Gesture Interface

A Naive Framework

Pan An (Andy)

November 2, 2016

National University of Singapore

Table of Contents

Introduction

Overview

Gestures

- Spatial Motions
- Finger Motions
- Muscles and other Biosignals
- Possibility with hybrid hardwares

Tasks(What we can do with machines)

- Motions and movements
- Objects and obstacles
- Complex tasks and command compressing

Gestures

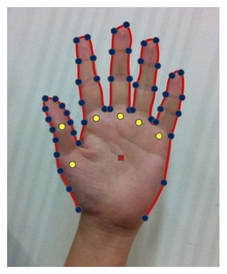


Figure 1: Example of Hand Gesture.

Color Based Method

The above picture is an example of color based gesture recognition. Color based algorithms are normally heavily relying on background color.

Spatial Motions

Swipe, tracking, clapping...

Finger Motions

Grabbing, rub, pointing and etc.

Muscles and Other Biosignals

All Involves hardwares.

Note

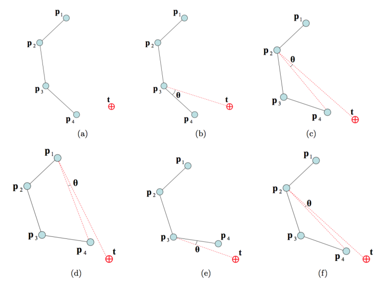
Naive camera based gestures are used.

Inverse Kinematics

IK Algorithms

- Cyclic Coordinate Descent
- Jacobian Transpose
- FABRIK
- Neural Network

Cyclic Coordinate Descent



Basic Idea

- Greedy
- Iterative
- Does not care whether the target is within range

Why

WangXin made the CCD work.

Jacobian Transpose

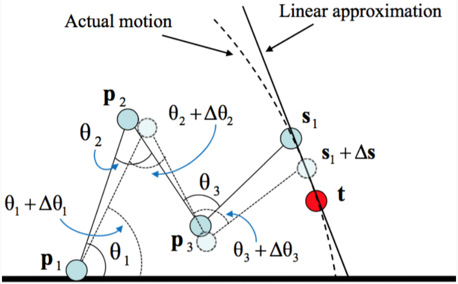


Figure 2: Jacobian Transpose

Definitions

J Partial Derivation of the entire chain system.

$\boldsymbol{\theta}$ Vector of θ values.

$\boldsymbol{s}$ Vector of end effectors.

$\boldsymbol{p}_j$ Position of the joints.

Jacobian Matrix

$$J(\boldsymbol{\theta}_{ij}) = (\frac{\partial \boldsymbol{s}_i}{\partial \theta_j})_{ij}$$

Where $i = 1, ..., k$ and $j = 1, ..., n$.
In this case $k = 1$ and $n = 3$.

Jacobian Transpose

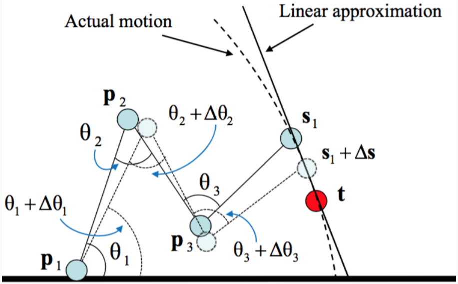


Figure 2: Jacobian Transpose

Jacobian Transpose

Jacobian Transpose is to move the angles with a step of

$$\Delta\theta = \alpha J^T \vec{\boldsymbol{e}}$$

Where $\boldsymbol{e}$ is the vector of direction of the step and α is the selected step size.

FABRIK

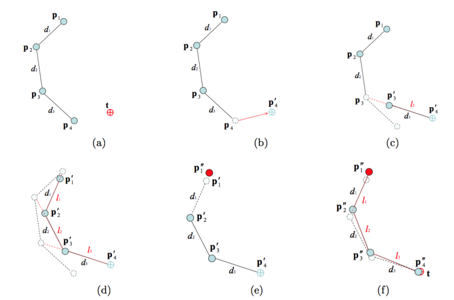


Figure 3: FABRIK

FABRIK

Stands for Forward And Backward Reaching Inverse Kinematics.

Definitions

- $\boldsymbol{t}$ Vector of targets.
- d_i Distance between each joint $d_i = |\boldsymbol{p}_{i+1} - \boldsymbol{p}_i|$

FABRIK

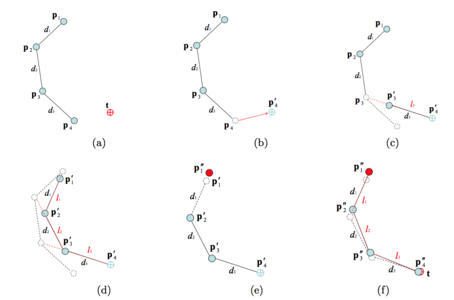


Figure 3: FABRIK

Data: $\boldsymbol{d}, \boldsymbol{t}, \boldsymbol{p}$

Result: The new joint positions $\boldsymbol{p}$

initialization;

if *Target is not within range* **then**

  point directly to the target and return;

else

  $\boldsymbol{b} = \boldsymbol{p}_1$ and $dif_A = |\boldsymbol{p}_n - \boldsymbol{t}|$;

  while $dif_A > tolerance$ **do**

    $\boldsymbol{p}_n = \boldsymbol{t}$;

    for $i = n-1, \ldots, 1$ **do**

      $r_i = |\boldsymbol{p}_{i+1} - \boldsymbol{p}_i|$;

      $\lambda_i = \frac{d_i}{r_i}$;

      $\boldsymbol{p}_i = (i - \lambda_i)\boldsymbol{p}_{i+1} - \lambda_i \boldsymbol{p}_i$;

    end

    $\boldsymbol{p}_1 = \boldsymbol{b}$;

    for $i = 1, \ldots, n-1$ **do**

      $r_i = |\boldsymbol{p}_{i+1} - \boldsymbol{p}_i|$;

      $\lambda_i = \frac{d_i}{r_i}$;

      $\boldsymbol{p}_i = (i - \lambda_i)\boldsymbol{p}_{i+1} - \lambda_i \boldsymbol{p}_i$;

    end

  end

end

FABRIK Constraints

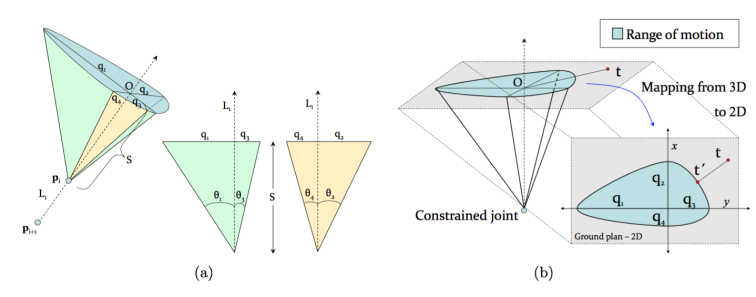


Figure 4: Example of Constraints in a Joint System

Reachable Space

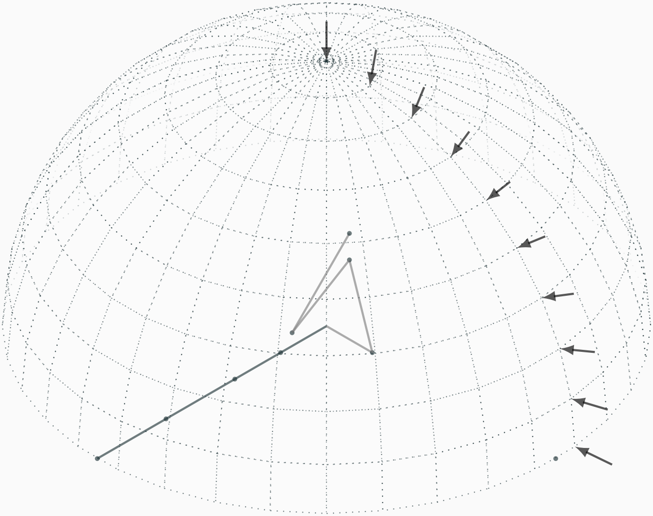


Figure 5: Reachable Space of a Fully Free Robot Arm

Reachable Space for UR10

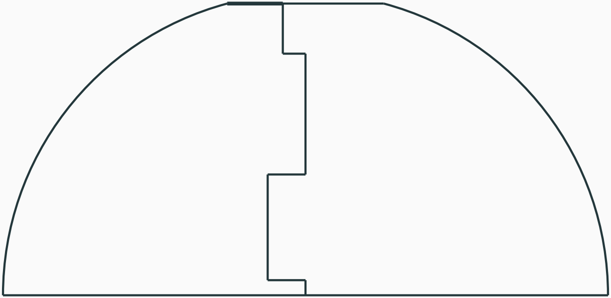
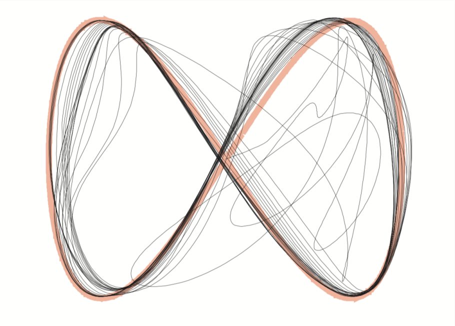


Figure 6: Reachable Space for UR10

Learning Methods

Actions(Scenarios)



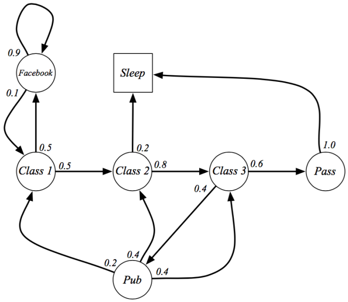
Natural Gesture of a Robot

- Unpredictability
- Robustness
- Gesture when reaching the end effector

Learning Method

- Monte Carlo(policy search)
- Dynamic Programming(value function)
- Neural Network

Model



$$
\mathcal{P} = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ Pass \\ Pub \\ FB \\ Sleep \end{array}
\begin{array}{c}
\begin{array}{ccccccc} C1 & C2 & C3 & Pass & Pub & FB & Sleep \end{array} \\
\left[\begin{array}{ccccccc}
 & 0.5 & & & & 0.5 & \\
 & & 0.8 & & & & 0.2 \\
 & & & 0.6 & 0.4 & & \\
 & & & & & & 1.0 \\
0.2 & 0.4 & 0.4 & & & & \\
0.1 & & & & & 0.9 & \\
 & & & & & & 1
\end{array}\right]
\end{array}
$$

Model

Definitions

π Policy: mapping from states to actions

S A set of States

A A set of Actions

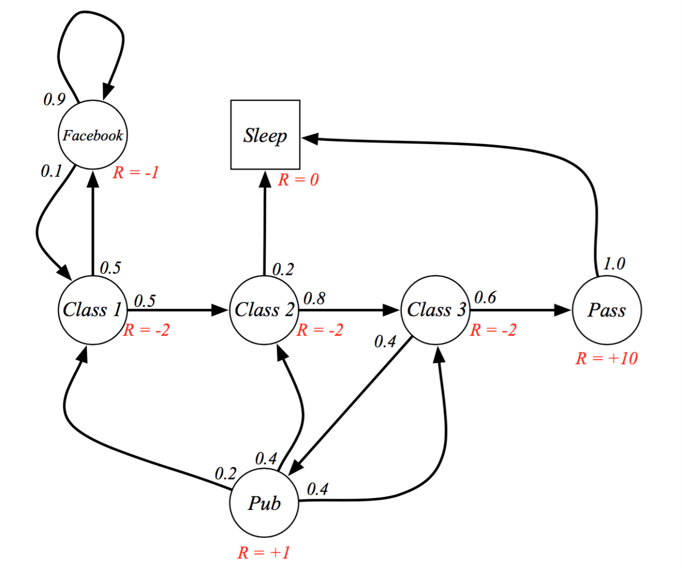
R Reward Function

P State transition function

$v(s)$ State Value Function of a MRP

γ Discount Function, $\gamma \in [0, 1]$

Model



0.9
Facebook
0.1
R = -1
Sleep
R = 0
0.5
0.2
1.0
Class 1
0.5
Class 2
0.8
Class 3
0.6
Pass
R = -2
R = -2
R = -2
0.4
R = +10
0.4
0.2
Pub
0.4
R = +1

Learning Methods

Goal

Discover an optimal policy that maximizes:

$$J = E\{\Sigma_{h=0}^{H} R_h\}$$

Where H is the steps the algorithm takes

Expanding

Expanding the above with reward settings:

$$\max_{\pi} J(\pi) = \Sigma_{s,a} \mu^{\pi}(s)\pi(s,a)R(s,a)$$

$$s.t. \mu^{\pi} = \Sigma_{s,a} \mu^{\pi}(s)\pi(s,a)T(s,a,s'), \forall s' \in S,$$

$$1 = \Sigma_{s,a} \mu^{\pi}(s)\pi(s,a)$$

$$\pi(s,a) \leq 0, \forall s \in S, a \in A$$

Where μ is the distribution of states.

Simulation

Environment

- Unity
- UR10 Avatar
- Simulated Data
- Ikpy

Questions?

Backup slides

Sometimes, it is useful to add slides at the end of your presentation to refer to during audience questions.

The best way to do this is to include the `appendixnumberbeamer` package in your preamble and call `\appendix` before your backup slides.

metropolis will automatically turn off slide numbering and progress bars for slides in the appendix.

P. Erdős.
A selection of problems and results in combinatorics.
In *Recent trends in combinatorics (Matrahaza, 1995)*, pages 1–6.
Cambridge Univ. Press, Cambridge, 1995.

R. Graham, D. Knuth, and O. Patashnik.
Concrete mathematics.
Addison-Wesley, Reading, MA, 1989.

G. D. Greenwade.
The Comprehensive Tex Archive Network (CTAN).
TUGBoat, 14(3):342–351, 1993.

D. Knuth.
Two notes on notation.
Amer. Math. Monthly, 99:403–422, 1992.

H. Simpson.
Proof of the Riemann Hypothesis.
preprint (2003), available at
http://www.math.drofnats.edu/riemann.ps, 2003.