

# CONTROLE ROBÓTICO BASEADO EM MAPEAMENTO 3D COM SENSOR RGB-D

Gabriel Marques Bandeira<sup>1</sup>; Judith Kelner<sup>2</sup>

<sup>1</sup>Estudante do Curso de Engenharia da Computação - CIn – UFPE; E-mail: gmb@cin.ufpe.br,

<sup>2</sup>Docente/pesquisador do Depto de Informação e Sistemas – CIn – UFPE. E-mail: jk@cin.ufpe.br.

**Sumário:** Este projeto teve por objetivo avaliar o nível de adequação dos algoritmos de reconhecimento de imagem mais utilizados na literatura em um sistema robótico proposto. Para este projeto, foi criado um sistema robótico que reconhece objetos com o uso de uma câmera móvel de forma a identificar imagens de objetos em tempo real e executar diferentes ações, de acordo com os objetos reconhecidos. Neste sistema, o robô pode se movimentar em áreas previamente conhecidas, utilizando imagens de objetos previamente adquiridas para se localizar e planejar os caminhos a serem seguidos posteriormente.

**Palavras-chave:** descritor; reconhecimento de objetos; robótica

## INTRODUÇÃO

Sistemas robóticos, normalmente, requerem a presença de um usuário dedicado ao controle dos mesmos por meio de controles remotos de comandos simples, onde cada comando ativa uma ação do robô. Entretanto, se o robô possuir a capacidade de extrair informações do ambiente onde ele se encontra, ele pode usar essas informações para planejar e realizar suas ações, autonomamente. Nesse cenário, onde o robô executa ações baseando-se no seu ambiente, um usuário pode controlá-lo estipulando objetivos a serem alcançados. Dado um objetivo ao robô, cabe a ele interpretar os comandos do usuário para desempenhar as ações desejadas, de acordo com informações de seu ambiente, como distância entre o robô e a parede ou presença de objetos em um corredor de passagem. Dessa forma, o controlador não precisa ficar dedicado apenas a um robô, uma vez que esse passa a planejar seu caminho até o objetivo, diminuindo a quantidade de comandos a serem enviados pelo usuário. O controlador pode, então, comandar vários robôs com comandos complexos que serão avaliados pelos diferentes robôs, e cada um deles usará as informações de seu ambiente para tomar decisões específicas para si.

Diversas informações podem ser extraídas do ambiente, como textura das superfícies, distância dos objetos ou formas dos objetos. Todas essas informações podem ser adquiridas usando uma câmera de vídeo ou um sensor RGB-D acoplado ao robô. Por meio de pesquisas, o bolsista pôde constatar que o sensor RGB-D permite que algoritmos complexos possam ser executados no robô, sendo uma plataforma de baixo poder computacional. Por este motivo, neste projeto de pesquisa decidiu-se utilizar uma câmera de vídeo para extrair informações do ambiente no qual o robô se localiza.

Esse projeto tem como objetivo principal o desenvolvimento de um sistema robótico que identifique objetos usando uma câmera. Os objetos identificados são utilizados para determinar as possíveis ações que o robô pode realizar, como andar ou desviar de objetos. Os dados gerados pelo robô durante a detecção dos objetos são armazenados e poderão ser comparados com diferentes técnicas de identificação de objetos em outros projetos.

## MATERIAIS E MÉTODOS

O levantamento do estado da arte foi realizado através do mapeamento de publicações em conferências e periódicos relevantes para a área que trataram do uso de câmeras para

identificação de objetos. Em seguida, o material coletado foi analisado, descrito e comparado. Concomitantemente, as técnicas de detecção de imagens contendo objetos conhecidos foram implementadas, tendo como objetivo a comparação dos seus resultados e avaliação do desempenho. Com o resultado da análise realizada e a implementação, foi possível identificar os aspectos positivos das técnicas estudadas para o cenário desejado.

O desenvolvimento dos módulos ocorreu em seguida, gerando os primeiros resultados funcionais integrados. Tais resultados foram avaliados através da execução conjunta com os demais módulos do sistema. Por fim, ocorreu uma etapa de aprimoramento funcional juntamente com a documentação de todo o processo e agrupamento dos resultados finais, a fim de serem publicados perante a comunidade científica.

Dois materiais foram necessários para a criação dos protótipos. O primeiro foi a plataforma Raspberry Pi [1], para executar os algoritmos de extração de características e cálculo de descritores. O segundo foi o kit LEGO Mindstorms NXT, de onde foram usados os motores e algumas peças para a montagem da estrutura. Com essas duas plataformas e uma webcam Microsoft Life Cam Cinema, foi possível a criação dos protótipos dos robôs utilizados nos testes.

## **RESULTADOS E DISCUSSÃO**

Durante o processo de pesquisa das técnicas utilizadas para reconhecer objetos em quadros de um vídeo, assim como de imagens previamente conhecidas nesses quadros, foi observado o grande uso dos algoritmos BRIEF [2], ORB [3], SURF [4] e SIFT [5]. Essas quatro técnicas foram avaliadas pelo bolsista, utilizando as implementações disponíveis na biblioteca OpenCV [6], e comparadas em três cenários diferentes utilizando um Raspberry Pi para executar os algoritmos.

Durante os testes realizados pelo bolsista, foi observada a complexidade e robustez do SIFT e do SURF, que obtiveram altas taxas de quadros identificados corretamente. Foi observado também que o uso do ORB foi o mais apropriado para o projeto, por ter uma taxa relativamente alta de quadros identificados corretamente e ter complexidade mais baixa que o SIFT ou SURF, sendo assim o algoritmo ideal para ser utilizado em plataformas com baixo poder computacional, como o Raspberry Pi.

## **CONCLUSÕES**

Com a pesquisa, foi possível observar que os melhores resultados foram dos algoritmos SIFT e SURF, seguido de ORB e por fim o BRIEF. No entanto, o bolsista conclui que, mesmo sendo alta a taxa de quadros onde o SIFT e o SURF conseguiram identificar os objetos corretamente, eles não são os mais adequados para o projeto, devido ao baixo poder computacional do Raspberry Pi não sendo compatível com a complexidade de execução do SIFT ou SURF, em tempo real. No entanto, é possível usar o ORB e obter um percentual aceitável de quadros corretos mantendo uma alta taxa de frames analisados por segundo, mesmo usando o Raspberry Pi.

Com a finalização do projeto, foi percebida a grande quantidade de pesquisas que já foram feitas pelas comunidades de robótica e visão computacional, mas também que ainda há muito a ser construído.

É esperado que, futuramente, o aluno bolsista possa dar continuidade ao projeto, integrando mais funcionalidades e aprimorando as que já estão presentes no protótipo, utilizando uma metodologia mais rigorosa.

### **AGRADECIMENTOS**

Ao CNPq por ter possibilitado e financiado esse projeto de pesquisa. E ao GRVM, por ter disponibilizado de recursos materiais para o andamento do projeto, assim como humano para suporte do mesmo.

### **REFERÊNCIAS**

- [1] - <https://www.raspberrypi.org> Acessado em 24/08/2015
- [2] - M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, “Brief: Computing a local binary descriptor very fast,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1281–1298, 2012.
- [3] - E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” *Computer Vision, IEEE International Conference on*, vol. 0, pp. 2564–2571, 2011.
- [4] - H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Surf: Speeded-up robust features,” *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp.346–359, jun 2008. [Online]. Available: <http://dx.doi.org/10.1016/j.cviu.2007.09.014>
- [5] - D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, nov 2004. [Online]. Available: <http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94>
- [6] - G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.