Model-free control of a robot with vision-based sensing: a deep neural network learning approach.

Thomas Mercier

1 Introduction

1.1 Motivation

Optimal control of a fully or under-actuated linkage is a widely studied field. In some cases, embedding sensors on the system to be controlled may not be possible or preferred. This can be for cost reasons, beaucase the sensor will disturb the system (especially at small scale) or because we want to limit its complexity, as for a surgical robot that require drastic certifications for any part going inside the human body.

Commercial vision systems are becoming cheaper, more precise and constitute a versatile measurement solution. They can work from micrometer to light-year scale depending on the optics associated and their time scale can vary from a few milliseconds to several months. The large amount of data generated make machine learning techniques natural candidates for post-processing.

1.2 Approach

The dimentionality of a movie can easily exceed 10^6 (the number of pixels) which makes any direct computation completely intractable. However, the movie of a robot only covers a manifold of dimention the number of degrees of freedom of the robot, which is typically much less (from 1 to 20). In this paper, we will take advantage of this low underlaying dimentionality to train a feedback controller to perform certain tasks. The system considered are a simple pendulum and an underactuated double pendulum ("acrobot").

2 Dealing with angles

Angles constitute a special type of numbers as they are only defined modulo 2π. From a computation point of view, it can create singularities or ambiguities. Stricly speaking, angles can be seen as an equivalence class for the relation "have a difference multiple of 2π":

$$a \sim b \text{ if } \exists p \in \mathbb{Z}, a - b = 2p\pi$$

which is inconvenient to implement in practice. In order to illustrate this problem, let us consider the following problem: a neural network has n input units $a_1 \dots a_n$. We feed the layer with data so that for each data point, there exists θ such that $a_i = \cos(\theta + 2i\pi/n)$. Obviousely, the data is on a 1-dimension manifold embedded in a n-dimension space. In order to find a function that reduces the dimensionality, we build a neural network with n inputs and one output (the encoder). Also, to make sure no information is lost, we build a second network with 1 input and n outputs and train them together so that the compose of both gives the identity. This principle of encoder-decoder (or autoencoder, illustrated fig ?????????????) has been used for dimensionality reduction or data compression.

There exist a few tricks to overcome this issue:

- Forcing all angles to be in $[0, 2\pi]$
- Replacing each angle θ with $\arctan(2\theta)$
- Replacing each angle θ with a pair $(\cos\theta, \sin\theta)$

For dynamic systems, however, the two first two solutions create a discontinuity between, for instance, 0.001 and $2\pi - 0.001$ which can affect the stability of the computation. The third one performs well in terms of smoothness but will double the number of dimensions required to code an angle which is catastrophic given the curse of dimensionality. In this paper, we will use an improved version of the third solution

3 Choice of a learning architecture

Learning a controller for our robot can be devided in three steps.

1. Learning of the sensor model to reduce the dimensionality of the problem
2. Learning of the dynamic model to determine the short term response of the robot to control inputs
3. Planning the long-term trajectory of the robot to acomplicomplish the required tasks

Depending on the learning architecture, they can be treated together or separetly.