Navegação autônoma de robôs móveis e detecção de intrusos em ambientes internos utilizando sensores 2D e 3D

Diogo Santos Ortiz Correa

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura:_____

Navegação autônoma de robôs móveis e detecção de intrusos em ambientes internos utilizando sensores 2D e 3D

Diogo Santos Ortiz Correa

Orientador: **Prof. Dr. Fernando Santos Osório**

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências - Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

USP – São Carlos
Agosto de 2013

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

S787n Santos Ortiz Correa, Diogo
Navegação autônoma de robôs móveis e detecção de intrusos em ambientes internos utilizando sensores 2D e 3D / Diogo Santos Ortiz Correa; orientador Fernando Santos Osório. -- São Carlos, 2013.
77 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2013.

1. Robôs móveis autônomos. 2. Robôs de vigilância. 3. Calibração de sensores. 4. Detecção de pessoas. I. Santos Osório, Fernando, orient. II. Título.

Aos meus familiares.

Agradecimentos

Aos que contribuíram diretamente e indiretamente, seja por contato virtual ou interpessoal.
Em especial agradeço
A Deus pela força,
Aos familiares pelo carinho,
Aos amigos e colegas do laboratório pelas ótimas experiências vividas,
Aos professores pelos ensinamentos, e
Ao meu orientador prof. Osório, que é um pai acadêmico e tem paciência infinita comigo.
Obrigado.

Resumo

Os robôs móveis e de serviço vêm assumindo um papel cada vez mais amplo e importante junto à sociedade moderna. Um tipo importante de robô móvel autônomo são os robôs voltados para a vigilância e segurança em ambientes internos (*indoor*). Estes robôs móveis de vigilância permitem a execução de tarefas repetitivas de monitoramento de ambientes, as quais podem inclusive apresentar riscos à integridade física das pessoas, podendo assim ser executadas de modo autônomo e seguro pelo robô. Este trabalho teve por objetivo o desenvolvimento dos principais módulos que compõem a arquitetura de um sistema robótico de vigilância, que incluem notadamente: (i) a aplicação de sensores com percepção 3D (Kinect) e térmica (Câmera FLIR), de relativo baixo custo, junto a este sistema robótico; (ii) a detecção de intrusos (pessoas) através do uso conjunto dos sensores 3D e térmico; (iii) a navegação de robôs móveis autônomos com detecção e desvio de obstáculos, para a execução de tarefas de monitoramento e vigilância de ambientes internos; (iv) a identificação e reconhecimento de elementos do ambiente que permitem ao robô realizar uma navegação baseada em mapas topológicos. Foram utilizados métodos de visão computacional, processamento de imagens e inteligência computacional para a realização das tarefas de vigilância. O sensor de distância Kinect foi utilizado na percepção do sistema robótico, permitindo a navegação, desvio de obstáculos, e a identificação da posição do robô em relação a um mapa topológico utilizado. Para a tarefa de detecção de pessoas no ambiente foram utilizados os sensores Kinect e câmera térmica FLIR, integrando os dados fornecidos por ambos sensores, e assim, permitindo obter uma melhor percepção do ambiente e também permitindo uma maior confiabilidade na detecção de pessoas. Como principal resultado deste trabalho foi desenvolvido um

sistema, capaz de navegar com o uso de um mapa topológico global, capaz de se deslocar em um ambiente interno evitando colisões, e capaz de detectar a presença de seres humanos (intrusos) no ambiente. O sistema proposto foi testado em situações reais com o uso de um robô móvel Pioneer P3AT equipado com os sensores Kinect e com uma Câmera FLIR, realizando as tarefas de navegação definidas com sucesso. Outras funcionalidades foram implementadas, como o acompanhamento da pessoa (*follow me*) e o reconhecimento de comandos gestuais, onde a integração destes módulos com o sistema desenvolvido constituem-se de trabalhos futuros propostos.

Abstract

Mobile robots and service robots are increasing their applications and importance in our modern society. An important type of autonomous mobile robot application is indoor monitoring and surveillance tasks. The adoption of mobile robots for indoor surveillance tasks allows the execution of repetitive environment patrolling, which may even pose risks to the physical integrity of persons. Thus these activities can be autonomously and safely performed by security robots. This work aimed at the development of key modules and components that integrates the general architecture of a surveillance robotic system, including: (i) the development and application of a 3D perception sensor (Kinect) and a thermal sensor (FLIR camera), representing a relatively low-cost solution for mobile robot platforms; (ii) the intruder detection (people) in the environment, through the joint use of 3D and thermal sensors; (iii) the autonomous navigation of mobile robots within obstacle detection and avoidance, performing the monitoring and surveillance tasks of indoor environments; (iv) the identification and recognition of environmental features that allow the robot to perform a navigation based on topological maps. We used methods from Computer Vision, Image Processing and Computational Intelligence to carry out the implementation of the mobile robot surveillance modules. The proximity and distance measurement sensor adopted in the robotic perception system was the Kinect, allowing navigation, obstacle avoidance, and identifying key positions of the robot with respect to a topological map. For the intruder detection task we used a Kinect sensor together with a FLIR thermal camera, integrating the data obtained from both sensors, and thus allowing a better understanding of the environment, and also allowing a greater reliability in people detection. As a main result of this work, it has been

developed a system capable of navigating using a global topological map, capable of moving itself autonomously into an indoor environment avoiding collisions, and capable of detect the presence of humans (intruders) into the environment. The proposed system has been tested in real situations with the use of a Pioneer P3AT mobile robot equipped with Kinect and FLIR camera sensors, performing successfully the defined navigation tasks. Other features have also been implemented, such as following a person and recognizing gestures, proposed as future works to be integrated into the developed system.

Sumário

Agradecimentos **i**

Resumo **iii**

Abstract **v**

Lista de Figuras **ix**

1 Introdução **1**
1.1 Motivação 4
1.2 Objetivos 6
1.3 Estrutura da Dissertação 6

2 Robôs Móveis Autônomos **9**
2.1 Mapeamento 10
2.2 Localização 13
2.3 Navegação 14
2.4 Sensores 15
2.4.1 Sensores Bidimensionais 15
2.4.2 Sensores Tridimensionais 18
2.5 Ferramentas 21
2.5.1 OpenCV 21
2.5.2 Player/Stage e Gazebo 23
2.5.3 PrimeSense OpenNI/NITE 24
2.5.4 ROS 25
2.5.5 JavaNNS 26
2.6 Considerações 28

3 Trabalhos Relacionados **31**
3.1 Robôs de Vigilância 32
3.2 Navegação Topológica 33
3.3 Detecção de Humanos 34
3.4 Detecção de Regiões Navegáveis 36
3.5 Considerações 38

4 Desenvolvimento e Resultados **41**
4.1 Detecção de Pessoas no Ambiente 43
4.1.1 Calibração de Sensores 44
4.1.2 Pré-processamento e segmentação das imagens 47
4.1.3 Processamento 49
4.1.4 Avaliação dos resultados da detecção de pessoas 56
4.2 Navegação Reativa com Detecção e Desvio de Obstáculos 58
4.3 Detecção de Estados e Transições junto ao Mapa Topológico 61
4.4 Trabalhos Complementares 64
4.4.1 Módulo *Follow Me* 64
4.4.2 Teleoperação com Visualização das Imagens no Cliente 66
4.4.3 Módulo de Reconhecimento de Gestos Utilizando Redes Neurais . . . 67
4.5 Considerações 68

5 Conclusão **69**
5.1 Trabalhos Futuros 70

Referências Bibliográficas **71**

A Publicações Decorrentes Desse Trabalho **77**

Lista de Figuras

1.1 Roomba – Robô aspirador de pó. Fonte: http://irobot.com/. 2
1.2 Spirit – Robô de exploração do planeta Marte. Fonte: NASA (2012). 2
1.3 Veículo autônomo desenvolvido pela Universidade de Stanford . Fonte: http://cs.stanford.edu/group/roadrunner/ 2
1.4 CaRINA – Carro Robótico Inteligente para Navegação Autônoma. Fonte: http://lrm.icmc.usp.br/carina/. 3
1.5 Robô móvel utilizado no Laboratório de Robótica Móvel da Universidade de São Paulo. Fonte: propriedade do LRM. 5

2.1 Robô Móvel Autônomo utilizado junto ao LRM para fins acadêmicos. Este robô possui um sensor laser e uma câmera de vídeo. Fonte: http://www.lrm.icmc.usp.br/ 10
2.2 Exemplo de mapa topológico à esquerda e de um mapa métrico à direita. Fonte: Thrun (2003). 11
2.3 Exemplo de mapa topológico de uma residência (Sales, 2012a). 12
2.4 À esquerda a lente acoplada ao chip da câmera. À direita o sensor CMOS da câmera. Fonte: http://en.wikipedia.org/wiki/File:Sweex_USB_webcam_PCB_with_without_lens_close_up.jpg 16
2.5 Câmera térmica FLIR PathFindIR à esquerda e FLIR TAU 640 à direita. Fonte: FLIR (2011a). 17
2.6 Visualização noturna de uma câmera de vídeo comum (à esquerda) e de uma câmera termal (à direita). Fonte: FLIR (2011a). 18
2.7 Um mapa de profundidade gerado pelo sensor Kinect. Cores escuras representam menor distância e cores claras representam maior distância da cena até o sensor. 18

2.8 Uma nuvem de pontos. Fonte: http://docs.pointclouds.org/trunk/group__registration.html 19
2.9 Sensor Kinect adotado junto ao XBOX 360 da Microsoft. Fonte: http://www.xboxplus.net/2010/06/14/projeto-natal-e-kinect-for-xbox-360/ 19
2.10 Imagens geradas pelo Kinect. À esquerda uma imagem gerada pela câmera de profundidade do Kinect, onde as cores quentes (avermelhadas) são de elementos mais próximos, e cores frias (azuladas) são de elementos mais afastados. À direita uma imagem gerada pela câmera de vídeo do Kinect. 20
2.11 Luz estruturada. À esquerda uma projeção completa da luz estruturada gerada pelo Kinect. À direita a projeção da luz estruturada em um objeto. Fonte: http://goo.gl/6Hs0sj 21
2.12 Mapa de profundidade gerado pelo Kinect: (a) Imagem original do mapa de profundidade com ruído e (b) Imagem do mapa de profundidade após a remoção de ruído. 22
2.13 Arquitetura do Player. Em (a) é apresentada a abstração de *hardware* e em (b) o modelo cliente/servidor. 24
2.14 (a) Interface gráfica do Stage exibindo alguns obstáculos, robôs e a atuação de um sensor laser e (b) Ambiente de simulação de robôs Gazebo. Adaptados de (Hata, 2010). 24
2.15 Rastreamento de um humano utilizando o módulo NITE e o sensor Kinect. . . . 25
2.16 Postura utilizada para a calibração do rastreador de humanos do OpenNI/NITE utilizando o sensor Kinect. Fonte: http://www.ros.org/wiki/openni_tracker. 26
2.17 Visualizador *rviz* do ROS. 26
2.18 Interface gráfica do JavaNNS. Na parte superior esquerda é apresentada a estrutura de uma rede neural, do lado esquerdo inferior a tela de configuração, onde podem ser alterados vários parâmetros. Do lado direito superior é mostrada a projeção de saída da classificação e do lado direito inferior o gráfico de erro. Fonte: http://www.ra.cs.uni-tuebingen.de/software/JavaNNS. 27

3.1 Robôs de segurança. Em (a) e (c) dois robôs comerciais, o Rovio e o Aerion e em (b) um robô de pesquisa, o SRV. Fontes: (a) Rovio (2012), (b) http://www.surveyor.com/SRV_info.html e (c) Aeryon (2012) 32
3.2 Simulação da navegação topológica. Fonte: (Sales *et al.*, 2011). 33
3.3 Navegação topológica real. Fonte: (Sales, 2012a). 33
3.4 Segmentação da imagem utilizando a câmera térmica À direita imagem orginal e à esquerda imagem segmentada. Fonte: (Nin, 2011). 35
3.5 Detecção de pessoas utilizando o Kinect: (a) Detecção de seres humanos utilizando a nuvem de pontos 4D (x, y, z, cor) gerada pelo Kinect (Loong, 2011). (b) Detecção de seres humanos utilizando a nuvem de pontos 4D gerada pelo Kinect. Imagem adaptada de http://www.youtube.com/watch?v=x--xlKWBTAE (Bonvin, 2010). 35

3.6 Mapa de navegabilidade obtido a partir de imagens com o uso de classificadores neurais – LRM (ICMC-USP). Em cada par de fotos as regiões em vermelho são classificadas como navegável na foto à esquerda e à direita o branco é navegável e o preto não navegável. Fonte: (Shinzato, 2010). 37
3.7 Quatro imagens correspondentes à classificação de terrenos quanto à navegabilidade. As duas imagens à esquerda correspondem à classificação utilizando redes neurais e as outras utilizando SVM. As regiões verdes na imagem representam as regiões navegáveis no mundo real, as regiões azuis representam as parcialmente navegáveis e as vermelhas as não navegáveis – LRM (ICMC-USP). Fonte: (Hata, 2010). 37
3.8 Desvio de obstáculos utilizando o algoritmo VFH para o esterçamento e o método RANSAC para a detecção das regiões navegáveis, as quais são representadas em verde. Fonte: (Mendes e Wolf, 2011). 38

4.1 Proposta do sistema de vigilância. Nesta representação são apresentados os módulos do sistema e outros módulos que poderão ser agregados ao sistema. . . 42
4.2 Subsistemas do sistema de vigilância. 43
4.3 Processo de detecção de intrusos utilizando os sensores Kinect e FLIR. 44
4.4 Vista superior do campo de visão horizontal dos sensores. Vermelho representa campo de visão da térmica e azul campo de visão do Kinect. A bola laranja indica a posição dos sensores. 45
4.5 Disposição dos sensores Kinect e térmica sobre a plataforma robótica. A câmera térmica FLIR TAU 640 está posicionada sobre o Kinect. 45
4.6 Imagens utilizadas na calibração horizontal dos dois sensores. Imagens dos sensores (a) Kinect e (b) câmera térmica utilizadas para a calibração horizontal. 46
4.7 Mapa de profundidade gerado pelo Kinect: (a) Imagem original com ruído e (b) Imagem após a remoção de ruído. 47
4.8 Imagem do sensor Kinect: (a) Mapa de profundidade tons de cinza, (b) Segmentação extraindo apenas a informação que representa o humano na cena. . . 48
4.9 Imagem da câmera térmica: (a) Original em tons de cinza, (b) Com pré-processamento utilizando o filtro de limiarização comum. 48
4.10 Exemplos do uso do histograma. Imagens (a) da térmica segmentada, (b) do histograma, (c) do Kinect segmentada, e (d) do histograma. 50
4.11 Casos de histogramas de representação de humanos na imagem. 51
4.12 Resultado sem humanos. Imagens (a) da térmica original, (b) da térmica segmentada, (c) do Kinect original, e (d) do Kinect segmentada. 52
4.13 Resultado com humano dentro do campo de visão de ambos os sensores. Imagens (a) da FLIR original, (b) da FLIR segmentada, (c) do histograma do ser humano na imagem segmentada da FLIR representando com uma bola vermelha a porção horizontal da imagem que se encontra o humano, (d) do Kinect original, (e) do Kinect segmentada, e (f) do histograma do ser humano na imagem segmentada do Kinect representando com uma bola vermelha a porção horizontal da imagem que se encontra o humano. 53

4.14 Resultado com humano apenas no campo de visão da FLIR. Imagens (a) da FLIR original, (b) da FLIR segmentada, (c) do histograma do ser humano na imagem segmentada da FLIR representando a porção horizontal que se encontra o humano na imagem térmica, (d) do Kinect original, e (e) do Kinect segmentada e sem humanos. 54
4.15 Resultado com humano visualizado apenas pelo Kinect. Imagens (a) da FLIR original, (b) da FLIR segmentada e sem humanos, (c) do Kinect original, (d) do Kinect segmentada, e (e) do histograma do ser humano na imagem segmentada do kinect, representando com a bola vermelha a porção que se encontra o humano na imagem. 55
4.16 Matriz de confusão da classificação da imagem do Kinect. 56
4.17 Taxa de acertos e erros da classificação da imagem do Kinect. 56
4.18 Matriz de confusão da classificação da imagem da câmera termal. 57
4.19 Taxa de acertos e erros da classificação da imagem da câmera termal. 57
4.20 Matriz de confusão da classificação da imagem do Kinect e da câmera termal. . 57
4.21 Taxa de acertos e erros da classificação da imagem do Kinect e da câmera termal. 57
4.22 Situações no mapa de profundidade do sistema reativo. As porções de imagem de cor mais escura representam obstáculos, que podem ser à frente, à direita e/ou à esquerda. 59
4.23 Sistema reativo de navegação em execução. Os valores mínimo e máximo representam, respectivamente, o valor mínimo e máximo de um pixel (distância) nas porções de imagem frente, esquerda e direita. Vídeo disponível em: `http://www.youtube.com/watch?v=SmqeO71hIfc` 60
4.24 Situações definidas de ambientes internos. 61
4.25 Em (a) imagem do sensor Kinect com destaque em vermelho para a região de 80 linhas utilizadas para o processamento da rede neural, e em (b) a topologia da rede neural artificial. 62
4.26 resultado de uma classificação. 63
4.27 Média do desempenho para os *5-folds* do treino de diferentes configurações de topologias de rede. 63
4.28 Matrizes de confusão representando a resposta do teste de validação da rede 632-316-8 para os *5-folds*. 63
4.29 Sistema follow-me em execução. Vídeo disponível em `http://www.youtube.com/watch?v=BwDHbBvVAUs` 65
4.30 Rastreamento do humano e em vermelho regiões utilizadas para o cálculo da distância e direção do deslocamento do robô. 65
4.31 Modelo do sistema de teleoperação. Um roteador é utilizado para fazer a comunidação sem fio entre o cliente (computador com *joystick*) e o servidor (computador conectado ao robô). Os sensores utilizados são o Kinect e a câmera térmica. 66
4.32 Sistema de reconhecimento de gestos. 67
4.33 Sistema de reconhecimento de gestos em execução. 68

CAPÍTULO

1

Introdução

Um dos principais objetivos da robótica é tornar a vida do ser humano mais confortável aumentando sua qualidade e propondo soluções para diversas tarefas do dia-a-dia, sejam elas domésticas, educacionais, empresariais, industriais ou militares. Os primeiros robôs eram fixos e constituídos basicamente de braços manipuladores (robótica industrial). Numa segunda fase de desenvolvimento começaram a surgir os robôs móveis, os quais possuem mecanismos que lhe permitem se deslocar no ambiente, seja por terra, pelo ar ou pela água. Um exemplo de robô criado para ajudar em uma tarefa doméstica é o *Roomba* (ver Figura 1.1), um robô aspirador de pó que se locomove no ambiente promovendo a limpeza de forma completamente autônoma. Outro exemplo de robô móvel é o *Spirit*, um robô de uso científico para a exploração do planeta Marte (NASA, 2012) (ver Figura 1.2).

Nas Figuras 1.1 e 1.2 são apresentados um robô de uso interno *indoor* e outro de uso externo *outdoor*, respectivamente. Um outro exemplo de pesquisa na robótica móvel externa são os veículos autônomos, onde o maior motivador desta pesquisa é o aumento da segurança no trânsito, evitando acidentes e melhorando a qualidade de vida das pessoas. Uma série de competições promovidas pela DARPA[1] impulsionou ainda mais a pesquisa de veículos autônomos. Na Figura 1.3 é apresentado o carro autônomo desenvolvido pela Universidade de Stanford[2]

[1]DARPA – Defense Advanced Research Projects Agency (Departamento de Defesa Americano) – Web: `http://www.darpa.mil/`

[2]Universidade de Stanford – Web: `http://www.stanford.edu`

Figura 1.1: Roomba – Robô aspirador de pó. Fonte: `http://irobot.com/`.

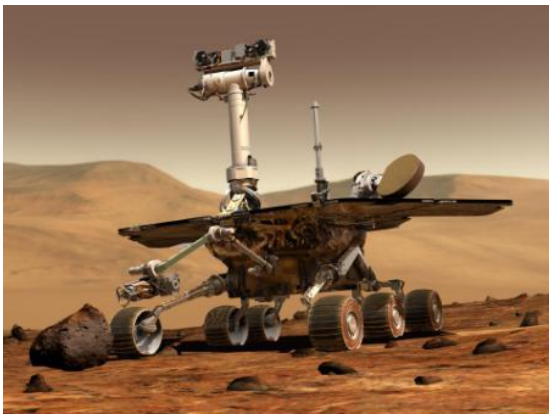


Figura 1.2: Spirit – Robô de exploração do planeta Marte. Fonte: NASA (2012).

para participar da competição DARPA Urban Challenge[3], na qual vários veículos deveriam, de forma completamente autônoma, navegar em um ambiente urbano realizando tarefas como estacionamento, ultrapassagem e manobras em cruzamentos.

Figura 1.3: Veículo autônomo desenvolvido pela Universidade de Stanford . Fonte: `http://cs.stanford.edu/group/roadrunner/`

Na Figura 1.4 é apresentado um dos carros autônomos do LRM/ICMC-USP[4] (Laboratório de Robótica Móvel/Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo). Este último é um exemplo de que o Brasil tem investido também na pesquisa de robôs móveis autônomos, neste caso se tratando de veículos autônomos (considerando também que

[3]DARPA Urban Challenge – Competição de veículos autônomos realizada em 2007 no estado da Califórnia (Estados Unidos da América) – Web: `http://archive.darpa.mil/grandchallenge/`

[4]LRM – Laboratório de Robótica Móvel - Web: `http://www.lrm.icmc.usp.br`

há outros projetos de veículos autônomos nesta e em outras universidades nacionais). Há vários projetos de iniciação científica, mestrado e doutorado deste laboratório envolvidos na automação dos veículos autônomos. Estes trabalhos estão envolvidos com o projeto CaRINA (Carro Robótico Inteligente para Navegação Autônoma), uma iniciativa do INCT-SEC[5], um instituto focado em "aplicações críticas" e com foco no desenvolvimento de robôs móveis terrestres, aéreos e aquáticos. Maiores detalhes sobre este instituto e suas atividades serão apresentados na seção 1.1.

Figura 1.4: CaRINA – Carro Robótico Inteligente para Navegação Autônoma. Fonte: `http://lrm.icmc.usp.br/carina/`.

Conforme apresentado acima, o INCT-SEC vem apoiando iniciativas na área de robótica autônoma, como é o caso do projeto do Veículo Autônomo e Inteligente - CaRINA. Por outro lado, o INCT-SEC também vem incentivando iniciativas na área de robótica móvel visando aplicações na área de segurança em ambientes *indoor*. É neste contexto que se insere o presente projeto de mestrado. Este trabalho teve o foco na robótica móvel autônoma *indoor*, na qual um robô deve operar de forma completamente ou parcialmente autônoma. O sistema desenvolvido aqui é composto por um robô vigilante voltado para o uso em um ambiente fechado, como, por exemplo, um depósito de materiais. O ambiente é caracterizado por salas e corredores e pela presença de diversos obstáculos estáticos (mobília, materiais, mercadorias). O robô navega pelo ambiente, sendo capaz de evitar colisões e danos, seja ao robô ou seja aos elementos do próprio ambiente. A tarefa de vigilância inclui a detecção de pessoas (intrusos).

[5]INCT-SEC – Instituto Nacional de C&T em Sistemas Embarcados Críticos - Web: `http://www.inct-sec.org/`

Uma vez que os robôs de vigilância podem ser encaixados na categoria de robótica móvel autônoma então, do ponto de vista da robótica, três principais questões devem ser abordadas para o desenvolvimento destes robôs: (i) o mapeamento de ambientes; (ii) a localização; e (iii) a navegação do robô. Portanto os robôs devem ser capazes de conhecer uma representação ou mapa do ambiente (ou até mesmo criar um mapa), a fim de poder se localizar (planejando sua trajetória) e assim realizar a navegação, desviando dos obstáculos e identificando situações anômalas (p.ex. intrusos).

Em relação a estas três funcionalidades (mapeamento, localização e navegação), neste trabalho foi abordado o uso de mapas topológicos, onde a construção e uso destes mapas considerou o trabalho anterior de outro membro do Laboratório LRM, Daniel Sales (Sales, 2012a), que também atuou junto aos projetos do INCT-SEC relativos aos robôs de vigilância. Este trabalho permitiu implementar um mecanismo de navegação topológica, resultando na publicação de um trabalho conjunto sobre este tema (Sales, 2012b). Em relação a localização, assume-se que a própria navegação topológica permite fornecer um controle e manutenção da localização, e por outro lado, os trabalhos desenvolvidos também junto ao Laboratório LRM por Leandro Couto (Couto, 2012), sobre localização baseada em imagens, e por Gustavo Pessin (Pessin, 2013), sobre localização baseada no uso de redes sem fio (Wi-Fi), são considerados como alternativas e trabalhos complementares referentes a esta questão. Por fim, a navegação com desvio de obstáculos e a detecção de intrusos são temas centrais abordados junto a esta dissertação, e que serão detalhados nos capítulos seguintes.

Nas duas próximas seções serão detalhados as motivações (Seção 1.1) e os objetivos (Seção 1.2) deste trabalho. A última seção (1.3) apresenta os próximos capítulos desta dissertação.

1.1 Motivação

Este projeto de mestrado se insere no contexto de pesquisas que já vem sendo desenvolvidas no LRM e faz parte de um projeto maior gerido pelo INCT-SEC (INCT-SEC, 2012) o qual é financiado e apoiado pelos órgãos FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo), MCT (Ministério da Ciência e Tecnologia), CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) e Capes (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior). O INCT-SEC possui atualmente 4 grupos de trabalho principais, onde os trabalhos de dois destes grupos vêm sendo fortemente desenvolvidos junto ao LRM, a saber, o GT 1 – "Robôs Táticos para Ambientes Internos" (ver Figura 1.5) – coordenado pelo prof. Osório e GT 2 – "Veículos Terrestres Autônomos" – coordenado pelo prof. Denis Fernando Wolf do

LRM, onde ambos envolvem o desenvolvimento de pesquisas sobre o controle e navegação de robôs e veículos autônomos (ver Figura 1.4).

O GT 1 do INCT-SEC visa desenvolver robôs de serviço, teleoperados e/ou autônomos, para o monitoramento e vigilância de ambientes fechados. O GT 2, por sua vez, visa o desenvolvimento de veículos autônomos para locomoção segura em ambientes urbanos. Este projeto se insere no grupo de trabalho GT 1, onde há projetos relacionados que já estão em desenvolvimento.

Figura 1.5: Robô móvel utilizado no Laboratório de Robótica Móvel da Universidade de São Paulo. Fonte: propriedade do LRM.

Com relação às tarefas de vigilância e patrulhamento um robô deve trazer conforto ao vigilante o qual poderia ficar em um ambiente seguro (Osório *et al.*, 2011) observando o sistema robótico e podendo tomar o controle do mesmo com uma teleoperação caso fosse necessário.

Um outro ponto interessante é referente a possibilidade de se desenvolver robôs móveis autônomos com um custo relativamente mais baixo e compatível com diversas aplicações a serem disponibilizadas no mercado, uma vez que o uso do sensor Kinect (ver capítulo 2.4) poderá viabilizar uma redução significativa dos custos dos robôs em termos de sensores, onde este sensor tem demonstrado que pode substituir com amplas vantagens o uso de outros sensores, inclusive os sensores laser que são bastante mais caros (p.ex. Hokuyo URG). Além do Kinect, foi adicionado ao sistema robótico uma câmera térmica, que permitirá a detecção de pessoas (intrusos).

Acredita-se também que este trabalho contribuiu de modo importante e significativo para o desenvolvimento de um sistema inteligente de controle e navegação para robôs móveis autônomos. Tema este relevante no cenário das pesquisas nacionais e internacionais em robótica móvel, bem como foco central das pesquisas em andamento no LRM e no INCT-SEC.

Considerando ainda que apesar da robótica estar se desenvolvendo há décadas ainda é pequena a parcela de robôs de vigilância de ambientes internos e quase não há soluções comerciais de robôs completamente autônomos e confiáveis. Portanto, os métodos desenvolvidos poderão ser aplicados em diferentes soluções robóticas, dadas as suas características, onde deverão permitir a criação de robôs móveis autônomos com custos adequados e com uma grande robustez de sua autonomia, garantindo que possam ser aplicados em diferentes problemas e tarefas.

1.2 Objetivos

Este trabalho tem como principal objetivo o desenvolvimento de um sistema de percepção do ambiente visando a navegação de robôs móveis autônomos em ambientes internos, para a execução de tarefas de monitoramento e vigilância de ambientes fechados. Com isto, busca-se uma solução que permita que o robô se desloque no ambiente evitando colisões com os objetos e obstáculos nele presentes, bem como deve inclusive auxiliar na detecção da presença de pessoas e de situações anômalas (p.ex. fontes de calor).

Este trabalho foca, portanto, na navegação do robô bem como em uma tarefa adicional que é a detecção de intrusos no ambiente em que está se locomovendo. Os dados do sensor Kinect são utilizados durante a navegação para que o robô possa se locomover de maneira segura e evitando colisões. Durante a navegação um mapa de navegabilidade local será criado permitindo assim o desvio de obstáculos. Durante a navegação o robô utilizará técnicas de inteligência artificial juntamente com processamento de imagens para se localizar no ambiente.

Este projeto visa ainda à utilização e integração das informações provenientes dos sensores câmera térmica e Kinect com a finalidade de detectar humanos. Um método de calibração será proposto para efetuar a correlação da informação destes dois sensores. Após este passo, são utilizados algoritmos para a detecção de humanos com os dados, integrados previamente, dos dois sensores utilizados. A utilização conjunta das informações dos dois sensores tem a finalidade de aumentar a confiabilidade e robustez durante a detecção de seres humanos, já que cada sensor tem suas limitações.

1.3 Estrutura da Dissertação

O documento está organizado em 6 Capítulos, cujos conteúdos se encontram estruturados abaixo:

- O Capítulo 2 descreve os principais conceitos de robótica e sensores utilizados ao longo da dissertação.

- O Capítulo 3 apresenta brevemente trabalhos relacionados a este projeto.
- O Capítulo 4 descreve o funcionamento do sistema, os algoritmos desenvolvidos, o uso das ferramentas e os experimentos realizados.
- O Capítulo 5 finaliza esta dissertação apresentando as conclusões.

CAPÍTULO

2

Robôs Móveis Autônomos

Os robôs móveis autônomos (ver Figura 2.1) fazem parte de uma classe de robôs, os quais possuem mecanismos que lhe permitem se movimentar em um ambiente de forma autônoma ou semiautônoma (Wolf *et al.*, 2009). Algumas ações e tarefas típicas incluem: navegar de um ponto a outro no seu ambiente de modo seguro; o robô pode seguir um ponto móvel no seu ambiente, como, por exemplo, seguir outro robô; um robô móvel autônomo pode monitorar um ambiente interno detectando intrusos e incidentes. Para todas estas e muitas outras tarefas é necessário que o robô tenha autonomia/inteligência para desviar de obstáculos, mapear os elementos do ambiente, saber sua própria posição e traçar trajetórias.

Dentro deste contexto, pode-se citar que as três maiores temáticas de pesquisa na robótica móvel autônoma são o mapeamento de ambientes, a localização e a navegação do robô (Wolf *et al.*, 2009). Estes assuntos serão apresentados nas próximas seções.

A inteligência, citada anteriormente, faz parte do controle inteligente o qual compõe o *software* do robô. Além disso, o robô deve ser dotado de sensores e atuadores (Bekey, 2005; Matari, 2007). Os sensores fazem parte do *hardware* do robô e permitem que seja feita a captura de dados (percepção) do próprio robô e do ambiente no qual ele está inserido. Os atuadores também constituem o *hardware* e permitem que o robô modifique o ambiente e/ou efetue o seu deslocamento no mesmo. O controle inteligente, portanto, é responsável por obter os dados dos sensores, processar e planejar possíveis ações para os atuadores (Dudek e Jenkin, 2000).

Figura 2.1: Robô Móvel Autônomo utilizado junto ao LRM para fins acadêmicos. Este robô possui um sensor laser e uma câmera de vídeo. Fonte: `http://www.lrm.icmc.usp.br/`

A finalidade deste capítulo é a de apresentar uma introdução à robótica móvel. Uma vez que o enfoque do trabalho é a integração de sensores, a navegação de um robô em um ambiente interno e a detecção de humanos, então serão apresentados nas próximas seções os conceitos básicos de robôs móveis autônomos, os sensores e as ferramentas utilizados.

2.1 Mapeamento

Para que um robô móvel se locomova no ambiente é necessário que exista um mapa no qual ele está inserido para que possa se localizar e navegar no ambiente. O processo de mapeamento envolve a captura de informações do ambiente onde o robô se encontra utilizando sensores (câmeras de vídeo, por exemplo) e a aplicação de algoritmos de inteligência computacional para a criação do mapa que representa a visão que o robô tem do ambiente (Thrun, 2002; Thrun *et al.*, 2005).

O mapa pode ser considerado local ou global. O primeiro é criado em tempo de execução e não é armazenado na memória, serve apenas para a navegação segura (regiões navegáveis e regiões não navegáveis). O mapa global representa o mapa completo do ambiente e pode ser utilizado para o robô planejar rotas/trajetórias de um lugar até outro dentro do mapa.

Com relação à estrutura os mapas podem ser classificados em: topológicos e métricos (Murphy, 2000; Thrun, 2002). Os primeiros representam o mapa em forma de grafo, onde os vértices são os lugares que o robô pode alcançar e as arestas são os caminhos que o robô pode seguir para chegar de um lugar a outro. Este tipo de mapa não leva em consideração a geometria do ambiente (não é métrico), nem descreve em detalhes a presença de obstáculos.

Devido à sua simplicidade de representação consome menos recursos computacionais (Hata, 2010; Wolf *et al.*, 2009).

Os mapas métricos, por sua vez, representam com mais precisão o ambiente. Os mais utilizados são chamados de grades de ocupação. Neste caso, são mapas bidimensionais onde cada célula tem um valor binário, no qual o valor 0 (zero) representa que esta célula é um espaço livre, não ocupado por obstáculos ou paredes, e o valor 1 (um) em uma célula representa uma porção do espaço que está ocupada. Portanto, 0 representa uma região navegável e 1 representa uma região não navegável. Em certos casos, os mapas representados por grades de ocupação podem substituir os valores binários (ocupado/livre) por uma representação indicando a probabilidade de uma determinada posição estar ocupada (Thrun *et al.*, 2005), onde sua visualização é representada por valores variando do certamente livre (0.0) ao certamente (1.0), tendo ainda a possibilidade de representar uma célula com estado desconhecido/não explorado (0.5). Da mesma forma que os mapas topológicos, um mapa métrico pode ser utilizado para planejar e estimar trajetórias.

Na Figura 2.2 são apresentados exemplos de um mapa topológico e de um mapa métrico representando uma grade de ocupação (probabilística).

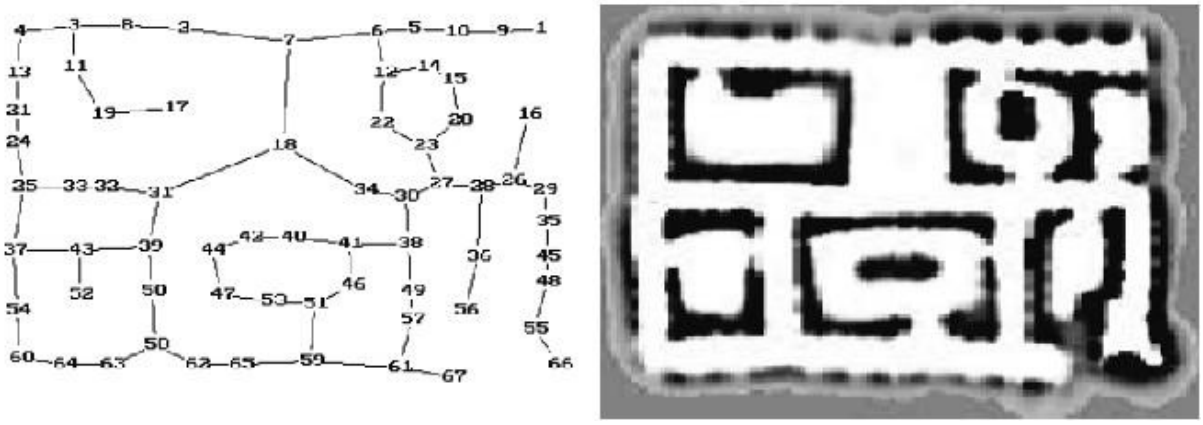


Figura 2.2: Exemplo de mapa topológico à esquerda e de um mapa métrico à direita. Fonte: Thrun (2003).

Neste trabalho foi considerado que o controle de navegação pode ser baseado justamente em uma sequência de etapas a serem seguidas (trechos a serem visitados), e que um mapa topológico armazena partes específicas do ambiente e pontos de referência. Esta sequência de trechos pode ser definida como um autômato finito (sequência de estados e ações associadas), e que permite representar uma trajetória definida em um mapa topológico. Esta abordagem foi proposta por Daniel Sales (Sales, 2012a), em sua dissertação de mestrado, tendo sido desenvolvida junto ao LRM/ICMC-USP.

A Figura 2.3 exemplifica essa abordagem. Dado o mapa topológico de uma residência, um caminho entre sala (posição inicial do robô) e área de serviço poderia ser descrito pela sequência {Sala-Corredor-Cozinha-Área de Serviço}, ou {Sala-Jardim-Área de Serviço}. Portanto, uma vez conhecida a posição de origem e destino no mapa topológico (que é previamente fornecido), o robô pode definir uma trajetória e navegar de modo a avançar de um nodo a outro do mapa topológico. De acordo com Sales (Sales, 2012a), são implementados comportamentos reativos simples de navegação para cada situação (estado), e ao ser detectada a transição de um nodo a outro (através de um reconhecimento dos padrões que definem as transições, como portas e passagens com curvas a esquerda ou a direita), o robô atualiza sua localização em relação ao mapa topológico, ao mesmo tempo em que seleciona um novo comportamento adequado para navegar neste ambiente (p.ex. avançar em um corredor). A abordagem proposta em (Sales, 2012a) realiza esta navegação baseada em mapas topológicos através da implementação de autômatos finitos e de redes neurais artificiais, que são responsáveis pela detecção da troca de estados (passagem de um nodo a outro do mapa).

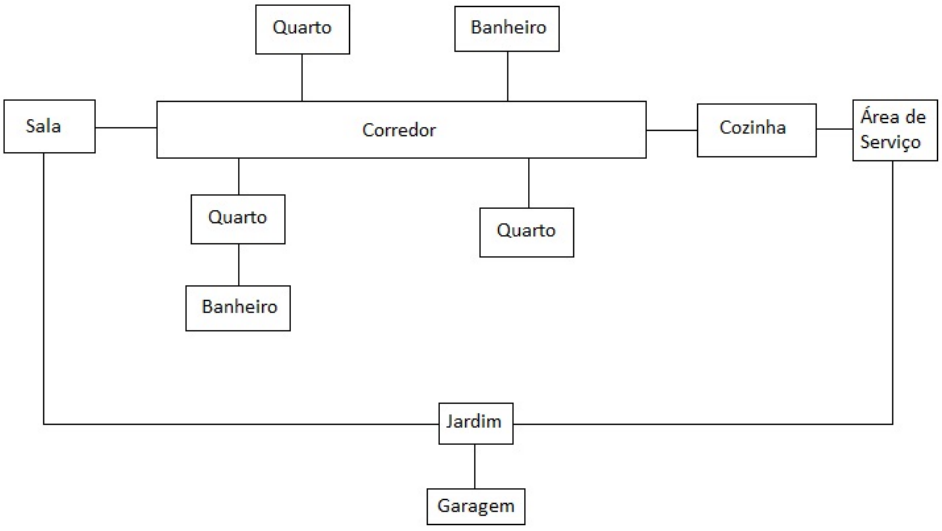


Figura 2.3: Exemplo de mapa topológico de uma residência (Sales, 2012a).

Nesta dissertação, considerou-se a proposta do uso da metodologia apresentada em (Sales, 2012a), sendo que as tarefas que necessitavam ser desenvolvidas eram: (i) a detecção de transições entre nodos do mapa topológico, e (ii) a implementação de um comportamento reativo de navegação que permitisse seguir de um nodo a outro do mapa, desviando de obstáculos. A detecção de transições entre nodos do mapa topológico foi desenvolvida usando uma rede neural artificial para identificar situações específicas (baseada no uso do Kinect), conforme será apresentado na seção 4.3. E a implementação do comportamento reativo para navegação com

desvio de obstáculos foi implementada (também baseada no uso do Kinect), conforme será apresentado na seção 4.2.

2.2 Localização

A localização na robótica móvel é a temática onde são pesquisadas técnicas precisas para identificar a localização do robô, ou seja, usualmente busca-se identificar da forma mais precisa possível sua pose, indicada pelas suas coordenadas e sua orientação no ambiente. Para efetuar a localização, da mesma forma que no mapeamento, o robô utiliza os sensores (odômetro[1], por exemplo) juntamente com um mapa ou sistema de referência (Siegwart e Nourbakhsh, 2004) e alguns algoritmos de estimação (Thrun *et al.*, 2005). Os sensores de odometria são muito utilizados para calcular o deslocamento do robô facilitando sua localização após a locomoção (Wolf *et al.*, 2009).

O mapeamento é útil para o processo de localização, já que o robô precisa de uma descrição concisa do ambiente existente para estar localizado. Quando não há um mapa previamente criado então é utilizada uma abordagem chamada SLAM (*simultaneous localization and mapping*) na qual o robô executa as tarefas de localização e mapeamento ao mesmo tempo.

O problema de localização na robótica pode ser dividido em local e global (Thrun *et al.*, 2005). No primeiro caso o robô já conhece sua posição inicial e ao longo do tempo deve manter a estimativa da sua localização no ambiente. No caso da localização global não é conhecida a posição inicial e algoritmos devem ser utilizados para estimar sua posição (Thrun *et al.*, 2005).

No mapa métrico a posição do robô deve ser geometricamente precisa, uma vez que este mapa leva em consideração a geometria do ambiente. Em um mapa topológico o algoritmo utilizado para a localização deverá posicionar o robô em algum dos vértices ou arestas do grafo que representa o mapa do ambiente, não sendo necessário a localização exata dentro de uma aresta, por exemplo.

Nesta dissertação, uma vez que foi considerado o uso de um mapa topológico, não havia a necessidade de se determinar a localização precisa do robô. A localização inicial aproximada, determinando em qual nodo do mapa está o robô, pode ser obtida por métodos como o proposto por Pessin (Pessin, 2012, 2013), que se utiliza do sinal Wi-Fi para determinar a posição aproximada de um robô. Uma vez determinada a posição inicial do robô no mapa topológico, a manutenção desta posição pode também ser feita baseada em técnicas utilizando o sinal de Wi-Fi, mas também através do acompanhamento da trajetória deste a medida que percorre os nodos do mapa topológico, como proposto por Sales (Sales, 2012a). Uma outra alternativa seria

[1]Odômetro: Dispositivo utilizado para medir deslocamento de um veículo.

o uso de localização baseada em imagens como proposto em Couto (Couto, 2012), entretanto esta técnica foi usada somente em ambientes onde havia uma boa iluminação que permitia identificar os pontos de referências em imagens capturadas por uma câmera normal. Portanto, nesta dissertação a localização do robô não será considerada como uma prioridade, uma vez que a mesma pode ser obtida através de métodos alternativos (pelo uso do próprio mapa topológico (Sales, 2012a), ou pelo uso de técnicas de localização por Wi-Fi (Pessin, 2013)).

2.3 Navegação

Durante a movimentação do robô móvel, este deve ser capaz de se deslocar evitando colisões com obstáculos e selecionando um caminho adequado para a navegação segura (Siegwart e Nourbakhsh, 2004). É necessário, portanto, definir uma trajetória, onde usualmente busca-se saber a sua localização e ter um mapa do ambiente a fim de planejar o caminho a seguir (Dudek e Jenkin, 2000). Este planejamento é feito pelo controle inteligente e faz parte da temática de navegação robótica.

Além de considerar o mapeamento e a localização é necessário também implementar o planejamento de trajetórias e o desvio de obstáculos (Hata, 2010). Segundo (Dudek e Jenkin, 2000) alguns aspectos que podem nortear planejamento de trajetória é a busca por um menor caminho. Algoritmos de busca em grafos, entre outras técnicas, são utilizados para efetuar o planejamento de trajetória. Alguns exemplos de algoritmos são Djikstra, A* e D* (Dudek e Jenkin, 2000; Hata, 2010).

O desvio de obstáculos, por sua vez, envolve a manipulação das velocidades linear (avanço) e angular (giro) do robô (Hata, 2010) com a finalidade de que a navegação ocorra de forma segura. Os métodos de desvio de obstáculos podem ser classificados em global e local. No primeiro caso todos os movimentos do robô são definidos antes mesmo de iniciar a navegação. Desta forma é necessário que um mapa seja construído anteriormente. Quando o método para desvio é local então será analisada apenas uma porção do mapa e os comandos de velocidade serão apenas para o desvio local do obstáculo (Siegwart e Nourbakhsh, 2004).

Nesta dissertação, a navegação é realizada com base em uma percepção local (mapa métrico local), que permite identificar os obstáculos e pessoas, e realizar uma navegação segura de modo a evitar colisões. Além disto, a percepção local permite também identificar os "marcos topológicos", ou seja, identificar a passagem por regiões que identificam a transição de um nodo a outro do mapa topológico como, por exemplo, portas ao longo de um corredor. Para que isto seja possível são necessários sensores capazes de prover uma boa representação do ambiente,

onde neste caso, foram adotadas câmeras e sensores especiais (Kinect, Câmera FLIR), porém de relativo baixo custo, e que serão detalhados na seção seguinte.

2.4 Sensores

Nesta seção serão apresentados os sensores utilizados neste projeto de mestrado: o sensor térmico FLIR PathFindIR e o sensor de distância Microsoft Kinect. Inicialmente será apresentada uma descrição geral sobre os sensores bidimensionais e tridimensionais, seguida da apresentação dos sensores específicos usados neste trabalho.

Os sensores, como citado no início deste capítulo, fazem parte do *hardware* dos robôs e podem ser classificados do ponto de vista da origem de sensoriamento como proprioceptivos e exteroceptivos (Bekey, 2005; Siegwart e Nourbakhsh, 2004). Os primeiros medem informações do próprio robô, como carga de bateria, deslocamento e velocidade (com o uso de encoders). Os sensores exteroceptivos recebem informações do meio em que o robô está inserido, como, por exemplo, sensores que detectam obstáculos à frente do robô, tal como sensores à laser, sonares e sensores infravermelhos (Matari, 2007; Murphy, 2000).

Os sensores podem ainda ter mais uma classificação, onde esta é com relação ao tipo de medição. Podem ser passivos, neste caso possuem um detector que apenas recebe informações do ambiente, a exemplo das câmeras de vídeo. Quando são ativos, os sensores tem um emissor para gerar um sinal e enviar ao ambiente e logo após recebem a reflexão deste sinal por um detector. Um exemplo é o sonar que utiliza ondas sonoras para detectar a distância dos objetos (Matari, 2007).

Finalmente, ainda podemos classificar os sensores de acordo com a grandeza da informação fornecida, podendo ser sensores pontuais (realizam uma única medida), sensores bidimensionais (representação em 2D, com projeção em um plano) e sensores tridimensionais (representação em 3D, com informações espaciais), conforme descritos a seguir.

2.4.1 Sensores Bidimensionais

Os sensores 2D, do tipo câmera de vídeo e câmera termal, criam mapas bidimensionais do mundo, ou seja, imagens. Uma imagem é uma grade de duas dimensões formada por elementos de mesmo tamanho chamados de pixel (do inglês *picture element*) (Murphy, 2000). As câmeras de vídeo são mecanismos que procuram uma aproximação e imitação do olho biológico, buscando ter uma percepção visual mais completa do mundo (Matari, 2007). Alguns sensores captam apenas a intensidade de luz, outros são capazes de captar informações de cor, e outros ainda percebem o calor através de radiação infravermelha.

As câmeras são dotadas de uma lente, por onde entra a luz, e um sensor que é um chip para processar as informações recebidas pela câmera (Matari, 2007). Na Figura 2.4 são visualizados os componentes de uma câmera de vídeo.

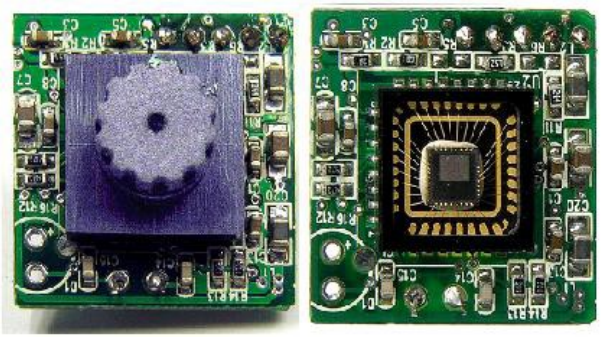


Figura 2.4: À esquerda a lente acoplada ao chip da câmera. À direita o sensor CMOS da câmera. Fonte: `http://en.wikipedia.org/wiki/File:Sweex_USB_webcam_PCB_with_without_lens_close_up.jpg`

A intensidade de luz pode ser representada na escala de cinza (tonalidades), neste caso irá utilizar um byte para representar a intensidade de um pixel. Chama-se este modelo de imagem monocromática de 8-bits. O valor armazenado zero significa preto e o valor armazenado 255 significa branco (intensidade máxima). Os valores intermediários são os tons de cinza. Um dos esquemas utilizados para representação de imagens coloridas é o RGB (do inglês *red*, *green*, *blue*). Este modelo de cor é chamado de 24-bits, pois se utiliza três bytes para representar a cor de um pixel, com seus componentes vermelho, verde e azul. Desta forma o preto é representado pelos três bytes com valor zero e o branco é obtido quando os três bytes de cor possuem o valor 255. Com as possíveis combinações de valores para cada byte de cor obtêm-se pixels coloridos (Murphy, 2000), podendo representar até 16 milhões de cores com o uso destes 24 bits.

As câmeras de vídeo utilizadas na robótica móvel para ambientes internos podem ser câmeras simples, onde algumas vezes são utilizadas as chamadas *webcams* pelo baixo custo e simplicidade de uso. Sua resolução é baixa, geralmente 320x240 ou 640x480 o que facilita o processamento. Um tipo particular de câmera de vídeo é aquele representado pelas câmeras de vídeo estéreo, onde é feita a aquisição simultânea de 2 imagens através de um conjunto binocular de câmeras, no entanto, este tipo de câmeras além de obter imagens 2D, é usado também para a aquisição de informações em 3D obtidas pela combinação das duas imagens (*Depth Map/Disparity Map*).

Existem câmeras capazes de captar imagens além do espectro visível, como é o caso dos sensores usados em satélites de monitoramento. A radiação infravermelha é um espectro de luz não visível onde sensores específicos podem ser capazes de perceber este tipo de radiação,

sendo capazes de indicar também a presença de calor. As câmeras termais (ou térmicas) são câmeras de vídeo que utilizam estes sensores infravermelhos (*near/far infrared*) para perceber o calor do ambiente e representar em forma de imagens a temperatura dos elementos presentes em uma cena (FLIR, 2011b). Estas imagens podem ser em tons de cinza ou utilizar falsas cores para destacar mais os elementos das imagens, usando uma paleta com cores "frias" e "quentes". As regiões mais quentes do ambiente são usualmente representadas com cores mais claras e regiões mais frias são representadas com cores mais escuras (Bräunl, 2008).

Duas câmeras térmicas que vem sendo utilizada em aplicações de robótica junto ao laboratório LRM são a FLIR PathFindIR e a FLIR TAU 640 (ver Figura 2.5). São dispositivos de relativo baixo custo e pequeno porte que são utilizados também para aplicações de visão noturna (FLIR, 2011a). Na robótica é de grande utilidade a possibilidade de verificar se um obstáculo à frente é um ser um humano ou um animal (seres de temperatura quente), ou um simples objeto sem vida (elemento frio).

(a) (b)

Figura 2.5: Câmera térmica FLIR PathFindIR à esquerda e FLIR TAU 640 à direita. Fonte: FLIR (2011a).

Na Figura 2.6 é visualizada à esquerda uma imagem gerada por uma câmera de vídeo comum e à direita imagem gerada por uma câmera térmica. Não seria possível ou seria muito difícil visualizar um animal na câmera de vídeo comum nestas condições de luminosidade, porém com o uso da câmera térmica o animal é facilmente identificado, pois o corpo deste animal é mais quente do que a temperatura do ambiente.

É importante destacar que o laboratório LRM conta atualmente com diferentes modelos de câmeras comuns (*webcams* e placas de aquisição de vídeo), possuindo também as câmeras térmicas PathFindIR e TAU 640, que serão alvo das pesquisas desenvolvidas junto à este projeto de mestrado (maiores detalhes no capítulo 4). As imagens obtidas a partir destas câmeras são adquiridas e tratadas com o uso da biblioteca OpenCV (Bradski e Kaehler, 2008), uma ferramenta

Figura 2.6: Visualização noturna de uma câmera de vídeo comum (à esquerda) e de uma câmera termal (à direita). Fonte: FLIR (2011a).

de código-aberto, robusta e flexível, que permite capturar, processar e salvar imagens obtidas a partir de diferentes tipos de dispositivos e em diferentes formatos (ver subseção 2.5.1).

2.4.2 Sensores Tridimensionais

Os sensores aqui apresentados são capazes de ter percepção tridimensional do ambiente. Sua representação pode ser em mapa de profundidade, o qual é uma imagem 2D com informação de distância, ou seja, cada pixel da imagem representa a distância daquele ponto. Um exemplo de mapa de profundidade é apresentado na Figura 2.7.

Figura 2.7: Um mapa de profundidade gerado pelo sensor Kinect. Cores escuras representam menor distância e cores claras representam maior distância da cena até o sensor.

Outra forma de representar o ambiente utilizando a informação proveniente destes sensores é utilizando uma nuvem de pontos. Na Figura 2.8 é visualizada uma nuvem de pontos 3D. Nesta representação é utilizado o sistema de coordenadas (x, y, z) no qual cada ponto do ambiente tem sua posição neste sistema. Quando informação de cor é adicionada na nuvem de pontos então tem-se uma nuvem de pontos 4D (coordenadas x, y e z e cor) (PCL, 2012) que também pode ser chamada de informação RGB-D (do inglês *red*, *green*, *blue*, *depth*) que é informação de cor adicionada de profundidade.

Figura 2.8: Uma nuvem de pontos. Fonte: `http://docs.pointclouds.org/trunk/group__registration.html`

Um sensor 3D que foi lançado recentemente pela Microsoft[2] é o Kinect (ver Figura 2.9). O dispositivo é um acessório de baixo custo projetado para o console XBOX 360 a fim de permitir que o jogador possa usar apenas o corpo como controle nos jogos. O *software* do jogo obtém os dados do Kinect e faz o reconhecimento do usuário rastreando os movimentos do mesmo e transformando-os em ações válidas no jogo.

Figura 2.9: Sensor Kinect adotado junto ao XBOX 360 da Microsoft. Fonte: `http://www.xboxplus.net/2010/06/14/projeto-natal-e-kinect-for-xbox-360/`.

Para possibilitar este reconhecimento de objetos, o dispositivo possui uma câmera RGB para obter imagens coloridas, um emissor infravermelho e uma câmera capaz de captar esta luz

[2]Microsoft – Web: `http://www.microsoft.com/`

infravermelha. Para obter a imagem em profundidade (ver Figura 2.10, o emissor preenche o ambiente com radiação infravermelha e o sensor capta o sinal refletido pelos objetos da cena possibilitando o cálculo de distância de cada ponto e gerando assim um mapa de profundidade (Kinect, 2012a,b). Esta abordagem é conhecida como "luz estruturada", onde um padrão conhecido é projetado no ambiente. Depois uma imagem é capturada com este padrão projetado sobre a cena, permitindo assim estimar a profundidade dos elementos deste ambiente. No caso do Kinect a luz e o padrão projetados não são visíveis a olho nu. A 2.11 apresenta um exemplo do padrão projetado pelo Kinect, onde podemos ver a variação da densidade de pontos da luz estruturada, conforme a distância do objeto até o sensor, permitindo assim estimar a distância para cada pixel da imagem.

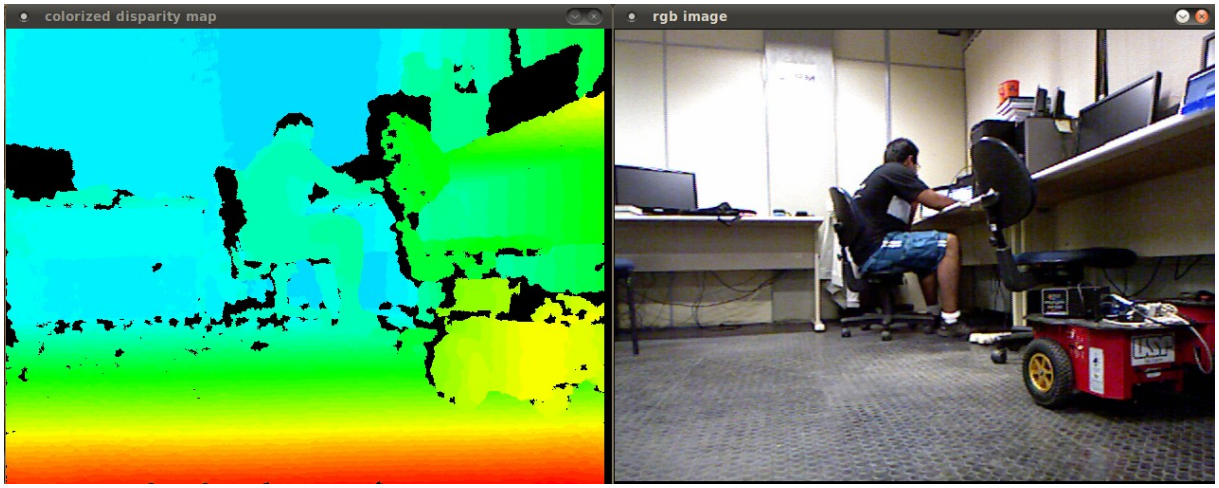


Figura 2.10: Imagens geradas pelo Kinect. À esquerda uma imagem gerada pela câmera de profundidade do Kinect, onde as cores quentes (avermelhadas) são de elementos mais próximos, e cores frias (azuladas) são de elementos mais afastados. À direita uma imagem gerada pela câmera de vídeo do Kinect.

Apesar do dispositivo Kinect ter sido lançado exclusivamente para ser utilizado no console XBOX foram criados diversos drivers (a princípio não oficiais) para sua comunicação direta com computadores. Além disso considera-se um baixo custo quando comparado à outros sensores de distância e de profundidade. Isso permitiu que o Kinect fosse adotado para pesquisas em diversas áreas em todo o mundo como saúde, educação e robótica (KinectHacks, 2012). Nesta área da robótica, vem sendo utilizado para, entre outras tarefas, criar mapas de profundidade, onde é possível perceber com bastante precisão os diferentes tipos de obstáculos à frente do sensor.

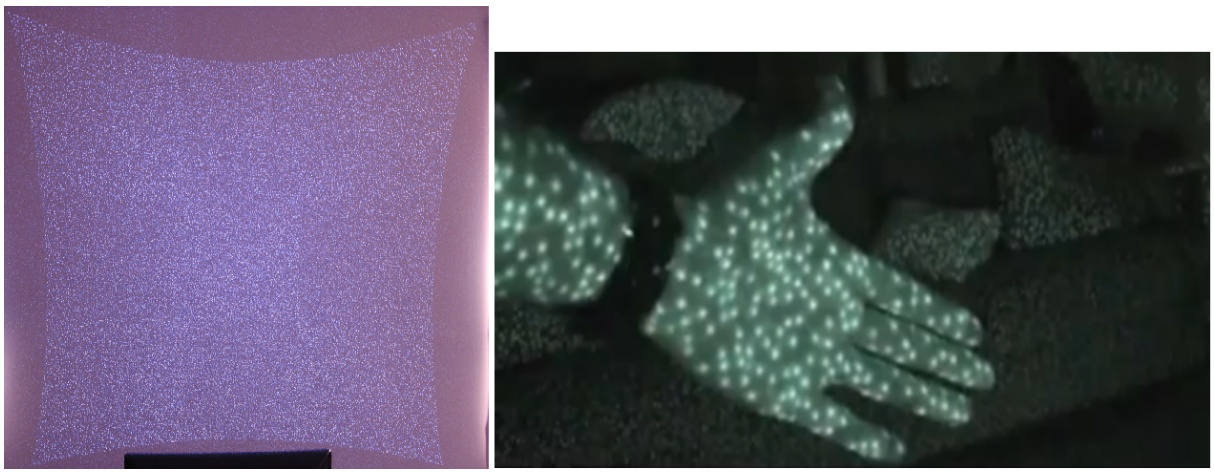


Figura 2.11: Luz estruturada. À esquerda uma projeção completa da luz estruturada gerada pelo Kinect. À direita a projeção da luz estruturada em um objeto. Fonte: `http://goo.gl/6Hs0sj`

2.5 Ferramentas

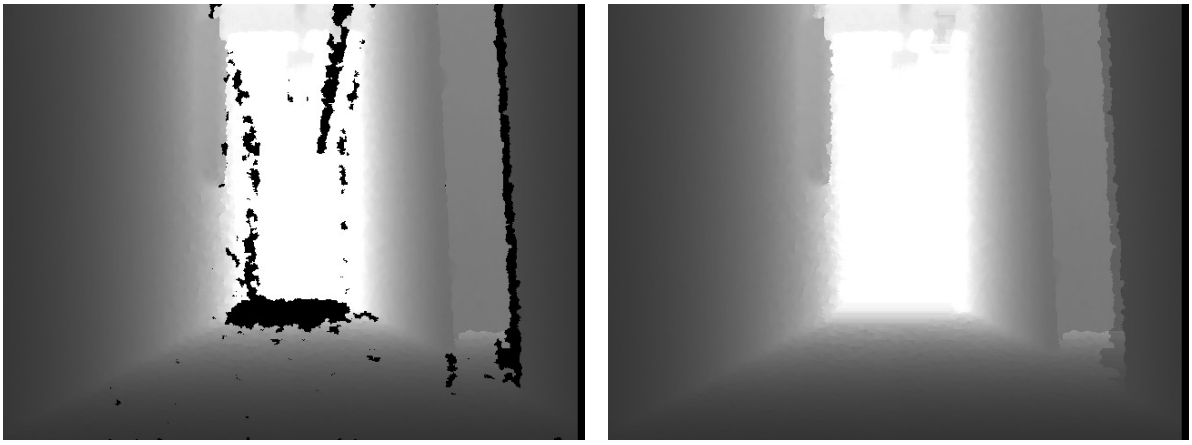
Do ponto de vista da Ciência da Computação (e consequentemente de *software*) o projeto de um robô móvel autônomo envolve a implementação/utilização de várias técnicas de aprendizado de máquina, processamento de imagens, visão computacional, busca em grafos, programação dinâmica, entre outras. Além disso é importante que sejam feitas simulações para validar os projetos de robôs, bem como utilizar drivers para a comunicação de computadores com os robôs e com os sensores. Existem diversas bibliotecas e drivers implementados em *software* livre (open source) que podem ajudar na criação e manutenção de sistemas robóticos, facilitando a sua implementação, como será visto nas próximas subseções.

2.5.1 OpenCV

Com o advento e a acessibilidade de computadores de cada vez maior poder de processamento várias áreas da computação tem usufruído deste desenvolvimento. Entre elas estão a visão computacional e o processamento de imagens. Áreas estas que estão envolvidas com a captura de imagens a partir de sensores, o seu processamento e a extração de informações importantes. Tendo isso exposto é apresentada nesta seção a biblioteca OpenCV (do inglês *Open Computer Vision*).

A OpenCV (OpenCV, 2012) é uma biblioteca de visão computacional que surgiu em 1999 na Intel Research[3] (hoje é mantida pela instituição de pesquisa Willow Garage[4] e conta com mais de 2500 algoritmos otimizados para execução em tempo-real. Suas aplicações estendem-se em várias áreas, como inspeção de produtos em uma fábrica, análise de imagens médicas, vigilância, interface de usuário, calibração de câmera, visão estéreo e robótica (Bradski e Kaehler, 2008; OpenCV, 2012).

Na robótica o uso da biblioteca OpenCV vai desde a captura dos dados dos sensores (câmera de vídeo, sensor Kinect), até a extração de características (encontrar pontos de interesse na imagem, por exemplo) e aplicação de técnicas de aprendizado de máquina que o OpenCV possui. Além disso, há rotinas de aplicação de filtros e segmentação de imagens (pré-processamento), úteis para facilitar o processamento das imagens ao longo de todo o processo de sensoriamento do robô. Nas Figuras 2.12(a) e 2.12(b), são apresentados exemplos de uma aplicação da biblioteca OpenCV na robótica móvel. A imagem de profundidade utilizada na navegação do robô foi submetida a um processo de remoção de ruído (porções indesejáveis da imagem e que podem prejudicar o processamento da mesma) , facilitando, entre outros aspectos, a identificação das regiões navegáveis. Detalhes sobre o pré-processamento das imagens serão apresentados na seção 4.1.2.



(a) (b)

Figura 2.12: Mapa de profundidade gerado pelo Kinect: (a) Imagem original do mapa de profundidade com ruído e (b) Imagem do mapa de profundidade após a remoção de ruído.

[3] `http://http://techresearch.intel.com/`
[4] `http://www.willowgarage.com`

2.5.2 Player/Stage e Gazebo

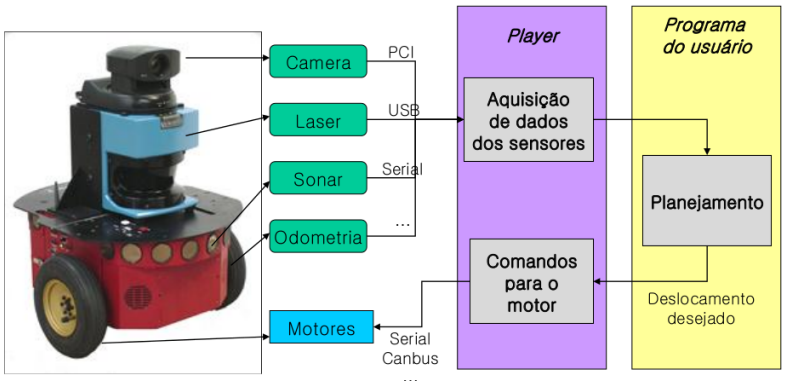
Para realizar o desenvolvimento e a programação de sistemas robóticos móveis -faz-se necessário o uso de simuladores para teste e validação dos métodos desenvolvidos. Um dos fatores motivadores disto é a segurança, ou seja, evitar danos ao robô. A agilidade na hora de testar mudanças nos códigos de programação dos sistemas robóticos também é uma motivação para o uso de simuladores, já que pode ser dispendioso montar robôs com os sensores e preparar o ambiente para os testes. Dentro deste contexto é apresentado a seguir um sistema que pode ser utilizado como controle e como simulação de robôs móveis, o Player/Stage (PlayerStage, 2012).

O Player é um conjunto de ferramentas de código aberto utilizados para o controle de robôs móveis. Foi criado em 2000 por um laboratório de robótica da Universidade do Sul da Califórnia e agora é mantido por muitas outras instituições. Seu uso é simples e é possível abstrair o *hardware* utilizado, sendo de alta compatibilidade com diversas plataformas comerciais, o que facilita muito o seu uso. A Figura 2.13(a) apresenta um esquema da arquitetura do Player e da camada de abstração do *hardware* provida por este *software*. O Player é baseado no modelo cliente/servidor, conforme apresentado na Figura 2.13(b), em que o servidor é o programa em contato com o robô e que faz a leitura dos sensores e envia comandos aos atuadores do robô. O cliente é um outro programa que se conecta e se comunica com o servidor por meio de um protocolo de comunicação em rede, por meio do qual recebe informações do servidor, interpreta e envia comandos (Wolf e Osório, 2011). Esta ferramenta suporta várias plataformas de robôs e tipos de sensores (Wolf *et al.*, 2009). O Player tem seu desenvolvimento concomitante com dois simuladores, o Stage e o Gazebo.

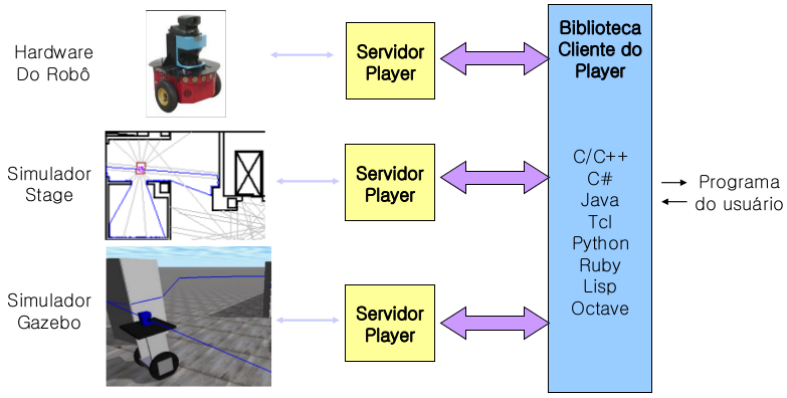
O Stage é um sistema que trabalha em conjunto com o Player para a simulação de um ambiente bidimensional de robôs e de sensores. Os robôs simulados são baseados no Pioneer (ver Figura 2.1), um robô fabricado pela empresa MobileRobots[5], pelo fato de serem robôs de pesquisa amplamente utilizados em todo o mundo (Wolf *et al.*, 2009). O Gazebo, por sua vez, é um sistema utilizado para a simulação de ambientes tridimensionais, além de simular a física dos objetos (robôs e obstáculos, por exemplo) presentes no ambiente. Desta forma é possível obter um maior nível de realismo na simulação.

Em ambos os *softwares* de simulação há modelos de ambientes e de robôs para serem utilizados, porém, é possível criar novos modelos de acordo com a necessidade do sistema robótico (Hata, 2010). Nas Figuras 2.14(a) e 2.14(b) são apresentados a tela do Stage e do Gazebo, respectivamente, simulando um robô e seus sensores.

[5]MobileRobots – Web: `http://mobilerobots.com`

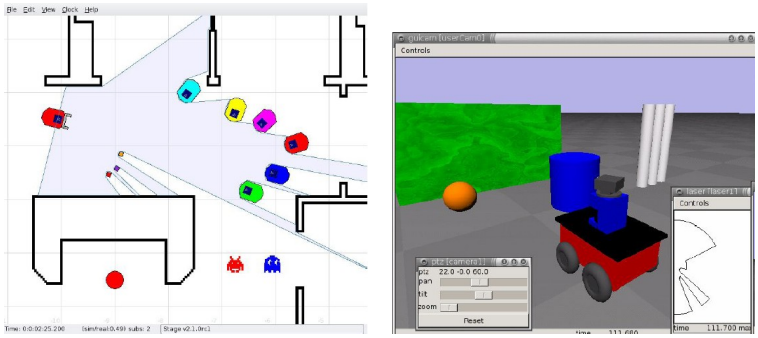


(a)



(b)

Figura 2.13: Arquitetura do Player. Em (a) é apresentada a abstração de *hardware* e em (b) o modelo cliente/servidor.



(a) (b)

Figura 2.14: (a) Interface gráfica do Stage exibindo alguns obstáculos, robôs e a atuação de um sensor laser e (b) Ambiente de simulação de robôs Gazebo. Adaptados de (Hata, 2010).

2.5.3 PrimeSense OpenNI/NITE

Como visto na seção 2.4.2 o sensor Kinect foi criado para o console Microsoft XBOX 360 e posteriormente foram criados drivers para sua utilização junto a um computador. Um destes

drivers é o OpenNI (do inglês *Open Natural Interaction*) (OpenNI, 2012), desenvolvido pela empresa PrimeSense[6], fabricante do *hardware* do Kinect. O driver foi criado em 2011 e possui rotinas de obtenção dos dados do Kinect que são a imagem RGB, o mapa de profundidade e o áudio. O NITE é um conjunto de métodos para interação humano/máquina e funciona juntamente ao OpenNI. Ele provê algumas funções para rastreamento do corpo humano, como pode ser conferido na Figura 2.15.

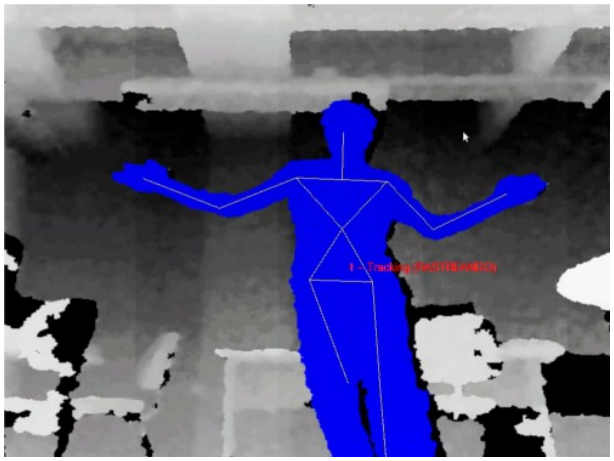


Figura 2.15: Rastreamento de um humano utilizando o módulo NITE e o sensor Kinect.

Quando o programa está em execução fica em constante procura por objetos na cena que tenham a forma de um ser humano. Quando algo em formato potencial de humano está presente no ambiente, este recebe uma coloração diferente e é marcado como sendo um usuário novo. Para que o rastreamento comece e esta pessoa receba uma identificação é necessário fazer o que se chama de calibração, na qual o humano deve permanecer rapidamente em uma determinada postura (ver Figura 2.16) chamada de *Psi Pose* (em português postura psi), uma referência à letra grega Ψ (psi). Em versões mais atuais do *software* não é necessário fazer a postura *psi*, o que facilita a detecção de pessoas.

2.5.4 ROS

Nas subseções anteriores (2.5.1, 2.5.2 e 2.5.3) foram apresentadas algumas ferramentas utilizadas para o processamento de imagens (OpenCV), controle e simulação de robôs (Player/Stage), bem com a obtenção e tratamento de dados de sensores (OpenNI). Com o grande número de drivers, bibliotecas surgiu a necessidade de um sistema que fizesse a integração destas ferramentas para facilitar o desenvolvimento de aplicações para robótica.

[6]PrimeSense – Web: `http://www.primesense.com`

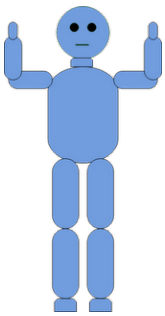


Figura 2.16: Postura utilizada para a calibração do rastreador de humanos do OpenNI/NITE utilizando o sensor Kinect. Fonte: `http://www.ros.org/wiki/openni_tracker`.

Neste contexto surgiu o ROS (do inglês *Robot Operating System*), o qual foi criado e é mantido pela empresa Willow Garage. O ROS é, portanto, um meta-sistema operacional com mais de 2000 bibliotecas para utilização no desenvolvimento de aplicações robóticas. Algumas de suas características incluem abstração de *hardware*, drivers de dispositivos, bibliotecas, passagem de mensagens entre processos e visualizadores (ROS, 2012). Na Figura 2.17 é apresentada uma tela do visualizador *rviz* do ROS mostrando informações bidimensional (à esquerda) e tridimensional (à direita) do sensor Kinect.

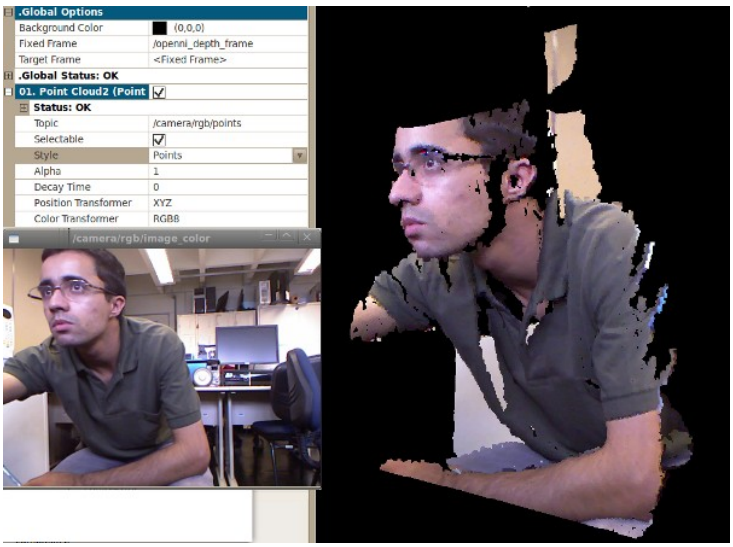


Figura 2.17: Visualizador *rviz* do ROS.

2.5.5 JavaNNS

As Redes Neurais Artificiais (RNA) são modelos computacionais baseados em redes neurais biológicas sendo utilizadas na área de Inteligência Artificial para o aprendizado de máquinas

(Rezende, 2003; Silva *et al.*, 2010). São estruturas complexas e a sua programação pode ser um pouco dispendiosa. Considerando isso é comum o uso de simuladores de redes neurais para integração em projetos. Três simuladores muito utilizados são o FANN[7] (do inglês *Fast Artificial Neural Network*), o SNNS[8] (do inglês *Stuttgart Neural Network Simulator*) e o JavaNNS[9] (do inglês *Java Neural Network Simulator*). Nesta seção será apresentado o JavaNNS, uma vez que este simulador esta sendo utilizado neste projeto de pesquisa.

O JavaNNS é um simulador de redes neurais artificiais escrito na linguagem de programação Java e baseado no simulador SNNS, um simulador de redes neurais escrito na linguagem de programação C pela Universidade de Stuttgart[10] e mantido pela Universidade de Tübingen[11], que por sua vez desenvolveu o JavaNNS.

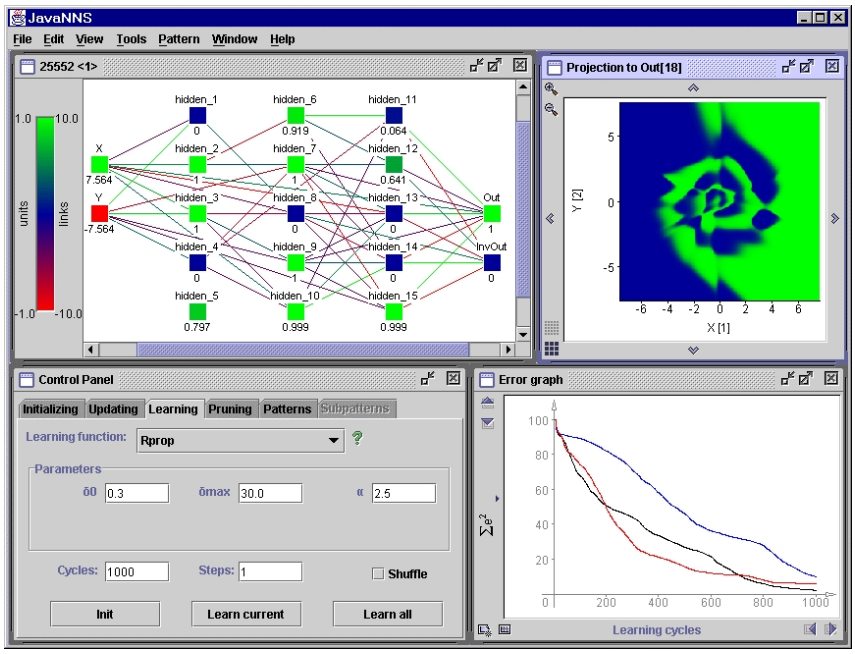


Figura 2.18: Interface gráfica do JavaNNS. Na parte superior esquerda é apresentada a estrutura de uma rede neural, do lado esquerdo inferior a tela de configuração, onde podem ser alterados vários parâmetros. Do lado direito superior é mostrada a projeção de saída da classificação e do lado direito inferior o gráfico de erro. Fonte: `http://www.ra.cs.uni-tuebingen.de/software/JavaNNS`.

Os projetos SNNS e JavaNNS foram criados para oferecer simulações eficientes e flexíveis em aplicações e pesquisa que utilizam redes neurais. Vários algoritmos e tipos de redes neurais

[7]FANN (Fast Artificial Neural Network) – Web: `http://leenissen.dk/fann/wp/`

[8]SNNS (Stuttgart Neural Network Simulator) – Web: `http://www.ra.cs.uni-tuebingen.de/SNNS/`

[9]JavaNNS – Web: `http://www.ra.cs.uni-tuebingen.de/software/JavaNNS/`

[10]Universidade de Stuttgart – Web: `http://www.uni-stuttgart.de/`

[11]Universidade de Tübingen – Web: `http://www.uni-tuebingen.de/`

foram implementados e podem ser utilizados modificando seus parâmetros de acordo com a necessidade. Durante o desenvolvimento de um sistema é possível, com um destes programas, realizar as fases que vão desde o treinamento de uma rede até a classificação utilizando a mesma.

Há uma interface gráfica para facilitar a visualização da estrutura da rede neural (neurônios de entrada, camada(s) oculta(s) e camada de saída) e dos gráficos de erros (ver Figura 2.18). É possível ainda exportar a rede neural criada no programa para ser utilizada como classificador em um projeto à parte. Esse é o caso deste projeto de mestrado. Uma vez treinada a rede neural, ela será exportada utilizando a linguagem C para ser integrada ao robô e utilizada durante o processo de navegação deste. Outros detalhes sobre a navegação utilizando redes neurais serão vistos no capítulo 4 sobre o desenvolvimento e resultados.

2.6 Considerações

A robótica móvel autônoma é uma subárea da robótica clássica e tem aplicabilidade em diversas áreas. Para a criação de robôs móveis autônomos é necessário considerar os três principais problemas desta subárea: o mapeamento de ambientes, a localização e a navegação do robô. Além disso, é necessário escolher os sensores, o *hardware*, os métodos e as ferramentas adequados.

Este capítulo apresentou, portanto, os conceitos básicos de robótica móvel autônoma, bem como alguns sensores e ferramentas de *software* utilizados ao longo deste projeto. Conceitos relacionados à construção e aquisição de mapas foram apresentados e discutidos, onde, neste trabalho foi necessária a opção por um tipo de representação do ambiente que pode ser facilmente integrado ao sistema robótico e sendo também adequado ao tipo de problema/tarefa tratada. Além disso, um método de localização e navegação também foram abordados e propostos para serem adotados na construção do sistema, bem como a implementação do desvio de obstáculos para garantir a navegação segura do robô.

Em relação às ferramentas apresentadas, o Player/Stage tem sido adotado para realizar testes de simulação, onde as facilidades oferecidas pelo Player permitem que depois o *software* testado em simulação possa ser conectado diretamente ao robô Pioneer, adotado nas pesquisas deste laboratório. O robô Pioneer possui uma câmera termal FLIR PathFindIR e um sensor Microsoft Kinect como principais sensores de obstáculos e de intrusos, sendo que a biblioteca OpenCV está sendo usada tanto para a captura e processamento das imagens térmicas quanto para a aquisição e tratamento dos mapas de profundidade obtidos a partir do Kinect.

Através do uso dos *softwares* OpenCV e Player/Stage, e dos *hardwares* do Pioneer, Kinect e FLIR, será possível desenvolver e testar diferentes algoritmos de navegação e detecção de obstáculos e de intrusos que servirão para o desenvolvimento deste trabalho. No capítulo a

seguir serão apresentados trabalhos, usando o *hardware* e *software* referidos acima, desenvolvidos principalmente junto ao LRM/ICMC-USP e que estão relacionados com as pesquisas desta dissertação.

CAPÍTULO

3

Trabalhos Relacionados

Entre os motivos para o desenvolvimento de robôs de vigilância para segurança de ambientes internos e externos estão: trazer segurança para os vigilantes humanos, economia de recursos, acessibilidade em locais de difícil acesso para humanos, além de realizar um trabalho que pode ser monótono para um ser humano.

No capítulo anterior foram apresentados conceitos sobre a robótica móvel, sobre os sensores e as ferramentas que são utilizadas no LRM/ICMC, e que serão utilizadas neste trabalho. Neste capítulo serão apresentados trabalhos relacionados a este projeto. Dentre estes trabalhos alguns são deste laboratório (LRM/ICMC), visto que este trabalho está contextualizado em um projeto maior (INCT-SEC, 2012; Osório *et al.*, 2011) (como citado no capítulo 1.1), do qual o LRM/ICMC faz parte. Além disto, considera-se que outros projetos deste laboratório também se encaixam em trabalhos/abordagens complementares e similares.

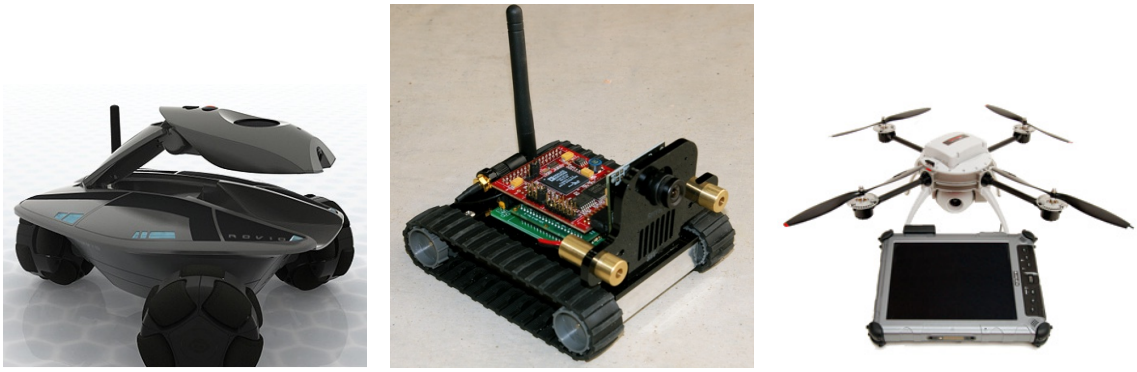
Uma vez que o tema deste trabalho são robôs de vigilância, serão abordados alguns trabalhos na seção 3.1 sobre este assunto. Na seção 3.3 serão abordados trabalhos relevantes sobre a detecção de humanos/intrusos utilizando o sensor de calor ou o sensor Kinect, já que estes sensores são utilizados na criação do robô de vigilância. Na seção 3.4 serão apresentados trabalhos para a detecção de regiões navegáveis, posto que neste projeto de pesquisa serão utilizados métodos para a criação de um mapa de navegabilidade local a partir da detecção de regiões navegáveis.

3.1 Robôs de Vigilância

A criação de robôs para vigilância de ambientes internos não é uma proposta totalmente nova (Crowley, 1987) , porém esta área de pesquisa ainda está crescendo bastante (Carnegie *et al.*, 2004; Monteiro, 2011; Osório *et al.*, 2011), visto que ainda há campo para crescimento das aplicações de robôs comerciais de segurança para ambientes *indoor* (internos). Um exemplo de robô comercial para vigilância é o Rovio (ver figura 3.1(a)) (Rovio, 2012).

Além dos robôs com aplicações práticas e comerciais, existem muitos robôs de pesquisa utilizados para estudos em robótica móvel de segurança. Alguns exemplos são o Pioneer e o Seekur, fabricados pela MobileRobots[1] e o SRV (ver Figura 3.1(b)), fabricado pela Surveyor Corporation[2]. As pesquisas nestes robôs devem ainda ganhar bastante maturidade para que produtos comerciais possam utilizar a sua tecnologia, e neste caso fazer jus aos objetivos da robótica, que entre eles estão aumentar a segurança, a autonomia, e o conforto humano.

Pode-se considerar também que algumas pesquisas em robôs de vigilância estão direcionadas a outros fins, como por exemplo, um robô para ambiente *outdoor* (externo) de aplicação militar que tem o objetivo de explorar lugares seguros em um combate e se esconder do inimigo (Hambling, 2012). Outros exemplos, também para segurança em ambiente externo, são os robôs aéreos dotados de hélices usados para captar imagens aéreas de zonas de interesse e para o monitoramento de ambientes (Aeryon, 2012) (ver Figura 3.1(c)).



(a) (b) (c)

Figura 3.1: Robôs de segurança. Em (a) e (c) dois robôs comerciais, o Rovio e o Aerion e em (b) um robô de pesquisa, o SRV. Fontes: (a) Rovio (2012), (b) `http://www.surveyor.com/SRV_info.html` e (c) Aeryon (2012)

Outros trabalhos recentes sobre robôs de vigilância em ambientes internos podem ser mencionados. Eles apresentam, porém, algumas deficiências no quesito vigilância de ambiente

[1]MobileRobots – `http://www.mobilerobots.com/`
[2]Surveyor Corporation – `http://www.surveyor.com/`

interno e detecção de intrusos. O autor Ribeiro (2011) criou um robô teleguiado por meio da internet para efetuar missões de vigilância dos ambientes de uma universidade. Não apresenta, portanto, total autonomia por ser teleguiado. O foco do trabalho de Monteiro (2011) é a localização baseada em câmeras fixas. Em Lee *et al.* (2011), o sistema também utiliza sensores no ambiente para a localização remota do robô. Nos trabalhos de Monteiro (2011) e Lee *et al.* (2011) o processo de instalação de sensores em todo o ambiente torna um sistema de alto custo. Além disso, há o risco do robô se deslocar para locais onde estes sensores estão fora do alcance (cobertura). De uma maneira geral, em todos estes trabalhos a detecção de incidentes e intrusos não é abordada em maiores detalhes.

3.2 Navegação Topológica

No trabalho de Sales (2012a) (também descrito em Sales *et al.* (2011)) – integrante do LRM/ICMC-USP e do INCT-SEC – um sistema de navegação topológica utilizando Redes Neurais Artificiais e Autômatos Finitos é apresentado. Nesse trabalho um robô deve se deslocar por um ambiente interno cujo mapa topológico é de conhecimento do robô e representado por um autômato finito, que descreve as localidades e as ações que o robô deve tomar. Para definir o contexto atual e a sua localização (saber em que posição do mapa se encontra), o sistema utiliza uma rede neural artificial treinada sobre dados de um sensor laser para reconhecer situações em um ambiente interno, como portas abertas para direita e esquerda, corredores e cruzamentos. Na Figura 3.2 é visualizada a simulação deste processo de navegação utilizando as informações de um sensor de distância to tipo laser SICK LIDAR (do inglês *Light Detection And Ranging*).

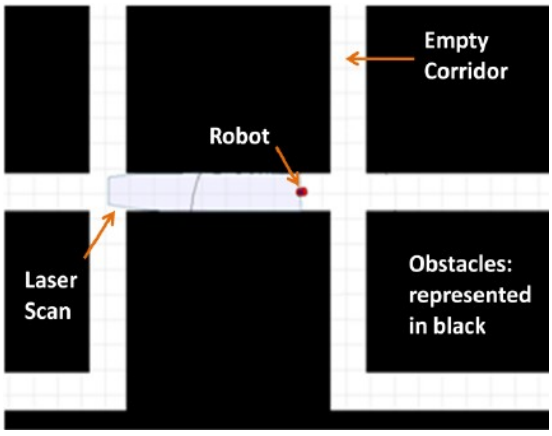


Figura 3.2: Simulação da navegação topológica. Fonte: (Sales *et al.*, 2011).

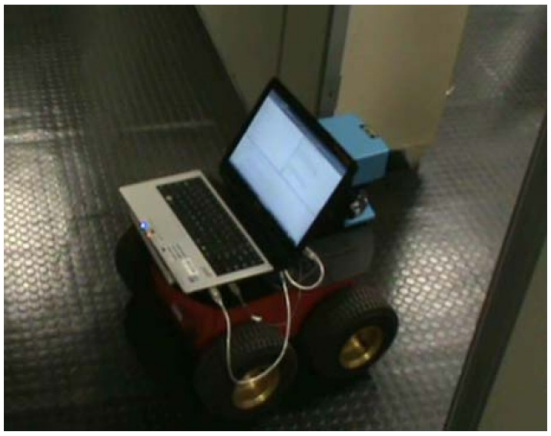


Figura 3.3: Navegação topológica real. Fonte: (Sales, 2012a).

Na Figura 3.3 é apresentada uma imagem da plataforma em tempo de execução. Esta plataforma está sendo utilizada neste projeto de mestrado e também deverá fazer a navegação topológica utilizando redes neurais artificiais, porém utilizando para este fim o sensor de distância tridimensional Kinect. Maiores detalhes serão apresentados no capítulo 4.2 o qual trata da metodologia deste trabalho.

3.3 Detecção de Humanos

Nos últimos anos tem aumentado a pesquisa no reconhecimento de humanos e de pedestres em aplicações de robótica (p.ex. detecção de intrusos, ou em aplicações de segurança no trânsito) e mais recentemente no reconhecimento de gestos de pessoas para interação natural em jogos, marketing e educação, por exemplo.

Há vários algoritmos para detecção e rastreamento de humanos. Além disso, várias ferramentas/sensores podem ser utilizados, como câmera de vídeo comum, câmera térmica e sensores RGB-D. Nesta seção serão apresentados alguns trabalhos que utilizam os sensores térmico e de profundidade para realizar a detecção de pessoas.

O autor Nin (2011) (tendo seu trabalho publicado também em Nin e Osório (2011)) é integrante do laboratório LRM/ICMC e utiliza o processamento de imagens (histogramas e segmentação de imagens) para a identificação de pessoas a partir dos dados provenientes da câmera FLIR PathFindIR (a mesma câmera utilizada neste projeto). Na Figura 3.4 é apresentado o resultado da segmentação de uma imagem térmica. Durante a segmentação as regiões mais escuras (regiões frias do ambiente, cujos valores dos pixels estão abaixo de um limiar) tem seus pixels igualados a zero a fim de focar o processamento apenas nas regiões de maior calor e potencialmente representando seres humanos. Objetos que emitem calor e possuem uma superfície relativamente grande podem ser detectados como humanos neste trabalho, e cujos pixels tem um valor acima de um limiar pré-definido.

Loong (2011) faz o reconhecimento de pessoas utilizando o sensor Kinect (ver Figura 3.5(a). Não há uma documentação mais detalhada disponível, porém é possível notar que objetos retangulares e em forma de humanos são considerados humanos. Loong (2011) disponibiliza um pacote de *software* para o ROS, que foi baixado e testado pelo autor deste projeto de mestrado e foram constatadas algumas deficiências durante a execução, já que alguns objetos não humanos são classificados como humanos. Uma das vantagens devido ao fato deste pacote ser de código aberto é a possibilidade de utilizá-lo e editá-lo para obter a melhora do seu desempenho.

Os autores Bonvin (2010) criaram um sistema que utiliza dois Kinects para verificar a moldura do corpo humano e fazer o rastreamento de pessoas, como pode ser conferido na Figura

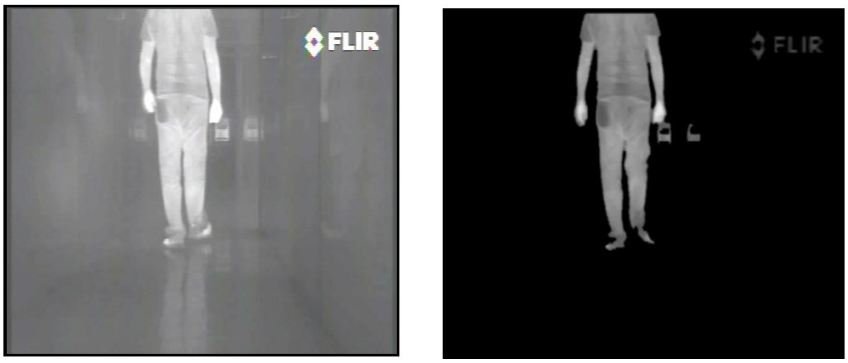


Figura 3.4: Segmentação da imagem utilizando a câmera térmica À direita imagem orginal e à esquerda imagem segmentada. Fonte: (Nin, 2011).

3.5(b). Na visualização das pessoas detectadas e rastreadas são utilizados paralelepípedos para representá-las.

(a) (b)

Figura 3.5: Detecção de pessoas utilizando o Kinect: (a) Detecção de seres humanos utilizando a nuvem de pontos 4D (x, y, z, cor) gerada pelo Kinect (Loong, 2011). (b) Detecção de seres humanos utilizando a nuvem de pontos 4D gerada pelo Kinect. Imagem adaptada de `http://www.youtube.com/watch?v=x--xlKWBTAE` (Bonvin, 2010).

Outros trabalhos que utilizam nuvem de pontos e mapas de profundidade gerados pelo Kinect mais recentes (já que o Kinect foi lançado há pouco mais de dois anos) estão em desenvolvimento (Hoshino e Morioka, 2011; Shotton *et al.*, 2011; Spinello e Arras, 2011; Xia *et al.*, 2011). Trabalhos que utilizam a câmera térmica são mais numerosos, uma vez que este tipo de sensor existe há mais tempo (Fernández-Caballero *et al.*, 2010; Suard *et al.*, 2006; Zhang *et al.*, 2007).

Com o propósito de evitar acidentes, um trabalho interessante e muito recente utiliza a fusão dos dois sensores utilizados neste projeto de mestrado (Kinect e FLIR) para a detecção e rastreamento de pessoas em um ambiente de mineração (Dickens e Wyk, 2011). Além disso, o sistema obtém a velocidade do veículo portador dos sensores e do humano à frente. A pessoa é detectada utilizando a câmera térmica, e o sensor de distância/profundidade é utilizado para calcular a trajetória e a velocidade do veículo e da pessoa. Uma vez detectado o risco de atropelamento um alarme é emitido.

3.4 Detecção de Regiões Navegáveis

Este projeto de pesquisa propõe a implementação de um sistema que irá navegar de forma autônoma e segura em um ambiente interno. É necessário, portanto, considerar alguns métodos para o reconhecimento de regiões navegáveis, de modo que não hajam colisões com os elementos do ambiente.

Nesta seção serão apresentados três trabalhos que envolvem a detecção de regiões navegáveis e a navegação de um veículo autônomo. Tais projetos podem ser adaptados para robôs de ambientes internos.

No trabalho de Shinzato (2010) foi desenvolvido, com o uso de redes neurais artificiais, e utilizando dados de uma câmera de vídeo monocular um sistema de detecção de regiões navegáveis. O resultado pode ser conferido na Figura 3.6. O autor utiliza um conjunto de classificadores (redes neurais) e em cada classificador um conjunto de atributos diferentes é utilizado. Ao final do processo é avaliado qual classificador tem melhor desempenho para que o resultado da classificação seja definido.

No entanto, este sistema foi testado mais exaustivamente com veículos em ambientes externos e em vias bem definidas (ruas pavimentadas e estradas de terra bem delimitadas). Alguns testes foram feitos em robôs móveis em ambientes fechados, obtendo resultados satisfatórios (Sales *et al.*, 2010).

Hata (2010) fez o mapeamento tridimensional do ambiente usando um sensor do tipo laser SICK (LIDAR) e criou mapas bidimensionais para efetuar a classificação de regiões quanto à sua navegabilidade. Ele utilizou redes neurais e SVM (do inglês *Support Vector Machine*) (Hearst, 1998) como classificadores. Na Figura 3.7 é apresentado o resultado do mapeamento juntamente com a classificação em navegável, não navegável e parcialmente navegável.

Outros trabalhos desenvolvidos neste laboratório se assemelham a este projeto de mestrado. Dois algoritmos distintos foram utilizados para a detecção de regiões navegáveis. O sensor utilizado foi uma câmera estéreo e devido a natureza dos dados ser a nuvem de pontos 4D

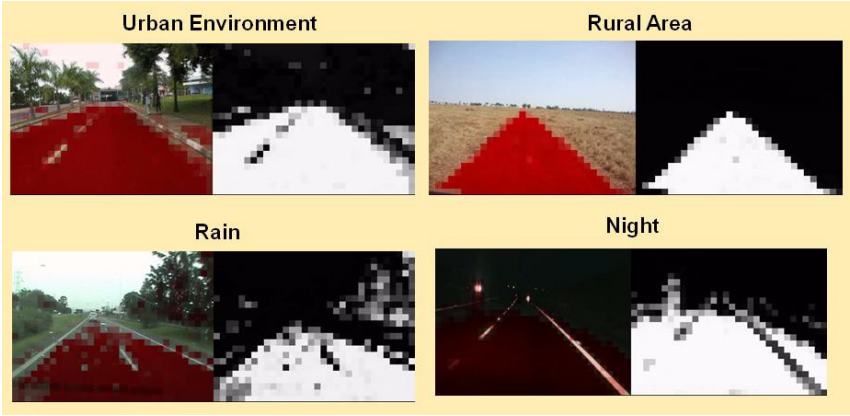


Figura 3.6: Mapa de navegabilidade obtido a partir de imagens com o uso de classificadores neurais – LRM (ICMC-USP). Em cada par de fotos as regiões em vermelho são classificadas como navegável na foto à esquerda e à direita o branco é navegável e o preto não navegável. Fonte: (Shinzato, 2010).

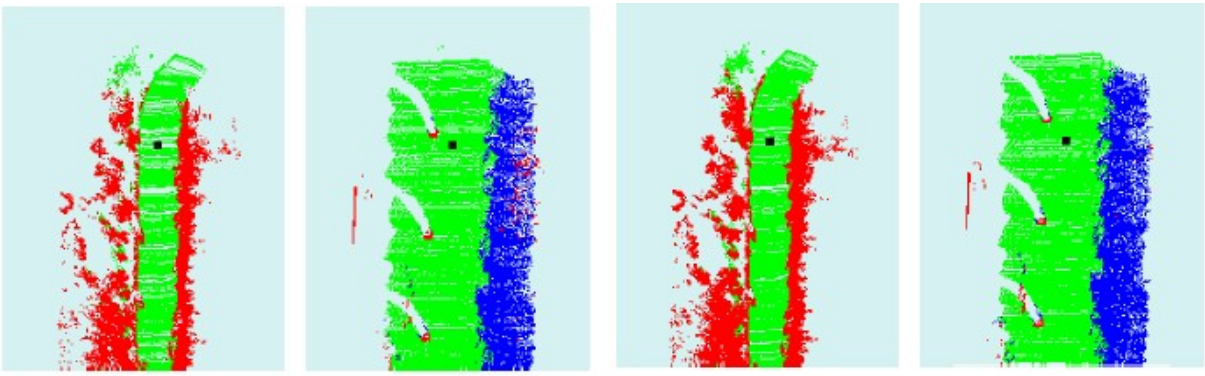


Figura 3.7: Quatro imagens correspondentes à classificação de terrenos quanto à navegabilidade. As duas imagens à esquerda correspondem à classificação utilizando redes neurais e as outras utilizando SVM. As regiões verdes na imagem representam as regiões navegáveis no mundo real, as regiões azuis representam as parcialmente navegáveis e as vermelhas as não navegáveis – LRM (ICMC-USP). Fonte: (Hata, 2010).

(coordenadas x, y e z e cor) verificou-se a possível utilização dos algoritmos utilizando o sensor Kinect. No primeiro trabalho (Mendes e Wolf, 2011; Mendes, 2012) é utilizado o algoritmo RANSAC (Fischler e Bolles, 1987; Konolige *et al.*, 2006) para detectar planos. Estes planos representam a área navegável.

Na Figura 3.8 é visualizado o resultado do trabalho de Mendes e Wolf (2011). Na imagem é apresentado o desvio de obstáculos utilizando o algoritmo VFH (do inglês *Vector Field Histogram*) (Borenstein *et al.*, 1991) para o esterçamento e o método RANSAC para a detecção das regiões navegáveis, as quais são representadas em verde na imagem superior. As regiões em

vermelho representam regiões não navegáveis (não plano). Na parte inferior da imagem há uma representação do VFH em funcionamento. A linha verde representa a posição dos obstáculos na imagem superior. A linha vermelha é um limiar usado para decidir o que deve ser considerado como obstáculo no caminho do robô e, portanto, deve ser evitado. A linha azul define a orientação que o veículo deve tomar a fim de evitar o obstáculo.

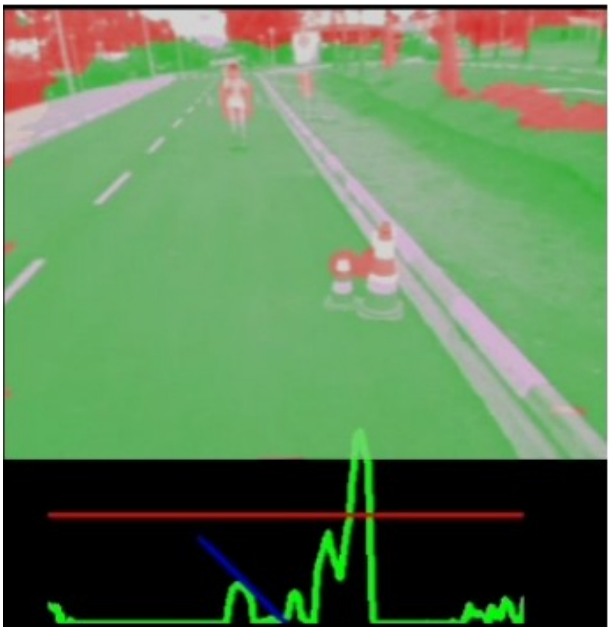


Figura 3.8: Desvio de obstáculos utilizando o algoritmo VFH para o esterçamento e o método RANSAC para a detecção das regiões navegáveis, as quais são representadas em verde. Fonte: (Mendes e Wolf, 2011).

Na segunda implementação, do mesmo autor, é utilizado um outro algoritmo para detectar planos, descrito em Talukder *et al.* (2002). Em ambos os trabalhos o objetivo é que o carro siga um ponto de GPS conseguindo desviar de obstáculos. Para isso precisa encontrar regiões navegáveis e utiliza o algoritmo VFH para desviar dos obstáculos efetuando o esterçamento do carro.

3.5 Considerações

Este trabalho de pesquisa envolve o desenvolvimento do sistema computacional usado em um robô móvel de segurança para ambiente interno, o qual deverá ser capaz de navegar de forma segura, desviando de obstáculos e detectando anomalias e intrusos no ambiente. Neste capítulo foram apresentados, portanto, trabalhos referentes à robótica de vigilância, à detecção de humanos e à detecção de vias navegáveis.

No capítulo 4 serão apresentados detalhes da metodologia de desenvolvimento do sistema proposto, que está baseada nas ferramentas e métodos que foram estudados previamente e nos trabalhos apresentados neste capítulo.

CAPÍTULO

4

Desenvolvimento e Resultados

O sistema de navegação e detecção de intrusos foi dividido em módulos executados separadamente. Os principais módulos desenvolvidos são Detecção de Humanos no Ambiente (descrita na seção 4.1), Navegação Reativa com Detecção e Desvio de Obstáculos (descrita na seção 4.2) e, por fim, Detecção de Estados e Transições junto ao Mapa Topológico (Navegação Topológica), na seção 4.3. Na seção 4.4 são apresentados alguns trabalhos complementares do autor que poderão ser agregados aos sistema robótico em trabalhos futuros.

A Figura 4.1 apresenta os módulos principais e os secundários que integram o sistema de vigilância proposto. Alguns destes módulos fazem parte de trabalhos desenvolvidos por outros integrantes do laboratório LRM e que deverão futuramente ser integrados ao trabalho desta dissertação. O módulo de Localização (item 1 na Figura 4.1) está relacionado aos trabalhos de localização *indoor* desenvolvidos por Pessin (2012, 2013) (utilizando redes sem fio), Couto (2012) (utilizando imagens), e Sales (2012a) (utilizando o mapa topológico). O trabalho desenvolvido por Sales (2012a) abordou a utilização de mapas topológicos e o desenvolvimento de um sistema de navegação topológica usando autômatos finitos, e que correspondem aos itens 2 e 3 na Figura 4.1.

Esta dissertação abrange os itens 3, 4 e 5 da Figura 4.1. O item 3 referente à navegação topológica é tratado na seção 4.3, sendo que foi adotada uma abordagem inspirada nos trabalhos de Sales (2012a). O item 4 referente à navegação reativa é tratado na seção 4.2. O item

5 referente à detecção de intrusos é tratado na seção 4.1. Além disto, foram realizados trabalhos complementares referentes aos itens 6, 7 e 8 na Figura 4.1 que tratam respectivamente da tele-operação, do *follow me* e do reconhecimento de gestos, sendo apresentados na seção 4.4.

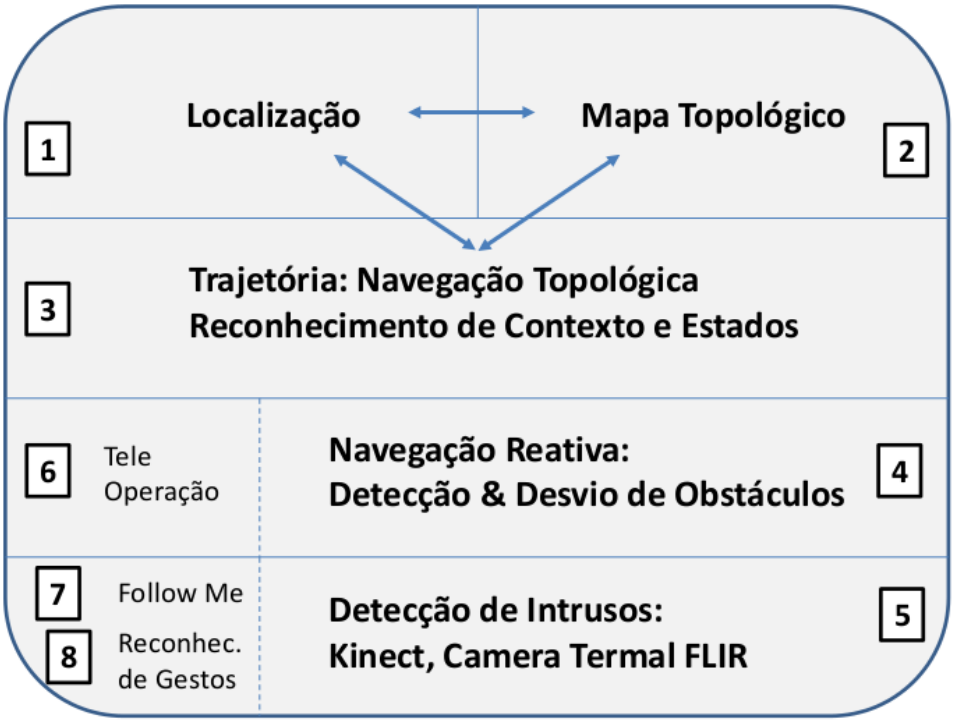


Figura 4.1: Proposta do sistema de vigilância. Nesta representação são apresentados os módulos do sistema e outros módulos que poderão ser agregados ao sistema.

O sistema de vigilância proposto deve agregar um módulo de navegação capaz de se localizar, utilizar o mapa topológico e navegar desviando de obstáculos. Este é o subsistema de patrulhamento do ambiente representado na Figura 4.2. O módulo de navegação reativa e topológica foram implementados junto a este trabalho. Em paralelo um outro subsistema é executado, sendo este o subsistema de detecção de intrusos apresentado na Figura 4.2, também implementado junto a este trabalho. A fim de realizar a detecção de intrusos é necessária a fusão dos sensores Kinect e câmera térmica, necessitando também de uma calibração prévia destes sensores.

Uma vez detectado um intruso é possível realizar diversas ações, entre elas enviar uma mensagem a uma central de operação notificando o evento. O operador pode então fazer o uso dos módulos de tele-operação e *follow me* para tomar o controle do robô ou seguir o intruso, respectivamente.

Para o desenvolvimento dos módulos do sistema foram feitos experimentos utilizando as ferramentas Player/Stage, OpenCV, ROS, o robô Pioneer e os sensores de profundidade (Kinect)

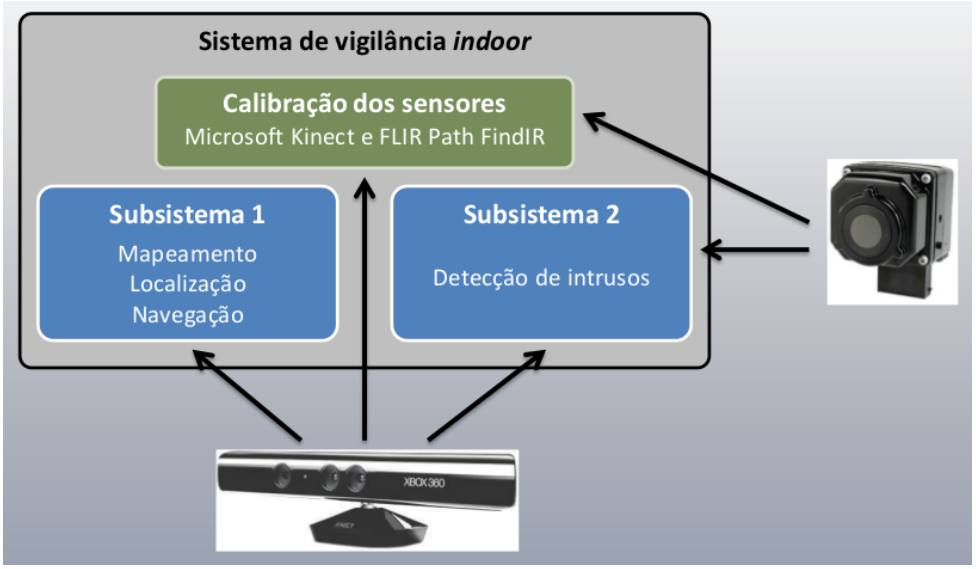


Figura 4.2: Subsistemas do sistema de vigilância.

e de calor (câmera FLIR TAU 640). Foram também utilizados métodos de aprendizado de máquina e de processamento de imagens. Detalhes sobre as ferramentas utilizadas podem ser conferidos na seção 2.5).

Uma das informações que o Kinect provê e que foi utilizada nos experimentos descritos nesta seção foi o mapa de profundidade, uma imagem de 640x480 pixels onde cada pixel fornece a informação de distância em mm (milímetros). A informação da câmera térmica utilizada nos experimentos também descrito nesta seção fornece uma imagem de 640x480 pixels representando o calor no ambiente.

Além das implementações e testes com *software* e *hardware* foi também desenvolvido, com a colaboração de outros mestrandos do mesmo laboratório, um estabilizador de tensão de 12 volts com a finalidade de alimentar o Kinect utilizando uma bateria externa ou utilizando diretamente as baterias do robô. Com este dispositivo o robô pode se deslocar livremente no ambiente sem a utilização de cabos de energia ligando-o a uma tomada fixa.

4.1 Detecção de Pessoas no Ambiente

Os sensores Kinect e FLIR (câmera térmica) foram utilizados em conjunto para a tarefa de detecção de humanos. A informação conjunta dos sensores foi utilizada com a finalidade de aumentar a confiabilidade dos resultados. A Figura 4.3 ilustra os passos realizados para a tarefa de detecção de pessoas. Primeiramente os sensores são calibrados apenas uma vez. Os próximos passos são executados constantemente conforme segue: inicialmente são obtidos os dados

dos sensores de distância e de calor. Estes dados são segmentados para descartar informação que não representa uma pessoa na cena. É realizado um cálculo de histograma para fazer o posicionamento horizontal da pessoa detectada em cada uma das imagens. Por fim é realizada uma análise conjunta dos histograma a fim confirmar a detecção de uma pessoa. Estes passos serão explicados com maiores detalhes nas próximas subseções.

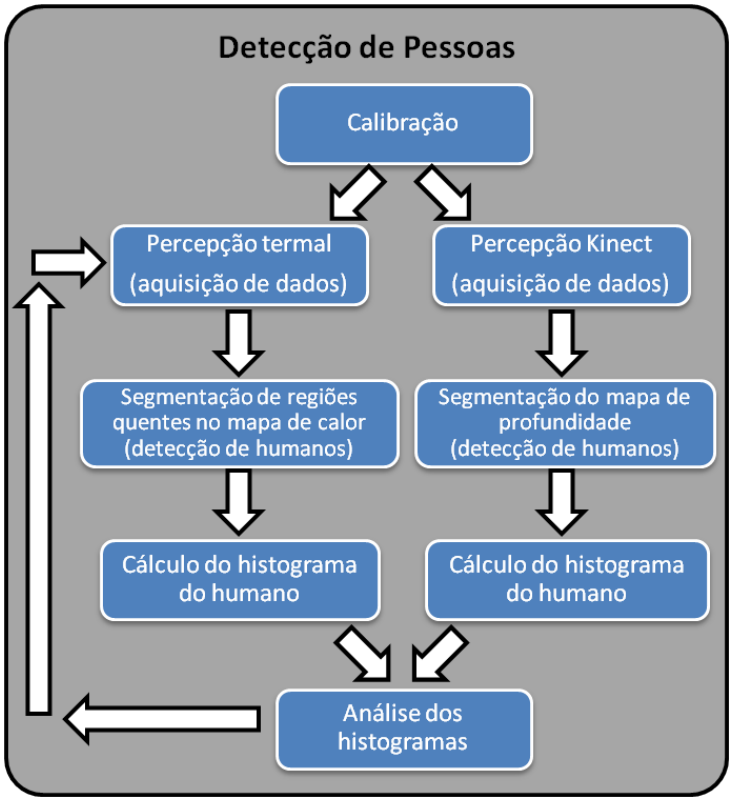


Figura 4.3: Processo de detecção de intrusos utilizando os sensores Kinect e FLIR.

4.1.1 Calibração de Sensores

A integração das informações (termal e profundidade) por meio da calibração dos dois sensores visa buscar uma correspondência aproximada entre as imagens fornecidas pelos dois sensores e assim aumentar o campo de visão e percepção do robô, já que cada sensor possui limitações. O Kinect tem maior campo de visão (maior ângulo de abertura das lentes), porém a distância de detecção dos objetos é limitada. A câmera térmica, por sua vez, tem menor campo de visão, porém percebe informações a distâncias bem maiores do que o Kinect . A área de abrangência horizontal dos dois sensores é representada na Figura 4.4 e a disposição dos sensores na plataforma robótica é exibida na Figura 4.5. Esta configuração é utilizada para a calibração, a navegação e a detecção de pessoas.

Característica / Sensor	Térmica	Kinect
Resolução da imagem	640 x 480 pixels	640 x 480 pixels
Taxa de frames	7.5 fps	30 fps
Ângulo de abertura da lente	36° H x 27° V	58° H x 45° V

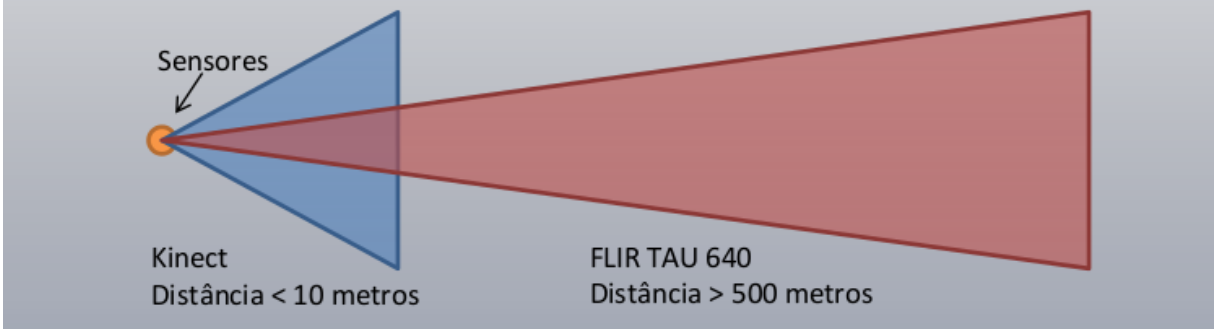


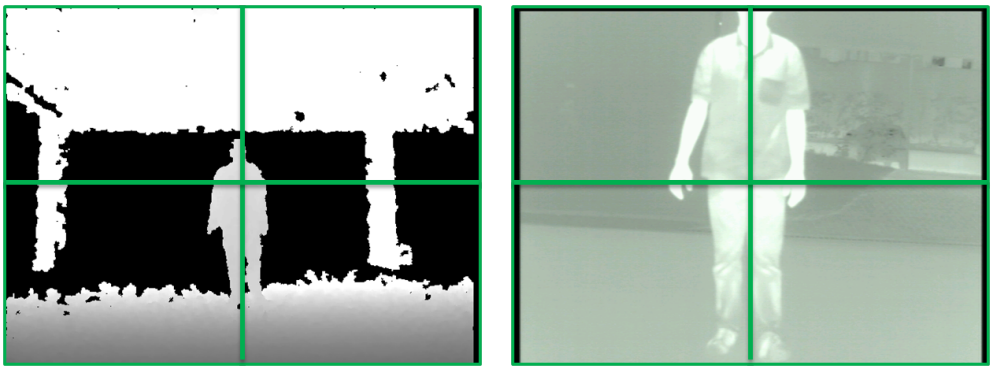
Figura 4.4: Vista superior do campo de visão horizontal dos sensores. Vermelho representa campo de visão da térmica e azul campo de visão do Kinect. A bola laranja indica a posição dos sensores.

Figura 4.5: Disposição dos sensores Kinect e térmica sobre a plataforma robótica. A câmera térmica FLIR TAU 640 está posicionada sobre o Kinect.

A calibração é um processo que é realizado apenas uma vez antes de iniciar a navegação (vigília) do robô a fim relacionar as informações dos dois sensores de tal forma que as imagens dos dois sejam correspondentes a mesma cena do ambiente. Foi optado por um método mais simples de calibração, chamado aqui de calibração horizontal ou alinhamento horizontal de

sensores. A calibração foi realizada de forma interativa, utilizando um programa onde o usuário visualiza uma marcação de referência e ajusta manualmente a posição e orientação dos sensores de forma a alinhar o centro das imagens.

O alinhamento dos sensores é realizado pelo usuário baseado nas informações que são exibidas contendo as imagens dos dois sensores com duas semi-retas desenhadas em cada, formando um alvo no centro das imagens (ver figura 4.6). Faz-se então a inspeção visual nas imagens e procura-se alinhar os sensores utilizando um ponto central no ambiente, por exemplo, a mão de uma pessoa ou até mesmo o corpo humano como referência. Desta forma obtém-se um alinhamento vertical e horizontal aproximado, uma vez que os campos de visão dos sensores são diferentes. Os sensores devem estar fixos um ao outro ou então ambos em uma plataforma rígida, de forma que não percam a calibração durante a movimentação do robô.

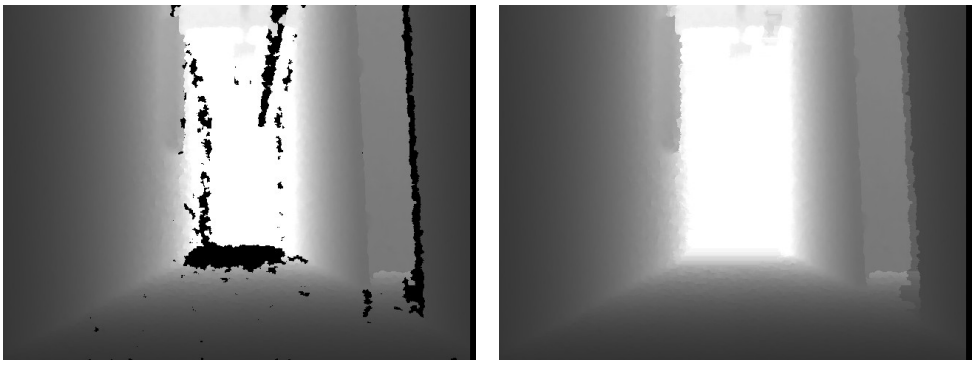


(a) (b)

Figura 4.6: Imagens utilizadas na calibração horizontal dos dois sensores. Imagens dos sensores (a) Kinect e (b) câmera térmica utilizadas para a calibração horizontal.

4.1.2 Pré-processamento e segmentação das imagens

O sensor Kinect é utilizado para duas tarefas: a navegação e a detecção de humanos. Uma vez que cada tarefa tem um uso diferente dos sensores então para cada tarefa foi realizado um processamento diferente das imagens. Para a navegação do robô foi tratado o ruído. No Kinect foi chamado de ruído regiões do mapa de profundidade (tons de cinza) que estão representados na cor preta, onde estes pontos do ambiente indicam que o sensor não foi capaz de estimar a sua distância. Para tanto foi implementada uma função usada para suprimir as regiões em que o Kinect não foi capaz de estimar a distância, como mostrado na Figura 4.7. Esta função processa a imagem, de modo que para cada reta horizontal desta imagem sem informação (pixel de cor preta na Figura 4.7(a)), calcula-se a média dos valores do primeiro pixel à sua direita que possua uma informação válida (diferente de preto) e o valor do primeiro pixel à sua esquerda com informação válida. Preenche-se então a região preta com a média calculada obtendo a imagem apresentada na Figura 4.7(b) na qual todos os pixels possuem um valor associado que representa a distância (representada através dos tons de cinza).



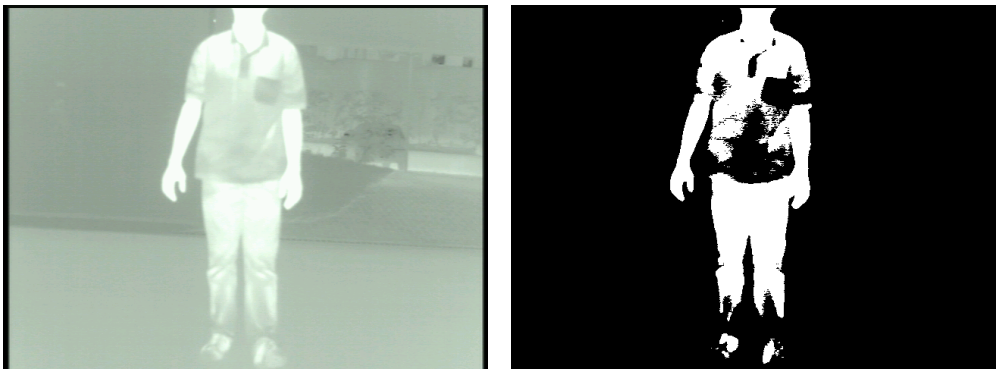
(a) (b)

Figura 4.7: Mapa de profundidade gerado pelo Kinect: (a) Imagem original com ruído e (b) Imagem após a remoção de ruído.

Para a realização da tarefa de detecção de pessoas o mapa de profundidade do Kinect foi processado para eliminar a informação de ambiente e considerar apenas a porção da imagem correspondente a um humano, realizando assim a segmentação do objeto de interesse (pessoa). Este processamento foi realizado em duas fases. A primeira teve como suporte o *software Human Tracker* do *framework/driver* OpenNI, na qual um ou vários humanos são identificados. A outra parte foi realizada para retirar informações da imagem que não representam o(s) humano(s), como ilustrado na Figura 4.8, que é a segmentação.

(a) (b)

Figura 4.8: Imagem do sensor Kinect: (a) Mapa de profundidade tons de cinza, (b) Segmentação extraindo apenas a informação que representa o humano na cena.



(a) (b)

Figura 4.9: Imagem da câmera térmica: (a) Original em tons de cinza, (b) Com pré-processamento utilizando o filtro de limiarização comum.

Com relação à câmera térmica, utilizada na detecção de pessoas, a quantidade de ruído é baixa e pode ser considerada desprezível. Porém, é desejado que seja feita a segmentação, processo este que irá gerar uma imagem com regiões de interesse, descartando o que não é considerado um humano/intruso na imagem (regiões frias). A Figura 4.9(a) apresenta a imagem

em tons-de-cinza. Na Figura 4.9(b) a imagem recebeu um filtro de limiar comum descartando assim as regiões mais frias.

As imagens apresentadas (Figura 4.9) demonstram os resultados de pré-processamento para representar um ser humano pelo calor gerado pelo mesmo. Estas imagens serão utilizadas para melhorar os resultados de detecção de intrusos utilizando o sensor Kinect.

4.1.3 Processamento

Neste ponto da tarefa de detecção de pessoas as imagens já estão segmentadas (imagem em preto-e-branco) e é possível afirmar se há ou não humano nas imagens de cada sensor. São consideradas quatro situações: um ser humano pode estar oculto para ambos os sensores, pode ser detectado apenas pela FLIR, pode ser detectado apenas pelo Kinect, ou pode ser detectado por ambos os sensores. Os casos em que uma pessoa é detectada por apenas um dos sensores ocorre quando o humano está fora do campo de visão do outro sensor. Esta situação ilustra bem uma das motivações do uso dos dois sensores, já que desta forma há um aumento do campo de visão do sistema robótico.

Quando ambos os sensores detectam um ser humano é necessário verificar se o resultado é consistente. Para isso é necessário calcular o quanto uma pessoa está deslocada para um dos lados na imagem de cada sensor. Este cálculo foi realizado criando e analisando um histograma, ou seja, obtendo a contagem do número total de pixels brancos para cada coluna da imagem. É importante destacar que o cálculo dos histogramas é realizado sobre as imagens, de distância obtida pelo Kinect e de calor obtida pela câmera térmica, que foram pré-processadas e estão binarizadas (preto-e-branco).

A contagem dos pixels brancos por coluna da imagem representa a posição horizontal do humano na imagem segmentada. A Figura 4.10 apresenta um exemplo do histograma com a contagem dos pixels brancos para cada coluna da imagem, onde pode ser constado que os picos do histograma ocorrem na região central para a Figura 4.10(a) e mais à direita para a Figura 4.10(c), onde o humano identificado pela segmentação se encontra, respectivamente, no centro e à direita.

Na Figura 4.11 são apresentadas as quatro situações de detecção, considerando ambos sensores, usando o método descrito acima. Nestes exemplos, os histogramas que representam pessoas foram apresentados no centro da imagem, exemplificando uma pessoa na parte central da cena. O histograma pode, de acordo com a situação real, estar deslocado para direita ou para esquerda.

A imagem do histograma é dividida em 10 seções verticais e são analisadas da esquerda para a direita com a finalidade de determinar em qual seção se encontra o maior valor da contagem de

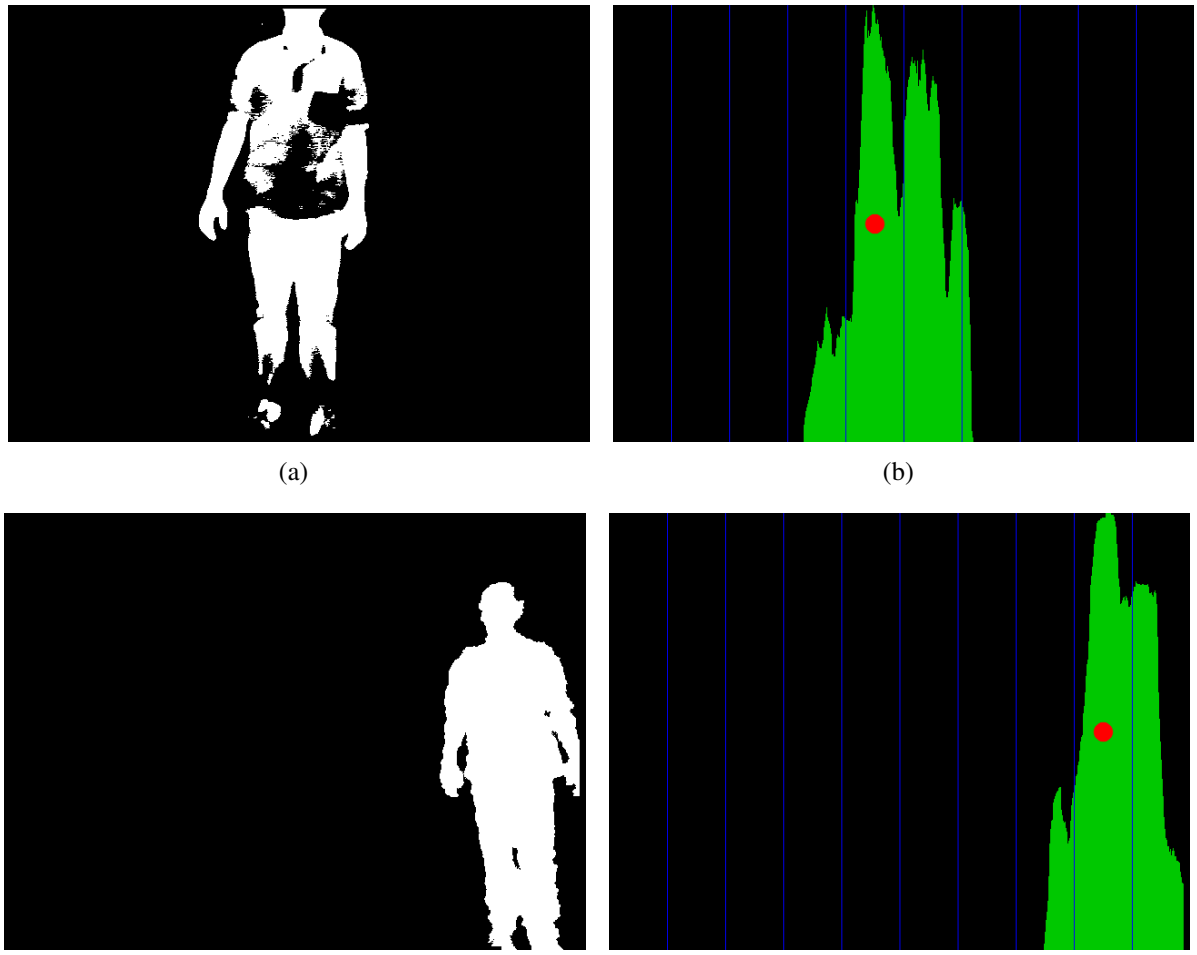


Figura 4.10: Exemplos do uso do histograma. Imagens (a) da térmica segmentada, (b) do histograma, (c) do Kinect segmentada, e (d) do histograma.

pixels brancos. A seção da imagem do histograma que possui o maior valor é marcada com uma bola vermelha no centro para representar que aquele é o ponto central do ser humano na imagem do sensor. Desta forma, é possível localizar horizontalmente o humano detectado em cada imagem de cada sensor e depois comparando ambas as informações que definem a localização do humano nas imagens, determinando assim a correspondência do humano detectado nas duas imagens.

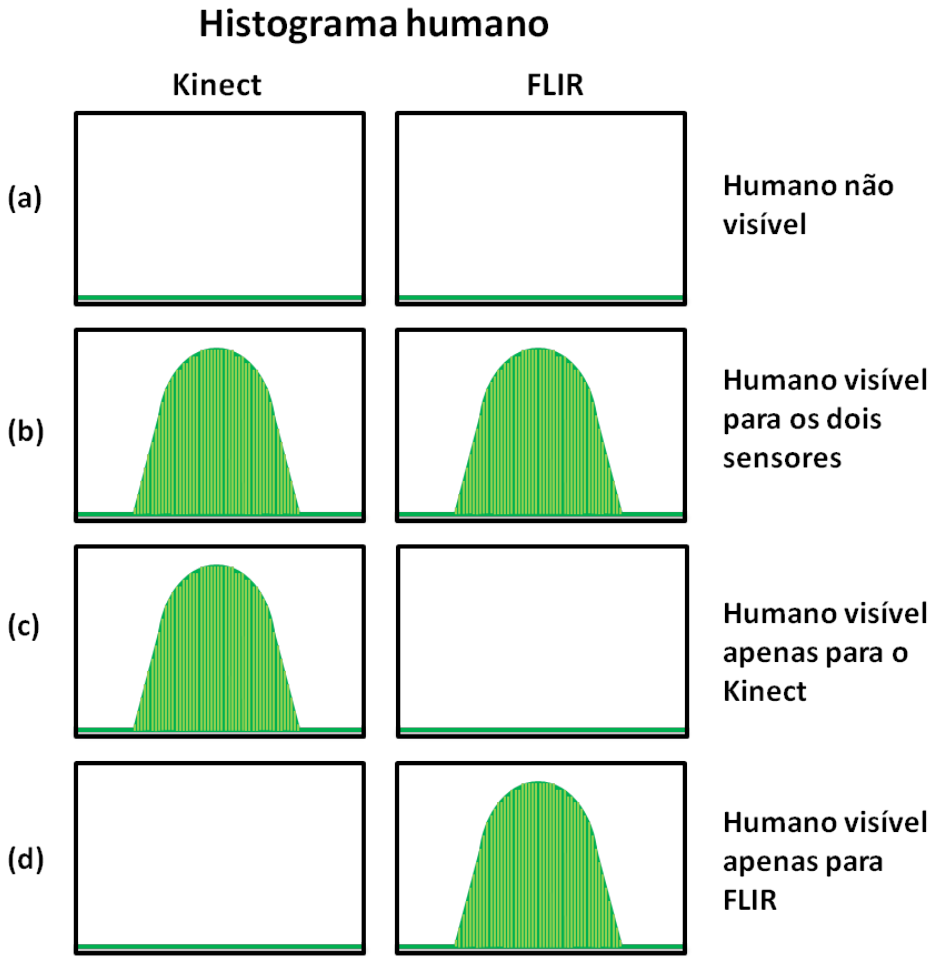


Figura 4.11: Casos de histogramas de representação de humanos na imagem.

Nas Figuras 4.12, 4.13, 4.14, 4.15 são apresentados quatro situações reais da detecção de pessoas, na mesma ordem das situações apresentadas na Figura 4.11.

A Figura 4.12 apresenta as imagens originais dos sensores (à direita) e as imagens segmentadas (à esquerda), confirmando que não há pessoas no campo de visão de ambos os sensores

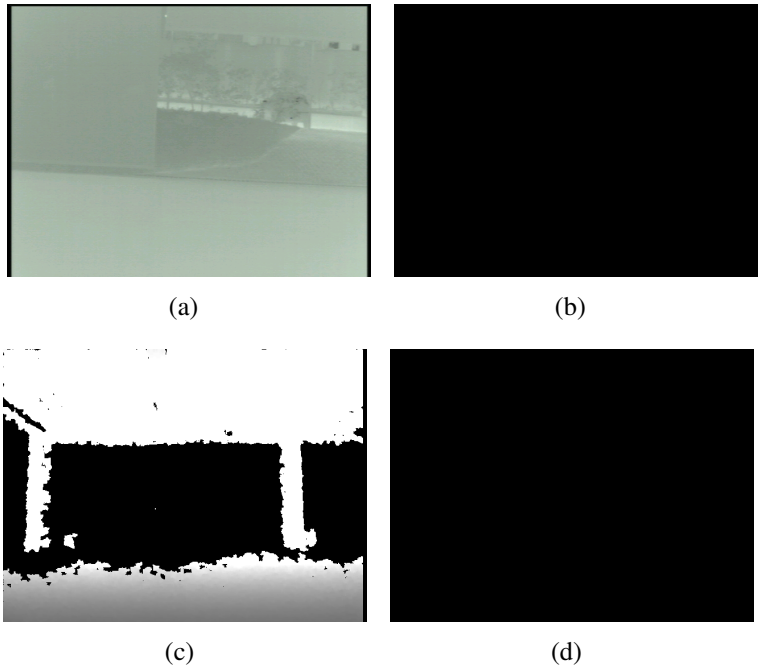
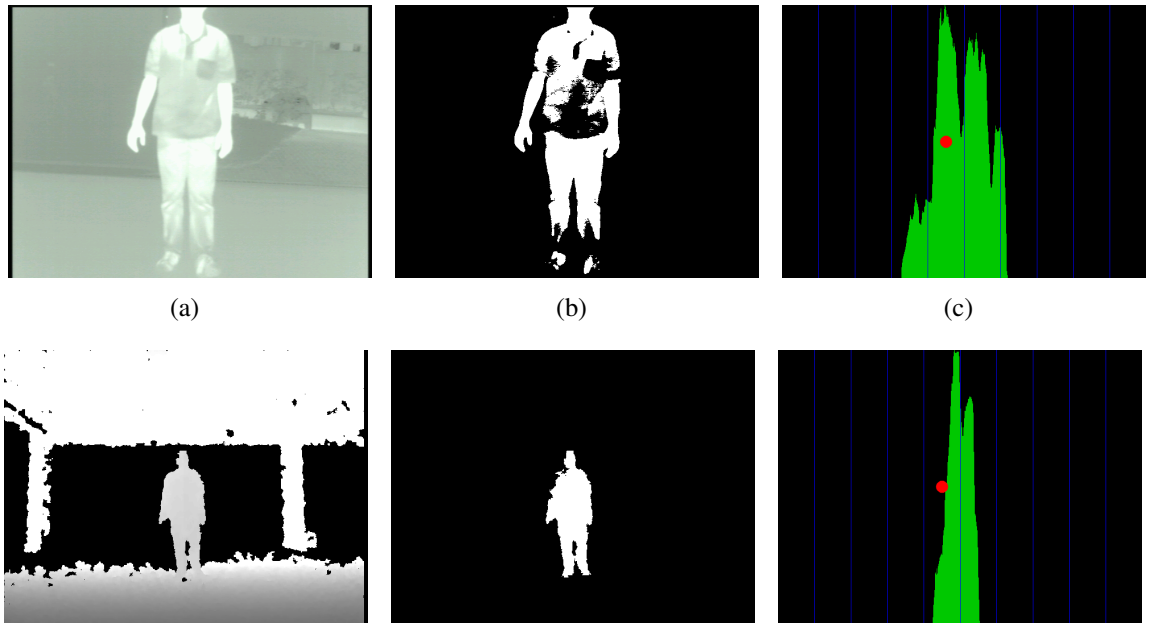


Figura 4.12: Resultado sem humanos. Imagens (a) da térmica original, (b) da térmica segmentada, (c) do Kinect original, e (d) do Kinect segmentada.

A Figura 4.13 apresenta a situação em que uma pessoa está dentro do campo de visão dos dois sensores. A esquerda são apresentadas as imagens originais da FLIR e do Kinect. No centro as imagens segmentadas e à direita a imagem do histograma dividida em 10 seções e com a bola vermelha representando a porção do humano na imagem dos dois sensores.

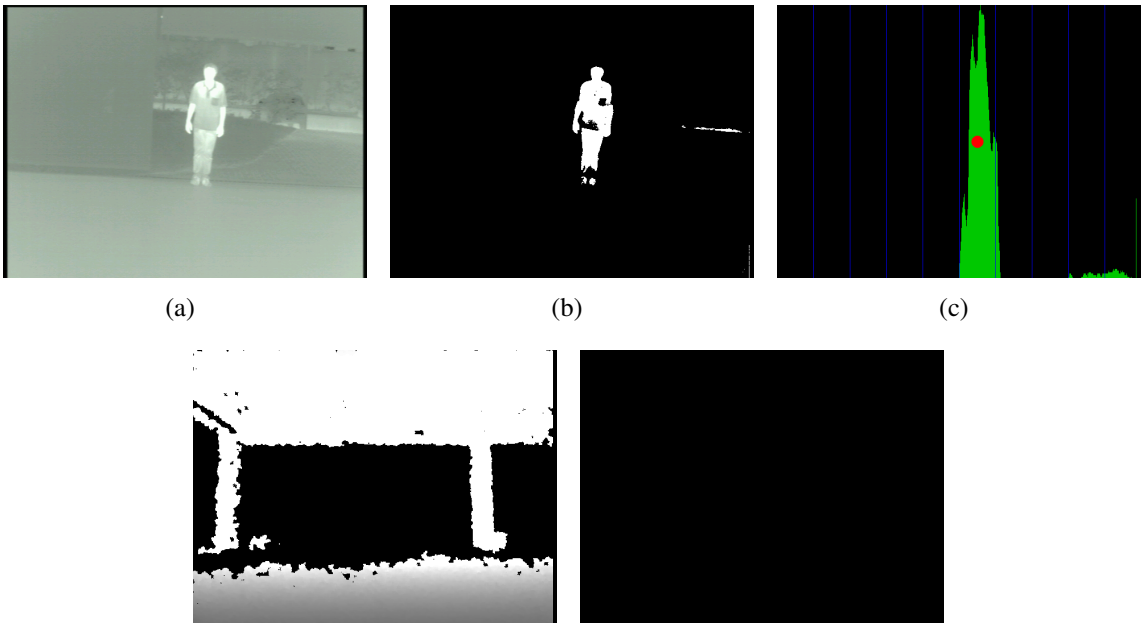


(a) (b) (c)

(d) (e) (f)

Figura 4.13: Resultado com humano dentro do campo de visão de ambos os sensores. Imagens (a) da FLIR original, (b) da FLIR segmentada, (c) do histograma do ser humano na imagem segmentada da FLIR representando com uma bola vermelha a porção horizontal da imagem que se encontra o humano, (d) do Kinect original, (e) do Kinect segmentada, e (f) do histograma do ser humano na imagem segmentada do Kinect representando com uma bola vermelha a porção horizontal da imagem que se encontra o humano.

A Figura 4.14 apresenta a situação em que uma pessoa está dentro do campo de visão apenas da câmera térmica, significando que está longe o suficiente para não ser detectada pelo Kinect. Neste caso é gerado o histograma apenas para a imagem da câmera térmica. A bola vermelha representa a porção horizontal que se encontra o humano na imagem térmica.

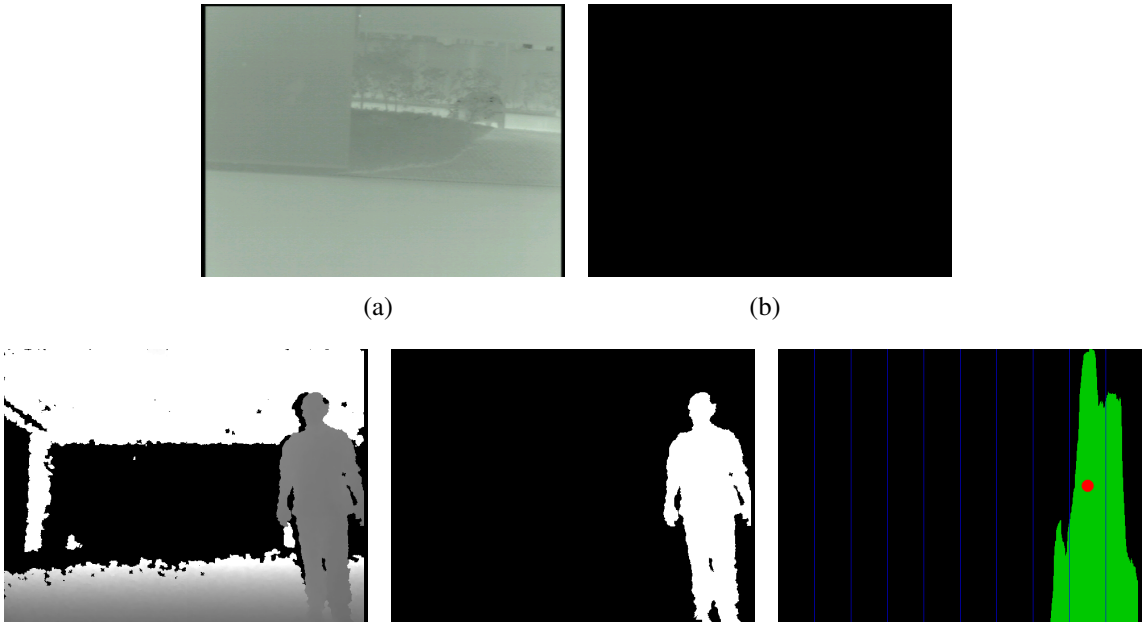


(a) (b) (c)

(d) (e)

Figura 4.14: Resultado com humano apenas no campo de visão da FLIR. Imagens (a) da FLIR original, (b) da FLIR segmentada, (c) do histograma do ser humano na imagem segmentada da FLIR representando a porção horizontal que se encontra o humano na imagem térmica, (d) do Kinect original, e (e) do Kinect segmentada e sem humanos.

A Figura 4.15 apresenta a situação em que uma pessoa está dentro do campo de visão apenas do Kinect, o que significa que está próximo dos sensores, porém afastado lateralmente não sendo visível para a térmica devido ao seu campo de visão inferior. Neste caso é gerado o histograma apenas para o Kinect mostrando a porção da imagem em que se encontra o humano. É possível perceber que o humano está na região periférica do robô e não sendo visível pela câmera térmica.



(a) (b)

(c) (d) (e)

Figura 4.15: Resultado com humano visualizado apenas pelo Kinect. Imagens (a) da FLIR original, (b) da FLIR segmentada e sem humanos, (c) do Kinect original, (d) do Kinect segmentada, e (e) do histograma do ser humano na imagem segmentada do kinect, representando com a bola vermelha a porção que se encontra o humano na imagem.

Por enquanto foi considerado um ambiente sem objetos quentes e com apenas uma pessoa para facilitar o processamento e a detecção do ser humano. Trabalhos futuros poderão envolver o processamento com mais de um objeto quente, além de detectar várias pessoas. Outro trabalho futuro previsto é a possibilidade de usar uma base do tipo “pan-tilt”, para mover a câmera térmica de modo a tentar orientá-la em direção ao objeto quente (enquadrar o objeto atualmente fora de seu campo de visão), e assim confirmar a sua presença humana indicada pelo Kinect.

4.1.4 Avaliação dos resultados da detecção de pessoas

Com o objetivo de validar a detecção de humanos através do uso dos sensores Kinect e câmera térmica foram realizados os experimentos, primeiramente com o Kinect, depois com a câmera térmica, e com ambos dispositivos sendo usados ao mesmo tempo. A coleta da base de dados foi feita obtendo-se simultaneamente 500 pares de imagens, uma para cada sensor. Cada imagem foi analisada e anotada como sendo uma imagem com a presença ou ausência de humano na cena.

Inicialmente as imagens do Kinect foram processadas para realizar a detecção de humanos e obteve-se uma matriz de confusão (Rezende, 2003) representando os acertos, erros, falsos positivos e falsos negativos relativos a classificação das imagens. A Figura 4.16 apresenta a matriz de confusão e a Figura 4.17 apresenta a taxa de acertos e erros obtida com o sensor Kinect.

Matriz de confusão - Resultado sensor Kinect		
esperado/obtido	sim kinect	não kinect
sim kinect	319	126
não kinect	2	53
	Total	500

Figura 4.16: Matriz de confusão da classificação da imagem do Kinect.

	soma	acertos	erros	% acerto	% erro
Presente	445	319	126	71,69%	28,31%
Ausente	55	53	2	96,36%	3,64%
Total	500	372	128	74,40%	25,60%

Figura 4.17: Taxa de acertos e erros da classificação da imagem do Kinect.

Posteriormente as imagens da câmera termal foram processadas para realizar a detecção de humanos e obteve-se uma matriz de confusão representando os acertos, erros, falsos positivos e falsos negativos relativos a classificação das imagens. A Figura 4.18 apresenta a matriz de confusão e a Figura 4.19 apresenta a taxa de acertos e erros obtida com o sensor termal.

Por fim, gerou-se uma matriz de confusão única representando quatro classes independentes de resultados, a saber: ausência de humano em ambos os sensores (não Kinect, não térmica), presença no Kinect e ausência na térmica (sim Kinect, não térmica), ausência no Kinect e presença na térmica (não Kinect, sim térmica), e presença em ambos os sensores (sim Kinect, sim térmica). É importante destacar que nesta matriz de confusão foram consideradas quatro classes independentes, ou seja, os sensores foram considerados de forma separada e sem uma

Matriz de confusão - Resultado Sensor térmica		
esperado/obtido	sim térmica	não térmica
sim térmica	396	0
não térmica	12	92
	Total	500

Figura 4.18: Matriz de confusão da classificação da imagem da câmera termal.

	soma	acertos	erros	% acerto	% erro
Presente	396	396	0	100,00%	0,00%
Ausente	104	92	12	88,46%	11,54%
Total	500	488	12	97,60%	2,40%

Figura 4.19: Taxa de acertos e erros da classificação da imagem da câmera termal.

fusão dos resultados dos mesmos. A matriz de confusão representando os acertos, erros, falsos positivos e falsos negativos relativos a classificação das imagens. A Figura 4.20 apresenta a matriz de confusão e a Figura 4.21 apresenta a taxa de acertos e erros obtida com o sensor termal. A taxa de erros indica, portanto, quando um dos sensores foi classificado de forma incorreta.

Matriz de Confusão - Resultado sensores kinect e térmica				
esperado/obtido	não kinect, não térmica	sim kinect, não térmica	não kinect, sim térmica	sim kinect, sim térmica
não kinect, não térmica	44	0	2	2
sim kinect, não térmica	4	44	0	8
não kinect, sim térmica	0	0	7	0
sim kinect, sim térmica	0	0	122	267
			Total	500

Figura 4.20: Matriz de confusão da classificação da imagem do Kinect e da câmera termal.

	soma	acertos	erros	% acerto	% erro
não Kinect, não térmica	48	44	4	91,67%	8,33%
sim Kinect, não térmica	56	44	12	78,57%	21,43%
não Kinect, sim térmica	7	7	0	100,00%	0,00%
sim Kinect, sim térmica	389	267	122	68,64%	31,36%
Total	500	362	138	72,40%	27,60%

Figura 4.21: Taxa de acertos e erros da classificação da imagem do Kinect e da câmera termal.

Analisando as matrizes de confusão apresentadas acima constata-se que o Kinect isoladamente tem uma taxa de acerto de $74,4\%$ e o sensor termal tem uma taxa de acerto de $97,6\%$. Portanto, o Kinect apresenta um erro maior, porém, possui um campo de visão mais aberto, por outro lado, a câmera térmica possui uma taxa de acerto superior e um campo de visão mais fe-

chado. Em função disto, justamente surge o interesse de unir ambos os sensores para a detecção de humanos.

A partir da Figura 4.20, se considerarmos apenas duas classes (presença e ausência de humano) unindo as respostas de ambos os sensores obtém-se uma taxa de acerto de $91,67\%$ para ausência de humanos (44 acertos e 4 erros por falso positivo) e uma taxa de acerto de $99,12\%$ para presença de humanos (448 acertos e 4 erros por falso negativo). Concluindo, a taxa geral de acertos para ambas as classes é de $98,4\%$.

Os resultados acima demonstram que o desempenho de ambos os sensores é superior ao desempenho obtido com os sensores utilizados individualmente. Em particular, o desempenho do Kinect é bastante beneficiado pelo uso da câmera térmica.

4.2 Navegação Reativa com Detecção e Desvio de Obstáculos

Como mencionado anteriormente (ver capítulo 1), o objetivo deste trabalho foi criar um robô de vigilância. Ele é capaz de navegar por um ambiente *indoor* desviando de obstáculos e detectando intrusos. Nesta seção será descrita a maneira como foi desenvolvida a navegação do robô no ambiente.

Para a realização dos movimentos do robô foi utilizado o comportamento reativo, no qual um robô recebe informações dos sensores e reage gerando comandos para os atuadores. Utilizando o mapa de profundidade gerado pelo Kinect (mapa métrico local) e utilizando a biblioteca OpenCV foi desenvolvido o módulo de navegação reativa no qual o robô se desloca no ambiente evitando colisões à sua frente.

Para efetuar a detecção de obstáculos pelo robô foram analisadas porções da imagem a frente do robô (mapa de profundidade) para definir qual ou quais segmentos de imagens que apresentam elementos que estão muito próximos ao robô. A imagem foi dividida em 5 seções verticais das quais três delas (esquerda, frente e direita) foram analisadas com relação à intensidade do pixel a fim de determinar as distâncias mínimas e máximas absolutas do sensor até os elementos e obstáculos do ambiente. Quando a distância mínima de uma seção é inferior a 60 cm então é considerado que naquela seção de imagem há um obstáculo e o robô deve reagir a esta situação. Considerou-se a distância de 60 cm, pois abaixo desta distância o sensor Kinect não apresenta um cálculo de distância confiável.

Foram definidas 8 situações que são descritas na Figura 4.22 e quatro comandos de velocidade para o robô: *FRENTE*, *DIREITA*, *ESQUERDA* e *PARADO*. No primeiro tipo de comando,

FRENTE, é setada uma velocidade linear pré-definida para frente. Os outros dois comandos, *DIREITA* e *ESQUERDA*, geram uma velocidade angular pré-estabelecida na qual o robô gira no próprio eixo para direita e para a esquerda, respectivamente. O último comando seta a velocidade zero.

O comando *FRENTE* é executado quando o robô se encontra diante à situação descrita na Figura 4.22(a) que é completamente livre de obstáculos (considerando a distância mínima de 60 cm). Nas situações das Figuras 4.22(b) e 4.22(c) o comando executado pelo robô é *ESQUERDA*, já que há obstáculos à direita e no centro. O comando *DIREITA* é executado quando há obstáculos à esquerda e a frente, como nas Figuras 4.22(d) e 4.22(e). Nas situações descritas nas Figuras 4.22(f) (obstáculos à frente), 4.22(g) (obstáculos nas laterais) e 4.22(h) (obstáculos à frente e nas laterais) então o robô fará uma escolha (que poderá ser aleatória) entre os dois comandos *DIREITA* e *ESQUERDA* para ser executado.

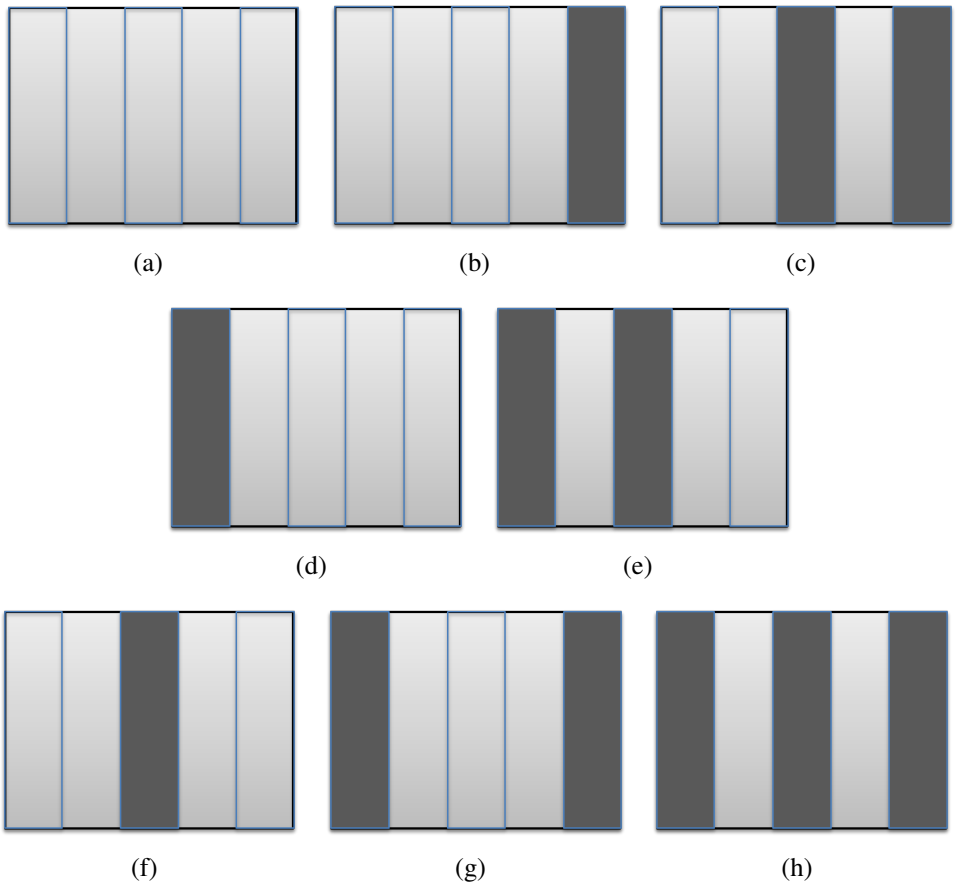


Figura 4.22: Situações no mapa de profundidade do sistema reativo. As porções de imagem de cor mais escura representam obstáculos, que podem ser à frente, à direita e/ou à esquerda.

Na Figura 4.23 é apresentada uma fotografia do robô em operação durante o processo de execução do sistema reativo em que o robô se desloca pelo ambiente de forma segura. No canto inferior esquerdo da imagem são mostrados o comando que o robô deve seguir e as distâncias mínimas e máximas de cada seção da imagem de profundidade.

Figura 4.23: Sistema reativo de navegação em execução. Os valores mínimo e máximo representam, respectivamente, o valor mínimo e máximo de um pixel (distância) nas porções de imagem frente, esquerda e direita. Vídeo disponível em: `http://www.youtube.com/watch?v=SmqeO71hIfc`

Este trabalho de navegação reativa resultou na autoria de um artigo (Correa, 2012) em evento nacional, além de servir como base para outro trabalho, o qual resultou na co-autoria de um artigo internacional (Sales, 2012b).

Devido ao ângulo limitado de visão do sensor Kinect há uma dificuldade de navegação em ambientes muito pequenos em que o robô deveria se deslocar muito próximo a paredes e/ou móveis. Entretanto, existe ainda a possibilidade de se adotar um esquema mais sofisticado de controle da navegação, com a adaptação da velocidade de deslocamento de acordo com a proximidade dos obstáculos e riscos de colisão, adotando, por exemplo, um algoritmo baseado no método VFH (Borenstein *et al.*, 1991). Outras abordagens mais complexas também podem ser consideradas, como o planejamento detalhado e antecipado de trajetórias, o uso de outros sensores complementares (como por exemplo, sonares e sensores infravermelho), e inclusive o uso de uma base tipo "pan-tilt" acoplada ao Kinect; no entanto considera-se que estas abordagens estão além do escopo previsto para este trabalho no presente momento, podendo gerar implementações para trabalhos futuros.

4.3 Detecção de Estados e Transições junto ao Mapa Topológico

Este módulo foi desenvolvimento com o apoio de Daniel Sales (Sales, 2012a,b) a fim de implementar a navegação topológica do sistema de vigilância. Durante a navegação topológica do robô, ele irá dispor de um mapa topológico global do ambiente (representado por um grafo) no qual deverá ser capaz de manter sua localização, isto é, saber em qual aresta ou vértice do grafo se encontra.

Foi desenvolvido um sistema de classificação utilizando redes neurais artificiais e o programa JavaNNS (ver subseção 2.5.5), que simula as redes. O sistema deve ser capaz de ler um mapa local de profundidade do ambiente interno (caracterizado por corredores e portas abertas) gerado pelo Kinect através do OpenCV e classificar a imagem em 8 situações (ver Figura 4.24). As situações são: "corredor livre", "curva esquerda", "curva direita", "curva direita e esquerda", "curva esquerda e corredor", "curva direita e corredor", "fechado" e "cruzamento".

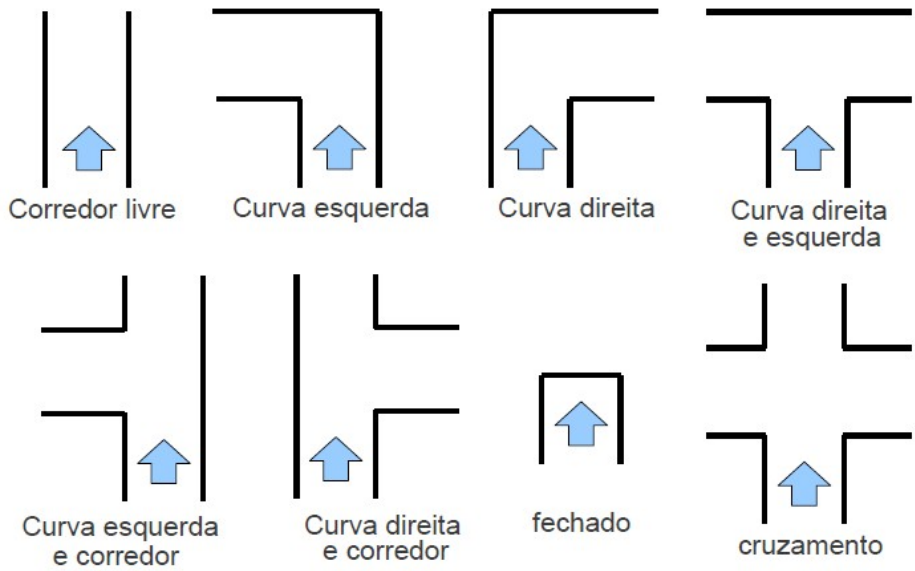


Figura 4.24: Situações definidas de ambientes internos.

Após realizar testes com diversas topologias de rede foi escolhida a seguinte topologia: 632 entradas, 316 neurônios na camada intermediária e 8 neurônios na camada de saída. A Figura 4.25 apresenta a topologia da rede neural bem como uma representação da área da imagem usada como entrada para rede neural. A área destacada na figura representa uma região com altura de 80 linhas, onde é calculada a média destas 80 linhas a fim de obter um valor único que as represente. Uma vez que a imagem possui 632 colunas, e que de cada uma destas colunas

é extraída a média das 80 linhas, obtém-se, portanto, 632 valores que são fornecidos como entrada da rede neural. As 8 saídas representam cada uma das situações dos ambientes internos, conforme apresentadas na Figura 4.24.

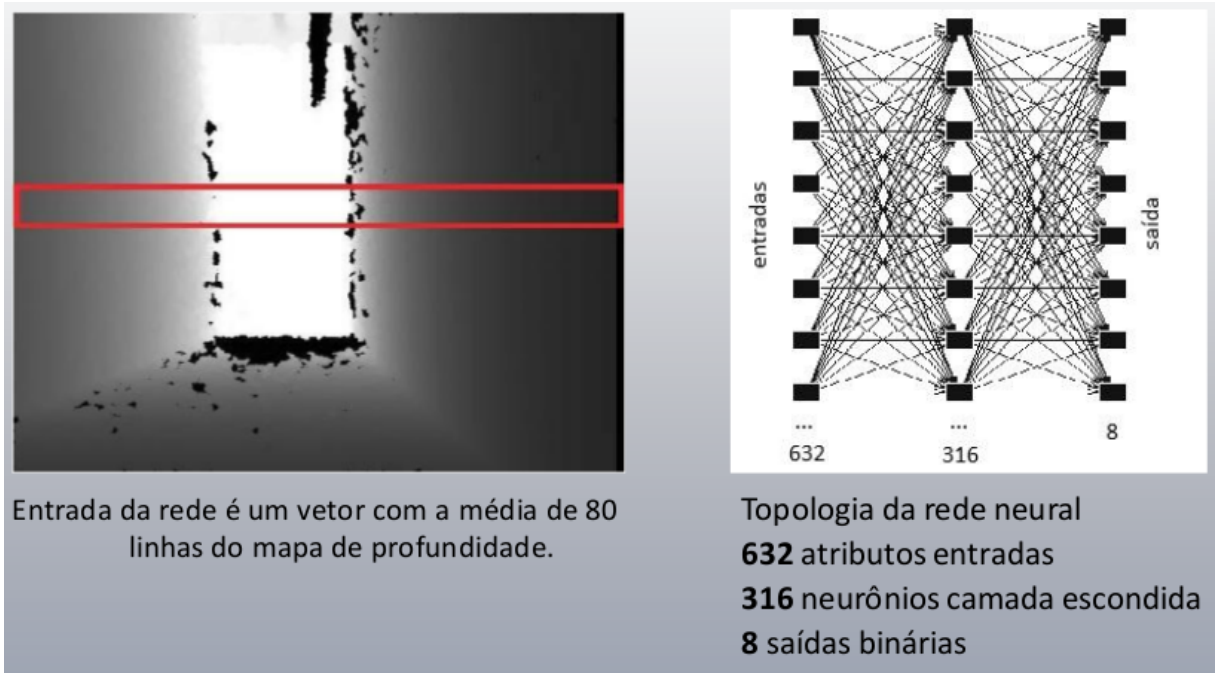


Figura 4.25: Em (a) imagem do sensor Kinect com destaque em vermelho para a região de 80 linhas utilizadas para o processamento da rede neural, e em (b) a topologia da rede neural artificial.

O programa de classificação em execução é visualizado na Figura 4.26, onde o lado esquerdo é o mapa de profundidade, o lado direito a imagem colorida (também fornecida pelo Kinect) apenas para visualização do ambiente e no meio o resultado da classificação da rede. Esta classificação é "curva esquerda e corredor", ou seja, tem uma porta à esquerda e à frente um corredor.

Foram realizados vários testes a fim de validar a rede neural e o resultado da classificação. Inicialmente foram realizados vários treinos usando uma base de dados de 1421 exemplos com um treino de *5-fold-crossvalidation*, onde cada partição dos dados continha em torno de 20% dos exemplos da base original. A Figura 4.27 apresenta a média do desempenho para os *5-folds* do treino de diferentes configurações de topologias de rede. Constatou-se que a melhor rede tinha a topologia 632-316-8, sendo, portanto, selecionada. A resposta do teste de validação desta rede para os *5-folds* é apresentada na Figura 4.28. Maiores detalhes podem ser encontrados no artigo Sales (2012b).

As funcionalidades realizadas neste módulo incluem a localização do robô no mapa topológico global (construído manualmente) a partir da classificação que utiliza redes neurais e o mapa

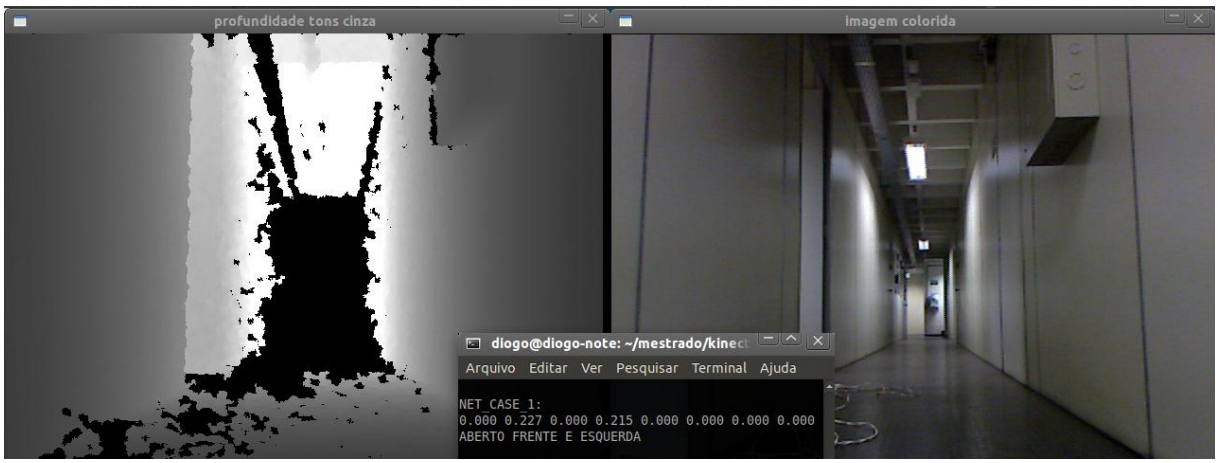


Figura 4.26: resultado de uma classificação.

Teste	1	2	3	4	5
Topologia rede	632-16-8	632-80-8	632-316-8	632-16-16-8	632-80-80-8
Taxa de acerto	85,4%	90,2%	92,2%	84,4%	**92%**

Figura 4.27: Média do desempenho para os *5-folds* do treino de diferentes configurações de topologias de rede.

34	0	1	0	0	0	0	0
0	34	5	0	0	0	0	0
2	3	36	0	0	0	0	0
0	1	1	22	0	1	0	0
0	0	0	0	41	0	1	1
0	0	0	0	0	42	0	1
0	0	0	0	0	0	25	0
0	0	1	0	1	1	0	31

Test 1

33	0	2	0	0	0	0	0
0	37	2	0	0	0	0	0
1	2	37	1	0	0	0	0
0	0	1	24	0	0	0	0
0	0	0	0	39	0	3	1
0	0	0	0	0	41	0	2
0	0	1	0	0	0	24	0
0	0	1	0	1	1	0	31

Test 2

33	0	2	0	0	0	0	0
0	33	5	1	0	0	0	0
1	1	36	1	0	1	1	0
0	1	0	23	0	0	0	0
0	0	1	0	41	0	0	1
0	0	0	0	1	40	1	1
0	0	0	0	0	0	24	0
0	0	0	0	2	0	0	32

Test 3

33	0	2	0	0	0	0	0
0	34	3	1	0	1	0	0
1	5	33	0	0	1	0	0
0	1	1	22	0	0	0	0
0	0	1	0	41	0	0	1
0	0	0	0	0	43	0	0
0	0	0	0	0	1	23	0
0	0	0	0	2	2	0	30

Test 4

33	0	1	0	0	0	0	0
0	35	4	0	0	0	0	0
0	5	34	0	0	0	0	1
0	0	1	23	0	0	0	0
0	0	0	0	42	0	0	0
0	0	0	0	0	42	0	0
0	0	0	0	0	0	24	0
0	0	0	0	2	3	0	29

Test 5

Figura 4.28: Matrizes de confusão representando a resposta do teste de validação da rede 632-316-8 para os *5-folds*.

local de profundidade. Este módulo pode ser utilizado pelo sistema de navegação autônoma do robô de vigilância desenvolvido neste trabalho presente.

A proposta e o desenvolvimento deste módulo de "Detecção de Estados e Transições junto ao Mapa Topológico"foi realizado em parceria com o trabalho de mestrado de Daniel Sales (Sales, 2012b). Este trabalho resultou na co-autoria de um artigo internacional (Sales, 2012b), o qual foi convidado para ser submetido em uma versão estendida para um journal internacional (atualmente aguardando a avaliação).

Considera-se esta aplicação de reconhecimento de situações em ambientes internos bastante relevante, pois esta é uma parte importante do sistema de navegação baseado em mapas topológicos. Acredita-se que tal proposta constitui-se de uma interessante contribuição desta dissertação, constituindo-se de uma proposta de um sistema mais geral de navegação em ambientes internos, baseado no uso do sensor Kinect.

4.4 Trabalhos Complementares

Aqui serão apresentadas implementações e experimentos que não compõem o sistema principal de navegação, mas que foram desenvolvidos ao longo deste trabalho de mestrado. A integração dos módulos apresentados nas subseções seguintes ao sistema de navegação desenvolvido faz parte dos trabalhos futuros. Esta integração visa o aprimoramento e a melhoria das funções realizadas pelo robô de vigilância.

4.4.1 Módulo *Follow Me*

Um robô *follow me* (siga-me, em português) tem a funcionalidade de seguir uma pessoa ou um objeto em movimento. Para tanto, o robô deve ser dotado de sensores que o permitam reconhecer o objeto/pessoa a ser seguido.

No módulo desenvolvido, o robô segue uma pessoa utilizando o sensor Kinect e o programa de rastreamento de pessoas do OpenNI/NITE (ver subseção 2.5.3). A partir do momento que uma pessoa é rastreada, o controle do robô pode gerar comandos de atuação. Os comandos são *FRENTE*, *TRAS*, *DIREITA*, *ESQUERDA* e *PARADO*, de acordo com a posição da pessoa que está sendo rastreada com o robô. Na Figura 4.29 é apresentada a foto de uma situação real do sistema em funcionamento.

O robô pode andar para frente quando a pessoa rastreada está distante (distância $>= 90$cm), pode andar para trás caso a pessoa esteja muito próxima (distância $<= 60$cm) e ficar parado em uma distância intermediária (distância entre 60 e 90cm). Quando o humano rastreado está nas laterais, um dos comandos de girar o robô é executado. A Figura 4.30 apresenta o rastreamento de uma pessoa utilizando o sensor Kinect, bem como uma divisão da cena em zonas que permitem orientar o robô em direção à pessoa conforme a zona que ela se encontra.

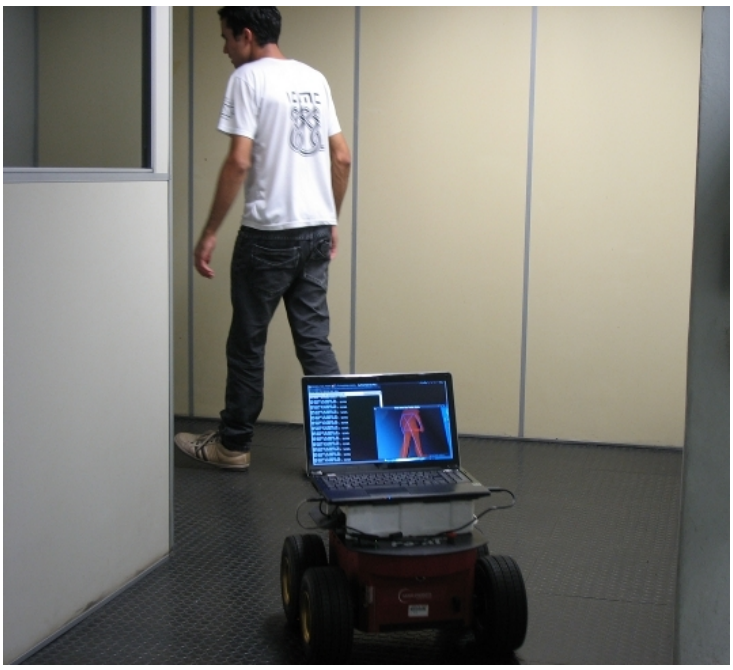


Figura 4.29: Sistema follow-me em execução. Vídeo disponível em `http://www.youtube.com/watch?v=BwDHbBvVAUs`

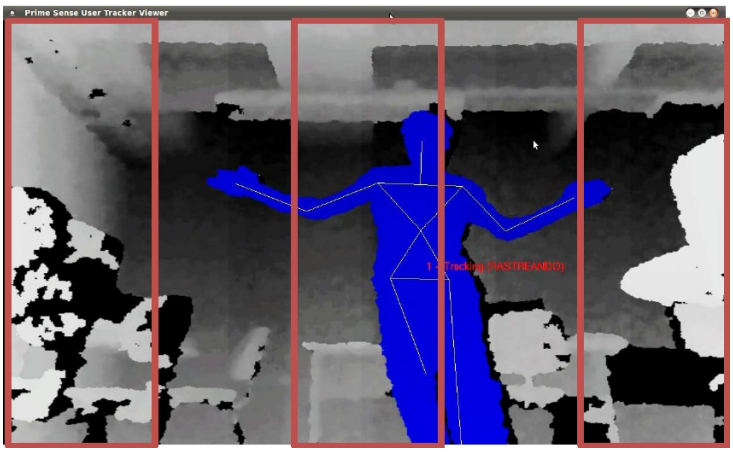


Figura 4.30: Rastreamento do humano e em vermelho regiões utilizadas para o cálculo da distância e direção do deslocamento do robô.

Outras situações que podem ocorrer são a oclusão do usuário rastreado e a disposição de um obstáculo entre o robô e o humano. No primeiro caso, o comando *PARADO* é executado até que a pessoa volte ser exibida ou, a partir de um tempo (10 segundos, por exemplo), o robô espera por novos humanos para efetuar a calibração e o rastreamento. Na outra situação, que ainda não foi tratada, o robô deve ser capaz de desviar de obstáculos.

Esta implementação *follow me* poderá ser agregada ao sistema de navegação adicionando uma nova funcionalidade, que é seguir um intruso pelo ambiente, ou até mesmo um vigilante humano, aumentando assim o leque de possibilidades de usos do sistema robótico.

4.4.2 Teleoperação com Visualização das Imagens no Cliente

Um sistema de vigilância autônomo pode ter a funcionalidade de operação remota em que um vigilante humano o controlará utilizando um *joystick*, ou mesmo pelo teclado do computador. A teleoperação pode ser importante quando o vigilante deseja mudar o robô de sua rota para, por exemplo, explorar um lugar suspeito dentro do ambiente.

Pensando nisso foi criada uma módulo no qual é feita a teleoperação do robô Pioneer utilizando o *software Player* (ver subseção 2.5.2) e um *joystick* conectado a um notebook, onde poderiam ser vistas as imagens de profundidade e coloridas do sensor Kinect. Os dados são trocados via mensagens (imagens codificadas em texto) utilizando uma rede sem fio mantida por roteador no qual são conectados os computadores cliente (notebook com o *joystick* conectado) e o servidor (computador conectado fisicamente com o robô).

Esta funcionalidade também poderá ser integrada ao projeto do robô de vigilância. Desta forma o robô teria um comportamento semi-autônomo, no qual um vigilante humano poderia controlar a distância o robô e depois retomar a navegação autônoma do mesmo. Na figura 4.31 é apresentado um esquema demonstrando a montagem da aplicação.

Figura 4.31: Modelo do sistema de teleoperação. Um roteador é utilizado para fazer a comunidação sem fio entre o cliente (computador com *joystick*) e o servidor (computador conectado ao robô). Os sensores utilizados são o Kinect e a câmera térmica.

4.4.3 Módulo de Reconhecimento de Gestos Utilizando Redes Neurais

O autor deste projeto de pesquisa participou do desenvolvimento de um trabalho para o reconhecimento de gestos utilizando o sensor Kinect. Como resultado deste trabalho foi publicado um artigo (Alvarenga *et al.*, 2012). Este trabalho fez parte do resultado do trabalho de conclusão de curso de Matheus Lin (Alvarenga, 2011) e contou com a contribuição do autor desta dissertação de mestrado.

O sistema foi desenvolvido utilizando o *software ROS – Robot Operating System* – que foi visto na subseção 2.5.4 e utiliza o sensor Kinect e o módulo de rastreamento de humanos provido pelo OpenNI (integrado ao ROS e visto na subseção 2.5.3). O sistema utiliza redes neurais para classificar três gestos humanos, que são *circle* (do inglês círculo), *come here* (do inglês venha aqui) e *goodbye* (do inglês adeus). Uma representação dos gestos é visualizada na Figura 4.32.

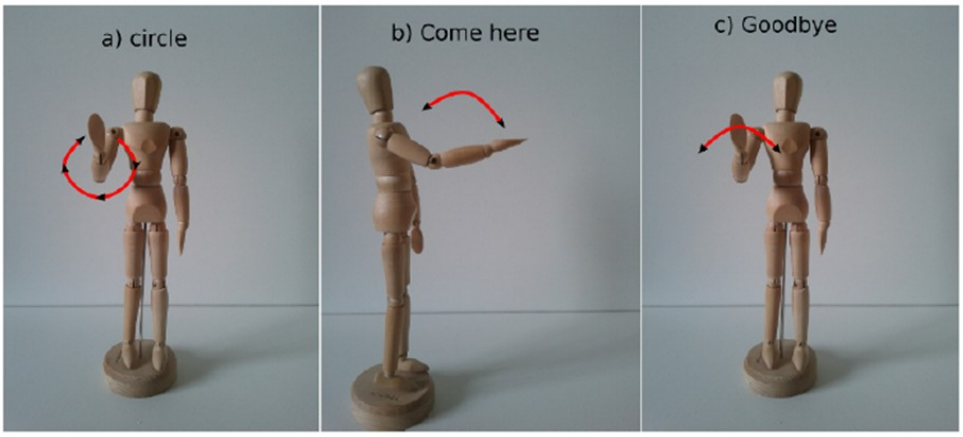


Figura 4.32: Sistema de reconhecimento de gestos.

Na Figura 4.33 é apresentada uma tela de execução do sistema de reconhecimento de gestos, onde o sistema está classificando o gesto como "goodbye".

Este módulo poderá ser utilizado para fazer a teleoperação do robô (citada anteriormente), porém, ao invés de utilizar um *joystick* seriam utilizados gestos humanos para o controle do robô. Além disso, também é possível reconhecer gestos da pessoa que foi detectada pelo robô vigilante.

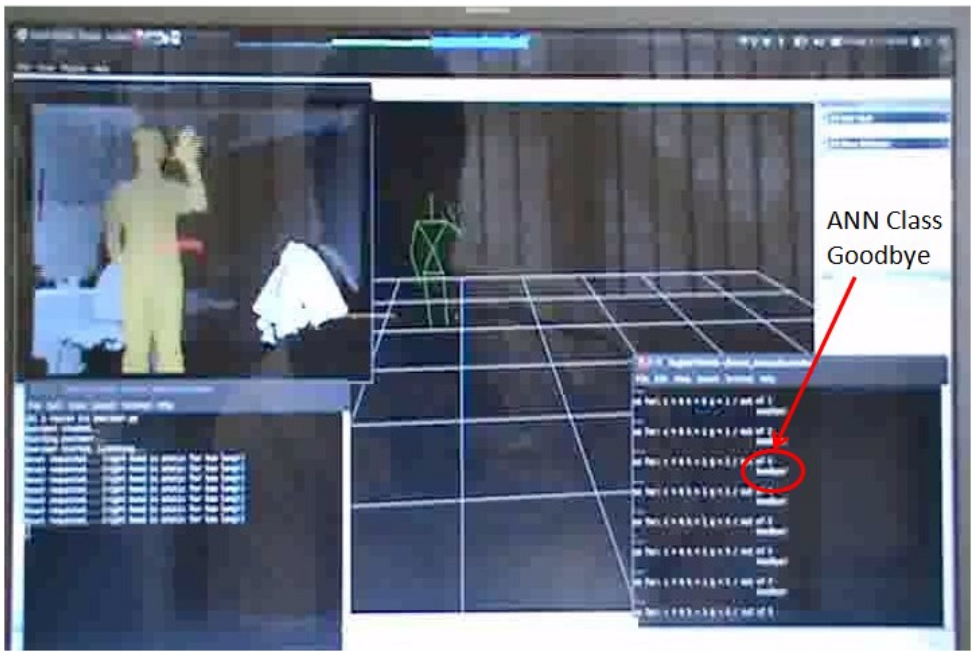


Figura 4.33: Sistema de reconhecimento de gestos em execução.

4.5 Considerações

Neste capítulo foram apresentadas aplicações desenvolvidas utilizando aplicações utilizando os sensores Kinect e FLIR para a construção de um sistema robótico de vigilância, que foi o propósito deste mestrado. Entre os *softwares* utilizados estão o OpenCV, o Player/Stage, o OpenNI/NITE e o ROS.

Portanto, neste capítulo foram detalhados os principais módulos que compõem o sistema de vigilância proposto no contexto desta dissertação. Estes módulos permitem a execução de tarefas de mais baixo nível, como a detecção de pessoas (intrusos) em ambientes internos, a navegação reativa com a detecção e desvio de obstáculos, e por fim, a detecção e classificação de situações (contexto) que permitem e tem sido utilizados em algoritmos de navegação baseada em mapas topológicos. Os métodos e algoritmos propostos são baseados no uso de sensores do tipo Kinect e câmera térmica, que em conjunto possibilitam uma melhor execução das tarefas de navegação e vigilância de ambientes internos.

Os diferentes módulos do sistema proposto foram validados através de experimentos práticos que demonstraram a funcionalidade de cada um deles (correta execução das tarefas junto ao robô real), bem como, foram descritos e melhor detalhados junto aos artigos que foram publicados. Estes artigos complementam as informações apresentadas nesta dissertação.

O próximo capítulo apresenta uma discussão final e as conclusões desta dissertação.

CAPÍTULO

5

Conclusão

Esta dissertação apresentou a proposta e o desenvolvimento de um sistema de navegação autônoma para robôs móveis de vigilância em ambientes internos. O sistema robótico desenvolvido é capaz de navegar de forma segura no ambiente, desviando de obstáculos; reconhecendo pontos de referência no ambiente, como portas e corredores, de modo a permitir a navegação em um mapa topológico; e detectando intrusos no ambiente.

Entre os *softwares* utilizados estão a OpenCV, uma biblioteca de visão computacional e processamento de imagens; o ROS, um meta-sistema operacional utilizado para o desenvolvimento de aplicações robóticas, o Player/Stage, uma ferramenta de arquitetura cliente/servidor utilizado para o controle de robôs móveis; e o JavaNNS, uma biblioteca construída na linguagem Java para a modelagem e o uso de Redes Neurais Artificiais. O *hardware* utilizado foi um robô Pioneer P3AT, um sensor de distância Kinect, um sensor de calor FLIR TAU 640 e um notebook, para a aquisição e processamento de dados e o controle do robô.

A navegação reativa foi realizada utilizando o sensor Kinect para medir a distância dos objetos presentes na cena e permitir que o robô se desloque de forma segura. Este sensor também foi utilizado para o reconhecimento dos pontos de referência no ambiente, como portas em um corredor.

Para a realização da detecção de pessoas foi necessário realizar a calibração dos sensores para garantir que tenham a mesma percepção do ambiente. Após este passo foi realizada a segmentação e o processamento das imagens de ambos os sensores para validar a detecção de

seres humanos. A utilização de dois sensores possibilitou um aumento considerável do campo de visão do robô, podendo ter este sistema grande aplicabilidade para vigilância de ambientes internos.

Conforme apresentado no apêndice A, foi possível apresentar diversos artigos em eventos nacionais e internacionais.

5.1 Trabalhos Futuros

Na seção 4.4 foram citados módulos desenvolvidos, mas que não foram integrados ainda ao sistema. Entre os trabalhos futuros estão a integração destes módulos ao sistema robótico com a finalidade de aumentar as funcionalidades do sistema.

- O módulo "follow me" poderá ser utilizado para seguir um intruso no ambiente quando detectado;
- O módulo "Teleoperação com Visualização das Imagens no Cliente" poderá ser utilizado para que um vigilante humano retomasse o controle do robô, podendo fazer a teleoperação do mesmo a distância utilizando um joystick, com a visualização dos dados dos sensores no próprio computador;
- O "Módulo de Reconhecimento de Gestos Utilizando Redes Neurais" poderá ser utilizado para a teleoperação e o controle do robô utilizando os gestos humanos. Neste caso seria utilizado um sensor Kinect no local onde o vigilante estaria operando o robô.

Referências Bibliográficas

AERYON Aerial vehicle systems - aeryon labs inc. Acesso em: 10 Fev, 2012. Disponível em http://www.aeryon.com/products/avs.html

ALVARENGA, M. L. T. *Reconhecimento de gestos 3d usando o dispositivo kinect*. Trabalho de conclusão de curso (bacharelado em ciências de computação), Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, São Carlos-SP, 2011.

ALVARENGA, M. L. T.; CORREA, D. S. O.; OSÓRIO, F. S. Redes Neurais Artificiais aplicadas no Reconhecimento de Gestos usando o Kinect. In: *Computer on The Beach 2012.*, Florianópolis/SC, 2012.

BEKEY, G. *Autonomous robots: from biological inspiration to implementation and control*. Cambridge, MA, USA: The MIT Press, 2005.

BONVIN, D. Connecting kinects for group surveillance. Acesso em: 10 Jan 2012. Disponível em actu.epfl.ch/news/connecting-kinects-for-group-surveillance, 2010.

BORENSTEIN, J.; KOREN, Y.; MEMBER, S.; MEMBER, S. The vector field histogram - fast obstacle avoidance for mobile robots. 1991, p. 278–288.

BRADSKI, G.; KAEHLER, A. *Learning opencv: Computer vision with the opencv library*. Cambridge, MA, USA: O'Reilly, 2008.

BRÄUNL, T. *Embedded robotics: Mobile robot design and application with embedded systems*. third ed. Australia: Springer, 2008.

CARNEGIE, D.; LOUGHNANE, D.; HURD, S. The design of a mobile autonomous robot for indoor security applications. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, v. 218, n. 5, p. 533, 2004. Disponível em http://pib.sagepub.com/content/218/5/533.short

CORREA, DIOGO O. ; SCIOTTI, D. . P. M. G. . S. D. O. . D. F. F. S. Mobile robots navigation in indoor environments using kinect sensor. In: *Brazilian Conference on Critical Embedded Systems - CBSEC*, New York: IEEE Press, 2012, p. 36–41.

COUTO, L. N. *Sistema para localização robótica de veículos autônomos baseado em visão computacional por pontos de referência*. Dissertação de mestrado em ciências da computação e matemática computacional (orientador: Fernando osório), Universidade de São Paulo, 2012.

CROWLEY, J. Coordination of action and perception in a surveillance robot. *IEEE Expert*, v. 2, n. 4, p. 32–43, 1987.

DICKENS, J.; WYK, M. V. Pedestrian detection for underground mine vehicles using thermal images. *AFRICON, 2011*, , n. September, p. 1–6, 2011.

DUDEK, G.; JENKIN, M. *Computational principles of mobile robotics*. Cambridge, Cambs., UK: Cambridge University Press, 2000.

FERNÁNDEZ-CABALLERO, A.; CASTILLO, J.; MARTÍNEZ-CANTOS, J.; MARTÍNEZ-TOMÁS, R. Optical flow or image subtraction in human detection from infrared camera on mobile robot. *Robotics and Autonomous Systems*, v. 58, n. 12, p. 1273–1281, 2010.
Disponível em `http://linkinghub.elsevier.com/retrieve/pii/S0921889010001168`

FISCHLER, M. A.; BOLLES, R. C. Readings in computer vision: issues, problems, principles, and paradigms. cap. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., p. 726–740, 1987.
Disponível em `http://dl.acm.org/citation.cfm?id=33517.33575`

FLIR Sistema de Imagem Térmica PathFindIR. Acesso em: 10 Out, 2011a.
Disponível em `http://www.flir.com/cvs/americas/pt/transportation/products/pathfindir/`

FLIR O que são raios infravermelhos? Acesso em: 10 Out, 2011b.
Disponível em `http://www.flir.com/thermography/americas/br/content/?id=14474`

HAMBLING, D. Next-Generation Surveillance Robots Can Analyze Their Environment. Acesso em: 10 Fev, 2012.
Disponível em `http://www.popsci.com/technology/article/2011-12/next-generation-surveillance-robots-can-analyze-their-environment`

HATA, A. *Mapeamento de ambientes externos utilizando robôs móveis*. Dissertação de mestrado em ciência da computação e matemática computacional, Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, São Carlos-SP., 2010.
Disponível em `http://www.teses.usp.br/teses/disponiveis/55/55134/tde-13072010-144634/pt-br.php`

HEARST, M. A. Trends & controversies: Support vector machines. 1998, p. 18–28.

HOSHINO, F.; MORIOKA, K. Human following robot based on control of particle distribution with integrated range sensors. In: *System Integration (SII), 2011 IEEE/SICE International Symposium on*, 2011, p. 212–217.

INCT-SEC GT 1 - Robôs Táticos para Ambientes Internos. Acesso em: 10 Jan, 2012. Disponível em `http://www.inct-sec.org/?q=node/2`

KINECT Kinect – Xbox. Acesso em: 10 Fev, 2012a. Disponível em `http://www.xbox.com/en-US/Kinect`

KINECT Microsoft Kinect Teardown – iFixit. Acesso em: 10 Fev, 2012b. Disponível em `http://www.ifixit.com/Teardown/Microsoft-Kinect-Teardown/4066/1`

KINECTHACKS Supporting the Kinect Hacking news and community. Acesso em: 10 Fev, 2012. Disponível em `http://www.kinecthacks.com/`

KONOLIGE, K.; AGRAWAL, M.; BOLLES, R. C.; COWAN, C.; FISCHLER, M.; GERKEY, B. Outdoor mapping and navigation using stereo vision. *Experimental Robotics*, , n. July, p. 179–190, 2006. Disponível em `www.ai.sri.com/~gerkey/papers/lagrnav.pdf`

LEE, H.-T.; LIN, W.-C.; HUANG, C.-H.; HOU-TSAN, L.; WEI-CHUAN, L.; CHING-HSIANG, H. Indoor Surveillance Security Robot with a Self-Propelled Patrolling Vehicle. *Journal of Robotics*, v. 2011, p. 1–9, 2011. Disponível em `http://www.hindawi.com/journals/jr/2011/197105/`

LOONG, S. D. People detection using the Microsoft Kinect. Acesso em: 10 Jan, 2011. Disponível em `http://www.ros.org/wiki/ppl_detection`

MATARI, M. *The robotics primer*. Cambridge, MA, USA: The MIT Press, 2007.

MENDES, C.; WOLF, D. Desvio de Obstáculos Utilizando um Método Estéreo Semi-global. In: *VIII Encontro Nacional de Inteligência Artificial*, Natal, RN, acesso em: 14 fev, 2011. Disponível em `http://www.dimap.ufrn.br/csbc2011/anais/eventos/contents/ENIA/ENIA_Sessao_Poster_Artigo_2_Mendes.pdf`

MENDES, C. C. T. *Navegação de robôs móveis utilizando visão estéreo*. Dissertação de mestrado em ciência da computação e matemática computacional, Universidade de São Paulo, 2012.

MONTEIRO, F. *Sistema de localização remota de robôs móveis baseado em câmaras de vigilância*. Tese de Doutorado, Universidade do Porto, 2011. Disponível em `http://paginas.fe.up.pt/~ee06199/wp-content/uploads/2011/03/Estado-de-Arte.pdf`

MURPHY, R. R. *Introduction to ai robotics*. Cambridge, MA, USA: The MIT Press, 2000.

NASA Mars Exploration Rovers. Acesso em: 10 Fev, 2012.
Disponível em `http://marsrovers.jpl.nasa.gov/home/index.html`

NIN, M. H. C. *Integração de câmera térmica e sensor laser para detecção de pessoas*. Trabalho de conclusão de curso (bacharelado em ciências de computação), Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, São Carlos-SP, 2011.

NIN, M. H. C.; OSÓRIO, F. S. Navegação de robôs móveis autônomos e detecção de humanos baseada em sensor laser e câmera térmica. In: *Mostra Nacional de Robótica*, 2011, p. 1–6.

OPENCV Welcome - opencv wiki. Acesso em: 10 Jan. 2012, 2012.
Disponível em `http://opencv.willowgarage.com/wiki/`

OPENNI Introducing OpenNI. Acesso em: 10 Jan, 2012.
Disponível em `http://www.openni.org`

OSÓRIO, F.; WOLF, D.; BRANCO, K. C.; UEYAMA, J.; PESSIN, G.; FERNANDES, L.; DIAS, M.; COUTO, L.; SALES, D.; CORREA, D.; NIN, M.; LOURENÇO, S. L.; BONETTI, L.; FACCHINETTI, L.; HESSEL, F. Pesquisa e Desenvolvimento de Robôs Táticos para Ambientes Internos. In: *Internal Workshop of INCT-SEC*, n. i, São Carlos, SP, 2011.

PCL About - Point Cloud Library. Acesso em: 10 Jan, 2012.
Disponível em `http://pointclouds.org/about.html`

PESSIN, GUSTAVO ; OSÓRIO, F. S. . S. J. R. . C. F. . U. J. . W. D. F. . B. T. . V. P. A. Evolving an indoor robotic localization system based on wireless networks. In: *International Conference on Engineering Applications of Neural Networks (EANN)*, New York: Springer CCIS Series, 2012, p. 1–6.

PESSIN, G. *Estratégias inteligentes aplicadas em robôs móveis autônomos e em coordenação de grupos de robôs*. Tese de doutorado em ciências da computação e matemática computacional (orientador: Fernando santos osório), Universidade de São Paulo, 2013.

PLAYERSTAGE http://playerstage.sourceforge.net/wiki/main_page. The Player Project. Acesso em: 10 Jan, 2012.

REZENDE, S. O. *Sistemas inteligentes: Fundamentos e aplicações*. Barueri, SP: Manole Ltda, 2003.

RIBEIRO, F. L. *Telepresença - Robô Vigilante*. Tese de Doutorado, Universidade do Porto, 2011.
Disponível em `http://paginas.fe.up.pt/~ee05067/`

ROS http://www.ros.org/. Documentation - ROS Wiki. Acesso em: 10 Jan, 2012.

ROVIO Rovio WowWee. Acesso em: 10 Jan, 2012.
Disponível em `http://www.wowwee.com/en/products/tech/telepresence/rovio/rovio`

SALES, D. O. *Neurofsm: aprendizado de autômatos finitos através do uso de redes neurais artificiais aplicadas à robôs móveis e veículos autônomos*. Dissertação de mestrado em ciência da computação e matemática computacional (orientador: Fernando osório), Universidade de São Paulo, 2012a.

SALES, D. O.; OSÓRIO, F. S.; WOLF, D. F. Topological autonomous navigation for mobile robots in indoor environments using ann and fsm. In: *1ª Conferência Brasileira de Sistemas Embarcados Críticos*, São Carlos, SP, 2011.
Disponível em `http://www.lbd.dcc.ufmg.br/colecoes/cbsec/2011/paper1.pdf`

SALES, D. O.; SHINZATO, P. Y.; PESSIN, G.; WOLF, D. F.; OSÓRIO, F. S. Vision-Based Autonomous Navigation System Using ANN and FSM Control. *2010 Latin American Robotics Symposium and Intelligent Robotics Meeting*, p. 85–90, 2010.
Disponível em `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5702186`

SALES, DANIEL OLIVA ; CORREA, D. O. . O. F. S. . W. D. F. 3d vision-based autonomous navigation system using ann and kinect sensor. In: *International Conference on Engineering Applications of Neural Networks (EANN)*, New York: Springer CCIS Series, 2012b, p. 1–5.

SHINZATO, P. Y. *Sistema de Identificação de Superfícies Navegáveis Baseado em Visão Computacional e Redes Neurais Artificiais*. Dissertação de mestrado em ciência da computação e matemática computacional, Universidade de São Paulo, 2010.
Disponível em `http://www.teses.usp.br/teses/disponiveis/55/55134/tde-16122010-162924/pt-br.php`

SHOTTON, J.; FITZGIBBON, A.; COOK, M.; SHARP, T.; FINOCCHIO, M.; MOORE, R.; KIPMAN, A.; BLAKE, A. Real-time human pose recognition in parts from single depth images. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 2011, p. 1297 –1304.
Disponível em `http://www.cs.dartmouth.edu/~cs104/BodyPartRecognition.pdf`

SIEGWART, R.; NOURBAKHSH, I. *Introduction to autonomous mobile robots*. Cambridge, MA, USA: The MIT Press, 2004.

SILVA, I. N.; SPATTI, D. H.; FLAUZINO, R. A. *Redes neurais artificiais para engenharia e ciências aplicadas - curso prático*. São Paulo, SP: Artliber, 2010.

SPINELLO, L.; ARRAS, K. People detection in RGB-D data. In: *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, IEEE, 2011, p. 3838–3843.
Disponível em `http://ieeexplore.ieee.org/search/freesrchabstract.jsp?navigation=no&arnumber=6095074`

SUARD, F.; RAKOTOMAMONJY, A.; BENSRHAIR, A.; BROGGI, A. Pedestrian detection using infrared images and histograms of oriented gradients. In: *Intelligent Vehicles*

Symposium, 2006 IEEE, Ieee, 2006, p. 206–212.
Disponível em http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1689629

TALUKDER, A.; MANDUCHI, R.; RANKIN, A.; MATTHIES, L.; CRUZ, S. Fast and reliable obstacle detection and segmentation for cross-country navigation. In: *Intelligent Vehicle Symposium, 2002. IEEE*, IEEE, 2002, p. 610–618.
Disponível em http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1188019

THRUN, S. Robotic mapping: A survey. In: *Exploring Artificial Intelligence in the New Millenium*, Morgan Kaufmann, 2002.

THRUN, S. Robotic mapping: A survey. In: *Exploring Artificial Intelligence in the New Millenium*, Morgan Kaufmann, 2003.

THRUN, S.; BURGARD, W.; FOX, D. *Probabilistic robotics*. Cambridge, MA, USA: The MIT Press, 2005.

WOLF, D. F.; OSÓRIO, F. S. *Player/stage controlador e simulador de robôs móveis - instalação e utilização*. Laboratório de Robótica Móvel - Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, São Carlos, SP, acesso em: 20 de Fev, 2011.
Disponível em http://wiki.icmc.usp.br/images/4/43/Player_man_302a.pdf

WOLF, D. F.; SIMÕES, E. D. V.; OSÓRIO, F. S.; TRINDADE JUNIOR, O. Robótica Móvel Inteligente: Da Simulação às Aplicações no Mundo Real. In: *JAI – Jornada de Atualização em Informática 2009 (Tutorial) - Congresso da SBC*, Bento Gonçalves, SP: Editora PUC do Rio, acesso em: 10 Out, 2009, p. 1–51.
Disponível em http://osorio.wait4.org/palestras/texto/JAI2009_Completo_Revisado.pdf

XIA, L.; CHEN, C.; AGGARWAL, J. Human Detection Using Depth Information by Kinect. In: *International Workshop on Human Activity Understanding from 3D Data in Conjunction with CVPR (HAU3D)*, Colorado Spring, 2011.
Disponível em http://www.nattee.net/sites/default/files/HumanDetectionUsingDepthInformationbyKinect.pdf

ZHANG, L.; WU, B.; NEVATIA, R. Pedestrian detection in infrared images based on local shape features. In: *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, IEEE, 2007, p. 1–8.
Disponível em http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4270450

APÊNDICE

A

Publicações Decorrentes Desse Trabalho

1. Correa, Diogo S. O.; Sciotti, Diego; Prado, Marcos G.; Sales, Daniel Oliva; Wolf, Denis F.; Osório, Fernando S.. Mobile Robots Navigation in Indoor Environments Using Kinect Sensor. In: *Brazilian Conference on Critical Embedded Systems - CBSEC*, 2012, Campinas. Proceedings of the 2md. CBSEC - 2012. New York: IEEE Press, 2012. v. 1. p. 36-41.

2. Sales, Daniel Oliva; Correa, Diogo S. O.; Osório, Fernando S.; Wolf, Denis F.. 3D Vision-based Autonomous Navigation System using ANN and Kinect Sensor. In: *International Conference on Engineering Applications of Neural Networks (EANN)*, 2012, Londres. Proceedings of the 13th EANN - 2012. New York: Springer CCIS Series, 2012. v. 1. p. 1-6.

3. Alvarenga, M. L. T.; Correa, Diogo S. O.; Osório, Fernando S. Redes Neurais Artificiais aplicadas no Reconhecimento de Gestos usando o Kinect. In: *Computer on The Beach 2012*. Florianópolis/SC, 2012.