

Botcem Team Description Paper (TDP) - IEEE VSSS Category – CBR Competição Brasileira de Robótica LARC – 2022

Arthur Haickel Nina*, Alberto Maciel de Oliveira Teles Filho*, Lucas Lucateli Panke*, Luís Henrique Allebrandt Schunemann*, Vinicius de Carvalho Monteiro Longo*

*Universidade Federal de Santa Catarina (UFSC)

Resumo—Este Team Description Paper descreve todo o processo por trás do desenvolvimento do time de robôs para a competição de Very Small Size Soccer da equipe Botcem da Universidade Federal de Santa Catarina, Centro Tecnológico de Joinville, incluindo toda a parte física de modelagem 3D, eletrônica, visão computacional e programação do time de robôs.

Palavras-chave—Modelagem, Solidworks, C++, Python, Algoritmos de Planejamento de Caminho, Visão Computacional, Placa de Circuito Impresso, PCB, Proteus, Motores, Câmera, ESP8266, Microcontrolador.

Abstract—The following Team Description Paper elaborates on the development process behind team Botcem's, from the Federal University of Santa Catarina, Technological Center of Joinville, robot team for the Very Small Size Soccer competition, presenting the topics of 3D modelling, electronics, computer vision and programming of such robot team.

Index Terms—Modelling, Solidworks, C++, Python, Path Planning Algorithms, Computer Vision, Printed Circuit Board, PCB, Proteus, Motors, Camera, ESP8266, Microcontroller.

I. INTRODUÇÃO

VERY Small Size Soccer, comumente referenciado por sua sigla (VSSS), é uma modalidade de competição de robótica que envolve duas equipes adversárias de três ou cinco robôs cujo objetivo é controlar a bola disponibilizada para marcar pontos no gol inimigo. Os robôs possuem seu tamanho limitado a 7.5cm x 7.5cm x 7.5cm e devem possuir autonomia total além da habilidade de se movimentar de acordo com os comandos recebidos.

As áreas abordadas pelos competidores podem ser divididas em diversas partes, a fim de se estabelecer uma lógica mais simples e modular para o aprendizado e desenvolvimento dos robôs em conjunto. As áreas de implementação podem ser divididas em: Parte física, Visão computacional, Estratégia e Comunicação.

A parte estratégica que comanda os robôs é efetuada em Hardware externo, mais comumente sendo um computador que contém o código de estratégia e comando, se comunicando tanto com o módulo de visualização quanto com o time de robôs autônomos. A visão do campo é tomada a partir de uma câmera que enquadra a parte do campo utilizado.

A câmera utilizada envia as imagens à parte estratégica, e a última aplica filtros e simplifica a interpretação da cor, e, por conseguinte, a direção do robô, a fim de identificar a posição e a velocidade de todos os autônomos presentes no campo, assim como a da bola. Toda a informação é passada por meio de pacotes de informação, que contêm os dados mencionados anteriormente de cada robô.

A fim de controlar a direção da partida e garantir uma competição justa, um juiz virtual deve ser implementado para marcar faltas e reposicionar os jogadores caso algum evento aconteça. Além do juiz virtual, cada equipe possui seu próprio juiz humano para apurar os acontecimentos da partida.

As partidas têm dois períodos iguais, cada qual com cinco minutos com um intervalo de meio tempo de dez minutos. Em caso de empate, caso a competição não permita este tipo de resultado, o jogo será prorrogado por mais cinco minutos, que podem ser estendidos por mais três. Caso o empate persista, a partida será decidida na disputa de pênaltis entre apenas um robô e um goleiro.

Portanto, tendo em vista o escopo da competição, a equipe decidiu por fazer a impressão 3D de seus robôs com filamento de PLA, acomodando a placa de circuito impresso com as medidas precisas para o suporte. Para controle e comunicação, serão utilizados módulos ESP8266 Wi-fi da nodeMCU, que possuem uma boa gama de canais de comunicação, o que impede interferência e garante boa velocidade na transmissão de dados do computador ao campo.

II. MECÂNICA

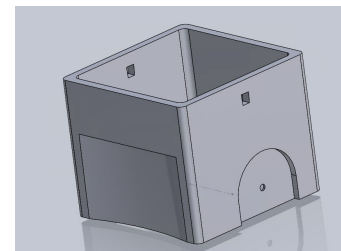


Figura 1. Modelo no Solidworks 2017. Fonte: Autor

O projeto mecânico foi desenvolvido no Solidworks 2017 e praticamente não fugiu do que se encontra no próprio GitHub do VSSS Brasil, o modelo possui concavidades na

frente e atrás para que o robô tenha algum controle de bola independente do sentido para o qual anda. As furações quadradas próximas ao topo servem para encaixar a tampa que cobre a área interna dele.

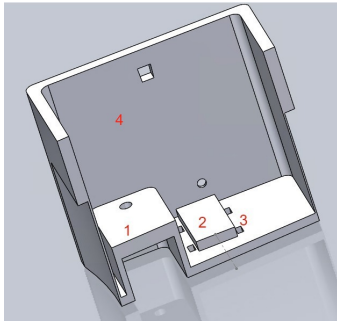


Figura 2. Corte numerado do modelo no Solidworks 2017. Fonte: Autor

Fazendo o corte no centro da carcaça percebem-se algumas furações e apêndices que serão listados:

- 1 - este é um suporte para um rolamento esférico. Como as rodas serão muito pequenas para dar suporte total ao robô, foi preferível criar um terceiro ponto de apoio, e o rolamento não agregaria atrito a níveis que trouxessem desvantagem ao time;
- 2 - encaixe para dar altura aos motores e eles se nivelarem com as rodas e o chão;
- 3 - essas furações foram feitas para passar abraçadeiras de *nylon*, percebeu-se que fixar os motores com cola quente ou fazer cavidades na impressão poderia dar muito mais trabalho caso eles sofressem dano e precisassem ser substituídos ou somente não coubessem. Essa solução garante fixação satisfatória e facilidade nos reparos;
- 4 - nesta área seria um suporte para a PCB como uma prateleira, porém na última modelagem houve algum erro e infelizmente o Solidworks não fez a visualização dessa extrusão.

III. ELETRÔNICA

A fim de suprir todas as necessidades que os três robôs autônomos possuem para efetivamente jogar uma partida, foram determinados os componentes eletrônicos utilizados e suas ligações.

A placa da Figura 3 foi desenvolvida a fim de suportar o microcontrolador utilizado, as conversões de tensão necessárias para o funcionamento de todos os componentes, as ligações para os motores que moverão o respectivo robô e a para a ponte H que irá controlá-los.

Desse modo, é importante salientar a escolha do microcontrolador ESP8266 como a base do funcionamento dos robôs. Tal controlador utilizado é relativamente compacto e possui alto poder de processamento, o que auxilia na agilidade em campo, velocidade da captação de dados e execução dos comandos, além de possuir módulos internos para comunicação sem fio, como Wi-Fi, que será utilizada para interligar a central estratégica e os robôs jogadores.

Seguindo essa esteira de raciocínio, pode-se notar também a inclusão da ponte H TB6612FNG para o controle dos

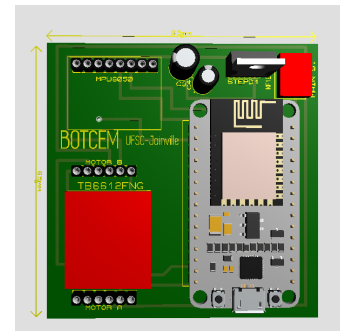


Figura 3. Placa eletrônica desenvolvida para o projeto. Fonte: Autor

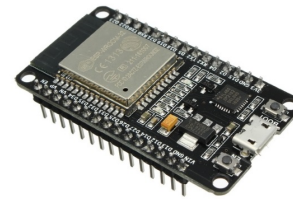


Figura 4. Microcontrolador utilizado para desenvolvimento dos robôs.[1]

motores. Por ser baseada em MOSFET e não em TBJs, é mais eficiente e pode controlar até dois motores DC independentes. É comumente utilizada em projetos open source e amadores, facilitando consideravelmente seu uso devido ao alto número de projetos disponíveis abertamente.

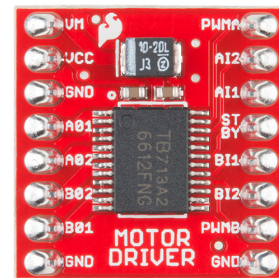


Figura 5. Driver utilizado para controle dos motores. [2]

O módulo MPU6050 foi incluído para a melhor aquisição de dados e controle dos dispositivos jogadores. Por possuir magnetômetro, acelerômetro e giroscópio inclusos, é um módulo eletrônico compacto que fornece grande quantidade de informações importantes para o controle da movimentação no campo.

IV. PROGRAMAÇÃO

A equipe se dividiu em duas linhas de pesquisa, visando aproveitar o melhor de 2 algoritmos de busca e planejamento de caminho e possivelmente mesclá-los em um só código,



Figura 6. Módulo MPU6050 para controle e captação de dados físicos.[3]

eles foram implementados para guiar os robôs dentro do campo em situação de jogo. Obstáculos e objetivos foram determinados para os jogadores perseguirem ou evitarem. Para isso, algoritmos de busca com amostragem de dados do jogo foram implementadas a fim de se otimizar a velocidade e eficiência dos robôs dentro de uma partida.

Um dos métodos de planejamento de trajetória implementados foi o método de pesquisa RRT* (Rapidly-Exploring Random Trees). Tal método se apoia em uma árvore de busca iterativamente expandida através de pontos randômicos e sua aproximação com o objetivo final.

O algoritmo RRT* começa definindo um número de pontos randômicos dentro de uma circunferência ao redor do ponto inicial (Centro do robô). O raio de tal circunferência é definido, assim como o número de pontos traçados.

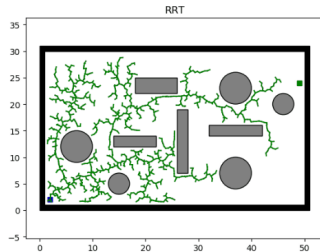


Figura 7. Método RRT* implementado em planejamento de caminho. Fonte: Huiming Zhou [4]

Os pontos que foram traçados em uma área inválida, como dentro de obstáculos ou fora do campo de jogo, são descartados, e aqueles dentro de áreas válidas são analisados. Dentre os pontos analisados, o que mais se aproxima do ponto objetivo é selecionado como próximo ponto da trajetória, e, caso todos os pontos sejam inválidos, outro sorteio de direções é realizado.

Estendendo as informações disponibilizadas anteriormente, entende-se que o método de busca através do Campo Potencial também foi implementado. Este consiste na criação de campos artificiais de vetores de ação.

O planejamento de trajetória se dá em tempo real, sendo alterado à medida em que os agentes se movem no campo. Eles podem se dividir em:

- o robô - ele é o agente principal, seu objetivo é caminhar para a configuração com menor energia potencial possível, isto é, seguir os vetores no espaço com gradiente negativo[5];

- obstáculos - quaisquer agentes móveis ou não que opunham o caminho traçado pelo robô, neste caso, os robôs do time adversário. Eles possuem gradiente mais positivo, ou seja, têm potencial que repele o robô;
- objetivo - quaisquer agentes móveis ou não que determinam o ponto final da trajetória. Possuem gradiente mais negativo, ou seja, potencial atrativo. Dependendo da situação de jogo, pode ser a bola, uma posição de defesa no campo ou o gol.

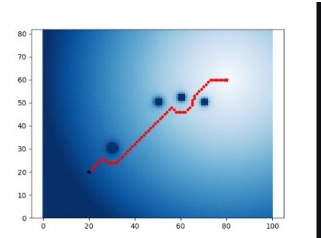


Figura 8. Mapa gaussiano de Campo Potencial Artificial. Fonte: Atsushi Sakai [6]

Comparando-se os dois algoritmos, o RRT* mostrou-se bastante otimizado e de fácil aplicação e grande flexibilidade dos seus parâmetros, enquanto o Campo Potencial Artificial apresentou alguns *bugs* e certa dificuldade em diminuir o tempo de suas tarefas a fim de realizar o mapeamento em $\frac{1}{120}$ segundo (tempo de um quadro no módulo de câmera escolhido). A proposta inicial é manter-se com o primeiro algoritmo funcional, mas o trabalho ainda não foi acabado e o segundo ainda tem potencial para ser explorado e melhorado.

V. VISÃO COMPUTACIONAL

A fim de possibilitar a captação de dados precisos da situação de campo, um sistema de visão computacional deve ser implementado. Para isso, é importante saber como a implementação da comunicação funciona e os protocolos utilizados.

O sistema de captação de imagem escolhido foi uma câmera de 120fps que será ligada diretamente a uma porta USB. A imagem produzida por essa câmera será tratada e simplificada por diversos filtros aplicados a fim de se extrair as coordenadas dos robôs presentes no campo e da bola.

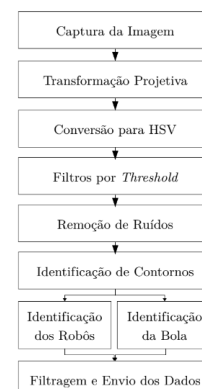


Figura 9. Esquema básico dos filtros de imagem utilizados. Fonte: Resende [7]

Desse modo, entende-se transformação projetiva como transformar a imagem recebida de modo que seu resultado inclua somente o campo centralizado, retirando os limites do campos e os colocando como limite da imagem. Nessa esteira de raciocínio, converte-se a imagem recebida RGB em uma imagem HSB, para a análise isolada dos campos de saturação. A diferenciação dos times se dá pelas posições recebidas dos pontos destacados e, depois, utiliza-se o canal matiz para separar suas cores.

Desse modo, utiliza-se um filtro por Treshold para filtrar cores e destacá-las, utiliza-se o filtro "Erosão" 7x7 para captar somente a região das etiquetas dos robôs para a remoção de ruído, usa-se o filtro "dilatação" para a resolução do problema de diminuição de imagem causada pelo filtro "Erosão". Por fim, utiliza-se o filtro de Kalman para filtrar os dados recebidos.

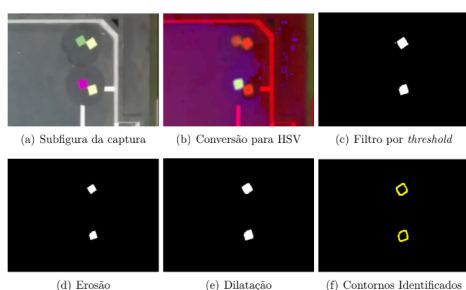


Figura 10. Processo de filtragem de imagem para extração de dados. Fonte: Chaves [8]

Os dados são enviados a central estratégica para a tomada de decisão, e o operador humano pode observar e estudar o comportamento do Software implementado através da API de código aberto OpenCV.

Sob essa perspectiva, é importante ressaltar que a aplicação dos filtros e o método de tomada de dados é baseado no sistema de visão computacional SSL-Vision. [9]

VI. CONCLUSÃO

Relembrando todo o desenvolvimento da categoria na equipe e tendo em vista a posição atual de estreantes na competição, as expectativas são bastante positivas para um primeiro projeto desse escopo.

Por isso, planos futuros, como a implementação de um software de aprendizado profundo para os jogadores, incremento na taxa de quadros da câmera, alteração dos motores utilizados e revisão das equações de controle implementadas, já estão sendo traçados.

REFERÊNCIAS

- [1] NodeMCU, "Esp8266," Disponível em: Link, acesso em: 23 fev. 2022.
- [2] Toshiba, "Bi-cd integrated circuit silicon monolithic tb6612fng," 2007.
- [3] I. Inc., "Mpu-6000 and mpu-6050 product specification revision 3.4," 2013.
- [4] H. Zhou, "Pathplanning," Disponível em: Link, 2020, acesso em: 26 de abril 2022.
- [5] A. Ojha, "Robot motion planning - artificial potential field method," Disponível em: Link, 2020, acesso em: 26 de maio 2022.
- [6] A. Sakai, "Pythonrobotics," Disponível em: Link, 2016, acesso em: 26 de abril 2022.
- [7] J. C. M. de Resende, "Mapeamento e controle de robôs móveis para a categoria ieev very small size de futebol de robôs," 2019.
- [8] R. A. Chaves, "Uaivision: um sistema de visão para futebol de robôs," 2019.
- [9] Robocup, "Ssl-vision," Disponível em: <https://github.com/RoboCup-SSL/ssl-vision/wiki>, 2008.