

## PS 2023 - Especificação projeto de Drones

A equipe de drones do RobôCIn é uma equipe responsável pela construção de drones autônomos com o objetivo de participar de competições e desenvolver tecnologias voltadas a aplicações reais. Dentre as áreas de pesquisas da equipe de drone se destacam o desenvolvimento de estudos em **Visão Computacional, Navegação e Controle, Inteligência Artificial e Odometria Visual**.

### Descrição do problema

Atualmente, o principal desafio da equipe de drones é a LARC/CBR, nela participamos da Flying Robot Trial League. Nessa competição, existem quatro fases nas quais devemos resolver problemas diferentes, porém, apesar das especificidades de cada um, é possível aproveitar boa parte do código de uma fase na outra. Por isso, é importante que a base de funcionamento seja bem feita e, para isso, será necessário um bom desenvolvimento na parte de **Visão Computacional, Inteligência Artificial, Navegação e Odometria**.

### Projeto da Seletiva

Dado que, no momento, os principais desafios da equipe de drones estão relacionados às áreas de **Decisão e Movimentação e Odometria Visual**, esse ano, serão propostos dois desafios para a seletiva:

### Desafio de Decisão e Movimentação:

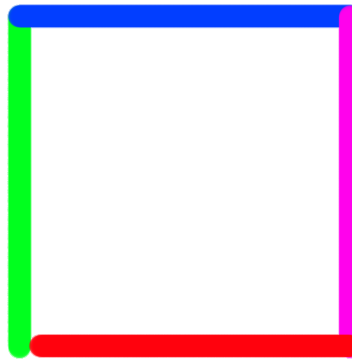
Robôs autônomos devem ser capazes de observar o ambiente ao seu redor através de sensores e, com base nas informações captadas, tomar decisões inteligentes e executar ações por conta própria. Todo esse processo muitas vezes envolve múltiplos softwares que possuem responsabilidades independentes e específicas, mas que precisam se comunicar para que o sistema funcione como um todo. A orquestração das diferentes partes de um sistema robótico autônomo é tão importante quanto as partes em si, e pode ser um fator decisivo para o sucesso do mesmo.

No contexto de drones autônomos, essa orquestração entre as partes pode ser exemplificada por um desafio que exige que as imagens capturadas por uma câmera à bordo do drone sejam processadas em tempo real para detecção de bases de pouso, e esses dados de localização das bases sejam utilizados para determinar como o drone deve se movimentar para chegar até tal base, movimentação essa que será executada através de comandos enviados do nosso software à controladora do drone.

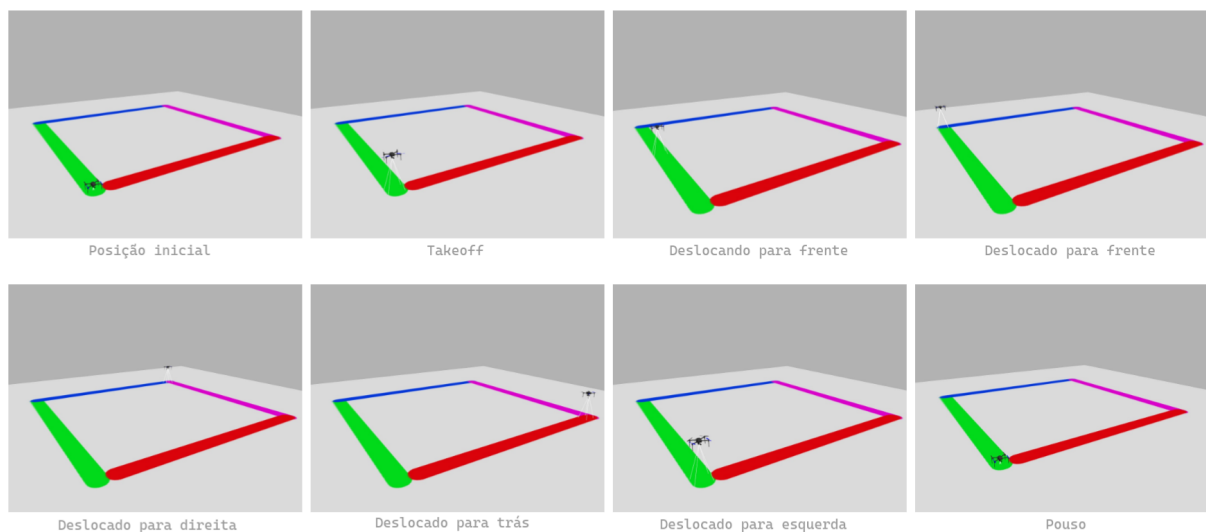
Devido à complexidade do desafio exemplificado acima (em especial a detecção de bases e o cálculo da posição real tridimensional das mesmas, além do consequente movimento necessário para chegar até a base, entre outros), é útil criar abstrações,

desafios menores que servirão como ponto de partida para construir um sistema maior. Dessa forma, **propomos um desafio simplificado**:

Um quadrado é pintado com quatro cores diferentes no chão. A imagem abaixo representa como esse quadrado foi pintado.



Um drone, munido de uma câmera RGB que aponta para o chão, é posicionado no canto inferior esquerdo do quadrado, e deve, ao ser comandado, realizar o **takeoff** (levantar vôo) e percorrer o quadrado, passando por cada um dos cantos uma única vez, até voltar à posição inicial, momento em que deve pousar.



Uma possível configuração do sistema para resolver esse desafio seria:

1. Um software que pega as imagens capturadas em tempo real pela câmera e detecta a cor a qual o drone está imediatamente em cima.
2. Um software que inicia mediante comando e decide qual movimentação deve ser realizada de acordo com regras de movimentação pré-definidas e a cor atual detectada através da câmera, e envia os comandos necessários para que o drone se desloque e eventualmente pouse.

As regras poderiam ser:

- ao receber um comando no terminal, dê takeoff (levante vôo)
- ao detectar verde, mova-se para frente
- ao detectar azul, mova-se para a direita
- ao detectar rosa, mova-se para trás
- ao detectar vermelho, mova-se para a esquerda
- ao detectar verde após vermelho, pare e pouse

Ao resolver esse desafio, é necessário pensar e mexer em áreas como visão computacional, comunicação com o drone, comunicação entre os diferentes softwares que compõem o sistema, máquinas de estado e simulação (já que não será possível utilizar um drone físico real durante a seletiva).

## O Projeto

O projeto consiste em desenvolver um sistema de decisão e movimentação para um drone autônomo capaz de reconhecer a cor do chão (na região logo abaixo do drone) e se mover de acordo com regras pré-definidas, com o objetivo de cumprir o desafio proposto acima. Para isso, será necessário fazer o setup completo do ambiente de simulação e da infraestrutura de comunicação entre os softwares, assim como a comunicação com o drone. O sistema desenvolvido deverá ser demonstrado em funcionamento dentro do simulador, com o quadrado “pintado” no chão, e o drone se movimentando como descrito.

## Formato da Entrega

A entrega será composta por 2 itens:

- Um repositório no github contendo a **implementação do seu projeto** para o desafio proposto.
- Uma **apresentação final** que você fará para nós contendo:
  - O seu projeto em execução resolvendo o desafio + breve explicação do seu código e design de projeto;
  - Seu relato contando como foi a realização do projeto, quais foram as maiores dificuldades e desafios que você enfrentou e, em especial, o que você aprendeu com a realização do projeto.

Mesmo que não complete o projeto / resolva o desafio completamente, faça a entrega! Será válido ver o que você conseguiu fazer, ouvir suas maiores dificuldades, o que você tentou fazer para resolvê-las e quais seriam seus próximos passos para tentar resolver o problema.

## Na sua apresentação, tente responder às seguintes perguntas:

1. Quais foram as maiores dificuldades enfrentadas durante a elaboração do projeto?
2. O que você aprendeu pesquisando e desenvolvendo o projeto?
3. Como você daria continuidade ao projeto? Existem pontos em que suas abordagens podem ser melhoradas ou otimizadas?

## Links Úteis

Preparamos os materiais abaixo para explicar um pouco melhor toda a infraestrutura base na qual você poderá construir o seu projeto. É altamente recomendado que você os leia/assista e busque entender.

1. Para uma visão de alto nível de como tudo funciona, disponibilizamos [essa página](#) que explica alguns conceitos importantes.
2. Recomendamos [essa playlist](#) para realizar o setup e aprender o básico a respeito da infraestrutura de comunicação entre softwares (ROS 2)
3. Disponibilizamos [esse tutorial](#) que explica, passo a passo, como realizar o setup do ambiente de simulação (Gazebo11 e Ardupilot SITL), comunicação com o drone (Dronekit) e toda a configuração necessária para capturar as imagens da câmera em tempo real. Em cada passo, damos também uma explicação conceitual breve.
4. Disponibilizamos o mundo gazebo com o quadrado pintado no chão + o drone [neste link](#).

## Desafio de Odometria Visual:

No desenvolvimento de veículos autônomos, uma das partes mais importantes para um bom funcionamento é a localização com relação ao ambiente. Para um robô se locomover para um determinado local através de algum comando ou livremente ele precisa saber o quanto ele está se movendo ao longo do tempo, em veículos terrestres isso pode ser estimado pela quantidade de revoluções feitas pela roda. Quando nos deparamos com veículos aéreos aumentamos o desafio pois não possuímos uma forma simples e precisa de analisar essa movimentação e, além disso, temos que considerar um novo eixo de movimentação, a altura.

Com isso em mente, existem técnicas de estimativa que utilizam câmeras acopladas a esses veículos aéreos que estimam a distância percorrida pela comparação entre imagens em diferentes instantes de tempo, essa técnica é chamada de odometria visual. Para o desafio que estamos propondo, queremos que você utilize alguma biblioteca open-source que faça uso de imagens de câmeras para estimar a posição ao longo do tempo, tendo em mente algumas particularidades do projeto que estamos desenvolvendo. É essencial que a aplicação funcione em tempo real, porém ela pode ser validada em datasets disponíveis na internet como o [KITTI](#) e o Euroc Mav Dataset. Para o

desafio, deve-se elaborar um documento com explicações conceituais, justificativas de escolhas para o projeto e validações.

## Especificações

- Utilizamos uma Jetson Nano 2Gb como computador de bordo, ela possui uma GPU dedicada com um poder de processamento razoável porém com limitações para algoritmos computacionalmente custosos.
- Utilizamos uma rede neural de detecção de objetos em tempo real que estaria rodando paralelamente ao algoritmo de localização, portanto, ele deve ser o mais computacionalmente eficiente possível, atendendo as necessidades.
- O mapa da competição possui algumas bases fixas e outras móveis, portanto, ele não pode ser previamente mapeado, já que algumas estruturas estarão em lugares aleatórios no momento dos desafios.
- No momento possuímos 2 câmeras monoculares e 2 câmeras de profundidade Intel R200, além de uma IMU interna da controladora. Atente-se a esse detalhe no momento de escolher formas de resolver o problema proposto.

## Formato da Entrega

- Um repositório no github contendo a implementação de seu projeto.
- Um documento com explicações conceituais e justificativas de escolhas na realização do desafio.

## Na sua apresentação, tente responder às seguintes perguntas:

1. Quais foram as maiores dificuldades enfrentadas durante a elaboração do projeto?
2. O que você aprendeu pesquisando e desenvolvendo o projeto?
3. Como você daria continuidade ao projeto? Existem pontos em que suas abordagens podem ser melhoradas ou otimizadas?

## Links Úteis

- Recomendamos [esse repositório](#) contendo várias implementações de odometria visual e SLAM
- [Esse site](#) contém um dataset muito utilizado para validação de algoritmos de odometria visual
- [Esse artigo](#) é um resumo da literatura sobre odometria visual e um bom ponto de partida para entender os conceitos importantes e que técnicas são usadas
- <https://ardupilot.org/copter/docs/common-non-gps-navigation-landing-page.html>
- <https://springerplus.springeropen.com/articles/10.1186/s40064-016-3573-7> -
- <https://guvencetinkaya.medium.com/visual-odometry-vs-visual-slam-cdda75df592>
- <https://www.youtube.com/watch?v=EbzfjNpLzvg&t=7s>

- <https://projects.asl.ethz.ch/datasets/doku.php?id=kmaavvisualinertialdatasets>
- <https://www.youtube.com/watch?v=JXRC SovuxbA>

Ficou com alguma dúvida que não foi respondida? Entra em contato com a gente!

- Instagram: <https://www.instagram.com/robocinufpe/>
- Gmail: [robocin@cin.ufpe.br](mailto:robocin@cin.ufpe.br)