

KINECT

画像データ

checkerboard-detector.launch

rostopic
/camera/rgb/ObjectDetection



chanbara.l

object detection-cb 関数

euslisp内部のworld座標系
に変換する

checker-coords

while (ros::ok)

相手の剣の情報を計算

座標変換行列を用いて、剣
の先端部の座標(their-pos)
と剣の向きベクトル
(their-vector)を計算する

their-posのx座標で場合分け
(自身から近い・遠い)

≥ 500

相手に近づく

自分の中心から、相
手への角度 θ を計算

$\theta > 20$

右に回る

$\theta < -20$

左に回る

$-20 \leq \theta \leq 20$

直進

< 500

parry 関数

相手の攻撃を防御する

ロボットの剣を表すベクトルを求める
their-vectorと、(1 0 0)の外積を取るこ
とで、相手の剣に直交するベクトルを
得る
長さは200にする

my-vector

get-hand-coordinates 関数

ロボットの手先の座標を得る

座標の位置

相手の剣の先端からロボットの剣のベ
クトルを引いたもの
(their-pos) - (my-vector)

座標の姿勢

ロボットの剣のベクトルの向きをz軸とす
る、姿勢行列を計算する。

相手の攻撃をブロック!

逆運動学を解く

z軸についてのみ回転拘束をかける

2軸に2つの逆運動学計算が必要である
(send *dxl-armed-turtlebot* :inverse-
kinematics hand-c :rotation-axis :z
:revert-if-fail nil)

← hand-c