**ACTIVE VISUAL TRACKING IN MULTI-AGENT SCENARIOS**

Yiming Wang and Andrea Cavallaro

Date: 29 August 2017

Requested citation acknowledgement when using this software:

Yiming Wang and Andrea Cavallaro, "**Active visual tracking in multi-agent scenarios**", Proceedings of the 14th IEEE International Conference on Advanced Video and Signal-based Surveillance, Lecce Puglia, August 2017.

1. Introduction

This software implements the proposed adaptive Optimal Reciprocal Collision Avoidance (A-ORCA) for multi-robot collision avoidance in active visual tracking applications. Each camera-equipped robot is following its target person with the objective of maintaining its target centered at the camera’s field of view. When targets are intersecting simultaneously or densely around, robots may not have accessible collision-free robotic controls which can worsen the on-target view maintenance. A-ORCA aims to increase the accessible collision-free controls by adapting the responsibility shared by a pair of robots during collision avoidance. The performance is compared with ORCA and Reciprocal Velocity Obstacle (RVO).

1. How to run the software?

* Start MATLAB on MS Windows XP or later (the software has been tested on MATLAB 8.1.0.604 (R2016a) on a PC with specifications: Intel (R) Core (TM) i5-2500K CPU @ 3.30 GHz, 8.0 GB RAM,64-bit).
* Set path of MATLAB to <.\code>.
* Run <.\code\main\_once.m>.

NB: Running the software clears the MATLAB workspace and closes the already opened figure(s). The main\_once.m file runs the experiment with one specific setting. You can vary the settings to check the performance of algorithms, e.g. the scenario type, the number of targets and the method. Resulted trajectories of mobile cameras and targets are visualised at run time to facilitate users to evaluate the performance. The figures generated at run time can be saved to folder <.\code\RawImage\ Screenshot> using an external figure saving tool in folder <.\code\figuremaker>

* Four scenarios presented in the paper can be generated with provided scripts and will be stored in <.\code\Setup> folder.
  + **Scenario I** with two targets intersecting with varying angles. Files are named as <EvaluationTracks\_TwoIntersect\_2\_1\_1\_*X*>, where *X* start from 1 to 10 with increasing angles.
  + **Scenario II** with increasing targets intersecting simultaneously. Files are named as <EvaluationTracks\_MultiIntersect\_*Nt*\_1\_1\_*X*>, where *Nt* is the number of targets intersecting. As the intersecting angle is randomly set, *X* is the trajectories generated at each run.
  + **Scenario III** with increasing targets moving with varying direction to create sparse target-intersecting cases. Files are named as <EvaluationTracks\_CircleIntersect\_*Nt*\_1\_0\_*X*>, where *Nt* is the number of targets intersecting. As the direction is varied randomly, *X* is the trajectories generated at each run.
  + **Scenario IV** with trajectories extracted from the PETS2009 dataset. The file is named as <EvaluationTracks\_PET>.
  + **Setups of mobile cameras** corresponding to each scenario are saved in files named as <MobileCameras\_*XXX*>, where XXX is the corresponding trajectory setup. For example, “MobileCameras\_CircleIntersect\_20\_1\_0\_10” is the mobile camera setup with respect to Scenario III with 20 targets at the 10th run.
* The source code for generating testing trajectories and setups for mobile cameras are provided:
  + TrajCamInitTwoIntersect.m is for scenario I
  + TrajCamMultipleIntersect.m is for Scenario II
  + TrajCamInitCircle.m is for Scenario III
  + TrajCamInitPET.m is for Scenario IV. TrajCamInitPET\_T.m provides a visualisation for the target’s position over time. GetTrajPET09.m extracts the trajectories from PETS2009 sequence and changes them to the format as other tested scenarios, i.e. the position at each second in meter unit.
* MakeVideo.m generates a video using the figures that are generated over time, and save the video file to a folder named as \RawVideo.
* ChangeCamSetting.m quickly changes the settings for a mobile camera including the car length, the view range and view angle.
* Detailed documentation of each script/function is provided inside each file.

1. License

This software is provided under the terms and conditions of the creative commons public license. Please refer to the file <License.doc> for more information.

1. Contact

If you have any queries, please contact [yiming.wang@qmul.ac.uk](mailto:yiming.wang@qmul.ac.uk) OR wangyimingkaren@gmail.com

Thanks for your interest,

Yiming Wang and Andrea Cavallaro