

# IMPLEMENTAÇÃO DE UM SISTEMA MULTI-AGENTE PARA A CAPTURA E PRÉ-PROCESSAMENTO DE IMAGENS DE AFÍDEOS

PE03191219/139

Daniel Nicolau Saito (Discente - IFSul Câmpus Passo Fundo– Ciências da Computação – daniel\_fuzion@hotmail.com )  
João Mário Lopes Brezolin (Docente Orientador - IFSul Câmpus Passo Fundo – DEPEX – joao.brezolin@passofundo.ifsul.edu.br)

PASSO FUNDO

13°  
JIC  
IFSul

JORNADA DE  
INICIAÇÃO CIENTÍFICA DO  
INSTITUTO FEDERAL  
SUL-RIO-GRANDENSE  
2020



## INTRODUÇÃO

Os insetos exercem um papel fundamental para o equilíbrio ecológico. Para garantir sua subsistência, entretanto, alimentam-se de culturas de grãos como o trigo, o que causa perdas significativas dessas culturas. Dessa forma, faz-se necessário o monitoramento das populações de insetos que atacam as plantações.

Este projeto, uma parceria com a Embrapa Trigo, busca estabelecer formas de automatizar esse monitoramento. Observou-se que, com o uso de técnicas de inteligência artificial, se-ria possível classificar os afídeos eficazmente. Com o decorrer da pesquisa, entretanto, percebeu-se que o uso de uma rede neural convolucional proporcionaria melhores resultados. No início do projeto foi modelada uma rede neural que foi treinada utilizando momentos invariantes, perímetro e área de insetos. Ela demonstrou bons resultados, com em torno de 86% de acurácia. A modelagem da rede neural e o processamento das imagens do dataset permitiram um desempenho de 93%, podendo chegar a valores maiores. Assim, conclui-se que é viável o uso de redes neurais para a classificação desses insetos. Entretanto, o custo computacional para o processamento dessas imagens era alto, uma vez que as imagens utilizadas para o treinamento da rede estavam no formato JPEG, com resolução 8000x8000 pixels e que não havia uma interface para a captura das imagens.

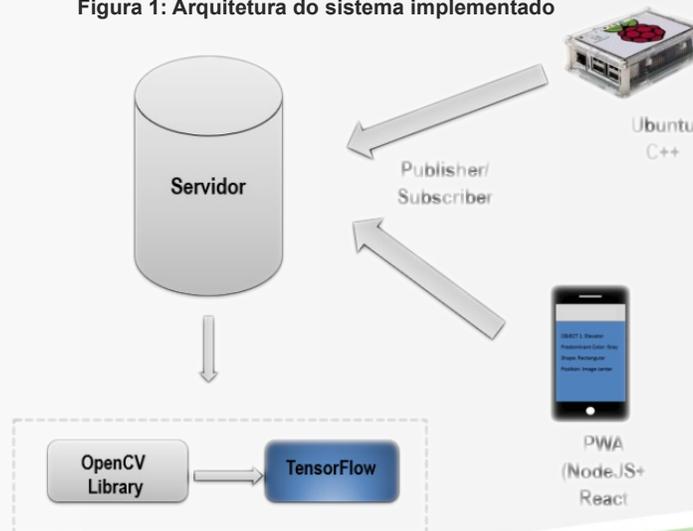
## OBJETIVOS

O presente projeto busca estabelecer uma infraestrutura local para captação e processamento de imagens. Para a implementação dessa infraestrutura utilizou-se o framework ROS (*Robot Operational System*) [O'Kane,2013]. O ROS estabelece as bases para a implementação de um sistema multi-agente composto por módulos independentes que controlam o processo de captação das imagens.

## METODOLOGIA

O ROS é um middleware que será a base da comunicação entre cada agente. Para validar o uso do framework foi implementado um protótipo composto por módulos, programados em C++ para sistemas embarcados, e em javascript ( Nodejs + ReactJs + rosnodejs) para smartphones. Ambos transmitem, via arquitetura publisher/subscriber, as imagens pré-processadas para o servidor. Essa transmissão ocorre através de canais dedicados que cada agente tem com o servidor, como demonstrado na Figura 1. O servidor as lê e as classifica. Com os smartphones a comunicação é feita com um plugin do ROS chamado ROSBridge, que cria uma conexão websocket para o recebimento das imagens.

Figura 1: Arquitetura do sistema implementado



Fonte: Autoria própria

Os testes realizados demonstraram correta a transmissão de imagens para o servidor através do ROS, mesmo com mais de um dispositivo ao mesmo tempo. Ou seja, não houve perdas, e as imagens foram armazenadas em pastas respectivas a cada canal de comunicação, que é exclusivo de seu dispositivo ativo. Essas imagens então foram encaminhadas para o módulo que realiza a classificação dos objetos, programado em Python, com as bibliotecas do TensorFlow e OpenCV.

## CONCLUSÕES

Observou-se, dessa maneira, que o framework ROS é adequado aos objetivos propostos, permitindo a implementação de um estrutura escalável que poderá ser expandida. O protótipo implementado será futuramente adequado a uma armadilha de sucção, o que poderá tornar o processo de monitoramento das populações de insetos mais preciso.

## REFERÊNCIAS

- Bordini, R. H., Hübner, J. F., and Wooldridge, M. (2007). **Programming Multi-Agent Systems in AgentSpeak Using Jason**. John Wiley & Sons.
- O'Kane, Jason M. **A Gentle Introduction to ROS**. CreateSpace Independent Publishing Platform, 2013. <http://www.cse.sc.edu/~jokane/agitr/>
- OpenCV. (2020) . **Open source computer vision and machine learning software library**. In: opencv.org.
- TensorFlow. (2020). **An open source machine learning library for research and production**. In: www.tensorflow.org.