

Descrição da Equipe SirSoccerVSS-B para a Categoria IEEE Very Small Size Soccer

Manoel Stilpen¹, Felipe Amaral¹, Samantha Vilaça¹, Marina Barros¹
Alberto Angonese^{1 2}, Eduardo Krempser^{1 3}

Abstract—Este artigo descreve os trabalhos feitos pela equipe SirSoccerVSS-B para a categoria *Very Small Size Soccer - VSSS* de Futebol de Robôs. Aqui estão descritos os principais componentes do sistema, o sistema de visão computacional, os robôs, o sistema de transmissão, questões de controle e navegação, e a estratégia de jogo.

I. INTRODUÇÃO

O Laboratório de Sistemas Inteligentes e Robótica - SIRLab, laboratório integrante da Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ, do município de Petrópolis (RJ), possui o curso de Tecnologia da Informação e da Comunicação, no qual, os alunos também tem a possibilidade de participarem de atividades extra curriculares nos demais projetos de professores da faculdade. Uma dessas iniciativas é o SIRLab, que teve sua primeira participação em competições universitárias em 2013, na Competição Brasileira de Robótica - CBR [5]. Hoje o laboratório conta com duas equipes de *Very Small Size Soccer - VSSS*, uma de simulação 2D e também com um projeto para a Mostra Nacional de Robótica - MNR.

A equipe SirSoccerVSS-B foi formada no início deste ano, com o objetivo de desenvolver uma plataforma de futebol de robôs [7] para participar da categoria IEEE *Very Small Size Soccer* na *Latin American Robotics Competition* [10]. A IEEE VSSS, é uma categoria de futebol de robôs que proporciona o estudo e aplicação em diversas áreas, como: eletrônica, visão computacional, movimentação e controle,

*Este trabalho é apoiado pelo Laboratório de Sistemas Inteligentes e Robótica - SIRLab e pela Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ

¹ Manoel Stilpen é aluno do curso de Tecnologia da Informação na Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ Petrópolis (RJ). mstilpen15@gmail.com

¹ Felipe Amaral é aluno do curso de Tecnologia da Informação na Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ Petrópolis (RJ). felipemelloamaral@gmail.com

¹ Samantha Vilaça é aluna do curso de Tecnologia da Informação na Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ Petrópolis (RJ). [santhavilaca@gmail.com](mailto:samanthavilaca@gmail.com)

¹ Marina Barros é aluna do curso de Tecnologia da Informação na Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ Petrópolis (RJ). mari.b.simas@gmail.com

^{1 2} Alberto Angonese é orientador dos projetos do SIRLab, professor do curso de Tecnologia da Informação na Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ Petrópolis (RJ), e mestre em Engenharia de Defesa pelo Instituto Militar de Engenharia - IME. angonesealberto@gmail.com

^{1 3} Eduardo Krempser é orientador dos projetos do SIRLab e professor do curso de Tecnologia da Informação na Faculdade de Educação Tecnológica do Estado do Rio de Janeiro - FAETERJ Petrópolis (RJ), e pesquisador da Fundação Osvaldo Cruz - FIOCRUZ. krempser@gmail.com

que são empregados no desenvolvimento de sistemas que devem atuar em um ambiente altamente dinâmico.

O sistema computacional utiliza uma câmera fixada a 2m acima do campo, dos robôs e da bola. O sistema é executado em um computador fora do campo, o qual envia comandos de arbitragem e posicionamento para os robôs, baseando-se na informação visual fornecida pela câmera. O computador desempenha o processamento necessário para a coordenação e controle dos robôs. A comunicação é sem fio e utiliza transmissor/receptor de rádio frequência. O sistema de visão funciona como o receptor de informações de estado do jogo, diferenciando os elementos do jogo (bola, robôs e campo) pela suas cores. A divisão de estratégia elabora quais serão as decisões tomadas pelos robôs, para as mais diversas situações que podem ser enfrentadas. Após tomada a decisão e realizada a atuação por parte de cada um dos jogadores, outra posição é captada pela câmera, e o ciclo de processo de informações ocorre novamente.

II. VISÃO COMPUTACIONAL

Uma das partes fundamentais no desenvolvimento do um time de futebol de robôs é a visão computacional [2], uma vez que ela é responsável pelo sensoriamento de todos os objetos de interesse. Visão Computacional é o conjunto de métodos e técnicas através dos quais sistemas computacionais podem ser capazes de interpretar e extrair informações a partir de imagens.

O processamento de imagens digitais consiste em melhorar ou recuperar determinadas características de uma imagem de forma a permitir ou facilitar o seu reconhecimento pelo ser humano ou por uma máquina [3], técnicas de visão computacional são utilizadas para reconhecer e obter informações de imagens através de sua análise (Figura 1).

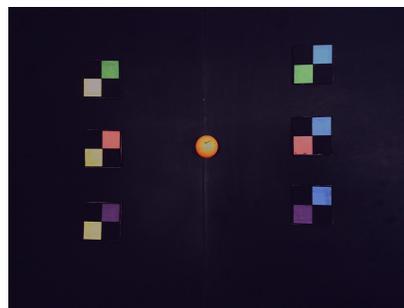


Fig. 1. Imagem captada pelo sistema de visão

Na construção do sistema de visão computacional da SirSoccerVSS-B utilizou-se a biblioteca OpenCV, pois trabalha com processamento de imagens em tempo real e pode ser desenvolvida em diferentes linguagens de programação, tais como: C, C++, CSharp, Java, Python dentre outras. Em nosso projeto optamos por trabalhar com C++, pois, além de ser leve é extremamente robusto e não promove problemas em relação à velocidade de processamento [1].

Para facilitar o processamento computacional é necessário o uso de um modelo que descreva matematicamente como uma determinada cor deve ser representada. Dentre as diversas opções que existem, nas primeiras implementações a equipe usou o modelo RGB (Figura 2). Esse modelo é usado pela maioria dos dispositivos de aquisição e visualização de imagens, podendo ser visto como um cubo de cores onde os componentes vermelho, verde e azul correspondem aos eixos x, y e z. Os tons de cinza ocorrem quando os três componentes possuem valores iguais, seguindo a diagonal principal do cubo, com a origem (0,0,0), preta, e a extremidade inversa (255,255,255), branca. Entretanto, existem outros padrões como o HSV (Hue, Saturation and Value), YUV, SCT e etc.

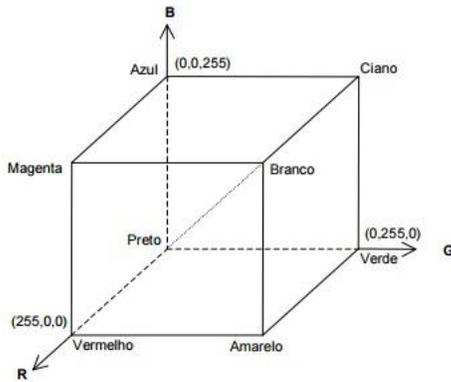


Fig. 2. Modelo RGB

Optou-se pelo HSV (Figura 3), pois contorna todos os problemas relacionados ao gradiente de luminosidade. No modelo HSV, do inglês Hue, Saturation e Value, utiliza os conceitos de crominância (tonalidade), saturação e luminância (brilho). A tonalidade representa a cor propriamente dita, a saturação representa o grau de pureza e a luminância, as informações do brilho da cor.

Na rotina de calibragem faz-se a seleção de um pixel pela sua cor (correspondente ao canal H), como propriedade dominante, mas levando em consideração também seus atributos de pureza e brilho (correspondentes respectivamente aos canais S e V). Contudo, esta técnica é aplicada apenas nas regiões de interesse definida pelos quadrados localizados na parte superior dos robôs e na bola, ao invés de toda a imagem.

O processamento ocorre através de duas etapas. Na primeira, os canais H, S e V das regiões de interesse são

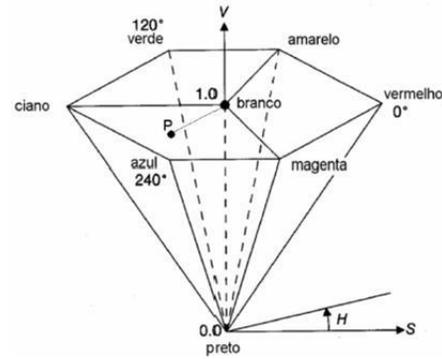


Fig. 3. Modelo HSV

limiarizados com limiares determinados empiricamente. Os limiares do canal H são intervalos de valores que representam a cor que se deseja selecionar. A limiarização do canal S seleciona os pixels com atributos de pureza maiores que determinado limiar. A limiarização do canal V elimina as regiões escuras, selecionando pixels com valores maiores que o limiar.

Na segunda etapa realiza-se operações AND entra as imagens resultantes da limiarização dos canais H, S e V. Esta técnica permite selecionar cores ao passo que filtra os ruídos provenientes do processo de aquisição de imagem, da iluminação não uniforme e da conversão do RGB para HSV que é implementada através de uma função disponibilizada na biblioteca da OpenCV. Onde, dentro do código desenvolvido, utilizamos padrões numéricos para capturar a escala padrão de cores propostas pelo range do HSV.

III. MOVIMENTAÇÃO E CONTROLE

O desenvolvimento da movimentação e controle do sistema está dividido em três partes. Estimativa de parâmetros base, estratégia, movimentação e técnicas de controle.

Para que a técnica de controle fosse implementada, e para que as técnicas de campos potenciais também pudessem ser usadas. É necessário anteriormente estimar alguns parâmetros básicos do jogo, como algumas distâncias e angulações. Como por exemplo a distância entre a bola e o robô, sendo (X_b, Y_b) as coordenadas da bola e (X_r, Y_r) as coordenadas do robô, tem-se a seguinte equação para a distância d entre esses dois pontos.

$$d = \sqrt{(X_b - X_r)^2 + (Y_b - Y_r)^2} \quad (1)$$

E a angulação Θ do robô dado um certo referencial (X_o, Y_o) no campo, é dada por.

$$\Theta = \tan^{-1} \frac{Y_r - Y_o}{X_r - X_o} \quad (2)$$

Esses parâmetros são a base para o desenvolvimento da estratégia. De posse dessas coordenadas é possível tomar algumas decisões, como por exemplo setar um ponto de projeção para o robô, ou seja, um ponto que de acordo com o

movimento da bola seja a posição ideal para fazer um gol no caso do atacante, ou para dar um passe no caso do zagueiro.

Com a coordenada correspondente ao robô e a coordenada do meio do gol adversário é possível calcularmos a função da reta entre esses dois pontos. Calculando a função dessa e a função da reta entre a bola e o meio do gol adversário e ainda as inclinações dessas retas, é possível saber se existe chance de empurrar a bola e se fazer um gol. Essa possibilidade ocorre quando as inclinações têm valores próximos. Essa é uma tática que tem trazido bons resultados durante os testes. Contudo, para que a estratégia seja efetiva os problemas de controle devem ser contornados.

O controle proporcional, integral e derivativo - PID [9] é usado para correção de erros, que são ocasionados por ruídos no sistema. O controle PID está dividido em três componentes: uma proporcional, uma integral e uma derivativa. Tendo cada um dessas componentes os respectivos parâmetros: K_p , K_i , K_d . Dada uma função $u(t)$ e uma função erro $e(t)$, o cálculo do PID é feito pela estimação do erro e aplicação deste em cada uma das componentes.

$$u(t) = kp.e(t) + kd.\frac{de(t)}{dt} + ki.\int_0^t e(t)dt \quad (3)$$

Para a movimentação tomou-se como base a metodologia de campos potenciais, em que cria-se campos de atração e repulsão para que o agente diferencie seus oponentes de um ponto objetivo e seja atraído ou repelido por eles [8]. A técnica de campos potenciais consiste na criação de campos artificiais que servem de guia na navegação dos robôs. O robô é visto como uma partícula que sofre influência de campos repulsivos ou atrativos, dependendo da estratégia adotada.

IV. TRANSMISSÃO DE DADOS

Nos trabalhos ao longo do ano foram tratados alguns aspectos no que diz respeito a segurança das informações transmitidas aos robôs. Para isso, foram implementados protocolos de segurança e criptografia, sendo eles: adição de chave pré-compartilhada entre os robôs e o transmissor principal, seleção de canais por meio de multiplexação de frequência, e definição de IP de rede diferente do padrão do XBee.

A rede se constitui de três receptores configurados como end-point, e um transmissor principal configurado como coordenador [11]. Tendo o coordenador uma frequência padrão de 2.4Ghz (Figura 4).

A rede Xbee opera na faixa de frequência de 2400MHz a 2486Mhz e é responsável por garantir toda a comunicação entre o computador e os robôs. A estrutura atual possui implementações de segurança por chave pré-compartilhada, seleção de canal por meio de multiplexação em divisão de frequências, e definição de IP de rede. Tal estrutura de rede permite que o sistema da SirSoccerVSS-B opere em um mesmo ambiente com outros sistemas que fazem uso de rádio frequência sem que sofra interferências. Os rádios Xbee, que são de baixa frequência, são os que mais sofrem interferência externa, causadas por telefones sem fio, aparelhos bluetooth e principalmente modems e roteadores *wireless*.

A multiplexação por divisão de frequências consiste em modificar a potência do rádio, somando uma constante na faixa de frequência (5MHz/canal) para produzir ondas de diferentes comprimentos, que são traduzidas pelo demultiplexador no rádio receptor.



Fig. 4. XBee S1

V. OS ROBÔS

A arquitetura foi trabalhada visando a diminuição de interferências eletromagnéticas entre os componentes. Os robôs da equipe SirSoccerVSS-B possuem Shields da DF Robot, placa Arduino Uno, shields XBee e motor Pololu.



Fig. 5. Robô SirSoccerVSS-B

O Arduino (Figura 6) [4] é a plataforma de desenvolvimento onde fica embarcada a codificação para a movimentação básica do robô, mais especificamente a codificação fica embarcada no microcontrolador ATmega328p, onde também é feito o processamento.

Os *Shields* (Figura 6) [6] servem para estender a capacidade do Arduino, podendo esses serem aplicados diretamente sobre a placa através dos conectores ou *headers*. O controle de *pulse wide-modulation* - PWM dos motores é feito pelo *shield* da DF Robot, utilizando o chip L298P.

Os motores são da marca Pololu de 6 volts, com caixa de transmissão integrada para ampliar o torque. A relação de transmissão é de 75:1, que resulta em um torque de 0,21N.m.

Por fim o *shield* XBee é responsável pela recepção de dados do computador por meio da rede ZigBee, que se disponibiliza de diferentes frequências de operação, o que pode evitar interferências.

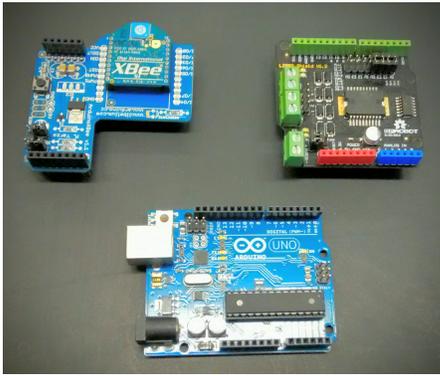


Fig. 6. Arduino e Shields do robô

VI. CONCLUSÃO

Por meio dos estudos realizados para a construção de um time de futebol de robôs da IEEE *Very Small Size Soccer - VSSS*, a equipe desenvolveu algoritmos e adquiriu experiência em temas antes nunca estudados. A interação em laboratório com membros mais antigos de outras equipes foi essencial para que alguns problemas fossem solucionados mais rápidos do que normalmente seriam. Assim, espera-se cada vez mais melhorar esse novo projeto do SIRLab afim de que a SirSoccerVSS-B consiga acompanhar as melhores equipes da categoria.

VII. AGRADECIMENTOS

Agradecemos a Faculdade de Educação Tecnológica do Estado do Rio de Janeiro (FAETERJ - Petrópolis) pelo apoio.

REFERENCES

- [1] Stroustrup, b. why c++ is not just an object-oriented programming language. disponível em: <http://www.stroustrup.com/oopsla.pdf>.
- [2] Machine vision fundamentals: How to make robots see. 2011.
- [3] Construção de um time de futebol de robôs para a categoria ieeec very small size soccer. 2015.
- [4] ARDUINO: Arduino HomePage. Disponível em: "<http://www.arduino.cc>". acessado em: 20 de maio de 2013.
- [5] CBR:Competicao Brasileira de Robotica. Disponível em: "<http://crobotica.org/>". acessado em: 30 de julho de 2014.
- [6] DFRobots HomePage. Disponível em: <http://www.dfrobot.com/index.php>. acessado em: 22 de junho de 2013.
- [7] E. B. Ferreira Filho. Desenvolvimento de um time de futebol de robôs categoria ieeec very small size. 2013.
- [8] M. A. Goodrich. Introduction potential fields tutorial, 2002.
- [9] W. A. Guerra. Implementação de controle proporcional, integral e derivativo digital em controladores lógico programáveis. 2009.
- [10] LARC:Latin American Robot Competition. <http://crobotica.org/>.
- [11] ZigBee Alliance. www.zigbee.org.