Polytechnique Montreal

GenTrO User Guide

Generic Tracker Optimization

Dariush Ettehadieh
12-17-2014

Overview

GenTrO is a Python tool for the optimization of trackers for specific scenes. This is accomplished by the fitting of tracker parameters to best match pre-established ground-truth tracks for a short calibration sequence.

Setup

The first step in optimizing a tracker for a given scene is to provide both a short segment of said scene and a manually-tracked ground-truth for performance evaluation. The segment should be selected so as to best represent the complexity of the scene (including lower complexity movement, if possible). It should also be at least one minute long, though a compromise must be made: longer segments are likely to produce better results, but may also take substantially longer to optimize.

Annotation of the ground-truth is perhaps most easily performed using the annotation app developed by JP Jodoin, which can be found here:

https://dl.dropboxusercontent.com/u/6705425/AnnotationApp_20140124_win32.zip

Setup of the algorithm is done through three text files: setup.ini, staticParameters, and variableParameters.

setup.ini

This setup file defines the parameters of the algorithm itself, most notably the commands required to run the tracker to be optimized.

[ConfigFiles]

nconfigs: integer number of configuration files used by the tracker and containing variables to be calibrated

configN: (0-4) name or path of the tracker's configuration files

[HomographyOptions]

no_homography: (0 or 1) if set to 1, no homography will be applied to the ground-truth. If the tracker itself uses homography, this should be kept at 0.

include_homo_altitude_mod: (0 or 1) whether the homography plane is to be predefined and fixed (0) or at a variable elevation to be optimized by GenTrO (1). For tall objects – e.g. pedestrians – using 1 is recommended.

shift_gt_homo: (0 or 1) whether or not to apply different homography planes for the ground-truth and tracker output. Set to 1 if objects are expected to be detected at points consistently higher or lower than the center of their bounding boxes.

metersperpixel: (float) meters per pixel in the world image used for point correspondence, if applicable

homo_filename: filename to use for the homography matrix

point_corr_filename: filename to use for point correspondence

gthomo_filename: filename for the homography of the ground-truth.

videoframefile: filename or path of the screenshot to be used for point correspondence

worldfile: filename or path of the scale map, to be used for point correspondence

[RunSettings]

nrunlines = (int) number of commands required to fully run the tracker

runlineN = single commands required to run the tracker

[GeneralSettings]

weight_mota: (float, 0-1) relative weight of MOTA during optimization; the weight of MOTP is this value subtracted from 1

max_iterations: (int) maximum number of iterations to run the algorithm for

relative_change: (float) relative size of changes to make to parameters at each iteration; should be left at 1

max_delta_i = (int) number of iterations after which, if no performance improvements are made, relative_change will automatically be reduced

max_n_changes: (int) maximum number of parameters to modify at each iteration

storage_filename: filename of CSV file within which to log optimization data

video_filename: filename or path of the calibration video

ground_truth_sqlite: filename or path of the sqlite file containing the ground-truth for the calibration video

sqlite_filename: filename or path to of the tracker's output

[OptimizationParameters]

These parameters affect the simulated annealing algorithm itself. Most should be left at default values, with the following exception:

max_match_dist: (float) maximum distance to use for matchmaking; is either in meters or pixels, depending on whether or not homography is being used

variableParameters.txt

This file defines the parameters to be optimized by GenTrO. Each parameter is defined by the following format:

Config ID, parameter name, variable type, change type, default value, min. value, max. value, change size

Config ID: ID number of the tracker configuration file in which the parameter is located. Corresponds to the IDs provided in setup.ini.

Parameter Name: Name of the parameter as it appears in the configuration file.

Variable type: data type of the variable, either **int, float** or **bool**

Change type: Type of operation to perform on the variable; options depend on data type:

- **int: add**
- **float: add** or **ratio**
- **bool: bool**

Default value: initial value to use

Min. value: minimum allowable value. If the variable is Boolean, this can be either possible value.

Max. value: maximum allowable value. If the variable is Boolean, this is the alternate value.

Change size: Maximum size of change to make to the parameter at each iteration.

An example is given below:

0,example-variable-parameter1,int,add,6,1,10,1

0,example-variable-parameter2,float,ratio,0.5,0,1,2

1,example-variable-parameter3,bool,bool,true,false,true,bool

static_parameter.txt

This file defines the tracker parameters contained in the modified configuration files (and defined in variable parameters) but which are not to be optimized. These are defined simply by specifying the configuration file in which they are to be written, along with the expected contents (including both the parameter name and its value) as in the example below:

0, example-static-parameter1 = 4

1, example-static-parameter2 = videoFilename.avi