colloquium exactarum

ROBÔ AUTÔNOMO MÓVEL SEGUIDOR DE SÍMBOLO

AUTONOMOUS MOBILE ROBOT FOLLOWER OF SYMBOL

Juliano Nishizima[1]; Leandro Luiz de Almeida[1]; Danillo Roberto Pereira[1]; Francisco Assis da Silva[1]; Flavio Pandur Albuquerque Cabral[1]

[1]Faculdade de Informática – FIPP, Universidade do Oeste Paulista – UNOESTE
e-mail: j.nishizima@hotmail.com, {llalmeida danilopereira, chico, pandur}@unoeste.br

RESUMO – Este artigo descreve o desenvolvimento de um robô móvel seguidor de símbolo do tipo AGV (*Autonomous Guiding Vehicles*). O robô é composto por duas partes: hardware e software. O hardware é constituído por um computador local responsável pelo processamento das imagens que são enviadas da câmera de um celular, com sistema operacional Android, e o Arduino, responsável pela leitura de todos os sensores e o controle dos atuadores. O software é composto por algoritmos de Visão Computacional, onde se utilizou a biblioteca OpenCV (*Open Computer Vision Liberary*) para aplicação dos algoritmos como: Transformada de *Hough*, Filtro *Smooth Gaussian* e operação *InRange*, para identificar o símbolo a ser seguido pelo robô.
Palavras-chave: Transformada de *Hough*; Veículo AGVs; Arduino.

ABSTRACT – This article describes the development of a mobile robot follower of the AGV type symbol (Guiding Autonomous Vehicles). The robot is composed of two parts: hardware and software. The hardware consists of a local computer that processes the images that are sent from a mobile camera with Android operating system and the Arduino responsible for reading all sensors and control actuators. The software consists of computer vision algorithms which we used the OpenCV library (Open Computer Vision liberary) for application of algorithms such as: Hough Transform, Gaussian Filter Smooth and InRange operation to identify the symbol to be followed by the robot.
Keywords: Transform Hough; Robot AGVs; Arduino.

Recebido em: 12/12/2014
Revisado em: 01/09/2015
Aprovado em: 05/11/2015

1. INTRODUÇÃO

O surgimento do termo robô, aparece pela primeira fez na obra de Karel Capek, no romance R.U.P (Robôs Universais de Rossum). Já palavra Robótica apareceu pela primeira vez nos livros de ficção científica de Issac Asimov em 1942, denominado de *Runaround*. Surgiu como uma maneira de representar a ciência dedicada aos estudos dos robôs (PEREIRA JÚNIOR, 2013). Robótica é uma ciência de manipulação e percepção do mundo físico com dispositivos controlados por computador.

Para conseguirem se movimentar, os robôs precisam ter as habilidades de navegação, exploração, mapeamento e de auto localização. A navegação é a habilidade de se chegar ao destino e o conhecimento do ambiente ao seu redor.

Os desafios para a construção de robôs móveis, nos dias de hoje, são cada vez maiores, pois não é suficiente apenas que sejam capazes de se mover, desviar de obstáculos e evitar colisões. De acordo com estudo realizado por Schneider (2012), é possível afirmar que o grande objetivo da robótica é o de criar robôs autônomos, para que não ocorra a intervenção do humano em serviços que podem causar danos a si mesmo. Eles podem substituir o ser humano em indústrias, no desarmamento de explosivos, em explorações espaciais, em explorações submarinas, e outros locais de difícil e perigoso acesso.

A visão do robô, de forma geral, tem um papel de extrema importância na locomoção, na busca e na identificação de perigos. A visão acoplada a um robô pode ser capaz de localizar pontos de interesse antes de qualquer outro modo de busca. Assim, se for possível fazer com que o robô processe e identifique algum objeto a

partir de uma imagem capturada por uma câmera, então o mesmo pode fazer tarefas similares as que os seres humanos fazem, pois suas aplicações e decisões são tomadas baseadas no que se pode enxergar. Por meio da Visão Computacional e da robótica, várias outras tecnologias podem ser desenvolvidas em conjunto. Isso é fundamentado por Delai e Coelho (2011a), que afirmam que a Visão Computacional é o ponto chave para o desenvolvimento da robótica, e por RUSS (1998) que descreve que: se for possível desenvolver um robô que possua uma semelhança ao modo de percepção humana do ambiente através da visão, isso a torna uma ferramenta extremamente poderosa.

A partir desse contexto, este trabalho apresenta um robô autônomo móvel seguidor de símbolo, que foi desenvolvido utilizando recursos de Visão Computacional e um microcontrolador Arduino[1], que futuramente pode ser implementado em uma plataforma que suporte elementos mais complexos. Como exemplo de aplicações para robôs desse tipo, tem-se algumas ideias que poderiam ser colocadas em prática e com isso, de alguma forma, melhorar a vida das pessoas. A indústria automobilística Volvo estuda dotar seus veículos automotores de maior segurança, incluindo a eles dispositivos que reconhecem objetos nas vias por onde trafega, permitindo que o motorista tenha mais recursos na tomada de decisões sobre uma situação que apresente um maior risco de acidente. O trabalho de Heinen (2004) mostra que um veículo pode seguir outro, sem a necessidade de um motorista. Delai e Coelho (2011b) evidencia um robô seguidor de símbolo, que pode ser utilizado como um robô cuidador que seja capaz de seguir um idoso ou uma criança com um determinado símbolo identificador em seu corpo, assim auxilia a pessoa responsável a monitorar sem a necessidade de estar no mesmo ambiente, ou de várias câmeras instaladas e dessa maneira evitar que algo grave possa ocorre. Outra aplicação de um robô cuidador foi desenvolvida por Paolieri Neto e Tozi (2013) que possibilita que uma cadeira de rodas motorizada possa seguir seu cuidador. A empresa Boston Dynamics, para auxílio a Marinha dos EUA tem desenvolvido robôs com funções baseadas em aptidões animais, como por exemplo, um cavalo robô utilizado para seguir um soldado, transportando todo o armamento militar necessário, isso ajuda o soldado no que tange ao gasto de energia, deixando-o mais resistente e mais ágil, pois não gastaria sua energia em situações desnecessárias. Um robô limpador com capacidade de recolher bolas de tênis em uma quadra após o final de uma partida, sempre visando à diminuição de gasto e mais praticidade.

2. OBJETIVOS

O objetivo principal é a obtenção, por meio da Visão Computacional, do uso do algoritmo de Transformada de Hough para que seja capaz de reconhecer e acompanhar um determinado símbolo a ser seguido, e a utilização de um robô autônomo móvel do tipo AGVs (*Autonomous Guiding Vehicles*) que possui, acoplado a ele, um celular com câmera que captura imagens e em seguida envia para o computador via *wireless* para que ocorra o processo de reconhecimento do símbolo. Um controlador de velocidade que coordena as rodas do robô e um sensor ultrassônico para evitar a colisão entre o robô e o símbolo.

3. MATERIAIS E MÉTODOS UTILIZADOS

Para a realização do trabalho, o robô (Figura 1) foi projetado para ambiente *indoor*, ou seja, em ambientes internos. O robô é dotado de: celular (sistema operacional Android) com câmera, microcontrolador Arduino ATmega2560, *Bluetooth Shield*, *Motor shield* L293D, sensores ultrassônicos e uma bateria de 9V com a função de fornecer toda energia necessária para o robô.

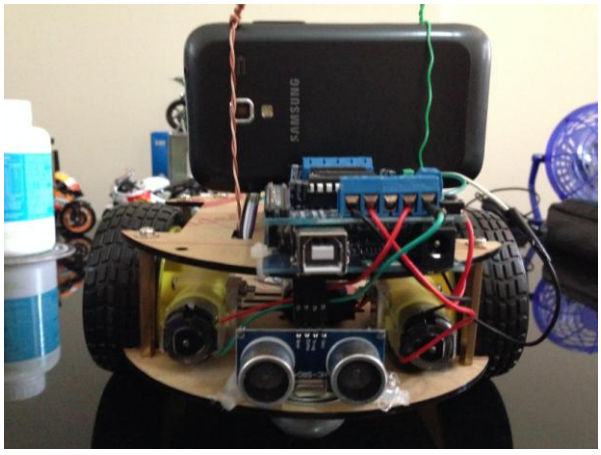

Figura 1. Robô desenvolvido.

[1] Arduino: é dito uma plataforma de computação física, onde sistemas digitais, ligados a sensores e a atuadores, são capazes de medir variáveis no ambiente físico, realizar cálculos numéricos e tomar decisões lógicas no ambiente computacional e gerar novas variáveis no ambiente físico. (FONSECA; VEGA, 2011).

O robô construído utiliza dois motores de corrente continua, cada motor trabalha individualmente, girar na mesma direção em caso de movimentar para frente ou para trás, ou ativa um motor de cada vez, se for virar para direita ou para esquerda.

A navegação do robô é baseada nas informações visuais capturadas pela câmera do celular e também pelas informações dos sensores ultrassônicos. A navegação visual permite que o robô siga um determinado símbolo, ou seja, conforme o objeto se movimenta no ambiente, o robô também se movimenta, mantendo o símbolo sempre no alvo.

3.1 ARDUINO

Segundo o site oficial do microcontrolador Arduino, possui uma ferramenta que permite controlar um grande número de elementos do mundo físico. Sua plataforma é considerada *open source*, arquitetada sobre uma placa microcontroladora simples que disponibiliza um ambiente de desenvolvimento.

Arduino pode ser utilizado para desenvolver objetos interativos, tendo entradas a partir de uma variedade de sensores ou interruptores, e controla uma variedade de luzes, motores e outras saídas físicas. Projetos Arduino podem ser independentes (*stand-alone*), ou eles podem se comunicar com software em seu computador (por exemplo, *Flash, Processing, MaxMSP*.) A placa pode ser montada ou adquirida pré-montada, o Arduino IDE (*Integrated Development Environment*) de código aberto pode ser obtido gratuitamente. O Arduino IDE é um software que simplifica a tarefa de programação e o código que controla a placa Arduino. Escrito em Java é um programa multiplataforma que permite a programação em C/C++, pois é montado sob o compilador avr-gcc, esse compilador é voltado para programas ou *Sketchs* como são popularmente chamados os códigos desenvolvidos para rodar em microcontroladores.

A Figura 2 mostra um exemplo de uma placa Arduino ATmega2560, que utiliza um voltagem de 5V interna, tem como entrada 7-12V, 54 pinos E/S digital (entrada/saída), 16 pinos E/S analógico, corrente elétrica de 40mA para os pinos de E/S e 50mA para os pinos de 3.3V, com memória flash de 256KB com 8KB reservado para gerenciador de *boot* (*bootloader*), SRAM 8KB, EEPROM 4KB e apresenta velocidade de 16MHz de clock.

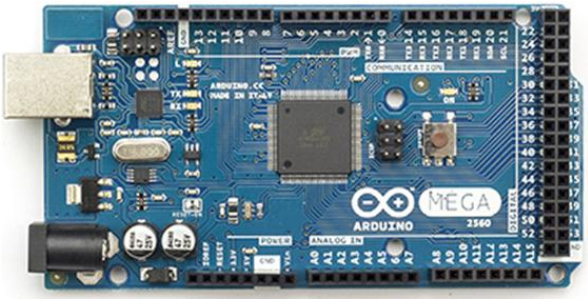

Figura 2. Arduino ATmega2560.
Fonte: http://arduino.cc/.

3.2 ROBÔS MÓVEIS

Robôs móveis são robôs dotados de um sistema de locomoção que os tornam capazes de navegar pelo seu ambiente de trabalho, interagindo com este, na realização de tarefas pelo uso de recursos de sensoriamento e tomada de decisão.

A Figura 3 mostra três tipos de robôs móveis que possuem características específicas, conforme o funcionamento dos motores acoplados.

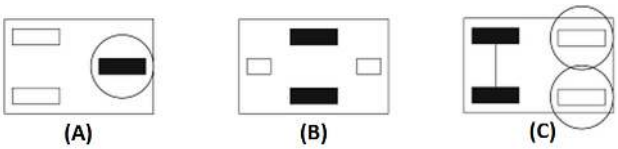

Figura 3. Tipos de robôs móveis com roda.
Fonte: (BRÄUNL, 2006)

O robô apresentado na Figura 3(A) possui a roda de propulsão (elemento sólido hachurado) juntamente com a direção, um motor que gira em torno de si mesmo, que direciona o sentido e o outro o movimento de propulsão do robô, na Figura 3(B) com o nome de Direção Diferencial (*differential drive*), pois consegue girar o robô em torno de si mesmo, sem a necessidade de direcionar as rodas para fazer as curvas, o que diminuir a complexidade do robô e na Figura 3(C) possui uma direção chamada Direção de *Ackermann* (*Ackermann Steering*), com dois motores, um para as rodas de propulsão e outro para as rodas de direção, esse é o padrão para os carros de passeio (BRÄUNL, 2006).

3.3 SENSOR ULTRASSÔNICO

Os princípios básicos da utilização do ultrassônico é a emissão de um som e a espera do retorno do mesmo, conhecido popularmente como eco. Por meio desse fenômeno é possível determinar se há um objeto próximo e também a sua distância, esse processo é chamado de

ecolocalização. Essa é a maneira que os morcegos e os golfinhos encontram os objetos no escuro e no fundo do mar, porém esses animais o utilizam em uma frequência mais baixa.

O sensor ultrassônico compreende dois componentes principais, um que envia o sinal e outro que o recebe, caso ocorra o retorno, como mostra a Figura 4. Também possui um componente adicional que tem a responsabilidade de determinar o tempo entre o envio e o recebimento de som. O valor desse tempo é codificado em uma tensão, quanto maior o atraso, maior a tensão, porém no Arduino sua tensão fica limitada 5V.

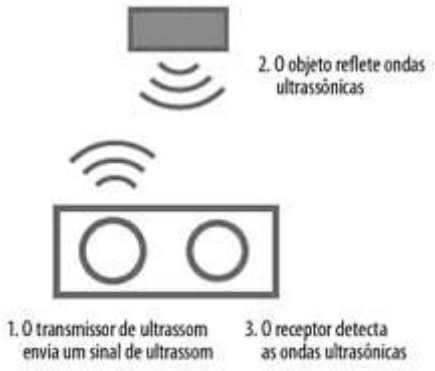

Figura 4. Como ondas de ultrassônico são transmitidas e recebidas por um sensor de distância.
Fonte: (MCROBERTS, 2013)

3.4 TRANSFORMADA DE HOUGH

A transformada de *Hough* é um método matemático utilizado para detecção de bordas, que resolve o problema encontrado na extração de primitivas em uma imagem, como a detecção de curvas analíticas do tipo segmentos de reta, círculos, elipses, etc. Tem como vantagem o fato de que pode ser aplicada a qualquer tipo de curva e apresenta grande eficiência em imagens com fortes ruídos. Já a sua desvantagem está associada a uma a curva complexa, onde há um maior número de parâmetros ocasionando maior esforço computacional.

3.4.1 HOUGH CIRCLES

A técnica de *Hough Circles* é encontrar por meio de um raio (r) estipulado, com coordenada de possível centro de circunferência (xc, yc), essa coordenada é encontrada por meio de um cálculo trigonométrico que é utilizado para gerar o espaço de Hough. O espaço de *Hough* é uma imagem com a mesma dimensão da imagem original, que inicialmente terá todos os seus pixels com o valor 0 (que corresponde a cor preta na imagem), que passarão a sofrer alterações de seus valores conforme as coordenadas encontradas, incrementando o seu valor em 1 (um, que representa a criação do espaço de *Hough*). Para a criação desse espaço é utilizado uma matriz de três dimensões representadas pelas coordenadas (x, y) do pixel e o raio de circunferência, em que o seu valor varia entre um intervalo de raio máximo e mínimo.

O seu algoritmo só realiza o cálculo trigonométrico quando é localizado o pixel que representa a borda de algum objeto, ou seja, depois do pré-processamento os pixels serão representados com o valor 255 (que corresponde a cor branca na imagem). Com uma imagem é possível encontrar vários objetos circulares de diversos tamanhos, portanto, é inviável a utilização de *Hough Circles* com o raio fixo.

A Figura 5, mostra um processamento de imagem com a utilização da transformada de *Hough Circles*, a imagem (A) mostra a imagem no seu estado original, (B) imagem pré-processada (binária), (C) aplicação do espaço de Hough, (D) Colônias de bactérias detectadas.

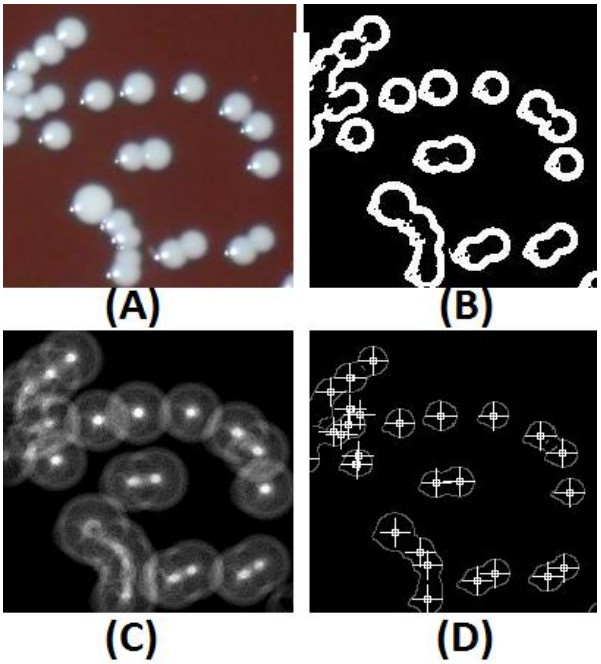

Figura 5. Processamento das Imagens de Colônias de Bactérias.
Fonte: (PINHEIRO et al., 2012)

3.5 GAUSSIAN SMOOTHING

O *Gaussian Smoothing* é um operador de convolação 2D que é usado para remover detalhes e ruídos de uma imagem. Neste sentido é semelhante aos outros filtros, mas usa um

núcleo diferente, que representa a forma de uma distribuição Gaussiana (‘em forma de sino’) corcova.

Fórmula de uma distribuição Gaussian em 2D é:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Aonde σ é o desvio padrão da distribuição.

A Figura 6 (A) mostra uma imagem original e a Figura 6 (B) um exemplo de aplicação de suavização gaussiana com σ = 10.



Figura 6. Exemplo de Aplicação Suavização Gaussiana.
Fonte: (GALVANIN, 2003)

3.6 TÉCNICA INRANGE

Esta função funciona como um filtro de passa faixa, que nesse caso irá passar somente a cor vermelha pelo filtro.

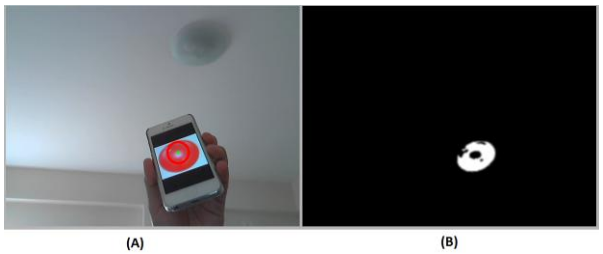

Figura 7. Exemplo utilizando a técnica InRange.

A Figura 7 (A) mostra uma imagem original e a Figura 7 (B) é o resultado do filtro de cor, o programa filtra somente a coloração vermelha na imagem, que é representada pela cor branca.

4. METODOLOGIA DE USO

O início do processo se dá com a captura de uma imagem por uma câmera, nessa imagem aplica-se a transformada de *Hough*, e a análise do resultado define se foi encontrado ou não o símbolo esperado, caso positivo é vinculado sua posição e enviado o comando para que o robô execute a função desejada.

A Figura 8 mostra uma visão geral da metodologia empregada para o desenvolvimento do trabalho.

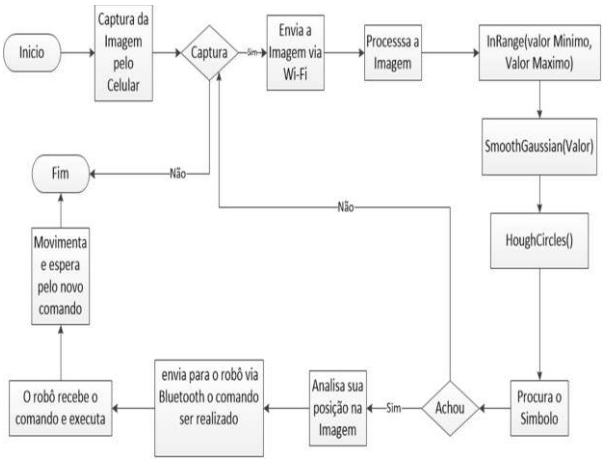

Figura 8. Fluxograma do funcionamento do sistema.

Os componentes envolvidos no desenvolvimento do trabalho são apresentados na Figura 9.



Figura 9. Funcionamento do sistema.

4.1 DESCRIÇÃO DO HARDWARE DO ROBÔ

Em virtude da complexidade desse trabalho, o hardware foi separado por dois módulos, cada um deles com a reponsabilidade de controle do robô. O primeiro módulo é chamado por Deliberativo, onde ocorre o processamento de imagens e a tomada de decisão. O segundo módulo chamado de Reativo, é responsável pelo controle dos atuadores e efetuadores do robô. A Figura 10 ilustra a arquitetura de hardware do robô AGVs.

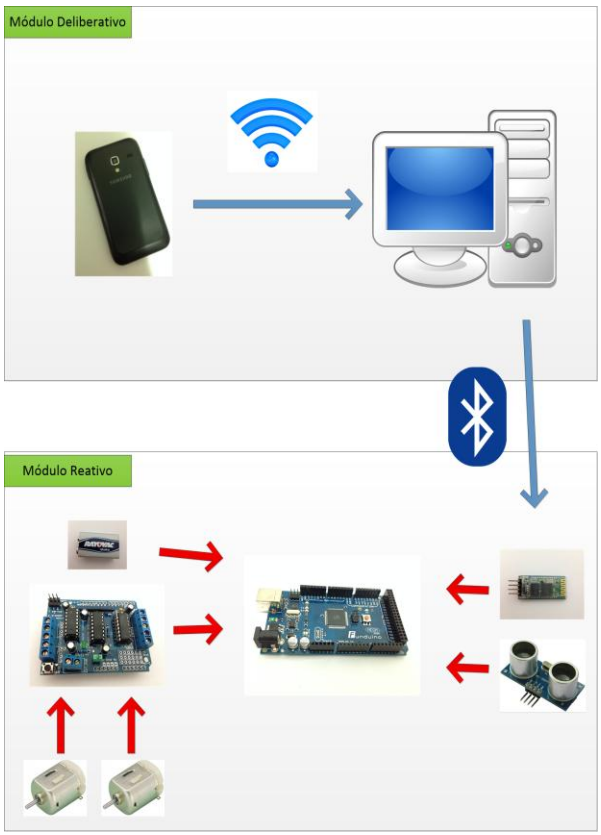

Figura 10. Arquitetura de hardware do robô.

O módulo Deliberativo é composto por um celular responsável por capturar informações do ambiente visualizado e enviar, via *wireless*, para o computador em tempo real. Esse computador, por meio do programa desenvolvido com o algoritmo da transformada de *Hough,* recebe e processa as imagens enviadas pelo celular com o intuito de localizar o símbolo e sua posição na imagem, onde a partir daí, é possível retornar para o robô, via *bluetooth,* os comandos a serem executados.

O módulo Reativo é composto pelo microcontrolador Arduino que é responsável por todo o processamento e controle de baixo nível do robô, como leitura de sensores e envio de informações aos atuadores e efetuadores. O *Motor shield* com a funcionalidade de controlar a velocidade das rodas do robô, um sensor ultrassônico para manter a distância do símbolo sem colidir e o *Bluetooth shield* para receber o comando enviado pelo computador.

Um detalhe importante, conforme consta na Figura 10, é que os módulos Deliberativo e Reativo se comunicam via *bluetooth*, que são áreas de redes pessoais sem fio.

4.2 DESCRIÇÃO DO SOFTWARE DO ROBÔ

Conforme a seção 5.1, o hardware é formado por dois módulos, o módulo Deliberativo composto por um computador local e o módulo Reativo por uma placa principal Arduino. Cada módulo é responsável por determinada função de controle do robô, desta mesma forma o sistema de controle também é modularizado, conforme mostrado na Figura 11.



Figura 11. Arquitetura de software do robô.

O módulo Deliberativo possui um celular com câmera, sistema operacional Android e um aplicativo que consegue enviar as imagens em tempo real para o computador com auxílio de uma conexão via *wireless* e um computador local com sistema operacional Windows e um programa desenvolvido que utiliza a transformada de *Hough*, para a identificação do símbolo seguidor, esse programa possui cinco etapas, a primeira recebe as imagens da câmera celular no programa via *wireless*; na segunda, aplica-se o *InRange* para identificar quais são as cores que estão dentro da faixa do símbolo procurado; na terceira, usa o filtro de *Smooth Gaussian* que é usado para imagens com "borrão", para remover detalhes e ruídos; na

quarta, executa-se a transformada de *Hough*, que neste trabalho foi utilizado o *Hough Circles* que localiza o símbolo circular; e na última e quinta etapa, é realizado o posicionamento do símbolo na imagem e o envio do comando correto para o Arduino via *bluetooth*.

O módulo Reativo é controlado por uma placa principal Arduino, onde também são executadas três etapas, a primeira etapa recebe o comando na placa do *Bluetooth shield* enviado pelo computador local pela comunicação via *bluetooth*; na segunda etapa, o Arduino interpreta o comando recebido e, se for necessário movimentar, verifica se o sensor ultrassônico não está ativado, caso esteja, significa que possui algum objeto ou símbolo próximo o suficiente para colidir com o robô. Caso esteja desativado, então o *Motor shield* é ativado; na terceira etapa o *Motor shield* recebe comando do Arduino e ativa somente os motores que necessários para se movimentar.

5 CONSIDERAÇÕES FINAIS

O sistema de processamento de imagens do robô seguidor foi desenvolvido com o uso da biblioteca OpenCV, que possui um conjunto de funções para aplicações em computação gráfica e processamento de imagens.

O principal objetivo do sistema de visão foi manter o foco da câmera no símbolo determinado, ao localizar o símbolo o sistema controla todo o processamento de imagem e informa ao sistema de controle do robô o comando necessário para que os atuadores que sejam executados no módulo Reativo. A Figura 12 mostra um exemplo do resultado de visão após localizar o símbolo.

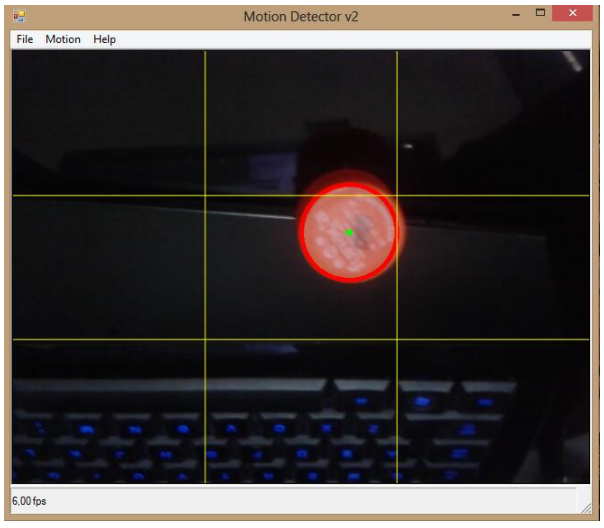

Figura 12. Exemplo de localização do símbolo.

No exemplo da Figura 12 o símbolo é um objeto circular, identificado pela cor vermelha. O sistema de visão do robô pode diminuir sua eficiência devido à alteração na luminosidade do ambiente, ou seja, ele pode não localizar o símbolo. Para a solução desse problema, foi adicionada fonte de luz para que ela tenha sua própria luminosidade e assim não depender somente da luminosidade do ambiente.

As conclusões obtidas são de que o algoritmo da transformada de *Hough* obteve um resultado satisfatório e rápido para o reconhecimento de um símbolo circular em tempo real, pois o robô funcionou corretamente conforme os comandos recebidos via *bluetooth* do controlador. O acoplamento de uma câmera celular no robô, para o envio das imagens a um computador externo, após a sua realização do reconhecimento do símbolo, é retornado para o robô via *bluetooth* um comando para ser interpretado e movimentar conforme o necessário para seguir o símbolo.

Foram realizados vários testes envolvendo fatores como luminosidade e tempo de resposta. No caso da luminosidade, foi necessária a adição de uma fonte de luz no próprio robô para que houvesse a melhora significativa na eliminação de falsos positivos. O tempo de resposta foi considerado aceitável, mas a utilização do processamento no próprio dispositivo seria o ideal para a eliminação do gargalo de comunicação entre o robô e o computador. Em termos qualitativos, definiu-se de maneira adequada todos os processos utilizados nos testes, consequentemente, a validação do artefato construído.

Para trabalhos futuros, espera-se que outros algoritmos para o reconhecimento de qualquer objeto, não apenas formas, como círculos em uma determinada cor, sejam aplicados, deixando assim, o sistema global totalmente flexível para aplicações gerais do mesmo. E também o uso de hardware que suporte elementos que forneça maior capacidade de armazenamento e processamento para melhor os resultados quanto a tempo de resposta.

REFERÊNCIAS

BRÄUNL, T. **Embedded robotics:** mobile robot design and applications with embedded systems**.** Australia, 2006.

DELAI, R.L; COELHO, A.D. **Visão computacional com a OPENCV –** material apostilado e veículo seguidor autônomo. São Caetano do Sul: Instituto de Tecnologia Mauá, 2011a.

DELAI, R.L; COELHO, A.D. Aplicação de ferramenta Open Source em sistemas de visão computacional – desenvolvimento de um veículo seguidor autônomo. In: CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA. **Anais...** Blumenau, 2011b.

FONSECA, E. G. P.; VEGA, A. S. **Tutorial sobre introdução a projetos utilizando o kit de desenvolvimento Arduino.** In: CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA. **Anais...** Blumenau, 2011.

HEINEN, F. et al. **Navegação de veículos de carga autônomos utilizando visão computacional com algoritmo de segmentação por cores.** São Leopoldo: Universidade do Vale do Rio dos Sinos, 2004.

MCROBERTS, M. **Arduino em ação.** São Paulo: Novatec, 2013.

PAOLIERI NETO F.; TOZZI C. L. A vision tracking system for robotic wheelchairs. In: WORKSHOP DE VISÃO COMPUTACIONAL. **Anais...** Campinas, 2013.

PEREIRA JUNIOR, C. A. et al. A concepção de robótica dos alunos de nível médio a partir da representação de protótipos relacionados ao conceito de titulação. In: ENCONTRO NACIONAL DE PESQUISA EM EDUCAÇÃO EM CIÊNCIAS – IX ENPEC, 9. **Anais...** Águas de Lindóia, 2013.

PINHEIRO, A. M. et al. **Contagem automática de colônias usando Hough Circles.** In: WORKSHOP DE VISÃO COMPUTACIONAL. **Anais...** Dourados, 2012.

RUSS, J. C. **The image processing handbook**. 5. Ed. Boca Raton: CRC Press, 1998.

SABBAGH, V.B. **Desenvolvimento de um sistema de controle para um veículo autônomo.** 2009. Monografia - Universidade Federal de Minas Gerais (UFMG) - Belo Horizonte - MG.

SCHNEIDER, I.R. **Uso de visão estereoscópica para detecção de obstáculos na locomoção de um veículo-autônomo**. 2012. Biblioteca Digital Universidade de Ijuí, Santa Rosa, RS.