



Uma proposta de MAV autônomo e inteligente

Escola Politécnica da Universidade de São Paulo

Prof. Dr. Felipe Miguel Pait

Lucas Penna Saraiva e João Henrique Helmeister



Visão geral

Vamos tentar esboçar o modelo de um drone autônomo e inteligente, ou seja, capaz de realizar missões de forma autônoma sem intervenção humana durante a execução dessa missão.

Para isso, vamos abordar os seguintes conceitos: engenharias de hardware e software para um drone capaz de ler dígitos escritos nos produtos de uma prateleira e conseguir mapear o estoque da empresa.



Visão Geral

01

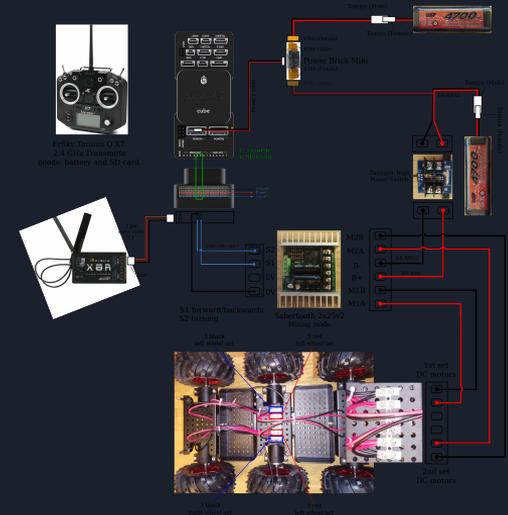
Hardware: Flight Controller Unity + Single Board Computer

02

Software: Linux Embarcado + Firmware + Rede Neural Convolutcional

01 Hardware

O hardware é composto por uma Flight Controller Unity, que nada mais é do que um hardware dedicado ao sistema operacional de tempo real do drone, responsável por rodar o firmware e executar tarefas de baixo nível, como aquisição e processamento de dados de sensores embarcados e controle dos atuadores do sistema (no caso dos drones, os motores).



01 Hardware

O hardware também é composto por uma Single Board Computer, responsável por rodar o sistema operacional de alto nível, que não atua em tempo real. Esse hardware possui linux embarcado e é onde conectamos a câmera. Sua arquitetura mais robusta nos permite, por exemplo, processar imagens onboard, tornando o veículo autônomo e independente da estação de base (computador de controle) para executar suas tomadas de decisão.





02 Software

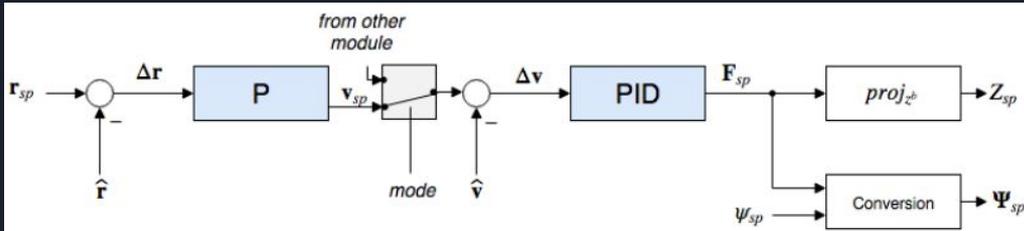
Para processamento das tarefas de baixo nível, adotamos a plataforma PX4. Trata-se do estado da arte para firmware de drones autônomos. É um projeto open Source, desenvolvido por Lorenz Meier na ETH Zurich, apoiado por empresas como Microsoft, Github, Boeing, etc.

O PX4 vai ser o firmware responsável por enviar os sinais para o electronic speed controller, dispositivo responsável pelo controle da velocidade e orientação dos motores do drone, além de executar os modelos de **CONTROLE** do drone.

02 Software

Controle

O modelo de sistema dinâmico para estabilização do multi-rotor funciona da seguinte maneira: a posição de interesse do drone é fornecida como input. Toma-se a diferença da posição de interesse (goal position) e a posição atual lida pelo drone. A diferença dessas posições passa por um controle proporcional, que transforma o delta posição em uma tupla de velocidades que o drone deve exercer para se movimentar à posição de interesse. Essa tupla de velocidades passa por um módulo que verifica o modo de voo. Diferentes decisões são tomadas baseadas no modo de voo. Essa tupla de velocidades de interesse (goal velocity) é subtraída da velocidade atual do drone. Esse delta de velocidades, então, passa por um controle proporcional integrativo derivativo cujo projeto transforma esse delta de velocidade goal e velocidade atual em uma força de empuxo. Essa força de empuxo, então, é convertida pelos blocos em aceleração vertical (Zs), e à attitude (ângulo do drone relativo ao solo). O ponto fraco é que para que o drone se estabilize em ambiente outdoor, é necessária uma boa conexão com GPS. Note que \hat{r} só pode ser estimada com confiabilidade caso o drone esteja conectado a no mínimo 10 GPS. Nem sempre isso é possível, pois há regiões de sombra de sinal; Por outro lado, quando há conexão adequada, o drone consegue calcular a posição atual com bastante precisão, sobretudo fazendo uso do filtro de Kalman estendido, que, por meio de inferência bayesiana, consegue fazer uma estimativa muito precisa da grandeza a ser calculada (nesse caso, posição).



- | | |
|-------------------------|--|
| \mathbf{r} - position | Δ - difference |
| \mathbf{v} - velocity | Z - body vertical thrust |
| Ψ - attitude | $\widehat{(x)}$ - estimated value (of x) |
| \mathbf{F} - thrust | ψ - yaw angle |
| | $(x)_{sp}$ - setpoint (of x) |
| | $proj_{z^b}$ - vector projected onto body Z-axis |



02 Software

Agora, vamos falar das tarefas de alto nível do drone.

Essencialmente, queremos um MAV capaz de identificar produtos (padrões em imagens) e montar um inventário de produtos, salvando-o em um arquivo.

Para isso, podemos treinar uma Rede Neural Artificial do tipo Convolucional para que seja capaz de classificar o que o drone está enxergando na prateleira.



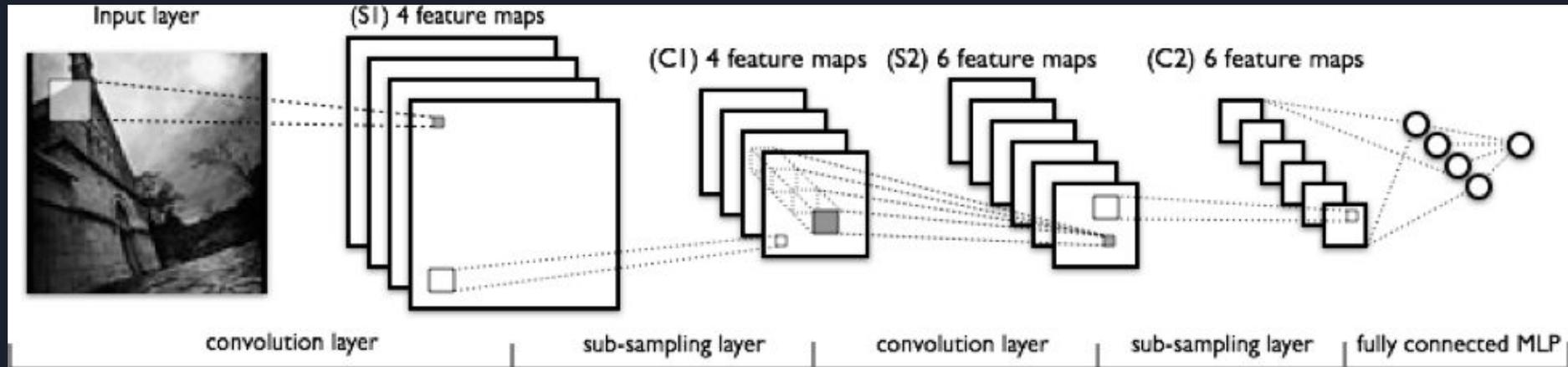
02 Software

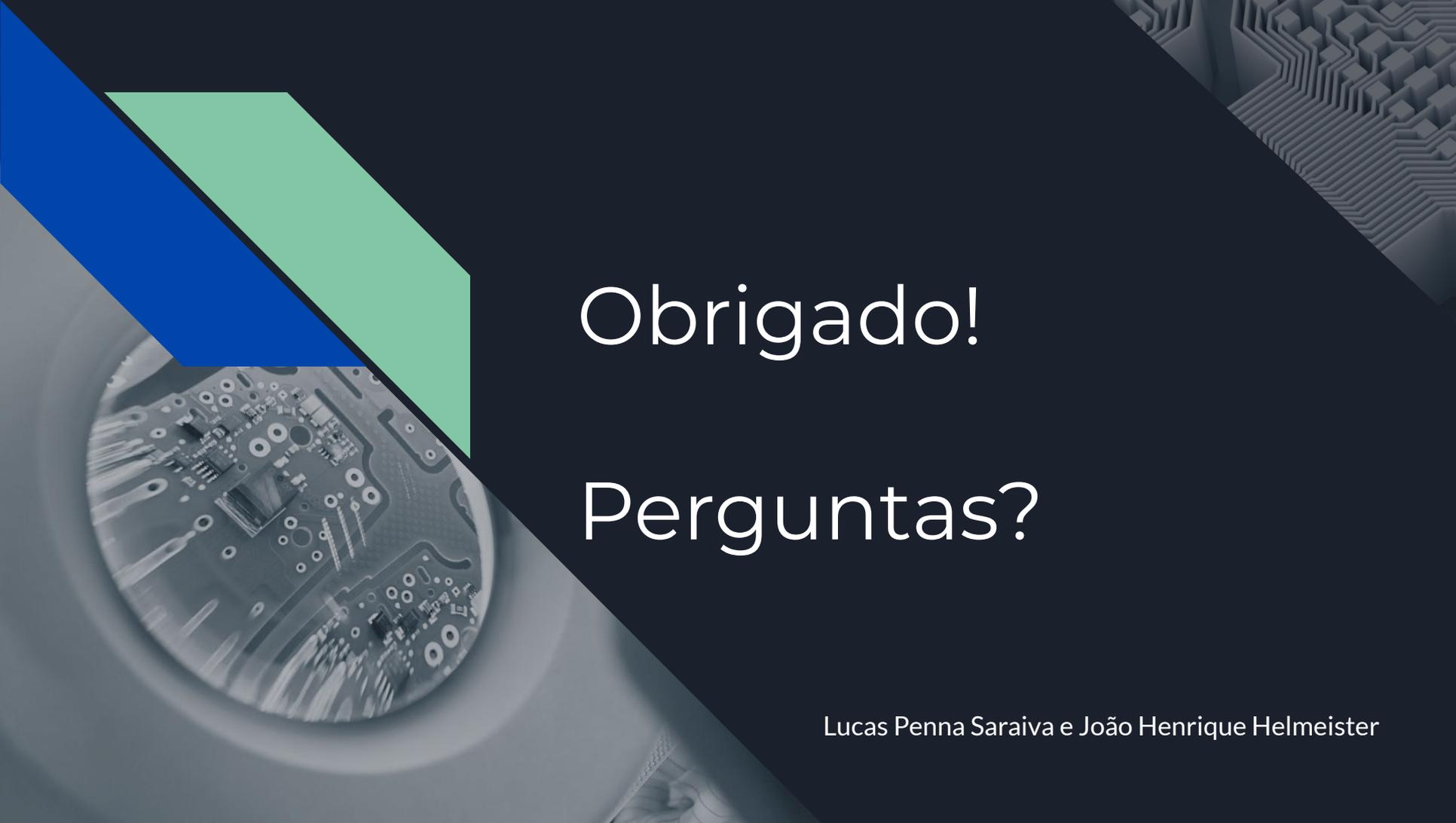
Como funciona uma Rede Neural Convolutacional?

Ela é composta por duas etapas: a etapa de feature extraction + uma etapa de classificação. A etapa de extração de features é uma série de operações de convolução entre um kernel (filtro aprendido na backpropagation) e a imagem de entrada na camada. Após uma série de convoluções, a imagem resultante entra nas camadas densas. De acordo com o Teorema da Aproximação Universal, se uma Rede Neural Densa estiver com seus parâmetros ajustados adequadamente (número de neurônios e camadas), ela é capaz de aproximar qualquer função, desde que o domínio do problema esteja suficientemente correto. Assim, ela deveria classificar corretamente os itens nas prateleiras, contando ao operador o estoque de sua fábrica, de forma autônoma e sem estresse.

02 Software

Como funciona uma Rede Neural Convolucional?





Obrigado!

Perguntas?

Lucas Penna Saraiva e João Henrique Helmeister