Note for running Julia on ROS:

Installation:

1. Install Julia terminal
2. Cd to current work folder: /Users/changliu/Documents/Git/Autonomous\_agent\_search/single\_agent\_search/exp\_julia/scripts
3. Start Julia: /Applications/Julia-0.4.3.app/Contents/Resources/julia/bin/Julia (on Mac)
4. Install packages that are used in script using Pkg.add(“package\_name”) (JuMP, RobotOS, MAT, Interpolations, Ipopt, Polynomials, etc.)
5. It’s a good habit to test packages after installing them using Pkg.text(“package\_name”)
6. Can use Pkg.update() to update packages after installation
7. Run a script in Julia:
   1. Method 1 (when Julia is not started): /Applications/Julia-0.4.3.app/Contents/Resources/julia/bin/julia testController.jl
   2. Method 2 (when Julia is on): include("testController.jl")

Note: after modifying a file, we need to run “workspace()” in julia terminal to clean previous workspace. Otherwise, new edits will not be used unless quitting and restarting Julia terminal.

Usage:

1. After writing the code in Julia, follow the “ROS integration” section at <https://github.com/phobon/RobotOS.jl> , especially including #!/usr/bin/env julia in the file (e.g. example.jl)that are used as rosrun file.
2. chmod +x example.jl (make the example.jl an executable)
3. if there are some modifications in example.jl, may need to catkin\_make (not sure if this is a must, but highly recommended)
4. several ways of running:
   1. rosrun mpc\_ros example.jl
   2. (in /mpc\_ros/src) julia example.jl
   3. launch julia terminal first, then

include (“example.jl”)

Sensor model:

Correct reading:

Missing reading:

out-of-FOV reading:

fake reading:

Remaining issues:

1. Conversion of different coordinates (camera, robot, global) is needed, e.g. fovModel is affected.

Files:

1. testController.jl is for testing controller.jl
2. map\_julia.jl is a previous version for running with ROS
3. controller.jl contain codes for connecting with ROS

Termination check

CarMpcUtils

CarMpc

Solve MPC

Update map

Controller.jl