFICIAL PARA UM ROBÔ EXPLORADOR**

Araranguá, dezembro de 2013.

Joildo Schueroff

**DESENVOLVIMENTO DE UM SISTEMA DE VISÃO
ARTIFICIAL PARA UM ROBÔ EXPLORADOR**

**Trabalho de Conclusão de
Curso submetido à Universi-
dade Federal de Santa Cata-
rina, como parte dos requisitos
necessários para a obtenção do
Grau de Bacharel em Tecno-
logias da Informação e Comu-
nicação.**

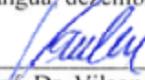
Araranguá, dezembro de 2013.

Joildo Schueroff

**DESENVOLVIMENTO DE UM SISTEMA DE VISÃO
ARTIFICIAL PARA UM ROBÔ EXPLORADOR**

Este Trabalho de Conclusão de Curso foi julgado aprovado para a obtenção do Título de "Bacharel em Tecnologias da Informação e Comunicação", e aprovado em sua forma final pelo Curso de Graduação em Tecnologias da Informação e Comunicação.

Araranguá, dezembro de 2013



Prof. Dr. Vilson Gruber
Coordenador do Curso



Prof. Dr. Anderson Luiz Fernandes Perez
Orientador

Banca Examinadora:



Prof. Dr. Anderson Luiz Fernandes Perez
Presidente



Profª Drª Analúcia Schiaffino Morales



Profª Drª Eliane Pozzebon

Aos meus pais, João Schueroff, e Bernar-
dina B. Schueroff, a minha companheira
Tânia Barbosa de Souza e ao meu filho
Lucas Joildo Schueroff.

AGRADECIMENTOS

Ao meu orientador, o professor Anderson Luiz Fernandes Perez, que como um mestre incansável, me apoiou e me orientou em vários aspectos como ser humano. A minha companheira Tânia que me ajudou em momentos difíceis e sempre me apoiou. A Universidade Federal de Santa Catarina pela oportunidade. Aos colegas do LARM (Laboratório de Automação e Robótica Móvel), que contribuíram com o desenvolvimento do projeto do robô explorador, especialmente ao Elder Tramontin. A minha família, amigos e ao meu irmão que estiveram sempre presentes nos momentos de dificuldades. Aos professores da Universidade Federal de Santa Catarina que contribuíram com minha formação durante a graduação.

Não há nenhum segredo para o sucesso. É o resultado de preparação, trabalho duro, e aprender com o fracasso.

Gen. Colin L. Powell

RESUMO

Um robô móvel explorador de ambientes deve ser dotado de um sistema de visão artificial que seja capaz de captar e processar imagens provenientes de uma câmera a fim de encontrar padrões que possam ser utilizados para identificar um ou mais objetos presentes no ambiente. Um robô móvel equipado com um sistema de visão artificial é de grande utilidade, uma vez que o robô pode tomar decisões baseadas nas informações visuais recebidas e processadas. Este trabalho descreve um sistema de visão artificial para um robô explorador de ambientes indoor e outdoor. O sistema é baseado nas informações visuais lidas a partir de uma câmera posicionada na parte superior do robô. As imagens são processadas com o objetivo de calcular a trajetória do robô até um objeto mapeado, com base na trajetória calculada o sistema determina um novo posicionamento para o robô.

Palavras-chave: Visão Artificial, Processamento Digital de Imagens, Robótica Móvel, Robô Explorador.

ABSTRACT

A rover should be equipped with an artificial vision system that is able to capture and process images from a camera to find patterns that can be used to identify one or more objects present in the environment. A mobile robot equipped with an artificial vision system is useful, since the robot can make decisions based on visual information received and processed. This paper describes an artificial vision system for a rover. The system is based on visual information read from a camera positioned on top of the robot. The images are processed in order to calculate the robot trajectory to an object, based on the calculated trajectory the system determines a new position to the robot.

Keywords: Artificial Vision, Digital Images Processing, Mobile Robots, Rover.

LISTA DE FIGURAS

Figura 1	Exemplo de uma das primeiras imagens digitais.....	28
Figura 2	Exemplo do uso de processamento digital de imagens na medicina.....	30
Figura 3	Etapas de um sistema de processamento de imagens ...	31
Figura 4	Exemplo de operação sobre máscaras para aproximação de uma imagem.....	34
Figura 5	Exemplo de uso do operador de adição.....	36
Figura 6	Exemplo de uso do operador AND em imagens binárias	36
Figura 7	Equalização de cores no histograma.....	38
Figura 8	Imagem do Robô Curiosity.....	42
Figura 9	Exemplos de robôs aplicados a exploração de ambientes	43
Figura 10	Ciclo percepção-ação.....	44
Figura 11	Exemplos de sensores utilizados na robótica.....	44
Figura 12	Exemplos de atuadores utilizados na robótica.....	45
Figura 13	Navegação baseada em mapas.....	49
Figura 14	Navegação baseada em comportamentos.....	50
Figura 15	Arquitetura de controle deliberativa.....	51
Figura 16	Arquitetura de controle reativa.....	52
Figura 17	Arquitetura de controle híbrida.....	52
Figura 18	Processo de determinação de rotas.....	53
Figura 19	Imagem do robô explorador.....	55
Figura 20	Modelo cinemático do robô explorador.....	56
Figura 21	Arquitetura de hardware do robô explorador.....	57
Figura 22	Imagem do Raspberry Pi modelo B.....	57
Figura 23	Freescale FRDM-KL25Z.....	58
Figura 24	Arquitetura de software do robô explorador.....	59
Figura 25	Processo de captura de uma imagem pelo robô explorador	60
Figura 26	Busca da diferença entre o centro e a posição do objeto	61
Figura 27	Projeção de ângulação do servomotor.....	61
Figura 28	Captura do quadrado de cor amarela em um cubo de Rubik.....	64
Figura 29	Gráfico das posições passadas para o servomotor.....	64

Figura 30	Captura de objetos circulares.....	65
Figura 31	Captura de um objeto circular de cor azul.....	66
Figura 32	Gráfico das posições passadas para o servomotor.....	67
Figura 33	Interface do ROS com o hardware.....	79
Figura 34	Interface do RViz.....	81
Figura 35	Reconhecimento de características com o OpenCV.....	83

LISTA DE TABELAS

Tabela 1	Operações sobre máscaras	33
Tabela 2	Operações aritméticas	35
Tabela 3	Operações lógicas	36
Tabela 4	Algumas funcionalidades do ROS.....	80

LISTA DE ABREVIATURAS E SIGLAS

ROS	<i>Robot Operating System</i>	24
OpenCV	<i>Open Source Computer Vision Library</i>	24
GPS	<i>Global Positioning System</i>	25
Pixel	<i>Picture Element</i>	28
3D	Espaço tridimensional.....	29
RGB	<i>Red, Green and Blue</i>	37
RMA	Robôs Móveis Autônomos.....	42
NASA	<i>National Aeronautics and Space Administration</i>	42
SLAM	<i>Simultaneous Localization and Mapping</i>	46
MHz	<i>Mega Hertz</i>	57
RCA	<i>Radio Corporation of America</i>	57
HDMI	<i>High-Definition Multimedia Interface</i>	57
MB	<i>Mega Bytes</i>	57
ARM	<i>Advanced RISC Machine</i>	57
RAM	<i>Random Access Memory</i>	57
SD	<i>Secure Digital</i>	57
GPIO	<i>General Purpose Input/Output</i>	57
I2C	<i>Inter-Integrated Circuit</i>	57
UART	<i>Universal Asynchronous Receiver/Transmitter</i>	57
mA	<i>Miliâmpères</i>	57
USB	<i>Universal Serial Bus</i>	57
CMOS	<i>Complementary Metal-Oxide-Semiconductor</i>	63
BSD	<i>Berkeley Software Distribution License</i>	79
Orocos	<i>Open Robot Control Software Project</i>	80

SUMÁRIO

1 INTRODUÇÃO	23
1.1 OBJETIVOS	24
1.2 MOTIVAÇÃO E JUSTIFICATIVA	24
1.3 METODOLOGIA	25
1.4 ORGANIZAÇÃO DO TRABALHO	25
2 PROCESSAMENTO DIGITAL DE IMAGENS	27
2.1 ORIGENS E PRINCÍPIOS DO PROCESSAMENTO DIGITAL DE IMAGENS	27
2.2 VISÃO COMPUTACIONAL	29
2.3 COMPONENTES DE UM SISTEMA DE PROCESSAMENTO DE IMAGENS	30
2.3.1 Aquisição de Imagens	30
2.3.2 Pré-Processamento de Imagens	31
2.3.3 Extração de Características	31
2.3.4 Detecção e Segmentação	32
2.3.5 Processamento de Alto Nível	32
2.4 RECONHECIMENTO DE PADRÕES EM IMAGENS	32
2.5 OPERAÇÕES SOBRE IMAGENS	33
2.5.1 Operações Aritméticas	34
2.5.2 Operações Lógicas	35
2.5.3 Histogramas	37
2.6 EXPLORAÇÃO DE DADOS DAS IMAGENS	38
3 FUNDAMENTOS DE ROBÓTICA MÓVEL	41
3.1 DEFINIÇÃO DE ROBÓTICA MÓVEL	41
3.2 PERCEPÇÃO E AÇÃO NA ROBÓTICA	43
3.2.1 Sensores	44
3.2.2 Atuadores	45
3.3 LOCALIZAÇÃO, MAPEAMENTO E NAVEGAÇÃO	45
3.3.1 Localização	46
3.3.2 Mapeamento	47
3.3.3 Navegação	48
3.4 ARQUITETURAS DE CONTROLE EM ROBÓTICA MÓVEL	50
3.4.1 Arquitetura Deliberativa	50
3.4.2 Arquitetura Reativa	51
3.4.3 Arquitetura Híbrida	51
3.5 ESTRATÉGIAS DE EXPLORAÇÃO EM AMBIENTES	53

4 PROJETO E DESENVOLVIMENTO DE UM SISTEMA DE VISÃO ARTIFICIAL PARA UM ROBÔ EXPLORADOR	55
4.1 DESCRIÇÃO DO ROBÔ EXPLORADOR	55
4.1.1 Arquitetura de Hardware do Robô Explorador	56
4.1.2 Arquitetura de Software do Robô Explorador	58
4.1.2.1 Sistema Reativo	58
4.1.2.2 Sistema Deliberativo	59
4.2 DESCRIÇÃO DO SISTEMA DE VISÃO ARTIFICIAL	60
5 AVALIAÇÃO DO SISTEMA DE VISÃO ARTIFICIAL	63
5.1 METODOLOGIA DOS EXPERIMENTOS	63
5.2 EXPERIMENTO I - BUSCA DE UM OBJETO COM UMA COR ESPECÍFICA	63
5.3 EXPERIMENTO II - BUSCA DE UM OBJETO COM UM FORMATO ESPECÍFICO	65
5.4 EXPERIMENTO III - BUSCA DE UM OBJETO COM COR E FORMATO ESPECÍFICOS	65
6 CONSIDERAÇÕES FINAIS E PROPOSTAS PARA TRABALHOS FUTUROS	69
6.1 PROPOSTAS PARA TRABALHOS FUTUROS	69
REFERÊNCIAS	71
ANEXO A – Ferramentas Utilizada no Desenvolvimento do Sistema de Visão Artificial do Robô Explorador ...	79

1 INTRODUÇÃO

Com o passar dos anos os sistemas digitais tem impulsionado diversos setores da economia mundial, com destaque para o processamento digital de imagens. Técnicas de processamento de imagens são empregadas para a obtenção de informações mais precisas, visando a melhoria da qualidade de diversos processos e o auxílio na tomada de decisões.

O processamento digital de imagens é útil em áreas como a medicina (RONG et al., 2011), indústria (SARDIS; VARVARIGOU, 2010) e a robótica (CABRE et al., 2013). Ao longo dos anos pesquisas vêm sendo realizadas visando contribuir e/ou melhorar as técnicas de processamento de imagens (OKASAKI; DÜCK; ALVES, 2013).

Segundo (GONZALEZ; WOODS, 2009) o processamento digital de imagens relaciona-se com:

- A melhoria na informação fotográfica para a interpretação humana;
- O processamento de dados, em forma de imagens para armazenamento, transmissão e representação para sistemas computacionais, com a capacidade de percepção automática.

A medida que a capacidade do processamento computacional aumenta, novas tecnologias de processamento de imagens surgem, gerando novos desafios, principalmente no campo exploração e monitoramento.

Informações proveniente do processamento digital de imagens podem trazer muitos benefícios importantes como a identificação de agentes patológicos de algumas doenças e o auxílio a tarefas do cotidiano humano. Por meio de imagens, sistemas computacionais podem realizar tarefas com um grau de acurácia ¹ maior do que os seres humanos.

Um dos desafios na robótica móvel é fazer com que robô conheça o ambiente em que está inserido, possibilitando recolher informações das mais diversas possíveis por meio de sensores. Com o auxílio da visão artificial, um robô pode executar tarefas complexas (BAHN et al., 2011). Segundo (HUANG, 2011) os robôs móveis tem atraído atenção devido a capacidade de mover-se em qualquer direção, em locais dos mais variados e podem ser utilizados para:

¹Aperfeiçoar, apurar, diagnosticar (FERREIRA, 2010).

- Monitoramento;
- Exploração;
- Exploração e monitoramento em conjunto.

1.1 OBJETIVOS

Para melhor descrever os objetivos deste trabalho, esta seção foi separada em Objetivo Geral e Objetivos Específicos.

- **Geral**

Propor e desenvolver um sistema de visão artificial para um robô explorador.

- **Específicos**

1. Levantar o estado da arte sobre o processamento digital de imagens.
2. Levantar o estado da arte sobre robótica móvel com ênfase em navegação baseada em visão artificial.
3. Estudar o framework ROS (*Robot Operating System*).
4. Estudar a biblioteca OpenCV (*Open Source Computer Vision Library*).
5. Desenvolver o sistema de visão artificial para um Robô Explorador.
6. Testar e validar o sistema de visão artificial desenvolvido em (5).

1.2 MOTIVAÇÃO E JUSTIFICATIVA

Com o passar dos anos houve um aumento significativo do uso de sistemas computacionais que utilizam processamento digital de imagens, estes sistemas visam obter informações das imagens com o propósito de usá-las em benefício de aplicações do cotidiano humano. Dados de imagens são necessários em diversas áreas da sociedade, com o intuito de extrair informações importantes que estão ocultas nas imagens.

Algumas áreas como a biologia (AL-FANDI et al., 2010), reconstrução de imagens (ZHENG et al., 2011), robótica (CABRE et al., 2013),

utilizam de alguma forma o processamento digital de imagens. Em particular, na área de robótica móvel, informações provenientes de imagens podem ser utilizadas para auxiliar a localização e a navegação de robôs em ambientes estruturados e não estruturados.

Além das informações provenientes do processamento digital de imagens, um robô pode se localizar em um ambiente com o auxílio de um GPS (*Global Positioning System*), entretanto, este não fornece informações precisas, como as imagens obtidas a partir de um sistema de processamento de imagens.

Sistemas de visão artificiais são uma tentativa de emular a visão humana, mas com uma capacidade de extrair informações mais ricas que os seres humanos, dados relevantes podem ser encontrados apenas com o processamento digital de imagens o que proporciona uma área de grande interesse para pesquisadores.

1.3 METODOLOGIA

Este trabalho é uma pesquisa tecnológica que visa propor e desenvolver um sistema de visão artificial para robôs móveis utilizados na exploração de ambientes.

O sistema foi desenvolvido na linguagem de programação C++ para o ROS (*Robot Operating System*). O hardware utilizado foi uma placa do tipo Raspberry Pi modelo B. Para a validação foram realizados experimentos visando aferir o comportamento do sistema de visão em diferentes cenários.

1.4 ORGANIZAÇÃO DO TRABALHO

Este documento, além desta **Introdução** está organizado em mais 5 (cinco) capítulos e 1 (um) anexo, que abordam os seguintes conteúdos:

O **Capítulo 2** descreve os fundamentos do processamento digital de imagens abordando o histórico, técnicas e áreas de aplicação. O capítulo também discorre sobre a visão computacional, área do processamento digital de imagens muito utilizada em robótica móvel.

O **Capítulo 3** descreve sobre os princípios da robótica móvel abordando conceitos de sensores e atuadores e as principais arquiteturas de controle utilizadas em robótica móvel.

O **Capítulo 4** descreve o sistema de visão artificial desenvolvido.

Para um melhor entendimento do capítulo, inicialmente, é descrito o hardware e o software de controle utilizados no robô explorador.

No **Capítulo 5** são apresentados os resultados de 3 (três) experimentos realizados com o sistema de visão artificial desenvolvido para o robô explorador.

O **Capítulo 6** descreve a conclusão do trabalho e apresenta algumas propostas para trabalhos futuros.

O **Anexo A** descreve as ferramentas utilizadas para o desenvolvimento do sistema de visão artificial do robô explorador.

2 PROCESSAMENTO DIGITAL DE IMAGENS

Este capítulo descreve os fundamentos do processamento digital de imagens abordando o histórico, técnicas e áreas de aplicação. O capítulo também discorre sobre a visão computacional, área do processamento digital de imagens muito utilizada em robótica móvel.

2.1 ORIGENS E PRINCÍPIOS DO PROCESSAMENTO DIGITAL DE IMAGENS

Um das primeiras aplicações das imagens digitais ocorreu na indústria de jornais, quando as imagens eram enviadas por cabos de submarinos entre as cidades de Londres e Nova York (GONZALEZ; WOODS, 2009). A implementação de sistema de transmissão de imagens por cabo submarino (conhecido como Bartlane) começou na década de 1920, e reduziu de mais de uma semana para menos de três horas o tempo para transportar uma foto pelo oceano Atlântico.

Um equipamento de impressão especializado codificava as imagens para a transmissão por cabo e depois a reconstruía no recebimento. Alguns problemas iniciais de melhora de qualidade visual dessas primeiras imagens digitais estavam relacionadas a seleção de procedimento de impressão e a capacidade de distribuição dos níveis de intensidade (FILHO, 1999).

Os primeiros sistemas Bartlane eram capazes de codificar imagens em cinco tons de cinza distintos, essa capacidade aumentou em 1929 para quinze tons de cinza. Bartlane não é considerado um modelo de processamento digital de imagens por não ter nenhuma relação com computadores.

O grande impulso para o processamento digital de imagens surgiu em três décadas mais tarde com o advento dos primeiros computadores digitais de grande porte e o programa espacial norte americano. Para (FILHO, 1999) de 1964 aos dias atuais, a área de processamento digital de imagens vem crescendo expressivamente e suas aplicações permeiam quase todos os ramos da atividade humana.

O processamento digital de imagens consiste em técnicas para capturar, representar e transformar imagens com o auxílio do computador. Segundo (GONZALEZ; WOODS, 2009) o campo do processamento digital de imagem se refere ao processamento digital de uma imagem por um computador. A Figura 1 representa uma das primeiras imagens

digitais.



Figura 1 – Exemplo de uma das primeiras imagens digitais
Extraído de (GONZALEZ; WOODS, 2009)

Uma imagem digital é composta por um número finito de elementos, com tamanho, localização e valores específicos. Esses elementos são chamados de elementos pictóricos, conhecido por pixel (*Picture Element*). Pixel é o termo mais utilizado para representar os elementos de uma imagem digital ou binária.

A visão é um dos sentidos mais avançados de várias espécies de seres vivos, e tem um papel importante na percepção. Aparelhos de processamento de imagem são uma tentativa de emular aspectos da visão dos seres vivos, capacitando a máquina a “enxergar”.

Segundo (PEDRINI; SCHWARTZ, 2008) o objetivo do processamento de imagens está relacionado com a obtenção de informações sobre quais frequências espaciais estão presentes em uma imagem ou em qual orientação ocorrem as maiores variações de intensidade de tons de cinza. Após a determinação desses dados pode-se, por exemplo, aplicar filtros sobre os dados visando eliminar componentes indesejados.

Para extrair informação de imagens digitais de uma forma adequada são utilizados dados estatísticos ou coeficientes de Fourier (funções de base sinusoidal) (XIMENES et al., 2010). Com essas medidas multidimensionais de distância reduz-se o tempo computacional gasto para o processamento das imagens (GONZALEZ; WOODS, 2009).

A visão da máquina geralmente se refere a combinação de análise de imagem com outros métodos da tecnologia como a reconstrução de

cena, detecção de eventos, monitoramento de vídeo, reconhecimento de objetos, aprendizagem, indexação, estimativa de movimentos e restauração de imagens (MILLER; FELS; OLDRIDGE, 2011).

Com exceção das etapas de aquisição e exibição, a maioria das funções de processamento de imagens pode ser implementada por software (FILHO, 1999). O uso de hardware especializado para processamento de imagens torna-se necessário em situações onde certas limitações do computador forem consideradas intoleráveis.

2.2 VISÃO COMPUTACIONAL

Segundo (GERONIMO et al., 2013) a visão computacional é um subcampo da inteligência artificial, que visa a compreensão de imagens fixas e sequências de vídeo, exemplos incluem reconhecer pessoas ou objetos, reconstruir formas das mais variadas possíveis, com um tamanho finito.

A visão computacional é uma importante ferramenta para o reconhecimento de ambientes, pois as informações resultantes do processamento das imagens podem ser utilizadas para o reconhecimento e a detecção de padrões. Durante a captura e o processamento de imagem são extraídas informações que podem definir qual ação deverá ser tomada, baseado em padrões encontrados nas imagens captadas.

Há várias décadas pesquisadores tentam construir aplicações com a capacidade de substituir os seres humanos em diversas tarefas que envolvam riscos, principalmente em ambientes hostis ou onde a rapidez e a precisão são essenciais. Somente nas últimas décadas, com o grande avanço em recursos computacionais, foi possível desenvolver aplicações onde a complexidade é exigida com mais rigor (QUILES; ROMERO, 2004).

Segundo (RICHA, 2013) o que ocorre na prática é que para resolver uma determinada tarefa, a interpretação de imagens é utilizada em conjunto com algoritmos específicos de processamento de imagens.

Esses algoritmos são divididos em grupos, como filtros de contraste, detectores de bordas de objetos, segmentadores de imagens em regiões e classificadores de texturas (PIRES; MEDEIROS; ELIAS, 2008).

Imagens são utilizadas em diversas áreas como medicina (RONG et al., 2011), biologia (AL-FANDI et al., 2010), exploração espacial (NASA, 2013), computação gráfica (ZHENG et al., 2011), entre outras.

A Figura 2 ilustra um exemplo do uso do processamento digital de imagens na medicina, juntamente com a tecnologia 3D (espaço tridimensional), ilustrando informações importantes de um rosto de feto

de 22 semanas sendo impresso em 3D.



Figura 2 – Exemplo do uso de processamento digital de imagens na medicina

Extraído de (FOLHA, 2013)

2.3 COMPONENTES DE UM SISTEMA DE PROCESSAMENTO DE IMAGENS

Sensores são utilizados para a captura de imagens, como uma câmera de monitoramento ou uma máquina fotográfica, esses dispositivos vão alimentar o sistema com informações provenientes das imagens sequencialmente captadas. Após a captura, as imagens são entendidas pelos sistema computacional, permitindo que o sistema extraia informações sobre o ambiente.

O diagrama apresentado na Figura 3 representa as etapas de um sistema de processamento de imagens.

As etapas ilustradas na Figura 3 determinam a ordem que as informações são passadas do meio abstrato do ambiente físico para um sistema computacional. Nas subseções seguintes cada etapa será descrita em detalhes.

2.3.1 Aquisição de Imagens

Esta é a fase de captação das informações provenientes do ambiente e tem como objetivo converter uma imagem em uma representação numérica adequada para o processamento subsequente, é a fase onde o



Figura 3 – Etapas de um sistema de processamento de imagens

sensor produz informações do ambiente e faz a transformação para um modelo compreendido pelo computador (FILHO, 1999).

Imagens podem ser produzidas por um ou mais sensores e podem ser representadas de forma bi ou tridimensional, ou ainda como uma sequência de imagens, dependendo do tipo do sensor que está sendo utilizada para a captura.

2.3.2 Pré-Processamento de Imagens

Segundo (ARAÚJO, 2009) o pré-processamento é uma melhoria da imagem visando aumentar as chances de sucesso das etapas seguintes. Consiste em aplicar técnicas para melhoria do contraste, remoção de ruídos, correção de foco, otimização da imagem e correção de defeitos gerados durante o processo de captura, proporcionando qualidade para as etapas posteriores.

2.3.3 Extração de Características

A extração de características trata da segmentação da imagem e visa isolar um conjunto de objetos e padrões de interesse. É importante ressaltar que é uma etapa complexa para um sistema de processamento digital de imagens, pois está relacionado com a solução do problema e a

identificação de dados específicos, tais como identificação de objetos, ou formas com medidas e características específicas (FELICIANO; SOUZA; LETA, 2010).

2.3.4 Detecção e Segmentação

O objetivo da etapa de detecção e segmentação é a análise das imagens digitais, ou seja a análise pixel a pixel de maneira isolada (FILHO, 1999). Neste processo, divide-se a imagem em regiões que devem corresponder às áreas de interesse da aplicação. Entende-se por regiões um conjunto de pixels contíguos que se espalham em várias direções do frame.

2.3.5 Processamento de Alto Nível

O processamento de alto nível tem a função de atribuir padrões de reconhecimento e ativar a validação que recebeu anteriormente, dependendo da situação é a etapa de exibição das informações da imagem após o processamento.

2.4 RECONHECIMENTO DE PADRÕES EM IMAGENS

Para (SIQUEIRA, 2010), o reconhecimento de padrões de uma imagem abrange diversas áreas como medicina (RONG et al., 2011), inteligência artificial (CABRE et al., 2013), entre outras, onde o reconhecimento de padrões tem um fator importante para extrair informações sobre determinados pontos das imagens que são relevantes para o usuário ou sistema que necessita das informações das imagens.

A dificuldade no reconhecimento de padrões se dá devido a interferências que podem surgir durante a aquisição da imagem pelo equipamento de processamento de imagens. Padrões de reconhecimento resultam em sistemas especializados, ou seja, um padrão é formado por um conjunto de características, por exemplo, cor, intensidade de cinza e textura (GONZALEZ; WOODS, 2009).

Fatores internos e externos podem influenciar e interferir no reconhecimento de padrão das imagens, como a luminosidade (RUSS, 2010). Esses problemas podem prejudicar a tarefa de reconhecimento de padrões, mas podem ser resolvidos com a aplicação de máscaras, durante o processo de aquisição das imagens.

Para (FELICIANO; SOUZA; LETA, 2010) ao interpretar uma imagem, variações de valores de intensidade devem ser analisadas. Os números de níveis de quantificação usados mais comumente para representar a intensidade em imagens digitais são cerca de 256 diferentes níveis de cinza. Isso produz um esforço computacional mais elevado, estas limitações encorajam o uso de sistemas de visão binária que contêm apenas dois níveis de cinza e que apresentam um melhor desempenho.

2.5 OPERAÇÕES SOBRE IMAGENS

Operações lógicas e aritméticas podem ser utilizadas para modificar imagens (PEDRINI; SCHWARTZ, 2008). Segundo (ARAUJO, 2009) em imagens binárias os pixels podem assumir valores 0 ou 1 . Para imagens em níveis de cinzas ou digitais normalmente, os valores dos pixels variam de 0 a 255 , e no caso de imagens coloridas, cada pixel pode ser representado por um conjunto de três a quatro valores dependendo do sistema de cores empregado.

Embora essas operações permitam uma forma simples de processamento, há uma grande variedade de aplicações em que tais operações podem produzir resultados de interesse prático. Além das operações lógicas e aritméticas é possível realizar o processamento orientado a vizinhança, que é chamado de operações sobre máscara.

A Tabela 1 lista as operações sobre máscaras que podem ser aplicadas em uma imagem digital.

Tabela 1 – Operações sobre máscaras

Região de Operação sobre Máscara		
$F(x-1, y-1)$	$F(x, y -1)$	$F(x+1, y -1)$
$F(x-1, y)$	$F(x,y)$	$F(x+1, y)$
$F(x- 1, y+1)$	$F(x, y+ 1)$	$F(x+1, y +1)$

Extraído de (PEDRINI; SCHWARTZ, 2008)

O objetivo principal da utilização de máscara é fazer com que o valor atribuído a um pixel seja a função do seu próprio valor e dos valores de seu vizinho (PEDRINI; SCHWARTZ, 2008). Esse tipo de operação está relacionado com componentes ao seu redor. Seus principais objetivos são:

- Homogeneização da imagem ou de alvos específicos;
- Extração de ruídos da imagem;
- Simulação de imagens com resoluções radiométricas menores;
- Melhora na discriminação de alvos da imagem;
- Detecção de bordas entre alvos distintos presentes na imagem;
- Detecção de formas, entre outros.

A aplicação de máscara em cada pixel é uma tarefa de alto custo computacional, o que requer um certo cuidado e deve ser utilizada com necessidades específicas. Dependendo da aplicação, operações sobre máscaras tem um importante efeito, tais como, realce da imagens, remoção de ruídos, aproximação, recuo, rotação, etc.

A Figura 4 ilustra uma operação de aproximação de uma imagem com a estratégia do uso de máscaras.

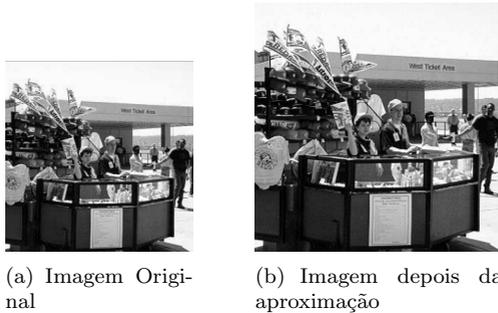


Figura 4 – Exemplo de operação sobre máscaras para aproximação de uma imagem

Extraído de (FILHO, 1999)

2.5.1 Operações Aritméticas

As operações aritméticas tem um importante papel no processamento de imagens digitais. Dadas duas imagens, F_1 e F_2 , as operações aritméticas mais comuns entre dois pixels $F_1(x, y)$ e $F_2(x, y)$ são a adição, a subtração, a divisão e a multiplicação, definidas conforme a Tabela 2.

A função das operações aritméticas consiste em:

Tabela 2 – Operações aritméticas

Operações Aritméticas	
Adição	$F_1(x, y) + F_2(x, y)$
Subtração	$F_1(x, y) - F_2(x, y)$
Multiplicação	$F_1(x, y) \cdot F_2(x, y)$
Divisão	$F_1(x, y) / F_2(x, y)$

Extraído de (PEDRINI; SCHWARTZ, 2008)

- Adição: sobrepor uma imagem.
- Subtração: identificar diferenças entre duas imagens.
- Multiplicação e divisão: ajustar o brilho e eventualmente corrigir problemas que possam surgir durante o processo de aquisição de imagens.
- Outras Operações: incluem a filtragem da imagem no domínio da frequência e na modelagem de ruídos, podem ter influência nas operações anteriores.

As operações aritméticas podem produzir imagens com valores fora do intervalo de níveis de cinza das imagens originais, e alguns cuidados devem ser tomados para contornar essa situação (PEDRINI; SCHWARTZ, 2008).

Uma adição de duas imagens pode resultar em um valor superior a 255 para um determinado pixel. A maneira de resolver este problema é, após o cálculo, realizar uma transformação da escala de cinza na imagem resultante para manter os valores dentro do intervalo adequado.

A Figura 5 (c) representa o resultado de uma adição da Figura 5 (a) com a Figura 5 (b).

2.5.2 Operações Lógicas

Operações lógicas são aplicadas apenas em imagens binárias e se descrevem como 0 (preto) e 1 (branco). As operações lógicas podem ser utilizadas para combinar informações entre as imagens e extrair regiões de interesse (PEDRINI; SCHWARTZ, 2008). A Tabela 3 lista as operações lógicas que podem ser aplicadas sobre imagens digitais.

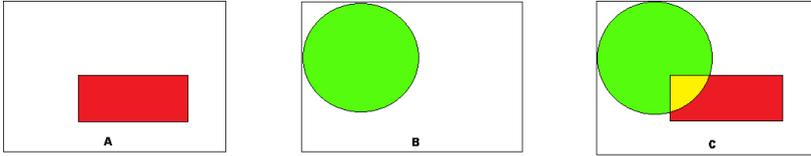


Figura 5 – Exemplo de uso do operador de adição
Adaptado de (FILHO, 1999)

Tabela 3 – Operações lógicas

Operações Lógicas	
AND	$F_1(x, y)$ and $F_2(x, y)$
OR	$F_1(x, y)$ or $F_2(x, y)$
XOR	$F_1(x, y)$ xor $F_2(x, y)$
NOT	NOT $F_1(x, y)$

Extraído de (PEDRINI; SCHWARTZ, 2008)

Esse tipo de operação é importante devido ao menor custo computacional envolvido, possui um desempenho mais satisfatório em relação as operações aritméticas, o que pode ser uma opção dependendo da aplicação de processamento de imagens. A Figura 6 (c) representa o resultado de uma operação AND da Figura 6 (a) com a Figura 6 (b).

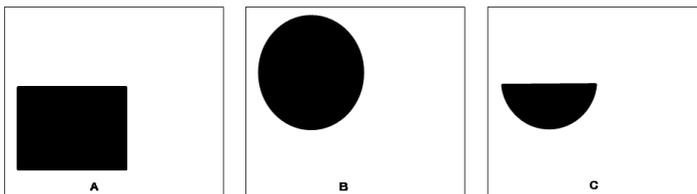


Figura 6 – Exemplo de uso do operador AND em imagens binárias
Adaptado de (FILHO, 1999)

2.5.3 Histogramas

Segundo (PEDRINI; SCHWARTZ, 2008) o histograma de uma imagem corresponde à distribuição dos níveis de cinza nesta imagem, e pode ser representado por um gráfico indicando o número de pixel para cada nível de cinza. Toda imagem possui um único histograma, que não contém informações espaciais, apenas valores de intensidade.

O histograma pode ser visto como uma distribuição discreta de probabilidade devido a este ser utilizado para calcular a probabilidade de se encontrar um pixels com um determinado valor de cinza na imagem. Para (FILHO, 1999) o histograma de uma imagem é simplesmente um conjunto de valores que indicam o percentual de pixel que apresenta um determinado nível de cinza na imagem.

Estes valores são normalmente representados por um gráfico de barras que representam um valor para cada nível de cinza, ou o percentual de pixels correspondentes aquela imagem. Através do histograma de uma imagem se obtém a indicação de sua qualidade quando ao seu nível de contraste e quanto ao seu brilho médio, bem como informações de cores que a imagem possui.

Um histograma possui 255 tonalidades de cores e sua utilização consiste no realce da imagem, podendo-se obter uma:

- Equalização global;
- Equalização regional por blocos;
- Equalização regional pontual.

Na equalização global todo o contexto da imagem é convertido conforme os tons de cinza, na equalização regional por blocos e a regional pontual é realizada uma varredura da área estabelecida pelo usuário. Com esse método é possível extrair informações relevantes conforme a finalidade do sistema de processamento de imagens. Um exemplo da quantidade de cores no padrão RGB (*Red, Green and Blue*) é ilustrado na Figura 7.

O histograma tem a função de restaurar informações para um determinada aplicação, como por exemplo o realce de imagens, possibilitando obter diversas informações com esse método, variando de acordo com a necessidade do sistema em questão, mostrando informações e o grau de qualidade de uma determinada imagem.

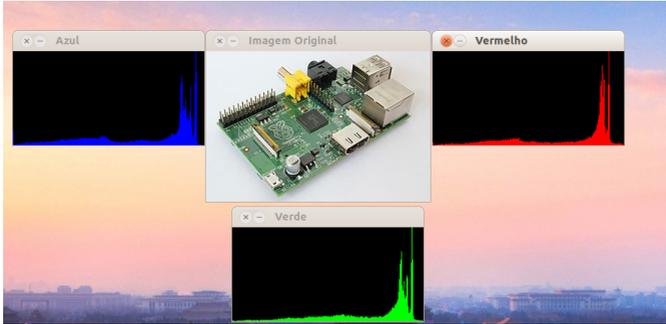


Figura 7 – Equalização de cores no histograma

2.6 EXPLORAÇÃO DE DADOS DAS IMAGENS

A exploração de dados sobre imagens é frequentemente utilizada, com o auxílio de filtros, realce e suavização durante o processamento de imagens e permite alterar ou extrair informações. A efetivação de um filtro em uma imagem deve ser realizada pela operação de convolução entre a imagem em questão e uma máscara utilizada em um determinado filtro (GOMES, 2009).

O principal objetivo das técnicas de realce é processar uma certa imagem de modo que a imagem resultante seja mais adequada que a imagem original (FILHO, 1999). Vários tipos de filtros são implementados de acordo com a necessidade do problema a ser resolvido. Esse processo de modificação inclui:

- Suavização de imagens;
- Eliminação de ruídos;
- Realce de características;
- Detecção de bordas.

As imagens podem conter ruídos aleatórios, que sobrepõem-se aos valores de brilho do pixel devido ao ruído gerado nos sensores durante a captura dos dados da imagem. Esse método mostra regiões de homogeneidade que podem ser removidos pelo processo de filtragem (RICHARDS, 2012).

A técnica de suavização propõe uma filtragem geral que produz uma imagem com menos detalhes que a original, seu resultado tende a

eliminar pequenos detalhes e destacar grandes estruturas da imagens (FILHO, 1999).

3 FUNDAMENTOS DE ROBÓTICA MÓVEL

Este capítulo descreve os princípios da robótica móvel abordando conceitos de sensores e atuadores e as principais arquiteturas de controle utilizadas em robótica móvel.

3.1 DEFINIÇÃO DE ROBÓTICA MÓVEL

A robótica pode ser definida como a arte de projetar e aplicar robôs ou sistemas robóticos em empreendimentos humanos. Seus conceitos envolvem, entre outros, mecânica, elétrica, eletrônica, computação e biologia (NIKU, 2010).

A robótica vem sendo estudada nos últimos anos em várias linhas de pesquisas afim de melhorar aspectos relacionados a sistemas mecânicos capazes de realizar tarefas independentes. A evolução tem sido constante, principalmente no projeto de novos sensores e atuadores mais eficientes.

Robôs móveis devem ter a capacidade de tomar decisões inteligentes para realizarem um planejamento do seus movimentos, evitando principalmente choques com obstáculos (SRIVASTAVA; NANDI, 2010). Segundo (STECKERT, 2013) robôs podem ser classificados em 3 (três) categorias, descritas abaixo:

- Robôs Manipuladores: provém a capacidade de manipular objetos em geral.
- Robôs Móveis: são dotados de equipamentos mecânicos, que possibilitam a movimentação física.
- Robôs Híbridos: são basicamente robôs móveis equipados com dispositivos manipuladores.

Para (RUSSEL; P.NORVING, 2004) os robôs manipuladores são os mais comuns e seus movimentos envolve uma cadeia inteira de articulações controláveis, permitindo que coloquem os atuadores em qualquer posição, são usados principalmente em trabalhos repetitivos.

Um robô móvel pode se deslocar no ambiente por meio de rodas, pernas, asas ou mecanismos semelhantes e são usados principalmente para monitoração e mapeamento de ambientes. Um exemplo de robô móvel é o Curiosity, ilustrado na Figura 8, enviado a Marte para explorar o solo daquele planeta.



Figura 8 – Imagem do Robô Curiosity
Extraído da (NASA, 2013)

Em especial os robôs híbridos tem recebido ao longo dos últimos anos um amplo destaque junto a mídia e a sociedade de um modo geral, principalmente os RMAs (Robôs Móveis Autônomos), que possuem características de locomoção e podem operar de forma semi ou completamente autônoma (WOLF et al., 2009). Robôs híbridos devem possuir algumas características específicas como:

- Fusão de sensores: adquirir e integrar diversos tipos de dados, a partir dos sensores compensando as limitações de alcance de cada sensor, através da fusão de dados.
- Desviar de obstáculos: detectar obstáculos e evitar as colisões, com intuito de preservar a integridade do robô.
- Auto localização: determinar sua posição e localização no ambiente.
- Mapeamento do ambiente: capacidade de realizar a exploração e conseguir realizar a construção de um mapa do ambiente.
- Planejamento de trajetória: estabelecer uma rota até o ponto de destino desejado.
- Planejamento de ações: estabelecer um plano de ações.

- Navegação robótica: realizar ações previamente definidas pelo plano de trajetória.
- Interação e comunicação: capacidade de interagir e se comunicar com outros agentes no ambiente.

Robôs híbridos são utilizados com intensidade no campo da exploração. A Figura 9 (a) ilustra o exemplo do robô Snake usado para desobstrução de tubulações, e a Figura 9 (b) o robô Ambiental Híbrido Chico Mendes que foi desenvolvido pela Petrobrás para uso em ambientes aquáticos.



(a) Robô Snake. Extraído de (SNAKE, 2013).



(b) Robô híbrido aquático. Extraído de (STEIN et al., 2007).

Figura 9 – Exemplos de robôs aplicados a exploração de ambientes

3.2 PERCEPÇÃO E AÇÃO NA ROBÓTICA

Dependendo da aplicação sistemas robóticos podem ser simples ou complexos. Um agente robótico é definido como uma entidade que influencia e é influenciada pelo meio, percebe o ambiente por meio de sensores e age sobre ele por meio de atuadores (RUSSEL; P.NORVING, 2004).

A percepção e a ação formam um ciclo que pode ser observado na Figura 10. A Seções 3.2.1 e 3.2.2 descrevem em detalhes sensores e atuadores.

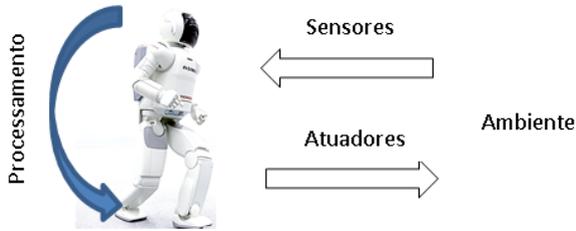


Figura 10 – Ciclo percepção-ação

3.2.1 Sensores

Para (STECKERT, 2013) sensores são responsáveis pela extração de informações do ambiente, que são interpretadas e transformadas em informações úteis. Sensores são responsáveis pela alimentação do sistema robótico de modo geral e sua principal função é fornecer ao robô informações necessárias para executar uma ação “inteligente” de forma autônoma (SICILIANO; SCIAVICCO; VILLANI, 2009).

Sensores permitem a um robô detectar a presença de um objeto no espaço de trabalho e para medir o alcance desde uma dada direção. São dispositivos que podem sentir e abstrair certas propriedades físicas do ambiente, como a temperatura, luminosidade, resistência ao toque, peso, tamanho, etc (KLIPP et al., 2013).

A Figura 11 ilustra 2 (dois) tipos de sensores usados na robótica.



Figura 11 – Exemplos de sensores utilizados na robótica

O principal problema a ser resolvido em um robô móvel é gerar trajetórias e guiar seu movimento de acordo com os dados recebidos pelos sensores.

3.2.2 Atuadores

Atuadores são mecanismos que, conectados através de juntas e articulações dos robôs, são responsáveis por gerar a força necessária para movimentar o sistema robótico, agindo como os músculos do sistema, e suas principais funções são de locomoção e manipulação de objetos no ambiente (STECKERT, 2013). A Figura 12 ilustra 2 (dois) tipos de atuadores usados na robótica.



Figura 12 – Exemplos de atuadores utilizados na robótica

O grau de autonomia de um robô depende em grande medida da capacidade do robô para abstrair informações ao seu redor visando realizar uma tarefa de maneira eficaz. As ações de um robô são desempenhadas diretamente pelos atuadores que entram em contato com o ambiente (KLIPP et al., 2013).

3.3 LOCALIZAÇÃO, MAPEAMENTO E NAVEGAÇÃO

A localização é um dos problemas mais pervasivos da robótica porque o conhecimento sobre locais em que o robô se encontra, e em que locais estão os objetos, é um fator fundamental para qualquer interação física bem sucedida (RUSSEL; P.NORVING, 2004).

Com as informações dos sensores e a integração das informações sensoriais, é possível que um ou mais robôs possam executar tarefas com uma eficiência desejada. As informações úteis do ambiente servem de entrada para um determinado algoritmo de controle robótico que percebe o que está em sua volta (STECKERT, 2013).

A percepção na robótica pode ser realizada com vários tipos de sensores, como de temperatura, som, infravermelho, câmera e outros, conforme apresentado na Seção 3.2.1. Para (ATHERTON; GOODRICH, 2011) o robô pode ser definido como uma máquina para detectar e agir

sobre o mundo, embora possam sentir o mundo através de sensores, podem ter a capacidade de realizar tarefas complexas em ambientes *outdoor* ou *indoor*.

Segundo (RUSSEL; P.NORVING, 2004) existem dois modos de um agente robótico utilizar suas percepções:

- Extração de características: o equipamento robótico percebe o ambiente através de alguns sensores e reage de acordo com as informações recebidas.
- Baseada em modelo: os sensores captam informações e reconstroem um modelo de forma abstrata em linguagem de máquina.

Um fator importante na robótica móvel é a localização, é fundamental que o robô conheça sua localização no ambiente na qual está imerso. Segundo (ERDTMANN, 2009) a questão da localização, mapeamento e navegação, foi um dos primeiros problemas a serem propostos nos estudos da robótica móvel.

O caráter de mobilidade acaba atrelando a ideia de mudar de posição e local, o que faz com que seja necessária alguma representação dos lugares e alguma estratégia para, dada a posição atual, chegar a algum lugar desejado.

3.3.1 Localização

Para (SANCHES, 2009) o problema da localização em robótica é a determinação da posição do robô em relação ao mapa do ambiente, esse mapa é definido em um sistema global de coordenadas. A localização é a transformação de coordenadas entre o sistema de coordenadas local do robô e o sistema de coordenadas global externo onde o mapa é extraído.

A posição do robô pode ser definida como um vetor contendo as coordenadas x e y com relação ao sistema cartesiano de coordenadas global e a orientação do robô. O robô deve conhecer sua posição no ambiente em qualquer ponto no tempo.

Determinar sua própria posição é uma capacidade básica para que qualquer tarefa de navegação possa ser executada. A partir de informações referentes a posição, é possível planejar uma trajetória até seu destino e cumprir as tarefas que lhe forem alocadas, por exemplo, com o uso do algoritmo de mapeamento e localização simultâneos (SLAM do inglês, *Simultaneous Localization and Mapping*)(THRUN,

2008). O problema de localização de robôs móveis pode ser dividido em:

- Local: o robô não conhece sua localização no ambiente que está inserido.
- Global: o robô não possui informações sobre sua localização global.

Ambos os problemas descritos acima podem ser resolvidos com sensores, mas as tarefas a serem executadas determinam a necessidade do robô conhecer sua localização a priori. De posse das informações de localização é possível reconhecer o ambiente através do mapeamento.

3.3.2 Mapeamento

(WOLF et al., 2009) argumenta que o levantamento de dados do ambiente no qual o robô está inserido é uma consequência das informações dos mapas que serão gerados e através destes que o robô verifica os possíveis caminhos que os levam a posição desejada e os obstáculos que devem ser evitados. Segundo (RUSSEL; P.NORVING, 2004) a maior parte dos robôs atuais utiliza algoritmos determinísticos para a tomada de decisões.

Segundo (ERDTMANN, 2009) a primeira etapa a definir quando se deseja implementar um sistema de posicionamento é o modelo de mapa. Com a ajuda de sensores pode-se reduzir erros no mapeamento. Para (SOUZA, 2012) um mapeamento pode ser realizado em diferentes tipos de espaço, seja eles em ambientes internos (*indoor*), ambientes externos (*outdoor*), ambientes subterrâneos ou subaquáticos.

Quanto a utilização, os mapas construídos podem ser considerados na realização de diversas tarefas desde as mais simples, tais como, desvio de obstáculos, planejamento de caminhos e localização, às de maior complexidade, por exemplo, exploração de minas subterrâneas, de instalações nucleares, limpeza de ambientes tóxicos, combate a incêndios e resgate de vítimas, entre outras atividades de maior dificuldade (SOUZA, 2012).

São criadas estruturas de dados que serão utilizadas pelos algoritmos, sendo as principais:

- Grade: consiste em dividir a área em várias células, onde é possível saber quais células estão ocupadas, permitindo assim traçar uma rota até o objetivo final.

- Vetorial: armazenam os obstáculos como uma lista de entidades geométricas (pontos, retas e outros) dentro do espaço.
- Topológico: armazenam informações em uma tabela, e usam algoritmos de roteamento para estabelecer uma rota.

Para que esse método tenha sucesso é comum extrair o estado mais provável da distribuição de estados produzida pelo algoritmo de localização. Tanto as técnicas de navegação, como as de localização são problemas desafiadores para os roboticistas.

Como um robô inicialmente não tem informações sobre o ambiente, onde a estrutura do mapeamento é dividida e classificada como desconhecidas, a medida que novas informações são obtidas pelos sensores, a classificação da célula é atualizada (WOLF et al., 2009). Na robótica vários tipos de mapas podem ser construídos, esses podem ser gerados anteriormente e pré-estabelecidos no robô, ou em tempo de execução, dependendo problema em questão.

Segundo (SOUZA, 2012) para que um sistema robótico tenha sucesso em seu mapeamento é interessante que possua sensores que capturem informações do seu entorno.

3.3.3 Navegação

O processo de navegação consiste em achar um caminho da posição atual até a posição final desejada, estabelecer uma trajetória e alterá-la caso necessário (ERDTMANN, 2009). Cada trajetória depende de um método ou plano estabelecido, que pode ter alterações caso o sistema encontre situações inesperadas. Esses métodos de trajetória são abordados na inteligência artificial.

Segundo (WOLF et al., 2009) a tarefa de navegação usualmente é implementada através do uso de uma arquitetura de controle do tipo reativo, deliberativo ou hierárquico/híbrido, descritas nas Seções 3.4.2, 3.4.1 e a 3.4.3, respectivamente, devendo incluir mecanismo para evitar colisões, desvio de obstáculos e o tratamento de situações inesperadas. Na navegação local, o robô depende das informações lidas dos sensores, desta forma pode encontrar o caminho para a meta final (SHAYESTEGAN; MARHABAN, 2012).

Embora o robô móvel possa estar equipado com sensores, ele é incapaz de contornar obstáculos de forma global, devido à desconhecer o ambiente, que muitas vezes é considerado complexo (QIAN; SONG, 2012). O modo de navegação global tradicional é difícil de aplicar

devido a dificuldade de representar o ambiente.

Um dos métodos utilizados é o baseado em mapas, onde o robô pode possuir informações sobre o ambiente, no qual está executando, esse modelo armazena informações sobre o ambiente com o intuito de usar posteriormente para uma eventual situação. A Figura 13 ilustra o esquema de navegação baseado em mapas.

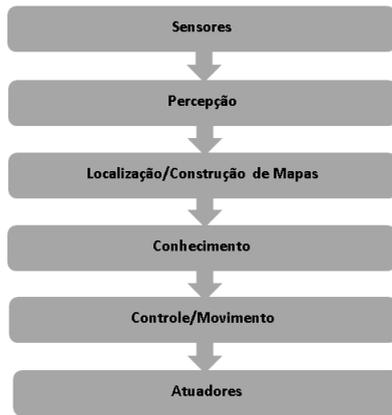


Figura 13 – Navegação baseada em mapas

Na navegação de um robô móvel há dois importantes fatores, a localização e a estimativa de posição. Portanto, o objetivo é estimar com precisão o local e a posição de um robô com a utilização de funções trigonométricas e coordenadas cartesianas (PARK; HASHIMOTO, 2009).

A navegação sem auxílio de mapas não requer nenhum conhecimento sobre o ambiente, onde as técnicas de navegação são baseadas em campos potenciais, navegação visual e navegação baseada em comandos ou comportamentos pré-definidos anteriormente pelo sistema reativo (MAJDIK et al., 2010). A Figura 14 ilustra o esquema da navegação baseada em comportamentos.

Nesse método não há qualquer tipo de informação armazenada previamente, o que pode ser um problema caso ocorra ações inesperadas durante o trajeto.

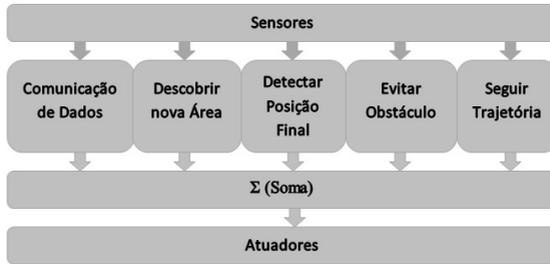


Figura 14 – Navegação baseada em comportamentos

3.4 ARQUITETURAS DE CONTROLE EM ROBÓTICA MÓVEL

Segundo (FREITAS et al., 2013) sistemas robóticos são usados nas mais diversas condições e, em alguns casos, em ambientes com certas restrições. O controle do sistema robótico depende do mapeamento entre a percepção e a ação sobre o ambiente.

As principais arquiteturas de controle de robôs móveis são: Deliberativa, Reativa e Híbrida. As Seções 3.4.1, 3.4.2 e 3.4.3 descrevem as características dessas arquiteturas em detalhes.

3.4.1 Arquitetura Deliberativa

Para (FREITAS et al., 2013) a arquitetura deliberativa realiza suas ações internas de maneira lógica e sequencial, no primeiro momento percebendo o ambiente, depois planejando a ação a ser realizada. Uma tarefa é dividida em sub-tarefas que são executadas uma por vez, a reorganização e a junção das subdivisões caracteriza o objetivo do sistema robótico.

Essa arquitetura consiste em um mecanismo de planejamento de ações, onde um plano prévio de sequência de ações já pode ser estabelecido inicialmente. O modelo deliberativo possui um processo de alto nível de raciocínio e tomada de decisões, e por ser complexo essa arquitetura exige um controle mais sofisticado (WOLF et al., 2009).

Esta arquitetura apresenta resultados positivos se o ambiente for muito dinâmico, já que situações inesperadas forcem o sistema a perceber e planejar novamente para então agir (FREITAS et al., 2013). A Figura 15 ilustra um modelo da arquitetura deliberativa.

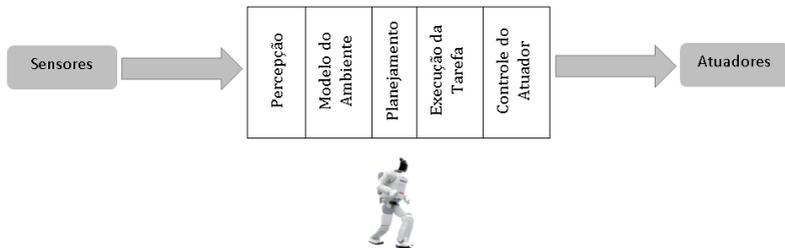


Figura 15 – Arquitetura de controle deliberativa
Adaptado de (FREITAS et al., 2013)

3.4.2 Arquitetura Reativa

Sistemas robóticos podem ser baseados no modelo percepção-ação, no qual suas ações serão relativas aos estímulos recebidos do ambiente. Este conceito é conhecido como arquitetura de software reativa (FREITAS et al., 2013).

Para (WOLF et al., 2009) o controle reativo consiste em um sistema de reação sensorial-motora, que normalmente é mais simples, não necessitando de muitos recursos computacionais. As etapas de controle relacionadas a este tipo de arquitetura são:

- Leitura dos sensores;
- Processamento imediato das informações;
- Geração de comando para os atuadores.

Esse tipo de arquitetura possui um desempenho satisfatório em tarefas como desviar de obstáculos, seguir objeto, seguir uma fonte luminosa (WOLF et al., 2009). A Figura 16 ilustra um exemplo de uma arquitetura reativa.

3.4.3 Arquitetura Híbrida

Sistemas de controle híbridos integram o modelo reativo e o modelo deliberativo. Neste caso, o sistema recebe informações do ambiente, e mantém a organização das ações de maneira lógica e sequencial, fazendo um planejamento prévio das ações tomadas durante a sua execução conforme a ilustração da Figura 17.



Figura 16 – Arquitetura de controle reativa
Adaptado de (FREITAS et al., 2013)

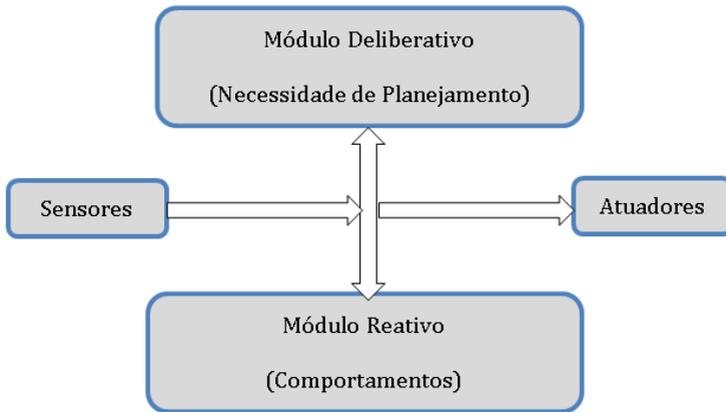


Figura 17 – Arquitetura de controle híbrida
Adaptado de (FREITAS et al., 2013)

Segundo (WOLF et al., 2009) sistemas híbridos apresentam vantagens por combinarem comportamentos obtidos em seus diferentes módulos da arquitetura deliberativa e a arquitetura reativa que podem ser executados em paralelos.

É possível que o controle deliberativo realize o planejamento e a execução de uma trajetória com o uso de um mapa, e ao mesmo tempo a arquitetura reativa, atua para o robô desviar de um obstáculo, e ainda

é possível ter um outro módulo que realiza auto-localização, permitindo ajustes no plano de trajetória através do constante monitoramento da posição atual do robô (WOLF et al., 2009).

3.5 ESTRATÉGIAS DE EXPLORAÇÃO EM AMBIENTES

A maioria de tarefas realizadas por robôs consideram a navegação e o mapeamento como elementos estáticos, um grande desafio é a constante mudança da posição dos robôs no ambiente e as mudanças no próprio ambiente.

Para um local já mapeado que sofre alterações em instantes de tempo incertos, determinadas mudanças nas estruturas locais podem não ser reconhecidas pelo robô instantaneamente, com isso há uma grande chance de o equipamento robótico sofrer erros na percepção do ambiente (SOUZA, 2012). A Figura 18 ilustra como este processo pode ser realizado.

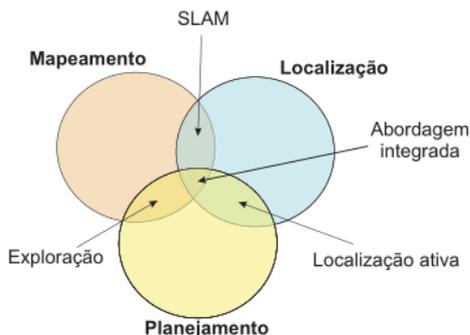


Figura 18 – Processo de determinação de rotas
Extraído de (SOUZA, 2012)

Segundo (SOUZA, 2012), durante o mapeamento, um sistema robótico deve escolher caminhos a serem seguidos. Um robô necessita ter algum conhecimento sobre o mundo, é essencial que saiba para onde ele deverá se mover para adquirir novas informações. Um método utilizado é o monitoramento constante do ambiente, atualizando dados, a partir do mapeamento e localização.

É responsabilidade da estratégia de exploração estabelecer os princípios para que o robô se movimente com o objetivo de fazer a

exploração em todo o ambiente (SOUZA, 2012). A estratégia de exploração admite que a posição do robô é conhecida, e tenta direcioná-lo de maneira eficiente ao longo do percurso.

4 PROJETO E DESENVOLVIMENTO DE UM SISTEMA DE VISÃO ARTIFICIAL PARA UM ROBÔ EXPLORADOR

Este capítulo descreve o sistema de visão artificial desenvolvido. Para um melhor entendimento do capítulo, inicialmente, é descrito o hardware e o software de controle utilizados no robô explorador.

4.1 DESCRIÇÃO DO ROBÔ EXPLORADOR

Um robô móvel tem como objetivo explorar um ambiente, seja ele *indoor* ou *outdoor*, a procura de objeto ou objetos específicos. Este tipo de robô deve ser dotado de um sistema de visão artificial que seja robusto o suficiente para que consiga identificar o objeto ao qual procura. Por exemplo, um robô para auxiliar pessoas com alguma deficiência (BETKE; GIPS; FLEMING, 2002), para auxiliar um motorista na condução de um veículo (STEIN, 2012) ou até mesmo para detectar e desarmar minas terrestres (GADER et al., 1998).

O robô explorador é baseado em uma caminhonete de controle remoto. É dotado de uma câmera do tipo webcam, responsável pela captura das imagens do ambiente e 6 (seis) sensores de ultrassom, sendo 3 (três) na parte dianteira e 3 (três) na parte traseira, que detectam a proximidade do robô a algum objeto presente no ambiente.



Figura 19 – Imagem do robô explorador

A Figura 19 ilustra uma imagem do robô explorador onde é possível perceber o posicionamento da câmera e dos sensores de ul-

trassom.

A navegação do robô é baseada nas informações visuais recebidas pela câmera e também pelas informações provenientes dos sensores de ultrassom. O objetivo da navegação visual é permitir que o robô siga um determinado objeto, desta forma, conforme o objeto se movimenta no ambiente, o robô também se movimentará, mantendo o objeto sempre no alvo, ou seja, no foco da câmera.

A Figura 20 ilustra o modelo cinemático do robô explorador onde é possível observar os deslocamentos, tanto do robô como do servomotor de posicionamento da câmera, nos eixos x e y , bem como os ângulos de movimentação.

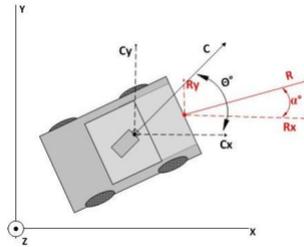


Figura 20 – Modelo cinemático do robô explorador

As Seções 4.1.1 e 4.1.2, descrevem, respectivamente, as arquitetura de hardware e de software do robô explorador.

4.1.1 Arquitetura de Hardware do Robô Explorador

Devido à complexidade do projeto do robô explorador, seu hardware é composto de dois módulos, sendo cada um responsável por uma etapa do controle. O primeiro módulo, onde acontece o processamento da imagem, é chamado de módulo Deliberativo.

O segundo módulo é chamado de módulo Reativo, é responsável pela leitura dos sensores de ultrassom e o controle dos atuadores e efetadores do robô. A Figura 21 ilustra a arquitetura de hardware do robô explorador onde é possível observar os módulos Deliberativo e Reativo.

O módulo deliberativo é composto por uma placa Raspberry Pi (PI, 2013) modelo B, ilustrado na Figura 22, que é considerando um computador de baixo custo, possui um processador da família ARM

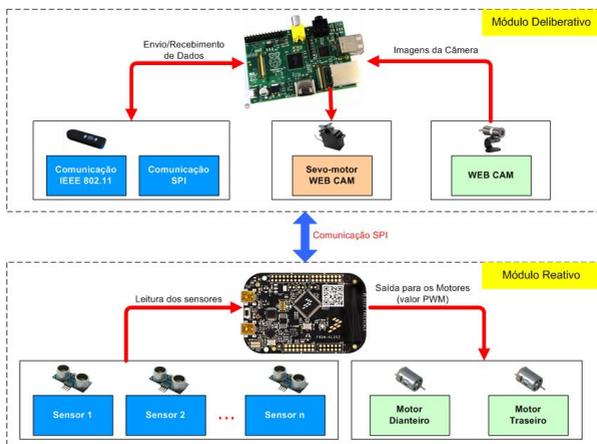


Figura 21 – Arquitetura de hardware do robô explorador

(*Advanced RISC Machine*) 11, de 32 bits com clock de 700 MHz (*Mega Hertz*), 512 MB (*Mega Bytes*) de memória RAM (*Random Access Memory*), interfaces RCA (*Radio Corporation of America*) e HDMI (*High-Definition Multimedia Interface*).



Figura 22 – Imagem do Raspberry Pi modelo B
Extraído de (PI, 2013)

O Raspberry Pi (PI, 2013) também conta com interface de entrada e saída, como Ethernet, uma interface para cartão do tipo SD (*Secure Digital*), onde é possível instalar o sistema operacional, possui ainda comunicação GPIO (*General Purpose Input/Output*), I2C (*Inter-Integrated Circuit*), UART (*Universal Asynchronous Receiver/Transmitter*), alimentação de 5 volts com um consumo de 800 mA (*Miliâmpères*).

O robô conta também com uma placa FRDM-KL25Z (FREESCALE, 2013), ilustrado na Figura 23, com microcontrolador Kinetis

CortexTM-M0+ Série L, presente no módulo reativo.



Figura 23 – Freescale FRDM-KL25Z
Extraído de (FREESCALE, 2013)

A comunicação entre os módulos deliberativo e reativo é realizada via protocolo SPI (*Serial Peripheral Interface*). A Seção 4.1.2 descreve a arquitetura de software dos módulos deliberativo e reativo do robô explorador.

4.1.2 Arquitetura de Software do Robô Explorador

A Figura 24 ilustra a arquitetura de software desenvolvida para o robô explorador. Esta arquitetura, assim como a arquitetura de hardware, é baseada em dois níveis de controle, sendo o primeiro, o nível deliberativo, responsável pelo processamento das imagens provenientes da câmera e o segundo nível, o reativo, responsável pela interpretação dos sinais recebidos pelos sensores de ultrassom e o controle dos motores.

4.1.2.1 Sistema Reativo

O módulo Reativo é gerido pelo sistema operacional MQX Lite (FREESCALE, 2013) ou o FreeRTOS (BARRY, 2009), onde são executadas três tarefas, sendo a primeira o sistema de controle principal deste nível, um sistema neurofuzzy (CHEN et al., 2011), que é baseado em uma rede neural do tipo Perceptron de Múltiplas Camadas com o ajuste de sua saída baseado em um sistema Fuzzy.

O sistema neurofuzzy é responsável por receber as informações provenientes do módulo Deliberativo, bem como informações das tarefas “controle da bateria” e “controle dos sensores e dos atuadores”. A tarefa “controle de bateria” é responsável por monitorar a carga

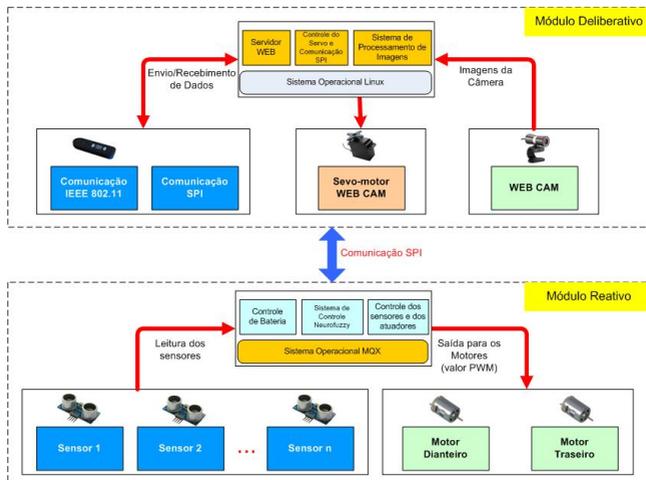


Figura 24 – Arquitetura de software do robô explorador

da bateria do robô, dependendo do nível de carga da bateria algumas funcionalidades podem ser desativas.

Já a tarefa “controle dos sensores e dos atuadores” é responsável por ler os dados provenientes dos sensores de ultrassom e os repassar ao controle neurofuzzy, este por sua vez, irá determinar uma saída para os atuadores que controlam a direção e a velocidade do robô.

4.1.2.2 Sistema Deliberativo

O módulo Deliberativo conta com um sistema operacional Linux, a distribuição Wheezy configurada para o Raspberry Pi (DEBIAN, 2013). Neste módulo são executadas três tarefas, sendo a principal o “sistema de processamento de imagens”, onde são processadas todas as imagens provenientes da câmera.

A segunda tarefa é o “controle do servo e da comunicação SPI” que é responsável por reposicionar o servomotor de acordo com as coordenadas fornecidas pelo sistema de processamento de imagens e também enviar e receber dados via protocolo SPI para/do módulo Reativo, a terceira tarefa é um “servidor WEB” que fica ativo para que um usuário possa enviar comandos remotamente para o robô via rede 802.11 (rede sem fio).

4.2 DESCRIÇÃO DO SISTEMA DE VISÃO ARTIFICIAL

O sistema de visão artificial do robô explorador é baseado em uma webcam que faz a captura das imagens. A câmera está posicionada sobre um servomotor que é responsável pelo reposicionamento da mesma caso o objeto ao qual está focando se desloque no ambiente. A Figura 25 ilustra o processo de captura das imagens pelo robô explorador.

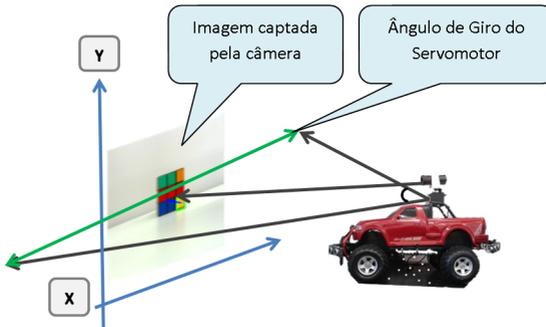


Figura 25 – Processo de captura de uma imagem pelo robô explorador

Para fazer com que o servomotor se posicione em direção ao objeto, é necessário transformar proporcionalmente os dados da imagem capturada pela câmera em relação ao ângulo do servomotor, mantendo assim o foco da câmera no objeto.

O processamento de imagens baseia-se em encontrar as coordenadas x e y em relação ao centro do frame, conforme ilustração da Figura 26. A linha amarela representa o centro do frame no qual é verificado a distância dos pixels em relação ao ponto captado pelo sistema de visão.

A localização do alvo na imagem implica em buscar informações da diferença do centro da coordenada x em relação a posição do objeto, esse valor será negativo ou positivo, dependendo da posição a direita ou a esquerda do frame, observando que os objetos devem possuir um tamanho mínimo e um tamanho máximo como padrão.

Após processar as imagens provenientes da câmera, o sistema de visão artificial calcula o ângulo de ajuste do servomotor a fim de manter o objeto focado no centro do frame. O gráfico ilustrado na Figura 27 representa as possíveis rotações que o servomotor pode executar de

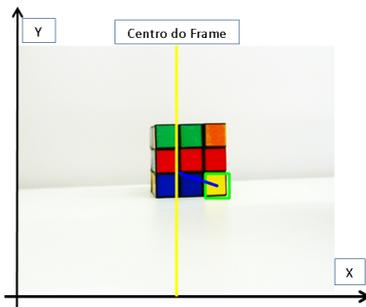


Figura 26 – Busca da diferença entre o centro e a posição do objeto

acordo com os dados recebidos do sistema de visão artificial.



Figura 27 – Projeção de ângulação do servomotor

Vale ressaltar que fatores externos podem comprometer a qualidade do sistema de visão, tal como a luminosidade ou algum defeito no hardware de visão.

O sistema de visão artificial do robô explorador foi desenvolvido em C++, baseado no framework ROS (*Robot Operating System*), e o OpenCV (*Open Source Computer Vision Library*), uma biblioteca de visão computacional. O Anexo A descreve em detalhes o ROS e a biblioteca OpenCV.

5 AVALIAÇÃO DO SISTEMA DE VISÃO ARTIFICIAL

Nesse capítulo são apresentados os resultados de 3 (três) experimentos realizados como o sistema de visão artificial desenvolvido para o robô explorador.

5.1 METODOLOGIA DOS EXPERIMENTOS

Para cada experimento descrito nas Seções 5.2, 5.3 e 5.4 foi utilizado uma câmera USB (*Universal Serial Bus*) da HP modelo HD-4110, que trabalha com uma taxa de 30 (trinta) frames por segundos, resolução de imagem de 13 (treze) megapixel e utiliza um sensor CMOS (*Complementary Metal-Oxide-Semiconductor*).

Para todos os experimentos descritos abaixo é necessário que seja definido o tamanho mínimo e o tamanho máximo de pixels, ou seja, o objeto a ser localizado pelo robô deve ter uma quantidade mínima e máxima de pixels para ser reconhecido pelo sistema de visão artificial.

Para todos os experimentos realizados foi utilizado um frame com a resolução de 172 X 140 pixels, onde 172 representa tamanho horizontal, e 140 representa tamanho vertical, ou seja, 172 representa x na imagem e 140 representa y na imagem dos experimentos, entretanto a resolução pode ser aumentada até a capacidade da câmera.

Os experimentos foram realizados com esse padrão devido ao custo computacional para fazer a varredura dos dados do frame. As técnicas utilizadas nos experimentos foram a binarização, juntamente com padrões matemáticos de cálculo de circunferência e a área de circunferência, e a técnica de cálculo sobre máscaras, ou seja processo orientado a vizinhança e reconhecimento de borda dos objetos.

5.2 EXPERIMENTO I - BUSCA DE UM OBJETO COM UMA COR ESPECÍFICA

O objetivo desse experimento foi fazer com que o robô localizasse um quadrado de cor amarela em um cubo de Rubik. Durante o experimento um quadrado de cor amarela foi colocado em posições diferentes a fim de testar a efetividade do sistema de visão e de localização do alvo pela câmera.

O cubo de Rubik foi utilizado por apresentar uma boa variedade

de cores, a pequenas distâncias, provando a capacidade de distinção das cores pelo sistema de visão. Uma imagem digital basicamente é uma matriz de pixels (*Picture Element*), cada qual contém valores de intensidade das 3 (três) cores básicas: vermelho, verde e azul. Para realizar o reconhecimento de padrões, primeiramente é realizada a binarização da imagem.

A binarização consiste em gerar uma nova matriz, de mesmo tamanho, com os pontos da matriz resultando em **1** caso os valores de RGB (*Red, Green e Blue*) estejam dentro dos limites específicos e **0** caso estejam fora desse intervalo.

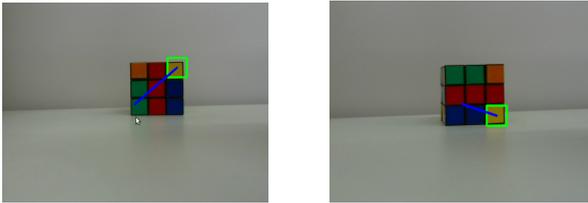


Figura 28 – Captura do quadrado de cor amarela em um cubo de Rubik

A Figura 28 ilustra a saída produzida pelo sistema. Com a posição do quadrado de cor amarela em dois experimentos que foram realizados. A borda de cor verde do quadrado amarelo indica que o sistema de visão detectou o alvo e o está perseguindo, ou seja, caso o objeto se movimente o sistema de visão fará com que a câmera se movimente também visando manter o foco no objeto que está perseguindo.

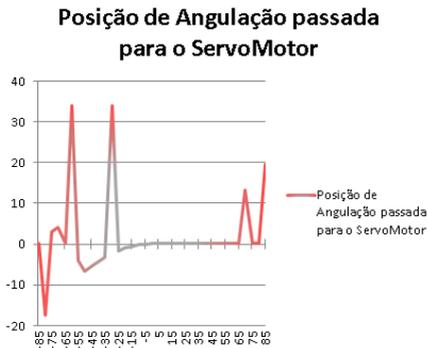


Figura 29 – Gráfico das posições passadas para o servomotor

A linha azul projetada nas figuras representa o vetor do centro da imagem em relação ao centro do objeto detectado. O gráfico da Figura 29 ilustra os dados de angulação determinados pelo sistema de visão para a movimentação do servomotor.

5.3 EXPERIMENTO II - BUSCA DE UM OBJETO COM UM FORMATO ESPECÍFICO

Esse experimento teve como objetivo buscar informações de uma imagem que representam objetos circulares. No processo de captura de um objeto circular é verificado o tamanho dos objetos captados, conforme ilustra a Figura 30, objetos de diferentes tamanhos foram localizados pelo sistema de visão artificial do robô explorador.

A Figura 30 ilustra o resultado da captura dos objetos do sistema de visão artificial do robô explorador.

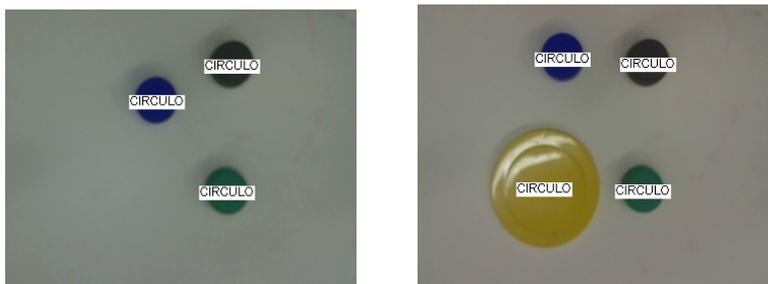


Figura 30 – Captura de objetos circulares

O método utilizado tem como objetivo extrair dados circulares do frame através de padrões matemáticos de cálculo de circunferência e a área de circunferência, juntamente com a técnica de cálculo sobre máscaras, ou seja um processo orientado a vizinhança, e a detecção de bordas dos objetos buscados.

5.4 EXPERIMENTO III - BUSCA DE UM OBJETO COM COR E FORMATO ESPECÍFICOS

O objetivo deste experimento foi buscar um objeto com cor e formato específicos. A Figura 31 ilustra o resultado da operações de busca de um objeto circular de cor azul. Ressalta-se que mesmo ha-

vendo outros objetos de cor azul no ambiente o sistema detectou e marcou somente o círculo de cor azul, ou seja, o objeto alvo, esse experimento foi utilizado a binarização e a operação sobre máscaras nas imagens.



(a) Primeiro exemplo de captura do círculo de cor azul



(b) Segundo exemplo de captura do círculo de cor azul

Figura 31 – Captura de um objeto circular de cor azul

Conforme a Figura 31 o sistema identificou a imagem circular de cor azul no frame, mesmo com diversos objetos de cores e formas específicas. A Figura 32 (a) e (b) ilustram respectivamente, as posições passadas ao servomotor durante o processo de localização do objeto.

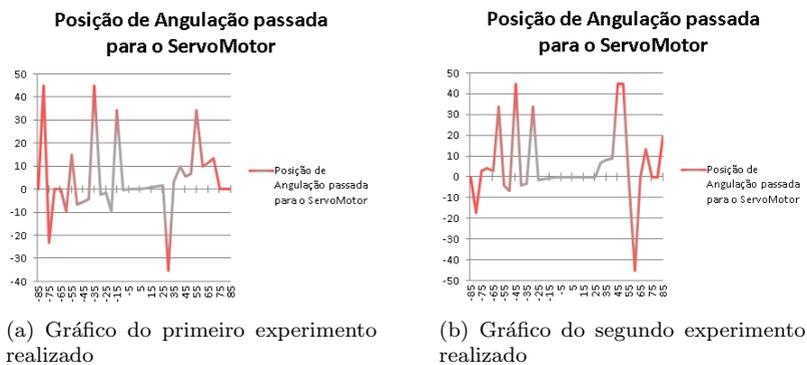


Figura 32 – Gráfico das posições passadas para o servomotor

6 CONSIDERAÇÕES FINAIS E PROPOSTAS PARA TRABALHOS FUTUROS

Este trabalho descreveu um sistema de visão artificial, desenvolvido para um robô explorador de ambientes indoor e outdoor. O robô utilizado possui 6 (seis) sensores de ultrassom e uma câmera do tipo webcam. O hardware do robô foi projetado de forma híbrida sendo composto por dois módulos de controle.

O objetivo do robô explorador é manter o foco sobre um objeto que pode estar em movimento no ambiente. Caso o objeto se desloque, fazendo com que a câmera perca a imagem do objeto, o sistema faz uma varredura no ambiente até localizar novamente o objeto e fazer com que a câmera mantenha este em foco.

Robôs exploradores são utilizados para as mais diversas tarefas, podendo ser aplicados em problemas simples como aspirar o pó de um residência ou problemas mais complexos, como inspecionar um duto de transporte de óleo ou gás natural. A visão é uma importante ferramenta para este tipo de robô, uma vez que permite que este localize e manipule determinados objetos reconhecidos no ambiente.

Sistemas de visão artificial também podem ser aplicados em outras áreas como a automação veicular. Nos dias atuais alguns modelos de veículos possuem sistemas que auxiliam o motorista em atividades como estacionar em uma vaga paralela. Num futuro espera-se que os veículos possam ser dotados de câmeras para identificar as placas sinalizadoras e faixas de uma autoestrada, desta forma, poderão auxiliar o motorista na tomada de decisões, como por exemplo, em uma ultrapassagem.

Os resultados obtidos com o sistema de visão artificial demonstraram que o robô identificou o objeto em diferentes situações e gerou um novo posicionamento para o servomotor de controle da câmera e os motores de controle de direção e velocidade, fazendo com que o robô se posicionasse de maneira correto em direção ao objeto mapeado.

6.1 PROPOSTAS PARA TRABALHOS FUTUROS

Esta Seção apresenta uma lista de propostas para trabalhos futuros que visam melhorar e/ou estender o sistema apresentado neste trabalho.

1. Implementar técnicas de Inteligência Artificial no sistema de pro-

cessamento de imagens do robô visando fazer com que o robô aprenda sobre determinados tipos de objetos;

2. Integrar os módulos reativo e deliberativo e testar o robô em ambientes abertos (*outdoor*) e em ambientes fechados (*indoor*);
3. Utilizar uma outra câmera que permita focar objetos com alta definição a fim de melhorar a qualidade do sistema de visão do robô.

REFERÊNCIAS

- AL-FANDI, M. et al. A real time vision feedback system for automation of a nano-assembly manipulator inside scanning electron microscope. In: *Systems Signals and Devices (SSD), 2010 7th International Multi-Conference on*. [S.l.: s.n.], 2010. p. 1–5.
- ARAUJO, S. A. de. *Casamento de padrões em imagens digitais livre de segmentação e invariante sob transformações de similaridade*. Tese (Doutorado) — USP, 2009.
- ATHERTON, J.; GOODRICH, M. Perception by proxy: Humans helping robots to see in a manipulation task. In: *Human-Robot Interaction (HRI), 2011 6th ACM/IEEE International Conference on*. [S.l.: s.n.], 2011. p. 109–110. ISSN 2167-2121.
- BAHN, W. et al. A motion-information-based vision-tracking system with a stereo camera on mobile robots. In: *Robotics, Automation and Mechatronics (RAM), 2011 IEEE Conference on*. [S.l.: s.n.], 2011. p. 252–257. ISSN 2158-2181.
- BARRY, R. *FreeRTOS reference manual: API functions and configuration options*. [S.l.]: Real Time Engineers Limited, 2009.
- BETKE, M.; GIPS, J.; FLEMING, P. The camera mouse: visual tracking of body features to provide computer access for people with severe disabilities. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, v. 10, n. 1, p. 1–10, 2002. ISSN 1534-4320.
- BRADSKI, G.; KAEHLER, A. *Learning OpenCV: Computer vision with the OpenCV library*. [S.l.]: O’reilly, 2008.
- CABRE, T. et al. Project-based learning example: Controlling an educational robotic arm with computer vision. *Tecnologias del Aprendizaje, IEEE Revista Iberoamericana de*, v. 8, n. 3, p. 135–142, 2013. ISSN 1932-8540.
- CHEN, C. et al. Machine condition prediction based on adaptive neuro-fuzzy and high-order particle filtering. *Industrial Electronics, IEEE Transactions on*, IEEE, v. 58, n. 9, p. 4353–4364, 2011.
- DEBIAN. Debian. 2013. <<http://www.debian.org/>>. Acessado em 30 Setembro 2013.

ERDTMANN, M. J. K. Desenvolvimento de uma plataforma robótica móvel, incluindo sistema embarcado de visão estéreo. v. 1, p. 140, 2009.

FELICIANO, F. F.; SOUZA, I. L. d.; LETA, F. R. Visão computacional aplicada à metrologia dimensional automatizada: considerações sobre sua exatidão. *ENGEVISTA*, v. 7, n. 2, 2010.

FERREIRA, A. de H. *Dicionário Aurélio*. Editora Positivo, 2010. <<http://books.google.com.br/books?id=r4WKuAAACAAJ>>. Acessado em 30 Setembro 2013.

FILHO, H. V. N. O. M. *Processamento Digital de Imagens*. [s.n.], 1999. 331 p. <<http://pessoal.utfpr.edu.br/hvieir/download/pdi99.pdf>>. Acessado em 30 Setembro 2013.

FOLHA. *Folha*. 12 2013. <<http://www1.folha.uol.com.br/fsp/saude/35772-bebe-impresso.shtml>>. Acessado em 30 Setembro 2013.

FREESCALE. Freescale. 2013. <<http://www.freescale.com/>>. Acessado em 30 Setembro 2013.

FREITAS, S. A. G. d. et al. Desenvolvimento de um sistema de controle embarcado para a equipe de futebol de robôs araranguá intruders. 2013.

GADER, P. et al. Landmine detection using fuzzy sets with gpr images. In: *Fuzzy Systems Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*. [S.l.: s.n.], 1998. v. 1, p. 232–236 vol.1. ISSN 1098-7584.

GARAGE, W. *Robot Operating System (ROS)*. 2010. <<http://wiki.ros.org>>. Acessado em 30 Setembro 2013.

GERONIMO, D. et al. Traffic sign recognition for computer vision project-based learning. *Education, IEEE Transactions on*, v. 56, n. 3, p. 364–371, 2013. ISSN 0018-9359.

GOMES, R. B. *Multi-resolução com Fóvea Móvel para Redução e Abstração de Dados em Tempo Real*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE, 2009. <<http://ftp.ufrn.br/pub/biblioteca/ext/bdtd/RafaelBG.pdf>>. Acessado em 30 Setembro 2013.

GONZALEZ, R. C.; WOODS, R. E. *Processamento Digital de Imagens*. [S.l.: s.n.], 2009. 623 p.

HUANG, H.-C. Intelligent motion controller design for four-wheeled omnidirectional mobile robots using hybrid ga-pso algorithm. In: *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*. [S.l.: s.n.], 2011. p. 2267–2272. ISSN 1062-922X.

KLIPP, T. d. S. et al. Proposta de uma arquitetura para alocação de tarefas em grupos de robôs móveis baseada em acordo bizantino. 2013.

MAJDIK, A. et al. Laser and vision based map building techniques for mobile robot navigation. In: *Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on*. [S.l.: s.n.], 2010. v. 1, p. 1–6.

MARENGONI, M.; STRINGHINI, S. Tutorial: Introdução à visão computacional usando opencv. *Revista de Informática Teórica e Aplicada*, v. 16, n. 1, p. 125–160, 2010.

MARTINEZ, A.; FERNÁNDEZ, E. *Learning ROS for Robotics Programming*. [S.l.]: Packt Publishing, 2013. ISBN 1782161449.

MILLER, G.; FELS, S.; OLDRIDGE, S. A conceptual structure for computer vision. In: *Computer and Robot Vision (CRV), 2011 Canadian Conference on*. [S.l.: s.n.], 2011. p. 168–174.

NASA. *National Aeronautics and Space Administration*. 2013. <<http://www.nasa.gov/>>. Acessado em 30 Setembro 2013.

NIKU, S. *Introduction to Robotics*. John Wiley & Sons, 2010. ISBN 9780470604465. <<http://books.google.com.br/books?id=2V4aGvIGt7IC>>. Acessado em 30 Setembro 2013.

OKASAKI, D.; DÜCK, D. C.; ALVES, E. d. S. R. Software de processamento de imagens para um sistema de visão de máquina. Curitiba, 2013.

OPENCV. *Open Source Computer Vision Library*. 2013. <<http://www.opencv.org>>. Acessado em 30 Setembro 2013.

PARK, S.; HASHIMOTO, S. Autonomous mobile robot navigation using passive rfid in indoor environment. *Industrial Electronics, IEEE Transactions on*, v. 56, n. 7, p. 2366–2373, 2009. ISSN 0278-0046.

PEDRINI, H.; SCHWARTZ, W. *Análise de Imagens Digitais: Princípios, Algoritmos e Aplicações*. [S.l.]: Thomson Learning, 2008. ISBN 9788522105953.

PI, R. *Raspberry PI*. 2013. <<http://www.raspberrypi.org/>>. Acessado em 30 de Setembro 2013.

PIRES, S. R.; MEDEIROS, R. B.; ELIAS, S. Qualim: software para treinamento na interpretação de imagens médicas digitais. *Radiol. Bras*, SciELO Brasil, v. 41, n. 6, p. 391–5, 2008.

QIAN, K.; SONG, A. Autonomous navigation for mobile robot based on a sonar ring and its implementation. In: *Instrumentation and Control Technology (ISICT), 2012 8th IEEE International Symposium on*. [S.l.: s.n.], 2012. p. 47–50.

QUILES, M. G.; ROMERO, R. A. F. Um sistema de visão computacional baseado em cores aplicado ao controle de um robô móvel. p. 5, 2004.

RICHA, A. von Wangenheim e Eros Comunello e R. *Introdução a Visão Computacional*. 2013. <<http://www.inf.ufsc.br/visao/>>. Acessado em 30 Setembro 2013.

RICHARDS, J. *Remote Sensing Digital Image Analysis: An Introduction*. Springer, 2012. ISBN 9783642300615. <http://books.google.com.br/books?id=CBn-x1j_Aj8C>. Acessado em 30 de Setembro 2013.

RONG, L. et al. Application research of computer adapted color vision test in chinese medicine inspection skill training. In: *IT in Medicine and Education (ITME), 2011 International Symposium on*. [S.l.: s.n.], 2011. v. 1, p. 597–600.

ROS. *Robot Operating System*. 2013. <<http://www.ros.org>>. Acessado em 30 Setembro 2013.

RUSS, J. *The Image Processing Handbook, Fifth Edition*. Taylor & Francis, 2010. (Image Processing Handbook). ISBN 9780203881095. <<http://books.google.com.br/books?id=Vs2AM2cW11AC>>. Acessado em 30 Setembro 2013.

RUSSEL, S.; P.NORVING. *Inteligência Artificial*. 2004. ed. [S.l.]: 2004, 2004. 1021 p.

SANCHES, V. L. M. Um sistema de localização robótica para ambientes internos baseados em redes neurais. v. 1, p. 107, 2009.

SANTOS, N. Atualização de demonstrador robótico para utilização do "ros". 2013.

SARDIS, E.; VARVARIGOU, T. Industrial workflows recognition by computer vision and ai technologies. In: *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference on*. [S.l.: s.n.], 2010. p. 587–590.

SHAYESTEGAN, M.; MARHABAN, M. Mobile robot safe navigation in unknown environment. In: *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*. [S.l.: s.n.], 2012. p. 44–49.

SICILIANO, B.; SCIavicco, L.; VILLANI, L. *Robotics: Modelling, Planning and Control*. Springer, 2009. (Advanced Textbooks in Control and Signal Processing). ISBN 9781846286414. <<http://books.google.com.br/books?id=jPCAfmE-logC>>. Acessado em 30 Setembro 2013.

SIQUEIRA, M. L. D. Reconhecimento automático de padrões em imagens ecocardiográficas. p. 107, 2010.

SNAKE, R. Robot snake. 2013. <<http://www.snakerobots.com/>>. Acessado em 30 Setembro 2013.

SOUZA, A. A. de S. *Mapeamento Robótico 2, 5-D com Representação em Grade de Ocupação-Elevação*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE, 2012.

SRIVASTAVA, S.; NANDI, G. C. Localization of mobile robots in a network using mobile agents. In: *Computer and Communication Technology (ICCT), 2010 International Conference on*. [S.l.: s.n.], 2010. p. 415–420.

STECKERT, T. Uma arquitetura para coordenacao de sistemas multirroboticos aplicada ao futebol de robos. 2013.

STEIN, F. The challenge of putting vision algorithms into a car. In: *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*. [S.l.: s.n.], 2012. p. 89–94. ISSN 2160-7508.

STEIN, P. S. et al. Sistema integrado de navegação para o robô ambiental híbrido na floresta amazônica. *Simpósio Brasileiro de Sensoriamento Remoto, 13 (SBSR)*, City São José dos Campos, p. 7103–7105, 2007.

THRUN, S. Simultaneous localization and mapping. In: *Robotics and cognitive approaches to spatial mapping*. [S.l.]: Springer, 2008. p. 13–41.

WOLF, D. F. et al. Robótica móvel inteligente: Da simulação às aplicações no mundo real. In: *Mini-Curso: Jornada de Atualização em Informática (JAI), Congresso da SBC*. [S.l.: s.n.], 2009.

XIMENES, C. et al. Predição de parâmetros estruturais de florestas tropicais a partir das técnicas de transformada de fourier e delimitação manual de copa aplicadas em imagens de alta resolução espacial/prediction the parameters of structural tropical forest with the fourier transformed techniques and manual delineation crown application in high-resolution imagery. *Caminhos de Geografia*, v. 11, n. 33, 2010.

ZHENG, X. et al. Contours-matching in building reconstruction from a single image. In: *Computational Intelligence and Security (CIS), 2011 Seventh International Conference on*. [S.l.: s.n.], 2011. p. 1104–1108.

**ANEXO A – Ferramentas Utilizada no Desenvolvimento do
Sistema de Visão Artificial do Robô Explorador**

A.1 ROS (*ROBOT OPERATING SYSTEM*)

O ROS (*Robotic Operation System*) é um framework para desenvolvimento de sistemas de controle robótico, foi criado em 2007 pelo Laboratório de Inteligência Artificial da Universidade de Stanford, atualmente é mantido pelo instituto Willow Garage (ROS, 2013).

Willow Garage é um laboratório/incubadora de projetos de pesquisas na área tecnológica, dedicada ao desenvolvimento de software e hardware para aplicações robóticas. Foi criada no final de 2006 por Scott Hassan, um ex-empregado da Google (GARAGE, 2010).

O ROS é uma ferramenta de código aberto (*Open Source*) e o seu núcleo está licenciado pela licença BSD (*Berkeley Software Distribution License*). Para (SANTOS, 2013) existem atualmente diversos plataformas que auxiliam o desenvolvimento de software para sistemas robóticos, e muitos com a licença de código aberto, entretanto o ROS se destaca pelas várias funcionalidades disponíveis.

Vários serviços são oferecidos na plataforma de desenvolvimento ROS, incluindo abstração de hardware, controle de dispositivos de baixo nível, transmissão de mensagens, gerenciamento de pacotes, ferramentas de bibliotecas adicionais (ROS, 2013). A Figura 33 ilustra o contexto do funcionamento do ROS.

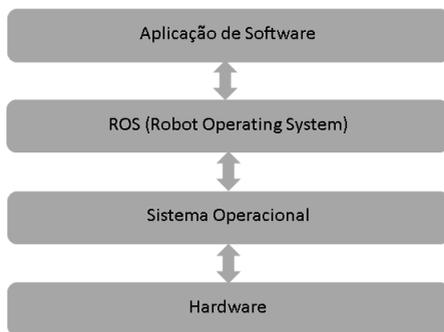


Figura 33 – Interface do ROS com o hardware.

A.1.1 Objetivos do ROS

Segundo (ROS, 2013) o ROS é um sistema federado de repositórios de código que possibilita a colaboração e a distribuição para o público em geral, possuindo uma estrutura distribuída de processos. As principais características do ROS são:

- Permitir o desenvolvimento de aplicações robóticas de maneira mais eficiente;
- Incorporar outros frameworks de sistema robóticos como o Orocos (*Open Robot Control Software Project*);
- Permitir a padronização de novas bibliotecas de maneira funcional;
- Ser compatível com várias linguagens de programação;
- Facilitar a depuração e testes;
- Permitir o desenvolvimento de sistemas complexos de controle de robôs.

Há muitos tipos diferentes de robôs com diferentes necessidades, onde cada desenvolvedor pode criar uma distribuição do ROS, assim é possível que cada desenvolvedor possa utilizar, de acordo com as suas necessidades, vários métodos e técnicas empregadas em sistemas de controle para robôs (ROS, 2013).

A.1.2 Ferramentas do ROS

O ROS oferece diversas ferramentas e funcionalidades para o área de robótica, que auxiliam no desenvolvimento de sistemas de controle robótico (SANTOS, 2013). A Tabela 4 lista algumas funcionalidades do ROS.

Tabela 4 – Algumas funcionalidades do ROS

Funcionalidades	Serviços	Linguagem C++	Linguagem Python
ROS(Sytem)	ROS	roscpp	rospy
Tipos de dados básicos	common_msgs	common_msgs	common_msgs
Processamento de Imagem		vision_opencv, image_common	vision_opencv
Planejamento de Movimentações	ompl, chomp-motion	action_lib, move_arm	actionlib, move_arm

Adaptado de (SANTOS, 2013)

Além das funcionalidades descritas na Tabela 4, o ROS disponibiliza ferramentas de simulação, tais como:

- RViz: permite combinar modelos de robôs com dados de sensores.
- Rxttools: um conjunto de ferramentas para visualização e análise de dados do sistema.
- Rxgraph: permite a visualização de dados e mensagens do sistema.
- Rxbag: permite criar um histórico das operações de dados do ROS para ser analisado posteriormente.

A Figura 34 ilustra a interface gráfica do RViz no ROS, que fornece dados de sensores para uma simulação do robô no ambiente.

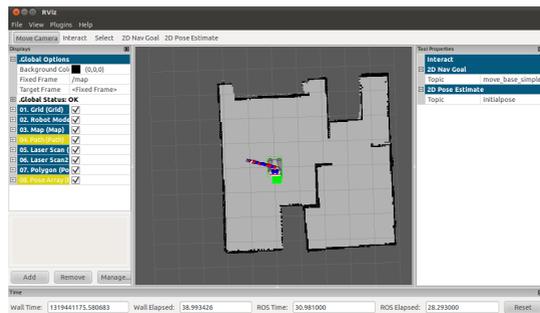


Figura 34 – Interface do RViz
Extraído de (ROS, 2013)

A.1.3 Funcionamento do ROS

O ROS é baseado em eventos e a comunicação entre os componentes do sistema é baseado no processo de publicar/escrever. Os nós são criados no sistema, geralmente, via console e gerenciados pelo servidor mestre do ROS (*roscore*) (MARTINEZ; FERNÁNDEZ, 2013).

O ROS é baseado no servidor principal o *roscore* que é responsável por iniciar as dependências de bibliotecas e o sistema de comunicação entre os nós do sistema (ROS, 2013).

A.2 OPENCV (*OPEN SOURCE COMPUTER VISION LIBRARY*)

O OpenCV (*Open Source Computer Vision Library*) é uma biblioteca para o processamento digital de imagens, compatível com diferentes tipos de sistemas operacionais. Foi desenvolvida pela Intel com a intenção de ser usada para fins acadêmicos e comerciais e surgiu a partir da necessidade de se desenvolver sistemas de visão artificial (BRADSKI; KAEHLER, 2008).

Além de ser multiplataforma o OpenCV também suporta o uso de diversas linguagens de programação. É utilizado em diversas áreas como, identificação, reconhecimento, gravação de vídeos e imagens, robótica, reconstrução 3D, realidade virtual aumentada e misturada, interação humano-computador em tempo real.

Segundo (MARENGONI; STRINGHINI, 2010) o OpenCV pode ser dividido em grupos funcionais como processamento de imagens, análise estrutural, análise de movimento e rastreamento de objetos.

A.2.1 Características do OpenCV

A biblioteca OpenCV possui 4 (quatro) módulos principais, que são:

- Cv: tem as principais funções do OpenCV.
- Cvaux: são funções experimentais da biblioteca.
- Cxcv: suporte as estrutura de dados e funções de álgebra linear.
- Highgui: funções para o desenvolvimento de interface gráfica.

Os sistemas de visão computacional, muitas vezes, necessitam de uma etapa de pré processamento envolvendo imagens. Algumas imagens precisam ser convertidas para um determinado formato ou tamanho e precisam ainda ser filtradas para que sejam removidos os ruídos provenientes do processo de aquisição da imagem (MARENGONI; STRINGHINI, 2010).

O OpenCV possui funções para a detecção de ruídos, filtros, conversão de formato e tamanho de imagens, extração de informações, como histogramas, etc (MARENGONI; STRINGHINI, 2010). A Figura 35 ilustra um exemplo da extração de dados do OpenCV.



(a) Imagem Original



(b) Imagem depois
do reconhecimento
das características

Figura 35 – Reconhecimento de características com o OpenCV
Extraído de (OPENCV, 2013)