

Byakugan: Visão Computacional

RESUMO

A Olimpíada Brasileira de Robótica (OBR) propõe que seus participantes desenvolvam robôs autônomos capazes de superar desafios em uma emulação de cenário de desastre. Com a intenção de resolver o desafio específico de resgate de vítimas, este trabalho propõe tecnologias de reconhecimento e tratamento de imagem, uma forma alternativa ao que geralmente é utilizado por outras equipes participantes. Foi adicionada uma câmera em um robô (previamente desenvolvido) com intuito de capturar imagens do ambiente onde são dispostas as vítimas e uma área para resgate. Em seguida, foi utilizado o Robotic Operating System (ROS) que permitiu a realização de algumas tarefas, tais como: a execução de programas do robô de forma distribuída em várias máquinas, a utilização de outros sensores além da câmera, o acionamento de atuadores presentes no robô e o processamento das imagens, feito com a biblioteca OpenCV. Os testes iniciais indicam resultados promissores na utilização das tecnologias propostas por este trabalho.

Palavras-chave: Robótica, Educação, Visão computacional, Competição.

ABSTRACT

The Brazilian Robotics Olympiad (OBR – Olimpíada Brasileira de Robótica), proposes their participants to develop robots capable to overcome challenges in a disaster scenario autonomously. This present work proposes an implementation of a computer vision system as an alternative way to those proposed by the OBR's challenge. This work added a camera to a robot (previously developed), aiming visual capture of the environment in which the victims and the safe area are disposed of. Then, was used the Robot Operating System (ROS), that allowed the accomplishment of some tasks like: run the robot programs in a distributed way on several machines, the utilization of some others sensors than the camera, the actuation of the actuators on the robot and the image processing, with the OpenCV library. Initial tests indicate promising results in the use of the technologies proposed by this work.

Keywords: Robotics, Education, Computer Vision, Competition.

1. Introdução

Sob um ponto de vista educacional, a robótica é uma tecnologia emergente que gera interessantes experiências. Essas experiências demonstram como a robótica é um agente inclusivo e que traz bons resultados em projetos coletivos que utilizam tecnologias para soluções de problemas. [OBR, 2019]

No contexto da competição da Olimpíada Brasileira de Robótica (OBR), em Marinato et. al [2017] e Amorim et. Al [2016], os autores desenvolveram projetos de robôs baseados em visão computacional. Esses projetos utilizam de tratamento de imagens para resolverem o desafio de uma emulação de resgate de vítimas proposta pela OBR. Este projeto também desenvolve uma proposta semelhante aos trabalhos anteriores; tendo como objetivo: aprimorar as estratégias utilizadas com a visão computacional e solucionar alguns problemas encontrados, como: a identificação de vítimas de cores distintas e o reconhecimento da área para o resgate. Além disso, a proposta utiliza o Robot Operating System (ROS, Sistema Operacional para Robôs) que permite um melhor controle do ambiente de execução e uma melhor precisão com relação aos sensores usualmente utilizados na competição da OBR.

Este trabalho contribuirá para a robótica do IFRN – *Campus* Santa Cruz, pois permitirá que novos alunos desenvolvam projetos que serão baseados em visão computacional. Além disso, o desenvolvimento desta proposta promoverá um avanço tecnológico nos robôs utilizados no *campus* em competições como a OBR.

Este artigo encontra-se organizado da seguinte forma: a seção 2 apresenta a metodologia do trabalho, o resultado e discussão do projeto são apresentados na seção 3 e as considerações finais apresentadas na seção 4.

2. Metodologia

A possibilidade de que um robô autônomo com processamento de imagens pudesse ser uma forma eficiente de resolver o desafio de emulação de resgate de vítimas proposto pela OBR, despertou a ideia de acoplar em uma plataforma robótica um sistema de visão computacional. Esse sistema deve ser capaz de auxiliar o robô na localização e resgate de objetos (vítimas) na sala de resgate (Figura 1) – local onde são dispostas as vítimas representadas por bolas) para serem resgatadas.

Figura 1 – Sala de resgate



Fonte: Própria

Inicialmente, o grupo estudou as ferramentas que seriam utilizadas: OpenCv, Sistema Operacional para Robôs (ROS). O OpenCv é uma biblioteca *open-source* que possui várias aplicações desenvolvidas para o reconhecimento de objetos e tratamento de imagens; essa biblioteca pode ser desenvolvida em C++ ou Python (linguagens utilizadas por este trabalho). O ROS é *middleware* que permite a criação de uma interface entre o *hardware* e o *software* do robô; distribuição de processos executando em várias máquinas e o monitoramento desses processos em tempo de execução. O ROS basicamente é formado por dois conceitos: nós e tópicos [EDNO, 2014]. O nó é a representação de um processo no ambiente de execução do ROS. Já os tópicos são interfaces que permitem a troca de informações entre os vários nós.

2.1 Etapas de desenvolvimento

Depois do estudo das ferramentas, a segunda etapa de desenvolvimento deste trabalho consistiu em desenvolver dois testes para validar que a proposta seria viável: o primeiro teste seria controlar um robô a partir do reconhecimento da movimentação de mãos humanas, utilizando o OpenCv; o segundo, controlar um robô remotamente por um joystick, com o objetivo de obter experiência no uso do ROS.

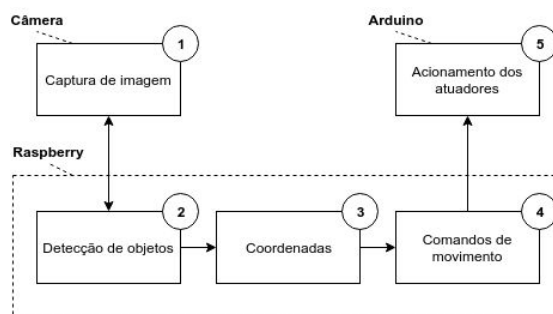
Logo após a realização desses testes, a próxima etapa foi o aprofundamento do conhecimento das tecnologias utilizadas e as suas aplicações iniciais no objeto de estudo deste projeto: o robô desenvolvido pela equipe Wall-E, do IFRN – *Campus* Santa Cruz. Essas aplicações iniciais no robô ocuparam a maior parte de desenvolvimento deste projeto. Nessa etapa, foi desenvolvida a identificação das vítimas (representadas por círculos). Além disso, foi possível diferenciar as cores das vítimas (pretas e prateadas, vivas e mortas, respectivamente). Ademais, foi realizada a identificação da área de resgate, sendo possível identificá-la em diferentes posições do robô em relação à de sala de resgate.

Na penúltima fase de desenvolvimento, o grupo conseguiu criar a comunicação entre o Raspberry e o Arduino, permitindo o Raspberry controle os atuadores e colete informações dos sensores do robô. Por último, o grupo está realizando o aprimoramento do software em testes iniciais de comportamento do robô sob diversas possibilidades de localização na sala de resgate.

2.2 Arquitetura lógica e processamento de imagens

Na elaboração da arquitetura lógica do sistema foram escolhidos dois componentes principais: o Raspberry Pi 3 e o Arduino Mega. Por razões econômicas e simplicidade é comum ter placas microcontroladas como o Arduino no comando de robôs na competição da OBR. No entanto, ler e processar imagens simultaneamente requer um poder computacional maior do que uma placa como o Arduino dispõe. Por essa razão, este trabalho decidiu utilizar o Arduino em conjunto com o Raspberry Pi 3, pois esse mostra eficiência para esse tipo de trabalho (Senthilkuma et al., 2014) e possui capacidade de processar vários dados dos componentes do robô. O diagrama simplificado representado na Figura 2 exibe a arquitetura lógica do sistema proposto.

Figura 2 – Arquitetura lógica do Sistema



Fonte: Própria

O processo começa quando o Raspberry solicita uma imagem da câmera e a disponibiliza para a próxima etapa. Na seção de detecção de objetos, utilizamos funções da biblioteca de visão computacional OpenCV para alterarmos as tonalidades e os formatos da imagem, fatores importantes para os próximos processos. É nessa etapa que imagem alterada é “binarizada”, etapa responsável pela transformação das tonalidades em apenas duas cores: preto e branco.

Com a imagem pronta para a identificação, determinamos os parâmetros para as funcionalidades de identificação de contornos e em seguida simplificamos os vários pontos encontrados. Dos contornos simplificados extraímos as coordenadas importantes para nosso trabalho: o centro dos objetos e suas posições na imagem. A partir das coordenadas, o Raspberry finalmente consegue enviar movimentos para o Arduino que serão necessários para a execução das tarefas.

3. Resultados e Discussões

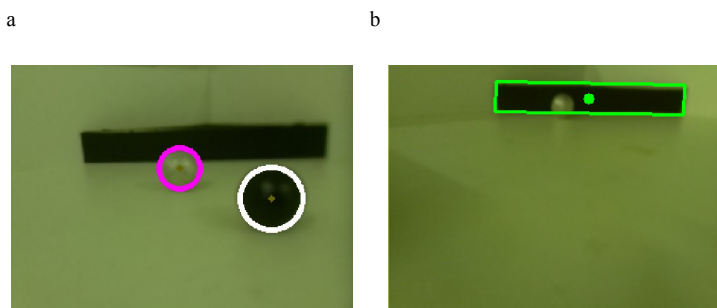
Nas fases iniciais de desenvolvimento, o uso da visão computacional no robô mostrou-se possível devido o processamento das imagens em paralelo com o uso de motores. Entretanto, em alguns casos foram notados altos níveis de processamento do Raspberry, o que evidenciou um ponto crucial na utilização da visão computacional: as imagens deviam ser de baixa resolução (320x240 pixels) para que fosse possível uma boa taxa de processamento.

3.1 Identificação dos objetos

No contexto da identificação de objetos, Marinato et. al [2017] e Amorim et. Al [2016] fazem uso de vítimas de coloração avermelhada pela facilidade de tratamento, o que explica o não reconhecimento de vítimas prateadas e pretas (vítimas vivas e mortas, respectivamente). O presente trabalho conseguiu a identificação de vítimas com as cores prateada e preta (Figura 3.a).

Marinato et. al [2017] deixa claro também a necessidade de um sistema de identificação da área de resgate (região segura para as vítimas). Em fase final de desenvolvimento, a detecção da área de resgate deste trabalho mostra que mesmo uma vítima prateada obstruindo a visão completa da área, essa é identificada na Figura 3.b por um grande retângulo preenchido preto (na perspectiva do robô) circundado por um retângulo verde.

Figura 3 - (a) Identificação das vítimas prateadas e pretas; (b) Identificação da área de resgate.



Fonte: Própria

3.2 Experimentos

Quando a identificação das vítimas e área de resgate foi desenvolvida, realizamos testes com vinte e nove possibilidades de posições, cores e tonalidades. Os resultados mostram que as vítimas mortas (pretas) foram identificadas cerca de 76% dos casos, já as vítimas vivas (prateadas) apenas 28%. Com os dados coletados, temos duas conclusões iniciais na identificação das vítimas: 1. As vítimas pretas são mais identificadas, pois seus contornos são mais evidentes no ambiente esbranquiçado da sala de resgate; 2. As vítimas prateadas são mais difíceis de se identificar por serem de coloração semelhante à sala de resgate, o que dificulta a detecção dos contornos.

Apesar da aparente facilidade de identificação das vítimas mortas na primeira conclusão, essas podem se tornar quase que impossíveis de se detectar quando estão em uma mesma perspectiva da área de resgate (de cor totalmente preta), o que não acontece com as vítimas brancas, que ficam muito evidentes com relação às tonalidades escuras.

Também foi notada uma facilidade de identificação quando as vítimas estavam mais próximas da câmera, o que permite uma discussão futura sobre qual seria a distância ideal para identificar as vítimas. Com relação à identificação da área de resgate, os testes iniciais apresentaram o resultado de cerca de 56% dos casos em que área foi identificada, o que demonstra um pouco de dificuldade em sua execução.

4. Considerações Finais

Na OBR 2019, a utilização da visão computacional para a resolução de um dos desafios da competição assegura a intenção deste trabalho: a equipe Wall-E (objeto de trabalho deste projeto) recebeu o prêmio de inovação, evidenciando que a proposta possui relevância tecnológica e que pode inspirar outras equipes participantes a utilizarem robôs desenvolvidos com base no reconhecimento e tratamento de imagens.

Apesar de algumas dificuldades e tarefas incompletas, este trabalho vem contribuindo de forma significativa para o desenvolvimento da robótica no IFRN - *Campus* Santa Cruz, permitindo que as tecnologias propostas possam ser desenvolvidas por novos alunos.

A proposta apresentada abre direções para trabalhos futuros que poderão superar os problemas deste trabalho e aperfeiçoar suas capacidades. Para trabalhos futuros, propomos:

1. Aperfeiçoamento das estratégias de utilização dos algoritmos de tratamento de imagem;
2. Utilizar esta proposta na Olimpíada Brasileira de Robótica (OBR);
3. Desenvolver documentação do uso de técnicas de visão computacional para robôs utilizados na OBR.

Agradecimentos

Gostaríamos de agradecer ao Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, por ajudarem no desenvolvimento de milhares de pessoas e por acreditarem que os robôs desenvolvidos em conjunto pela comunidade acadêmica são de extrema importância para a expansão do uso de tecnologias educacionais.

Agradecemos também o nosso orientador, familiares, apoiadores, Olimpíada Brasileira de Robótica e a Mostra Nacional de Robótica.

Referências

- MARINATO, Gabriela P.; SOARES, Joyce A. P.; AMARAL, Eduardo M. A. **Sistema de detecção e resgate de vítima para um robô autônomo seguidor de linha baseado em visão computacional**. Mostra Nacional de Robótica (MNR), Serra - ES - Brasil, 2017. Disponível em: <http://sistemaolimpico.org>. Acesso em: 16 abr. 2019.
- AMORIM, Junior Aguilar et al. **Implementação de um robô para competição baseado em visão computacional**. Anais da Mostra Nacional de Robótica, <http://www.mnr.org.br/>, 2018.
- BARBOSA, André; FAGUNDES, Elizandro Gomes; DARÓS, Renan Rocha. **Uma metodologia para a resolução do desafio da olimpíada brasileira de robótica – O robô Volk**. 5º Simpósio de Integração Científica e Tecnológica do Sul Catarinense – SICT-Sul, 2016. Disponível em: <http://docente.ifsc.edu.br/werther/publicacoes/2016SITCSUL-2a.pdf>. Acesso em: 10 ago. 2019.
- ARRUDA, Gabriel Augusto; MARIANO, Vinícius Barbosa Pereira Mariano; SALES, Vinicius Gambi. **Otimização de detecção de rotas e reconhecimento de objetos para Robocup Junior Rescue por meio de processamento digital de imagem e visão computacional**. MNR 2017, Londrina – PR, 2017. Disponível em: <http://www.mnr.org.br/wpcontent/uploads/2019/06/MNR-Anais2017.pdf>. Acesso em: 11 ago. 2019.
- EDNO, Francisco. **Tutoriais sobre ROS**. [S. l.], 14 dez. 2014. Disponível em: http://wiki.ros.org/pt_BR/ROS/Tutorials. Acesso em: 15 mar. 2019.
- G. Senthilkuma, K. Gopalakrishnan, V. Sathish Kumar. **EMBEDDED IMAGE CAPTURING SYSTEM USING RASPBERRY PI SYSTEM**. Volume 3, Issue 2, March – April 2014.
- OPENCV DEV TEAM. **OpenCV 2.4.13.7 documentation**. [S. l.], 12 jul. 2018. Disponível em: <https://docs.opencv.org/2.4.13.7/>. Acesso em: 15 mar. 2019.
- ORGANIZAÇÃO OBR. **Por que uma olimpíada de robótica?** [S. l.: s. n.], 2018. Disponível em: <http://www.obr.org.br/>. Acesso em: 16 abr. 2019.