

Proposta de Trabalho Prático

Controlador *AR.Drone*

André Harder (2011048910) *andreharder@dcc.ufmg.br*

Jean Luc Coelho (2011049207) *jeanlucncoelho_1@hotmail.com*

1 Introdução

Neste trabalho, propõe-se como meta primária a criação de um controlador em *python* para o *AR.Drone*, um quadricóptero remotamente controlável (figura 1).

Em contraste com outros controladores já existentes, propomos um controlador que ofereça uma flexibilidade maior, provendo um nível de abstração a nível de (velocidade \times sentido).

Havendo êxito nesta tarefa, propomos como metas desejáveis, porém opcionais (considerando a complexidade das mesmas tal como possíveis imprevistos na execução da proposta central), a criação de um segundo controlador, usando redes neurais, e um uso prático de auto-locomção do robô, construído sobre algum modelo simples de localização por câmera.

2 Metodologia

O controlador será baseado em algum controlador pré-existente. Estamos atualmente considerando 2 controladores: *python-ardrone* [1] e *ardrone-autopilot* [2]. A meta é adaptar um destes controladores para possibilitar um controle mais baixo-nível. A implementação será validada por via da constatação da execução exitosa de comandos de movimento.

Havendo sucesso no anterior, deseja-se construir um segundo controlador, baseado em um controle mais primitivo do hardware, que seja aprendido por uma



Figura 1: Um *AR.Drone*

rede neural. A priori, considera-se o uso de uma rede neural recorrente, junto a extratores de fatores latentes (e.g.: *Convolutional Neural Networks*, *Stacked Denoising Autoencoders*) para auxiliar na generalização do aprendizado. A rede receberá como entrada o estado do robô e uma representação do comando do usuário, e gerará como saída uma sequência de ações a serem executadas pelo hardware do robô. Adotaremos um protocolo de aprendizado dividido em duas fases, em que (1) primeiro usamos aprendizado supervisionado para mapear as entradas (comando \times estado \rightarrow ação), durante o controle do robô por um humano, e (2) aprendizado por reforço, quando a rede neural pré treinada é posta a controlar o robô.

Em paralelo a isto, novamente na condição de não houverem imprevistos na implementação do controlador em *python*, propõe-se uma demonstração deste controlador por via de um experimento simples de localização. Estamos considerando 2 métodos de mapeamento para possibilitar a localização do robô: uso de marcadores coloridos ou relacionar localidades a fotos em arranjos tridimensionais (parecido com o que o *google street view* faz). O algoritmo de localização em si será probabilístico e ainda não foi definido.

3 Cronograma

19/10 - 25/10	Familiarização com a API/Protocolos do drone Pesquisa bibliográfica
26/10 - 01/11	Controlador em python Testes e depuração
02/11 - 08/11	(assumindo êxito no anterior) Controlador por rede neural (André) Localização com câmera (Jean Luc)
09/11 - 10/11	Elaboração da apresentação do trabalho
11/11 ou 13/11	Apresentação do trabalho

Referencias

- [1] Projeto *pythonArdrone*
<https://github.com/venthur/python-ardrone>
- [2] Projeto *ardroneAutopilot*
http://home.wlu.edu/~levys/software/ardrone_autopilot/
- [3] Página oficial do *AR.Drone*
<http://ardrone2.parrot.com>