DECEMBER 12, 2014

USING GENTRO WITH TRAFFIC INTELLIGENCE

A QUICK GUIDE

DARIUSH ETTEHADIEH
POLYTECHNIQUE MONTREAL
drushkey@gmail.com

The version of GenTrO available on Github is already setup for use with Traffic Intelligence (TI). Use with said tracker therefore only requires a number of simple steps in order to prepare it for new scenes.

1 – Prepare the calibration sequence

Calibration of TI is performed on short, representative segments taken from the larger video – or set of videos – to be analyzed. This segment can be extracted by any number of video editing tools, e.g. Movie Maker, available free from Microsoft.

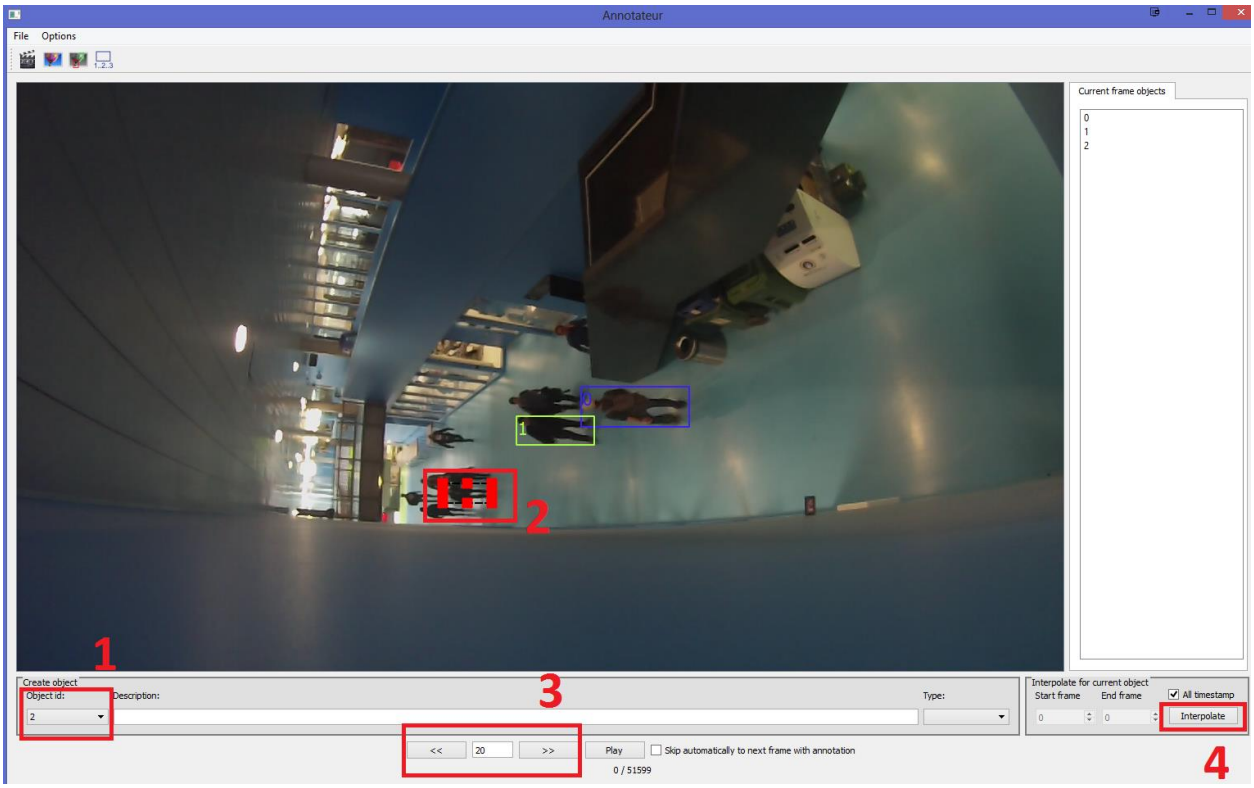
Ideally, this segment should represent the full range of complexity of the studied scene, including if possible the full range of common movements, as well as a variety of object densities if applicable. For example, if studying an intersection, this segment should include all signal phases. Longer segments are likely to produce slightly better-performing parameters for TI, but will also prolong the optimization process. As a rule of thumb, the video should be kept between one and two minutes in length.

2 – Prepare the ground-truth tracks

These tracks will provide the "correct" tracks to which TI will be calibrated. These are best extracted with the annotation application developed by Jean-Philippe Jodoin and available at:

http://www.clubic.com/telecharger-fiche35534-k-lite-codec-pack.html

Using this application, generating the ground-truth tracks is simple:



1: Create a new object.

2: Define the bounding box for the object at the current frame.

3: Jump ahead N frames; since the app will interpolate positions between defined frames, N can be 20 or more frames unless the object is moving in a particularly complex manner. Repeat this step until either the object leaves the frame or the video ends; once either of these occurs:

4: Select the object, and click “Interpolate” to automatically generate the object positions in the previously skipped frames.

Repeat these steps until all objects have been tracked, then save the track database in the PolyTrack format.

3 – Extract a single frame from the video

A sample image is required to establish point correspondence, and thus translate in-frame positions to real-world coordinates. Most video playing applications have an option to generate an image from the current frame; the only requirement is that the size (in pixels) of this image match that of the video frames.

4 – Generate the world-image

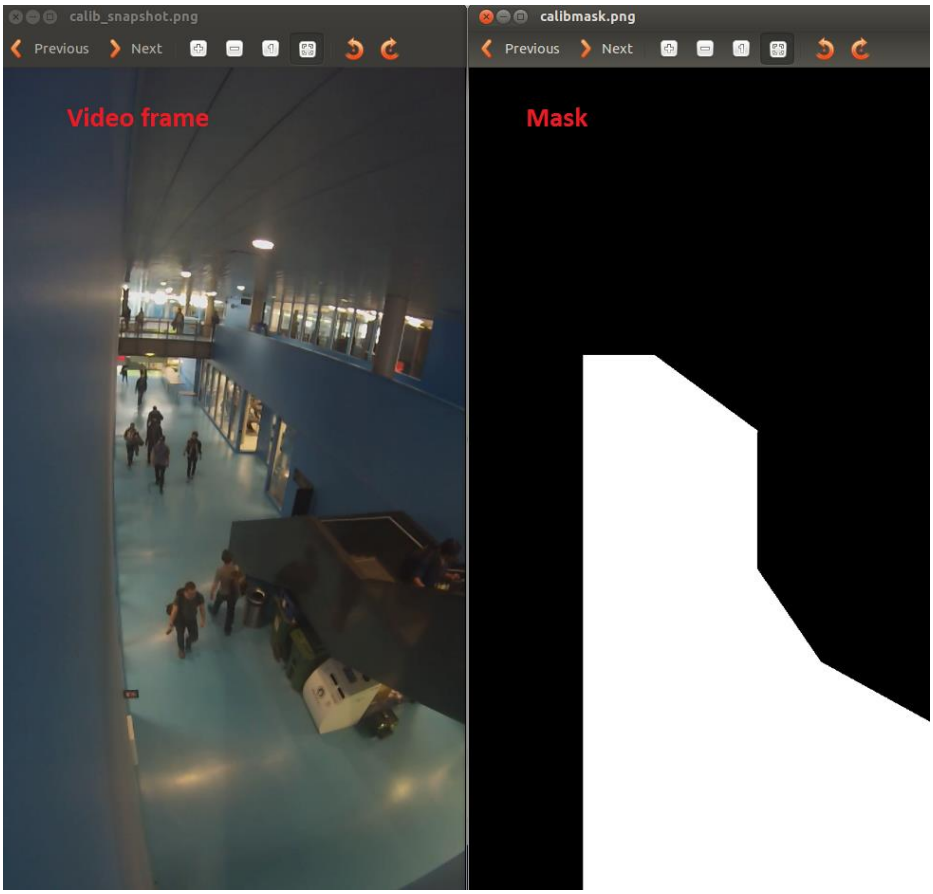
This is the second image required for point correspondence. This must be a to-scale representation of the studied area, either manually generated or taken from existing maps (e.g. Google Maps). In reality, the only parts of the area that must be accurately represented are those that will be used for point correspondence, so a schematic world-image can suffice.

5 – Determine the number of meters per pixel in the world-image

This is the final step in preparing point-correspondence. This can easily be accomplished by counting the number of pixels between two points of known distance, and dividing said distance by this number.

6 – Generate a mask image if required

A mask defines which areas of the video frames for TI to ignore. This is simply an image of identical dimensions to the video frames, where areas to track are in solid white and those to ignore in black, as in the example below:



7 – Edit the setup.ini file

This file contains the parameters of GenTrO itself, including the information required to run TI with the calibration video. Only the following parameters require adjustment:

1: Modify the filename to those desired for the current optimization. videoframefile, worldfile, ground_truth_sqlite and video_filename must correspond to the files generated in the previous steps; the other files are generated by GenTrO and can therefore be named anything.

2: Edit metersperpixel to correspond to the value from step 5.

3: Select the maximum matching distance, in meters.

staticParameters.txt × setup.ini × variableParameter

```
[ConfigFiles]
nconfigs: 1
config0: atrium.cfg

[HomographyOptions]
no_homography: 0
include_homo_altitude_mod: 1
shift_gt_homo: 1
metersperpixel: 0.01123
homo_filename: atriumhomo.txt
point_corr_filename: atriumpc.txt
gthomo_filename: atriumhomogt.txt
videoframefile: PolyAtriumSnapshot.png
worldfile: PolyAtriumSnapshot.png

[RunSettings]
nrunlines = 2
runline0 = feature-based-tracking atrium.cfg --tf
runline1 = feature-based-tracking atrium.cfg --gf

[GeneralSettings]
weight_mota: 1
max_iterations: 1500
relative_change: 1
max_delta_i = 100
max_n_changes: 3
storage_filename: atriumstorage.csv
video_filename: atrium_video.avi
ground_truth_sqlite: atrium_gt.sqlite
sqlite_filename: atriumtest.sqlite

[OptimizationParameters]
prob_constant: 1
t_init: 16
max_match_dist: 0.9
lamda: 0.4
emax: -100
```

1 2 1 1 3

8 – Edit the staticParameters.txt file

This file defines non-optimized parameters in the TI configuration file. The only changes required are the filenames below, which must correspond to those in the setup.ini file where applicable.

staticParameters.txt × | setup.ini × | variableP

```
1 0,video-filename = atrium_video.avi
2 0,database-filename = atriumtest.sqlite
3 0,homography-filename = atriumhomo.txt
4 0,intrinsic-camera-filename = none
5 0,distortion-coefficients = -0.11759321
6 0,distortion-coefficients = 0.0148536
7 0,distortion-coefficients = 0.00030756
8 0,distortion-coefficients = -0.00020578
9 0,distortion-coefficients = -0.00091816
10 0,undistorted-size-multiplication = 1.31
11 0,interpolation-method = 1
12 0,load-features = false
13 0,display = false
14 0,video-fps = 30
15 0,measurement-precision = 3
16 0,frame1 = 0
17 0,nframes = 0
18 0,max-nfeatures = 1000
19 0,use-harris-detector = false
20 0,k = 0.133561
21 0,nframes-velocity = 3
22 0,max-number-iterations = 20
23 0,min-feature-eig-threshold = 0.0001
24 0,max-distance = 5
25 0,min-velocity-cosine = 0.188628
26 0,max-predicted-speed = 50
27 0,prediction-time-horizon = 5
28 0,collision-distance = 1.8
29 0,crossing-zones = false
30 0,prediction-method = na
31 0,npredicted-trajectories = 10
32 0,min-acceleration = -9.1
33 0,max-acceleration = 2
34 0,max-steering = 0.5
35 0,use-feature-prediction = true
36 0,max-normal-acceleration = 2
37 0,max-normal-steering = 2
38 0,min-extreme-acceleration = 2
39 0,max-extreme-acceleration = 3
40 0,max-extreme-steering = 3
41 0,mask-filename = none
```

9 – Run GenTrO

```
Please select what to do:

1 - Begin or continue tracker optimization.
2 - Extract best parameters to config file(s).

Traffic Intelligence specific commands:
3 - Display tracks for an input SQL file.
4 - Create heatmap for an input SQL file.
5 - OD counts over thresholds.

q - Quit.
```

1: Runs GenTrO to optimize TI for the scene. This process can take upwards of a thousand iterations.

If this is the first run on a given calibration video, the user will be required to input points for point correspondence. Three windows will appear, requiring four points to be input on each – note that the input order must be the same in each window for points to be correctly associated.

The first two windows represent the sample frame. In the first, the points for correspondence should be defined at ground level. In the second, points should be selected directly above the previous ones, at an elevation of approximately 1.5 meters (if unsure, it is preferable to place the points slightly further than closer to the ground).

Figure 1 - Points for homography calculation in the video frame. Red points are at ground level, green ones approximately 1.5 meters above.

The third window is the world-image. Again, points must be placed in the same order, at the position where they are in reality.

2: Extract best parameters to config file; the last iteration is not necessarily the best set of parameters, so this command generates configuration files for TI using optimized parameters. Note that this file will still refer to the calibration scene, so it must be modified to run on the desired, full video.

3: Displays the raw tracks for a given SQL file in PolyTrack format.

4: Generates a heat map, representing densities of object detections (not necessarily pedestrian densities).

5: Counts tracks going between screen lines defined by the user.