A Semi-Automated Image Processing Solution for Extracting Microscopic Traffic Data

Paper Number 22

Bharathiraja Muthurajan, **Rushikesh Amrutsamanvar***
and
Lelitha Devi Vanajakshi

Transportation Engineering Division
Department of Civil Engineering
Indian Institute of Technology Madras

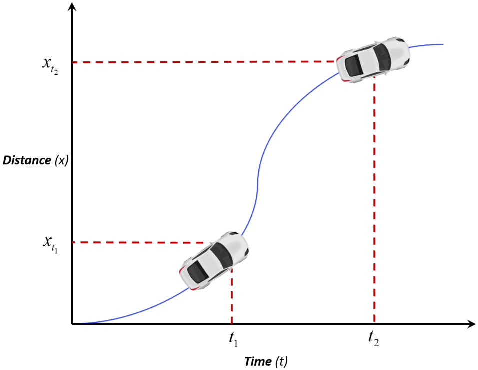
10^{th} **Urban Mobility India Conference 2017**

Outline

Trajectory data

What is that and why we need it?

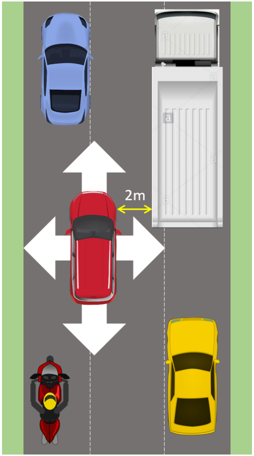
- Primitive element of microscopic traffic flow modeling
- Inter-vehicular distances
- Key to characterize any type of traffic



Trajectory data

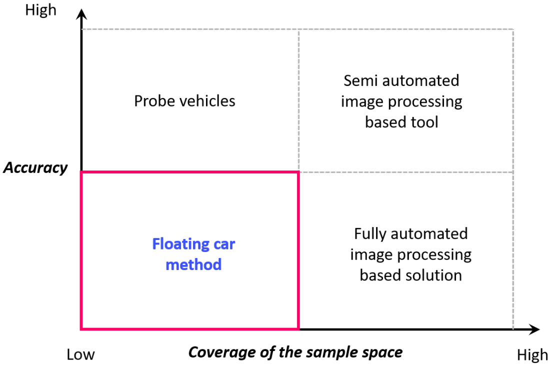
What is that and why we need it?

- Primitive element of microscopic traffic flow modeling
- Inter-vehicular distances
- Key to characterize any type of traffic



Challenges involved in trajectory data collection

- Drivers under surveillance
- Different classes of vehicles?
- Re-examine?
- Practically impossible to track every vehicle

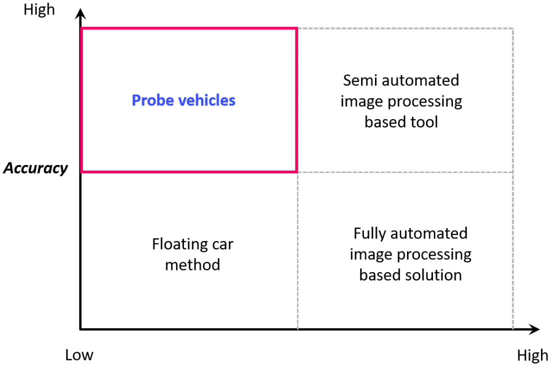


Methods for collecting the trajectory data

Ahemed,(1999); Lee, (2008)

Challenges involved in trajectory data collection

- Different classes of vehicles?
- Re-examine?
- Practically impossible to track every vehicle

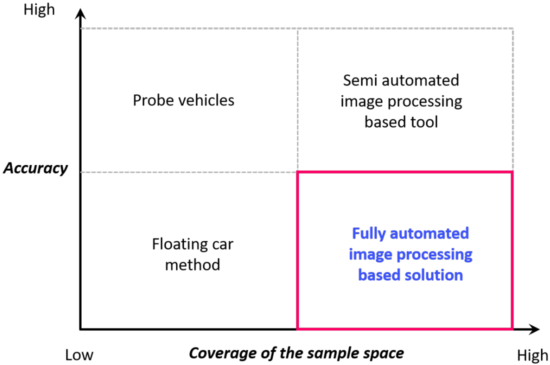


Methods for collecting the trajectory data

Ahemed,(1999); Lee, (2008)

Challenges involved in trajectory data collection

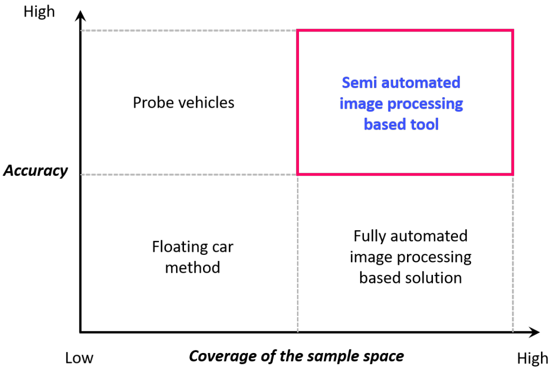
- Looses the track of vehicle
- Duplicate entries
- Smaller trap length



Methods for collecting the trajectory data

Munigety *et al.*, (2013); Fulari, (2015)

Challenges involved in trajectory data collection



Methods for collecting the trajectory data

- Tracking only one vehicle at a time
- **one known point per image frame**

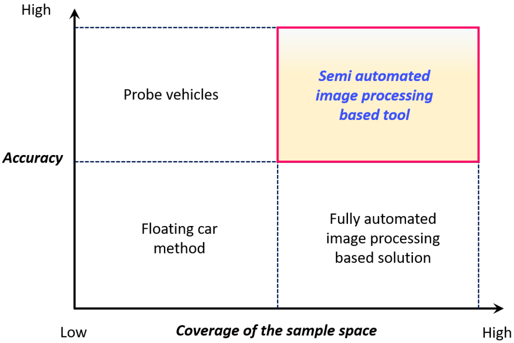
Approach of present study

Developed semi automatic tool - Vehicle Trajectory Extractor(VTE)

- A semi automated tool is developed

- User input vehicle classification and identification
- Once selected, automated tracking of vehicle
- Tracks multiple vehicles simultaneously

✓ Efficient trajectory extraction



Methods for collecting the trajectory data

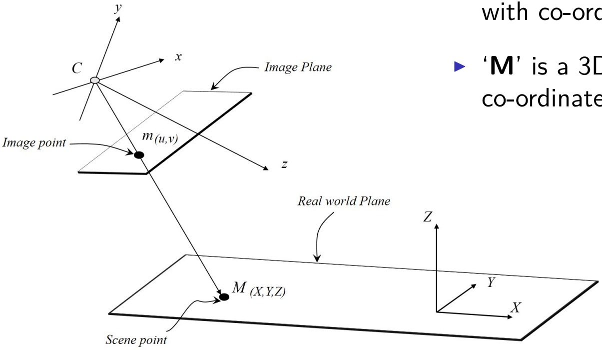
Camera calibration and coordinate Conversion

Pinhole camera model (Zhang, 2000)

- Process of mapping the co-ordinates of two-dimensional (2-D) digital image to the three-dimensional (3-D) real-world coordinates
- Based on:
  - Intrinsic parameters
    - optical, geometric, and digital characteristics of the camera
    - obtained from chess board video footage
  - Extrinsic parameters
    - defines the location and orientation of the camera
    - obtained from the co-ordinates of four reference points marked in the field
- Once these parameters are obtained, mapping from 2D to 3D is possible using equations from projective geometry

Pinhole camera model

(Zhang, 2000)



- '**m**' is the point on 2D digital image with co-ordinates (u, v)
- '**M**' is a 3D point in real world with co-ordinates (X, Y, Z)

Pinhole camera model
(Zhang, 2000)

- Now mapping from $2D$ to $3D$ can be carried out as:

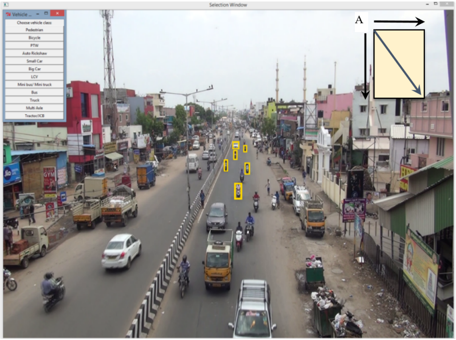
$$s \begin{bmatrix} u \\ v \\ z \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{1}$$

where,

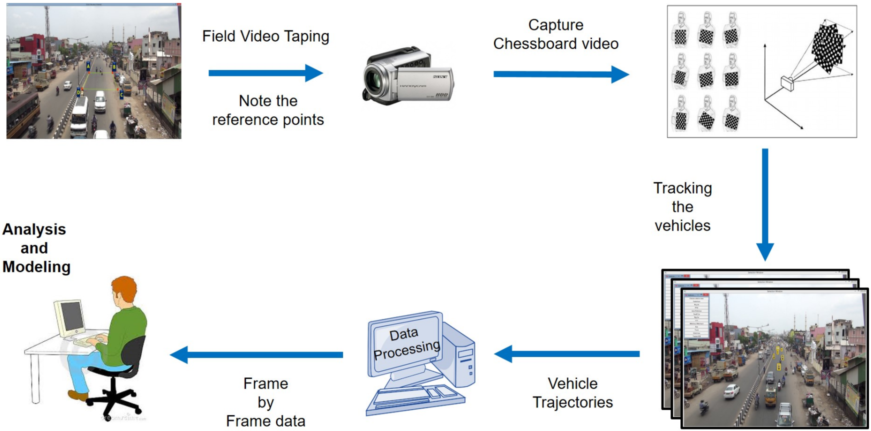
- $c_x, c_y \longrightarrow$ co-ordinates principal point that is usually at the image center
- r_{12} to $r_{33} \longrightarrow$ rotational coefficients
- $s \longrightarrow$ scale factor.
- $f_x, f_y \longrightarrow$ are the focal lengths expressed in pixel units
- t_1 to $t_3 \longrightarrow$ translational coefficients

Vehicle tracking

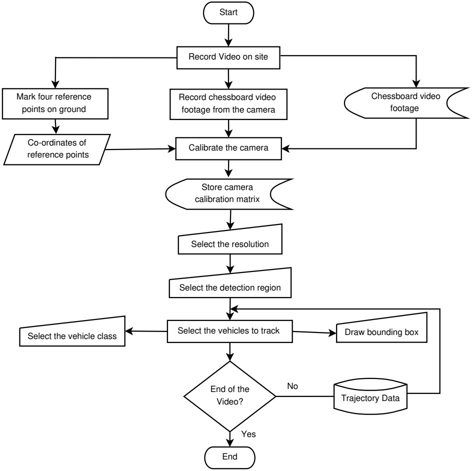
- Discriminative correlation filter (Danelljan *et al.*, 2014) for vehicle tracking
- Based on histograms of oriented gradients (HOG) features
- Accepts vehicle selection as user input in the form of a bounding box
- Tracker learns a discriminative correlation filter to localize this target in a new frame



Data collection process using VTE



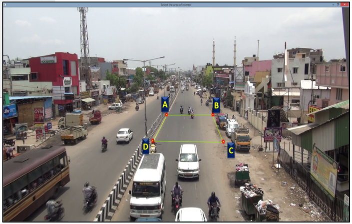
Working with VTE



Study area and data collection

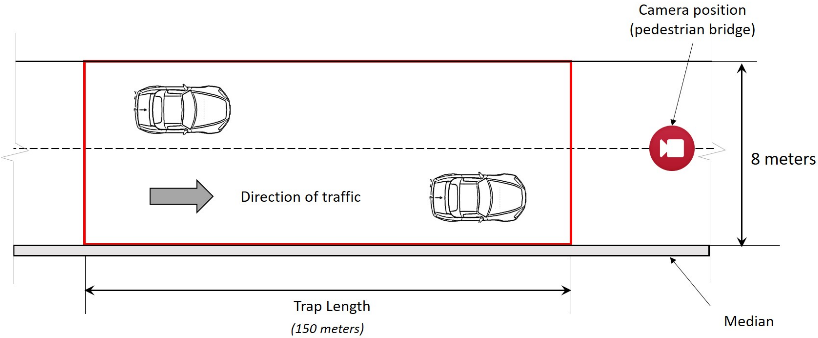
Source: www.google.com

Data collection site



- Two lane divided urban road
- Afternoon - off peak period
- 20 minutes of video footage
- 150*m* trap length
- 1/25 *seconds* resolution
- 1136 Vehicle trajectories

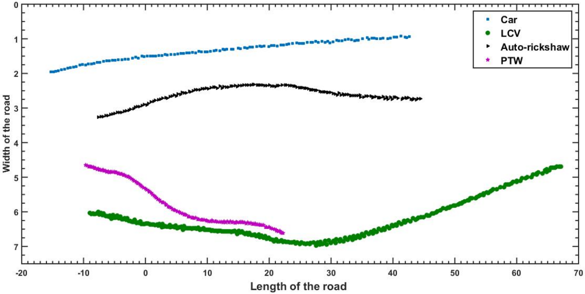
Study area and data collection



- Two lane divided urban road
- Afternoon - off peak period
- 20 minutes of video footage
- 150m trap length
- 1/25 *seconds* resolution
- 1136 Vehicle trajectories

Sample output from VTE

Vehicle ID	Frame Number	Vehicle Type	X	Y
1	47	4	31.88	1.56
2	47	2	53.60	3.06
3	47	6	10.82	2.42
4	47	2	20.66	0.65



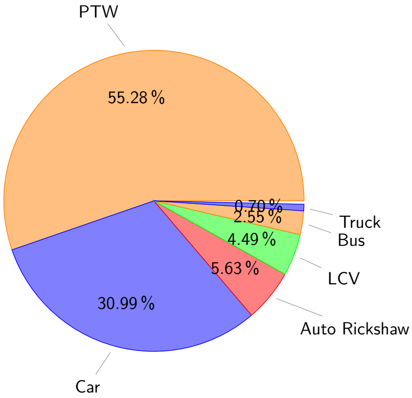
sample vehicle trajectories

*The number in vehicle class column refers to particular vehicle class

Preliminary analysis

Vehicle composition in the database

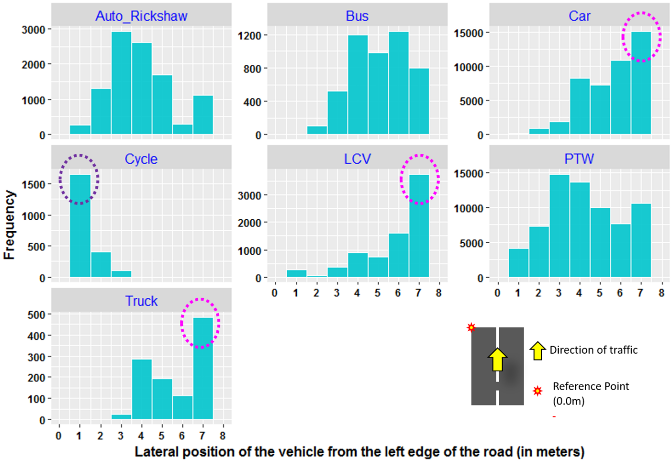
Vehicle Type	Number of vehicles in dataset
Bicycle	4
PTW	628
Auto Rickshaw	64
Car	352
LCV	51
Bus	29
Truck	8
Total	**1136**



Powered two wheelers (PTW)
Light commercial vehicles (LCV)

Preliminary analysis

Vehicle Positioning



- cycle and auto rickshaws prefer left side
- PTW are using both sides
- heavy vehicle and cars prefer right side

Summary and conclusion

- A novel semi-automatic tool is developed for extracting vehicle trajectory data

1. Easy to handle
2. Improved accuracy
3. Time efficient
4. Cost effective

- Future tasks:
  - Validation for accuracy
  - Building a graphical user interface (GUI)

Summary and conclusion

- A novel semi-automatic tool is developed for extracting vehicle trajectory data

1. Easy to handle
2. Improved accuracy
3. Time efficient
4. Cost effective

- Future tasks:
  - Validation for accuracy
  - Building a graphical user interface (GUI)

Summary and conclusion

- A novel semi-automatic tool is developed for extracting vehicle trajectory data

1. Easy to handle
2. Improved accuracy
3. Time efficient
4. Cost effective

- Future tasks:
  - Validation for accuracy
  - Building a graphical user interface (GUI)

Summary and conclusion

- A novel semi-automatic tool is developed for extracting vehicle trajectory data

1. Easy to handle
2. Improved accuracy
3. Time efficient
4. Cost effective

- Future tasks:
  - Validation for accuracy
  - Building a graphical user interface (GUI)

Summary and conclusion

- A novel semi-automatic tool is developed for extracting vehicle trajectory data

1. Easy to handle
2. Improved accuracy
3. Time efficient
4. Cost effective

- Future tasks:
  - Validation for accuracy
  - Building a graphical user interface (GUI)

Summary and conclusion

- A novel semi-automatic tool is developed for extracting vehicle trajectory data

1. Easy to handle
2. Improved accuracy
3. Time efficient
4. Cost effective




- Future tasks:
  - Validation for accuracy
  - Building a graphical user interface (GUI)

References

1. Ahmed, K. I. (1999) Modeling drivers' acceleration and lane changing behavior (*Doctoral dissertation*, Massachusetts Institute of Technology).
2. Bradski, G., and Kaehler, A. (2008) Learning OpenCV: Computer vision with the OpenCV library. " *O'Reilly Media, Inc.*"
3. Dalal, N. and Triggs, B. (2005) Histograms of oriented gradients for human detection. *In IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (1), pp. 886-893.
4. Danelljan, M., Häger, G., Shahbaz Khan, F. and Felsberg, M. (2014). Accurate Scale Estimation for Robust Visual Tracking. *Proceedings of the British Machine Vision Conference,* 2014.
5. Fulari S. G. (2015) Traffic state estimation under uncertain automated sensor data (*MS dissertation*, Indian Institute of Technology Madras).
6. King, D.E. (2009) Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*,pp.1755-1758.
7. Kovvali, V. G., Alexiadis, V., and Zhang, P. E. (2007) Video-based vehicle trajectory data collection. In *Transportation Research Board 86th Annual Meeting* (No. 07-0528).
8. Lee, T. C., Polak, J. W., and Bell, M. G. (2008) Trajectory Extractor user manual version 1.0. *Centre for Transport Studies, Imperial College London*, United Kingdom. Available at: <www.cts.cv.ic.ac.uk/documents/publications/iccts01155.pdf>.
9. Mallikarjuna, C., Phanindra, A., and Rao, K. R. (2009) Traffic data collection under mixed traffic conditions using video image processing. *Journal of transportation engineering*, 135(4), 174-182.
10. Munigety, C., Vicraman, V. and Mathew, T. (2014) Semiautomated Tool for Extraction of Microlevel Traffic Data from Videographic Survey. *Transportation Research Record: Journal of the Transportation Research Board*, (2443), pp.88-95.
11. Wei, H., Feng, C., Meyer, E., and Lee, J. (2005) Video-capture-based approach to extract multiple vehicular trajectory data for traffic modeling. *Journal of Transportation Engineering*, 131(7), 496-505.
12. Zhang, Z. (2000) A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11), 1330-1334.

Thank you for your attention!

Questions??

Contact:

Rushikesh Amrutsamanvar
Indian Institute of Technology Madras
rushikesh.amrut@gmail.com

Pinhole camera model

Considering the lens distortion coefficients

$$x_{corrected} = x'(\frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6}) + 2p_1 x' y' + p_2(r^2 + 2x'^2) \tag{2}$$

$$y_{corrected} = y'(\frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6}) + 2p_2 x' y' + p_1(r^2 + 2y'^2) \tag{3}$$

where,

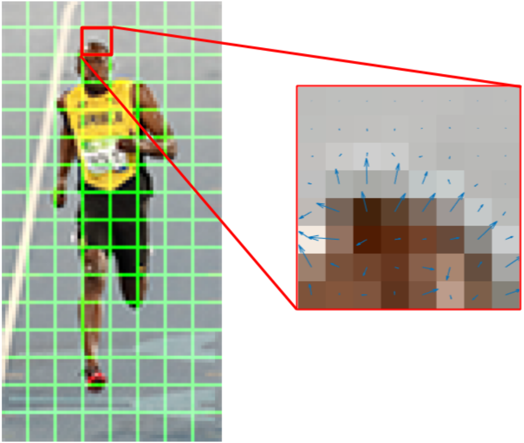
- k_1 to k_6 are radial distortion coefficients
- p_1 and p_2 are tangential distortion coefficients
- $r^2 = x'^2 + y'^2$
- $x' = x/z; \qquad y' = y/z$

$$u = f_x x_{corrected} + c_x \tag{4}$$

$$v = f_y y_{corrected} + c_y \tag{5}$$

HOG Features

How it works?



2	3	4	4	3	4	2	2
5	11	17	13	7	9	3	4
11	21	23	27	22	17	4	6
23	99	165	135	85	32	26	2
91	155	133	136	144	152	57	28
98	196	76	38	26	60	170	51
165	60	60	27	77	85	43	136
71	13	34	23	108	27	48	110

Gradient Magnitude

80	36	5	10	0	64	90	73
37	9	9	179	78	27	169	166
87	136	173	39	102	163	152	176
76	13	1	168	159	22	125	143
120	70	14	150	145	144	145	143
58	86	119	98	100	101	133	113
30	65	157	75	78	165	145	124
11	170	91	4	110	17	133	110

Gradient Direction

* Figure source: Satya Malilicks blog

Tracking algorithm

Algorithm 1 Proposed tracking approach: iteration at time step t.

Input:

Image I_t.
Previous target position $\boldsymbol{p}_{t-1}$ and scale s_{t-1}.
Translation model A_{t-1}^{trans}, B_{t-1}^{trans} and scale model A_{t-1}^{scale}, B_{t-1}^{scale}.

Output:

Estimated target position $\boldsymbol{p}_t$ and scale s_t.
Updated translation model A_t^{trans}, B_t^{trans} and scale model A_t^{scale}, B_t^{scale}.

Translation estimation:

1: Extract a translation sample z_{trans} from I_t at $\boldsymbol{p}_{t-1}$ and s_{t-1}.
2: Compute the translation correlation y_{trans} using z_{trans}, A_{t-1}^{trans} and B_{t-1}^{trans} in (6).
3: Set $\boldsymbol{p}_t$ to the target position that maximizes y_{trans}.

Scale estimation:

4: Extract a scale sample z_{scale} from I_t at $\boldsymbol{p}_t$ and s_{t-1}.
5: Compute the scale correlation y_{scale} using z_{scale}, A_{t-1}^{scale} and B_{t-1}^{scale} in (6).
6: Set s_t to the target scale that maximizes y_{scale}.

Model update:

7: Extract samples f_{trans} and f_{scale} from I_t at $\boldsymbol{p}_t$ and s_t.
8: Update the translation model A_t^{trans}, B_t^{trans} using (5).
9: Update the scale model A_t^{scale}, B_t^{scale} using (5).