**Near Miss Project – User Manual**

Iris Guo, Civil and Environmental Engineering, Rutgers

|  |  |  |
| --- | --- | --- |
| No. | Process | Detail |
| 1 | Web Camera Setup | After camera setup, users will be able to access real-time traffic data through the following link  <http://165.230.153.22/index.shtml>  ID: nearmiss  PW:Rutgers2017 |
| 2 | Download Traffic Data | * Click the ‘setup’ in the top right corner ; * Then click ‘Recordings’-‘List’ on the left side and user will be able to see the real-time recording data; * Click the ‘Export’ button and the ‘Export Recording Window’ will pop up; * Adjust the ‘start time’ and ‘stop time’ to control the duration of video, the system will show the estimated file size at the same time * Click ‘Export’ to save the target video(.mkv) to download folder   The downloaded video will be name after the start/end time, the frame size is 2592\*1944, frame rate will be 12 fps:  Duration:20170801\_103007 – 20170801\_103815  Frame width and height: 2592\*1944  Frame rate: 12 frame/second |
| 3 | Download Source Code | https://github.com/gmy0317/RU\_NearMiss.git |
| 4 | Video Clipping | * Copy the raw .mkv video to following folder   Nearmiss\_summer2017\RU\_NearMiss\0-Video&PointCloud\1\_Raw\_Video   * Using sublime to open the ‘Video\_Pre.bat’ and edit line-10 str1(Video name) to your raw video name      * Edit the duration of the target video in line-60 and line-63. For example, 180 means 180 seconds.      * Save the bat file. * Double Click ‘Video\_Pre.bat’, then system will automatically clip the raw video into short videos |
| 5 | Camera Calibration | * Open folder: Nearmiss\_summer2017\RU\_NearMiss\1-CameraCalibration * Print out a chessboard and record a short video using the web camera. Copy your video to the ‘1-CameraCalibration\example\_input’ folder * Run the following code in command line:   calibrate.py .\example\_input\YOURVIDEONAME.mp4 calibration.yaml --debug-dir out   * The camera matrix will be saved to calibration.yaml   We have finished camera calibration for our web camera, the camera matrix information are as follows. So if you are using the same camera, there is no need to do camera calibration again.  K\_undistort2 = np.array([[1822.7410517557776, 0.0, 1380.9286407307004], [0.0, 1823.2135457248423, 1107.4325084607897], [0.0, 0.0, 1.0]]) K\_undistort1 = np.array([[1822.7410517557776, 0.0, 1380.9286407307004], [0.0, 1823.2135457248423, 1107.4325084607897], [0.0, 0.0, 1.0]]) K\_array = np.array([[-0.3995256101289808, 0.1860076410786205, -0.01028873144026083, -0.003397377103708964, -0.0476809580344696]]) |
| 6 | Video Un-distortion | * Open the ‘0-Video&PointCloud\2\_Clip\_Video’ folder, you will be able to see several short videos generated in step 4.      * Double click the following batch file   RU\_NearMiss\0-Video&PointCloud\2\_Clip\_Video\Undistort.bat   * Then the system will do the video un-distortion frame by frame.     Video un-distortion(Raw video: left; Undistorted video: right) |
| 7 | Link Video with PointCloud | The projection information are calculated by JavaScript and the result is listed in ‘Image2PC.csv’, ’GPS\_x.csv’, ’GPS\_y.csv’, ’GPS\_z.csv’ |
| 8 | Trajectory Tracking | * Edit ‘iris/filepath.cfg’, edit line-5, change video-filename to your video path      * Using command line to run the Tracking.exe   Tracking.exe iris/main.cfg |
| 9 | SQL Data Processing | Run python code: Sqlite2py.py  After running the code, the exported result will be save to ‘5-result’ folder  Output attributes includes: object id, object location, object frame, GPS info, velocity, acceleration.. |