Thesis overview

Ashesh Vasalya

November 19, 2018

Abstract

- Abstract
- Contents
- Acknowledgments
- List of Figures
- List of Tables

- Introduction

- chapter 1 — State-of-the-Art
  - Contents
  - Cognitive Robotics
  - Humanoid Robots and Neuroscience
  - Motor Contagions
  - Human Robot interaction: Physical & Non-Physical
- chapter 2 — Distinct Motor Contagions
- chapter 3 — Motor Contagions Influence Performance
- chapter 4 — HRI: human-human & human-robot dual arm object handover

- Conclusion/Future possibilities

- Apendix
- Bibliography

Algorithm 1 linear prediction controller

Input: $\mathcal{T}_R^{efL}, mocapFrameData$
Output: wp_{obj}^{efL} —way points to predicted handover location
Initial require: $t_{observe} = 10ms, t_{predict} = 1sec, i = 1$

if $mocapStart$ **and** $mocapFrameData(i)! = NaN$ **then**
  if $(i\%t_{observe}) == 0$ **then**

    $\mathcal{T}_R^{efL} \leftarrow endEffectorTask$
    $\mathcal{T}_M^{efL_{marker}} \leftarrow mocapFrameData.robotMarker((i - t_{observe}) + 1)$

    for $j \leftarrow 1$ to $t_{observe}$ **do**
      $\mathcal{T}_M^{obj_{marker}} \leftarrow mocapFrameData.objectMarker(i - t_{observe}) + j)$
      $\mathcal{T}_{obj_{marker}}^{efL} = \mathcal{T}_R^{efL^{-1}} \times \mathcal{T}_M^{obj_{marker}} \times \mathcal{T}_M^{efL_{marker}^{-1}} \times \mathcal{T}_R^{efL}$
      if $j == 1$ **then**
        $P_0^{obj_{marker}} \leftarrow \mathcal{T}_{obj_{marker}}^{efL}.translation()(1)$
      end if
      if $j == t_{observe}$ **then**
        $P_{t_{observe}}^{obj_{marker}} \leftarrow \mathcal{T}_{obj_{marker}}^{efL}.translation()(t_{observe})$
      end if
    end for

    $\mathcal{C} \leftarrow \mathcal{F}_c(P_0^{obj_{marker}}, P_{t_{observe}}^{obj_{marker}}, t_{observe})$
    $\mathcal{V}_{obj_{marker}}^{efL} \leftarrow \mathcal{F}_{diff}(\mathcal{T}_{obj_{marker}}^{efL}.translation())$
    $\bar{\mathcal{V}}_{obj_{marker}}^{efL} \leftarrow \mathcal{F}_{avg}(\mathcal{V}_{obj_{marker}}^{efL})$

<u>*predict position of object handover at* $t_{predict}$</u>
    function PREDICTPOS($\bar{\mathcal{V}}_{obj_{marker}}^{efL}, \mathcal{C}, t_{predict}$)
      $P_{t_{predict}}^{obj} \leftarrow \bar{\mathcal{V}}_{obj_{marker}}^{efL} \times t_{predict} + \mathcal{C}$
    return $P_{t_{predict}}^{obj}$
    end function

<u>*generate way points between robot left end effector and object handover location*</u>
    function GENERATEWP($P_{t_{observe}}^{obj_{marker}}, P_{t_{predict}}^{obj}, t_{predict}$)
      for $k \leftarrow 0$ to $t_{predict}$ **do**
        $wp_{obj}^{efL}(k) \leftarrow (P_{t_{predict}}^{obj} - P_{t_{observe}}^{obj_{marker}}) \times k + P_{t_{observe}}^{obj_{marker}}$
      end for
    return wp_{obj}^{efL}
    end function

  end if
  $i = i + 1$
end if

Our prediction controller behavior can be tuned by two initially required constant time periods, $t_{observe}$ —which defines the time period required to observe the motion of object and $t_{predict}$ —required to predict the object handover location in advance.

Inputs of the controller are robot left end effector pose $\mathcal{T}_R^{efL}$ and mocap markers position data in the mocap frame of reference $mocapFrameData$. $\mathcal{T}_M^{efL_{marker}}$ is the left end effector marker pose in the mocap frame. Similarly, object marker pose given by $\mathcal{T}_M^{obj_{marker}}$. For simplicity, at the moment, we have assumed zero rotation of the markers and end effector, therefore rotation part of $\mathcal{T}_R^{efL}$, $\mathcal{T}_M^{efl_{marker}}$ and $\mathcal{T}_M^{obj_{marker}}$ are Identity matrix $\mathcal{I}$.

The transformation matrix $\mathcal{T}_{obj_{marker}}^{efL}$, provides the relative pose of object marker w.r.t. robot left end effector in the robot coordinate system. $P_0^{obj_{marker}}$ $P_{t_{observe}}^{obj_{marker}}$ are the updated observed positions of object whenever the condition $(i\%t_{observe} == 0)$ satisfy. Based on the $P_0^{obj_{marker}}$ $P_{t_{observe}}^{obj_{marker}}$, average velocity $\bar{\mathcal{V}}_{obj_{marker}}^{efL}$ of the observed object motion calculated. Where, $\mathcal{F}_c$, $\mathcal{F}_{diff}$, $\mathcal{F}_{avg}$ are the *helper functions*.

Function $PREDICTPOS$, returns the predicted position of the handover $P_{t_{predict}}^{obj}$ at time $t_{predict}$.

Function $GENERATEWP$, returns the way-points wp_{obj}^{efL} between robot left end effector and object handover location, which is the final output of the controller.

Later, wp_{obj}^{efL} is being fed in the `mc_rtc positionTask`.