

Preparation of Papers for IEEE Sponsored Conferences & Symposia*

Abstract—This paper describes the electronic, mechanical and software designs developed by the RoboIME Team in order to join the LARC 2017. The overall concepts are in agreement with the rules of Small Size League 2017. This is the fourth time RoboIME participates in the LARC.

I. INTRODUCTION

RoboIME é um time de Very Small Size do Instituto Militar de Engenharia, IME, no Rio de Janeiro. Essa é a primeira vez que a equipe participa de uma competição.

Todos os estudantes que trabalham no projeto da VSS são membros do Laboratório de Robótica e Inteligência Computacional do IME.

Esse artigo descreve as informações e avanços que a equipe obteve nesse ano, abordando os seguintes tópicos:

II. DESIGN DE SOFTWARE

Nosso software atual foi desenvolvido em Labview 2017 devido ao seu potencial de escalabilidade e às ferramentas que ele dispõe para a medição de entidades físicas e para análise gráfica de sistemas. O fluxo de informações do software - como descrito na figura 1 - composto de cinco estágios: Recepção, Pré-processamento, Tomada de Decisão (Módulo de Inteligência Artificial), Controle e Transmissão. Na Recepção, pacotes do SSLVision são recebidos e, em seguida, pré-processados no módulo Gamestate. No Gamestate, a informação da Visão é filtrada usando Filtro de Kalman.

O estágio de Controle, no qual são recebidos os comandos do sistema de IA, executa algoritmos com o intuito de controlar o movimento dos robôs através do cálculo das velocidades apropriadas. Em seguida, no estágio de Transmissão, o módulo de Transmissão envia essas velocidades para cada robô, para que eles possam desempenhar o movimento desejado. O estágio de Tomada de Decisão - que é composto pela Estratégia (Módulo de Personalidades) - e a Tática são descritos posteriormente.

A. Gamestate

1) *Estimativa usando Filtro de Kalman*: O filtro de Kalman usa os dados disponíveis para estimar valores com maior precisão e obter informações que não eram disponíveis antes (uma boa referência inicial para filtro de Kalman encontrada em [1]). O modelo cinético aplicado nos objetos, o que melhora nossos valores instantâneos de posição, velocidade e aceleração e ajuda a prever situações futuras, mesmo sem o auxílio do SSLVision. Outras funcionalidades podem ser incluídas nesse filtro, como por exemplo o recebimento das velocidades instantâneas dos robôs.

B. Sistema de IA

1) *Personalidades*: A atual arquitetura do nosso sistema de IA recebe a informação pré-processada do sistema de visão e, então, executa a personalidade definida para cada robô dentro de campo. Existem três personalidades principais - Atacante, Defensor e Goleiro -, cada uma das quais executa uma função diferente na dinâmica de jogo. Abaixo, temos a estratégia utilizada / descrição do comportamento de cada personalidade.

C. Módulo Tático

- **Atacante**: Com a informação fornecida pelo sistema visual, possível estimar as aberturas no gol inimigo, para então determinar em que intervalo do gol ele deve jogar a bola. Após definido o alvo, aplicado o método GoTo para definir as velocidades do atacante de modo a posicionar o robô apropriadamente com o intuito de levar a bola até o ponto-alvo, tenha ele a posse de bola ou não. Se o atacante estiver próximo o suficiente da bola e posicionada como desejado, o robô irá empurrá-la em direção ao gol.
- **Defensor**: O papel do defensor no time é evitar problemas para o goleiro e ajudar o atacante a ter a posse de bola. Sendo assim, ele irá fazer uma barreira entre o gol e a bola, como se fosse um goleiro fora de posição, caso o atacante inimigo não esteja sendo pressionado pelo nosso. Caso contrário, ele irá se posicionar do lado do atacante inimigo para atrapalhar seu deslocamento e facilitar o roubo de bola.
- **Goleiro**: A principal função do goleiro é interceptar a bola antes que ela alcance o gol. Desse modo, quando o inimigo tem a posse de bola, o goleiro é enviado para onde o robô inimigo está direcionado. Se esta posição estiver fora do gol, o goleiro permanecerá na extremidade do gol mais próxima posição final estimada para a bola. Se o inimigo não tiver a posse da bola, mas a bola estiver se movendo em direção ao nosso lado, a posição final da bola é estimada e o goleiro é enviado para lá. Se esta posição estiver fora do gol, o goleiro permanecerá na extremidade do gol mais próxima posição final estimada para a bola.

D. Controle

1) *Desvio de obstáculos*: Com o intuito de evitar colisões com outros robôs, o algoritmo RRT utilizado para o planejamento de trajetórias (Leitura posterior pode ser encontrada em [2]). Ele busca uma trajetória a partir de um estado inicial até o objetivo, expandindo uma árvore de busca. Caso uma colisão ocorra, ele retorna um estado vazio e estende a árvore utilizando como alvo o objetivo ou um ponto aleatório, o que garante que o algoritmo encontre um caminho sem obstáculos.

III. PROJETO ELETRICO

A. Firmware

For LARC 2016, the team developed a new firmware in C++ applying concepts of hardware abstraction and object orientation. Thus, there are abstract classes, representing the robots sensors, actuators and its microcontroller resources, and concrete classes, which implement the methods defined in the abstract classes. This way, the firmware becomes closer to a development platform that can be studied, improved and can support changes as well as serving as a foundation to new projects. The firmware is embedded in a STM32F4 Discovery board with an ARM cortex-M4 microcontroller [9]. The C++ language was chosen due to its object oriented nature, as well as other benefits to the project, such as inheritance and polymorphism. The main functions of the firmware are receiving and sending information to our computer and the control over the robot. The communication occurs inside an infinite loop, while the control is performed through interrupts. 1) Abstract classes: The functionality of classes is declared by abstract classes and implemented by specific codes for each hardware. For example: abstract class motor, implemented by the classes motor brushless and motor ponteH The firmware has the following classes that implement functionalities:[2]: GPIO (sets usable IO pins), SPI, ADC (analog to digital converter), PWM, Interrupt-Timer (implements an interruption through a timer), Encoder, Motor, Robo (implements an interface to all robots sensors and actuators), NRF24L01P (transmits information between computer and robot).

IV. PROJETO MECNICO

Nossos robs da VSS, com estrutura cbica comportando as dimenses mximas de 75x75x75mm, foram estruturados de forma simples e compacta, tendo como foco principal maior agilidade e facilidade de transporte de bola.

A. Design

Utilizamos o software Solid Works para o design 3D (desenho e junção das partes) em CAD dos robs. Como forma dos robs alojarem e controlarem a bola com mais preciso durante a partida, foram adicionadas cavidades ligeiramente rasas nas partes frontal e traseira. Os robs so compostos de duas rodas laterais de tamanho ..., que permitem mudana de direo apenas pela diferença de velocidade entre elas. A tampa superior dos robs encaixada através de im, para facilidade de desmontagem caso seja necessário mexer em algum componente interno.

B. Manufatura

Pela simplicidade dos robs da VSSL, após feito o design 3D em CAD, a equipe optou pelo uso de impressora 3D com material ABS, para impresso das peças, tendo como vantagem a fácil substituição e troca de componentes quebrados da carenagem do rob. Foram realizadas diversas alterações na estrutura do rob até que conseguíssemos chegar no formato atual

APPENDIX

Appendices should appear before the acknowledgment.

ACKNOWLEDGMENT

The preferred spelling of the word acknowledgment in America is without an e after the g. Avoid the stilted expression, One of us (R. B. G.) thanks . . . Instead, try R. B. G. thanks. Put sponsor acknowledgments in the unnumbered footnote on the first page.

References are important to the reader; therefore, each citation must be complete and correct. If at all possible, references should be commonly available publications.

REFERENCES

- [1] Bzarg <http://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/>, 2015
- [2] Ramon Jansen. Waypoint navigation with obstacle avoidance for mavs. 2016.