REPORT FOR TASK 1
LITERATURE SURVEY

MACHINE LEARNING ALGORITHMS FOR TURN MOVEMENT COUNT PREDICTION

Sanjay Ranka
Professor
Computer and Information Science and Engineering
University of Florida

Contents

List of Figures

1 Introduction

Turning Movement Counts are used for wide variety of intersection analyses, intersection design, and transport planning applications. Efficient design of operations at signalized intersections at different traffic conditions can be done by the information obtained from turn movement counts. The traditional method of obtaining turn movement count include manual count by people, which is very time consuming and monotonic work. Approach volumes can be easily obtained from the loop detectors, but they only give information on how many vehicles are entering the intersection. Thus an automated way of calculating turn movement counts is very important and it is challenging in case of having limited traffic data. They require information on type of movement, origin-destination for each vehicle. One other challenge is when the case of shared lanes. We can impute turn movements from one detector, multiple detectors are needed to track the turns-another detector should be placed at departure lane. However, many states do not use departure detectors.

Some studies have used mathematical techniques like linear programming to impute turning movement counts which balances inflow, outflow of traffic with some constraints. Some studies used artificial neural networks to model dependencies among different traffic variables and thereby imputing turn movement counts. Some studies also used video data to impute turn movement counts. Trajectories are extracted from raw video data by tracking moving objects and thus inferring turn movement counts. But relying on this method for large number of intersections is not cost effective, also considering how computationally expensive is video processing. Further, applicability of these approaches depends on geometry of the intersection, camera view.

2 Purpose and Scope

The purpose of this research is to summarize the existing literature on turning movement counts prediction. We broadly categorize the existing literature and summarize papers in each category. We filtered existing literature on how relevant it is to suit our needs

3 Heuristic Methods for TMC

Network Equilibrium: In this method unknown movement is estimated based on equilibrium of volume in two intersections. By modelling the flow with some some constraints as linear equations, we can solve for every unknown movement if we have enough equations. To calculate westbound through volume at intersection i, the following equation can be used.

$$WT_i^t = -NL_i^t - SR_i^t + WR_j^{t+\delta t} + WL_j^{t+\delta t} - \delta_{ij}^t \tag{1}$$

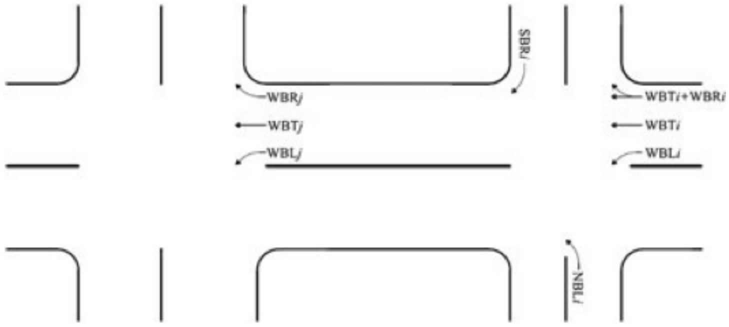

Figure 1: Example network with two intersections.

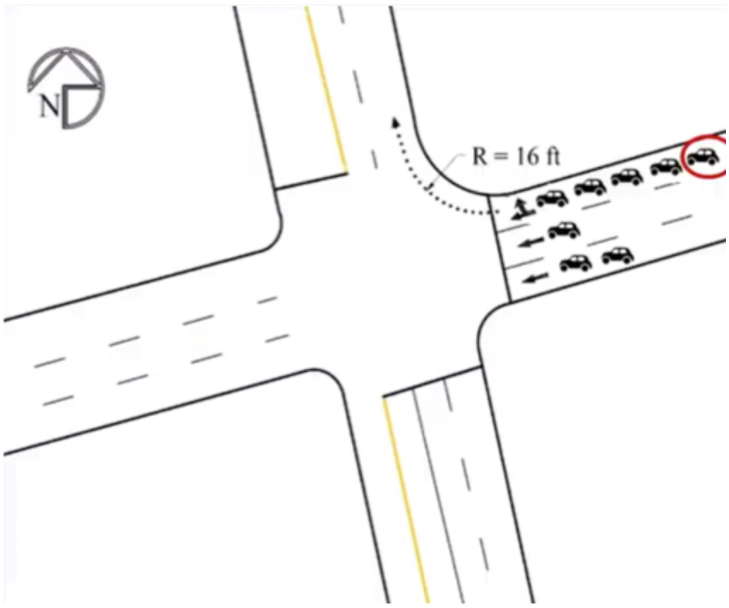

Figure 2: Example intersection.

WT: west bound through
NL: north bound left
SR: south bound right
WR: west bound right
WL: west bound left
t: time interval t
i,j: intersections number
Δt time taken to travel between i,j
$\delta_{ij}^{t+\Delta t}$: additional trips generated between intersections i,j during time interval t

Δt is assumed to be zero if the distance between two intersections is small as the volume fluctuations would be significantly low. δ_{ij} can be assumed to be zero if there is no trips generated between these two intersections

Volume and Queue Length of Shared Lanes: This is based on the assumption that a newly entered vehicle in a shared line decided to turn if there are more vehicles in that lane compared to adjacent lanes. This assumption also depends on distance to the stop bar, intersection geometry. Figure 2 shows a similar situation.

Flow Characteristics of Shared Lanes: This method is based on the assumption that vehicles which have smaller head-ways from their front car when they pass stop bar are more probable to go through. High resolution stop bar detector data along with signal timing plan should be used for this method. Applicability of this method also depends on the intersection geometry. The probability of a vehicle i turning in some direction is modeled as function of position of vehicle in the line, headway of front vehicle, type of front vehicle and vehicle type. This relation is modelled as shown in equation 2.

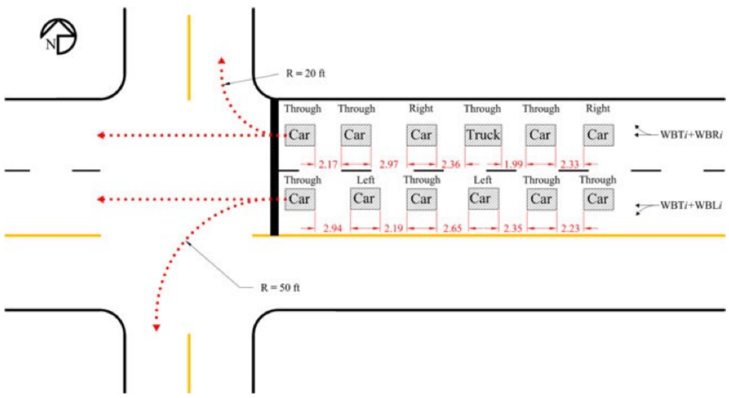

Figure 3: Headway in shared lane.

$$P(t_{i,r}) = f(h_i, hf_i, cp_i, ct_i, cf_i) \tag{2}$$

P($t_{i,r}$): probability that vehicle i turns right or left
h_i: headway when passing a stop bar, the time difference of vehicle i from its front vehicle
hf_i:summation of each headway with previous car headway
cp_i: position of vehicle i in the line
ct_i: type of vehicle i
cf_i: type of front vehicle of vehicle i

If the turning radius is large then relying on head-ways is not a good idea. Thus geometry of intersection plays a important role in this method.

4 Broad categories of existing literature

Algorithms based on Network Flows: Some studies have used mathematical techniques like linear programming to impute turning movement counts which balances inflow, outflow of traffic with some constraints. These system of linear equations are solved using different optimization techniques to impute turn movement counts.

Neural Network Based Models:: Neural Networks are computing systems that are vaguely inspired by biological neural networks that constitute animal brains. It is like a artificial nervous system receiving, processing and transmitting data. Basically, neural networks consists of a input layer, hidden layers and a output layer. The fundamental unit that comprises a neural network is called a neuron.

- **Input Layer**. This layer takes the input in required shape and passes it to the hidden layer. No computations are done here.
- **Hidden Layer**. They receive input from the input layer, perform computations and then transfer its output to the following layer.

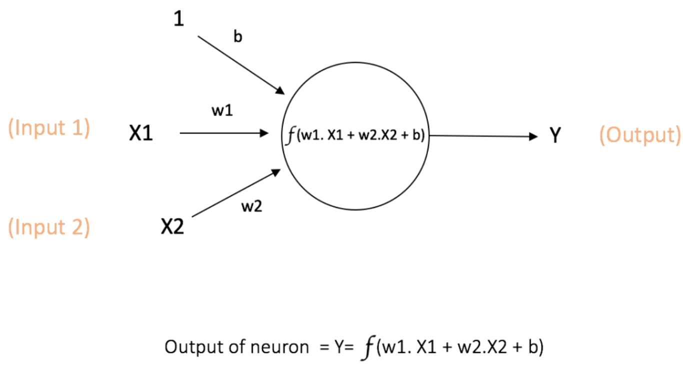

Figure 4: Single neuron with input and output

- **Output Layer**. With use of a specific activation function, this layer is used to map output to desired format.
- **Connections**. Output of each neuron is transferred to a neuron in the following layer with a connection. A weight is associated with each connection. If output of i is connected to input of j, then corresponding weight is W_{ij}
- **Activation Function**. Output of each neuron is a function of its input. This function is called activation function of a neuron. Some activation functions we use in practice are Sigmoid, ReLu, Tanh.

Figure 4 shows a single neuron and the input, output mechanism. It takes two inputs X1, X2 and they are connected with weights W1, W2. So, the input to the neuron becomes W1X1 + W2X2. Output of this neuron is a function of this input fed to it. This functions are called activation functions. This is important because it models non-linearity in the relation between input and output.

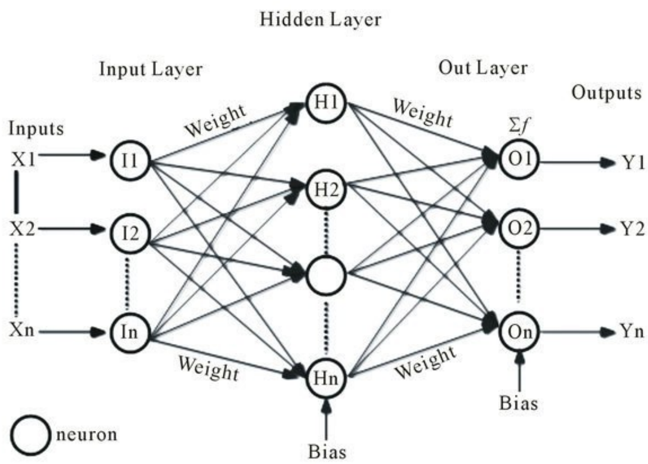

Figure 5: Example 3 layer network

Figure 5 shows a three layer network with all the labelling. In our case the non linear relationship between flow variables like approach volumes etc. and the corresponding turning movements is modelled with the use of neural networks(ANN).

Video Based Models: Video data from signalized intersections is used to extract information by detecting moving objects and computing their trajectories. These trajectories can be used to get turning movement counts.

4.1 Algorithms based on Network Flows

In [1] they propose a path flow estimator based solution to estimate turn movement counts. The proposed path flow estimator is used to compute complete link flows along with turn movement counts when some traffic counts at selected intersection are known. They claim that their method has advantage because of the single level convex programming formulation, when compared to other bi-level programming approaches. Their proposed algorithm iterates over entropy and equilibrium terms to obtain the final solution. The first term, entropy, distributes trips to multiple paths. While the second term, equilibrium, make those trips cluster together on minimum cost paths.

[2] uses linear programming approach to compute turning movement counts. This approach required measurements like detector flow, weight given to each link, constraints on flow for each link. A system of linear equations are modelled with the above constraints/information. These system of equations are solved to get turning movement counts.

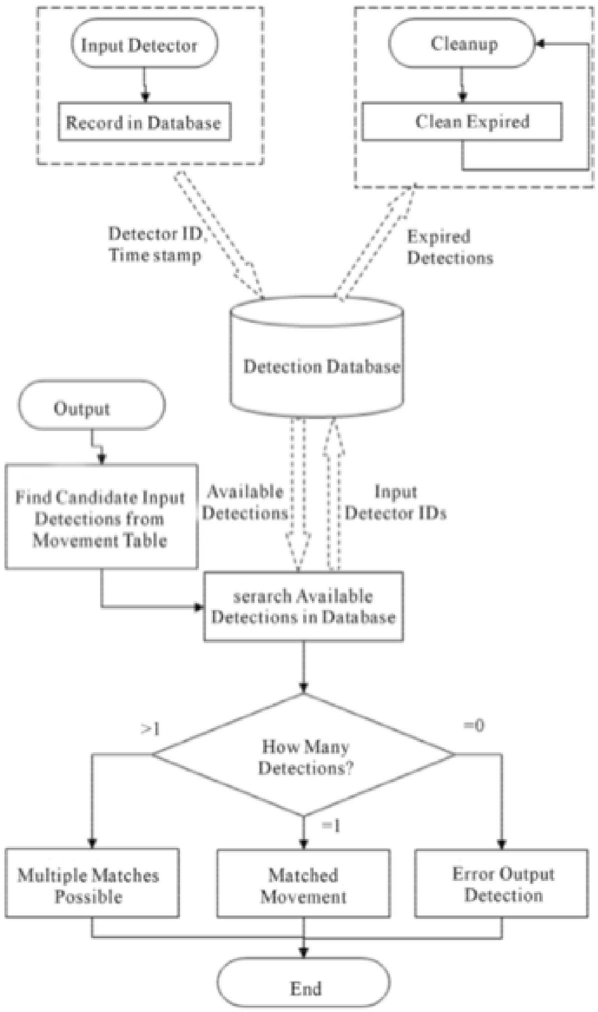

Figure 6: Proposed ATIMS Algorithm Flowchart

[3] used origin-destination matrices to compute turning movement counts. They uses already observed O-D matrices to predict future O-D matrices. They use travel demand model, Furness

Method and then use equilibrium principle to assign future O-D matrices to the network.

[4] developed a Automatic Turning Movement Identification System (ATMIS). In the proposed architecture, they uses input and output detectors along with signal timing plan to impute turning movement counts. They also claim that this can be used irrespective of intersection geometry. Figure 6 shows their proposed framework.

4.2 Genetic Algorithm based Model

In [5] a genetic algorithm based framework is developed to obtain a dynamic relation between turning movement proportions and traffic counts at intersections at each time. The proposed objective function minimizes sum of absolute differences between observed and predicted traffic counts. The objective function is made to converge using a revised parameter optimization model. Then they finally propose a genetic algorithm to impute turning movement counts. Figure 7 shows their proposed flow chart. They conclude that Genetic Algorithm achieved the

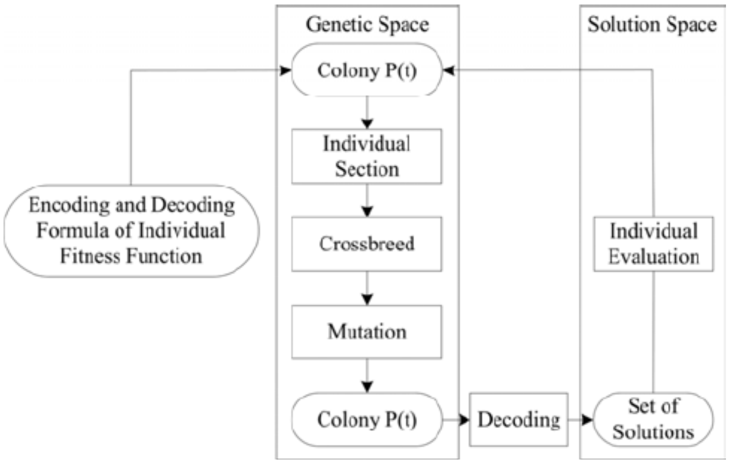

Figure 7: Genetic Algorithm Flow Chart

best results when compared to Kalman FIltering(KF), Least Square(LSQ methods.

4.3 Neural Network Based Models

In [6] the non linear relation between approach volumes and the corresponding turning movements is modelled by neural networks. They use only approach volumes to predict corresponding turning movement counts.

Figure 8 shows how a four leg intersection is represented as a node. Number of vehicles leaving the intersection i be P_i, $P_i = \sum_{j=1}^{4} T_{ij}$. The turning movement counts between node i, node j are number of trips from i to j. So, when a vehicle is approaching a intersection, they have four possible turns-left, through, right, and U-turn. For the four approaches there are 16 possible movements. For intersection j, the number of vehicles attracted to it be A_j, $A_j = \sum_{i=1}^{4} T_{ij}$, this is sum of all the trips with destination j. So 16 different turning movement counts should be modelled with the help of known values.

Each T_{ij} is modelled as function of P_i and A_j. For the turning movement counts, the incoming and outgoing traffic volumes are calculated. Neural network is used to model dependent variable turning movement counts with dependent variables-inbound and outbound approach volumes.

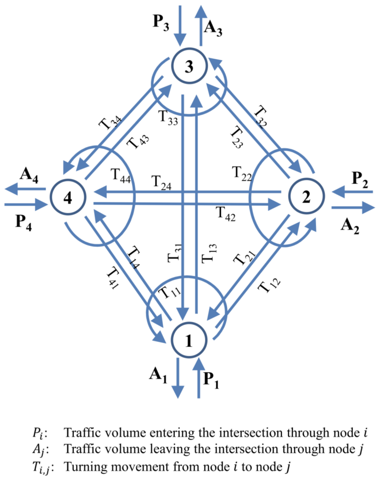

Figure 8: Representation of Signalized Intersection

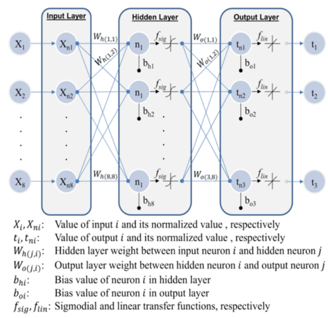

Figure 9: Neural Network Layout

[7] uses stacked auto encoder model to predict traffic flow. They trained their model with three months of data of 15,000 detectors. The generic traffic flow features are modelled with stacked auto encoder model. They also take into account spatial, temporal correlations. They use logistic regression layer as the top layer for supervised traffic prediction. Figure 10 shows the architecture they used.

Autoencoder is a neural network that tries to reproduce its input. It acts as a identity function. It tends to learn features that form a good representation of its input. Stacked auto encoder models are created by stacking auto encoders stacked in a hierarchical way.

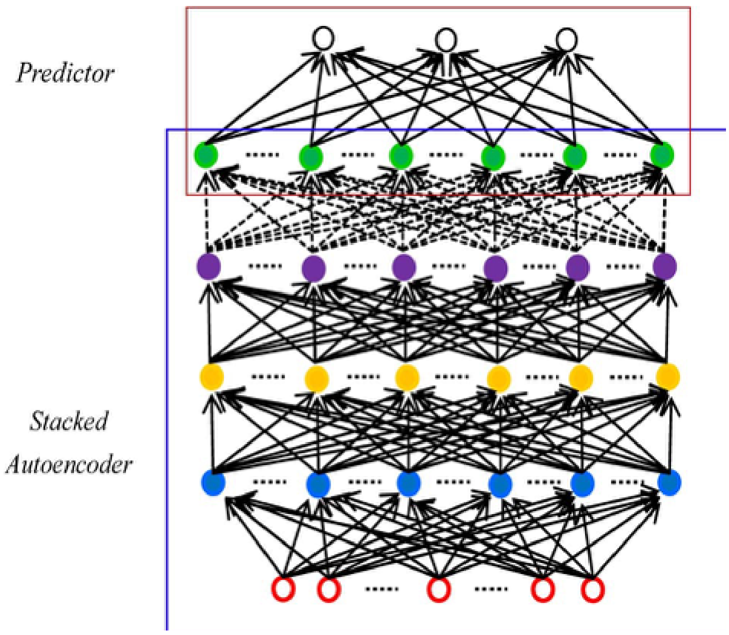

Figure 10: Stacked autoencoder model

4.4 Video Based Models

In [8] they propose a method for automated turning movement prediction. They uses vision based tracking of vehicles to impute turning movement counts.

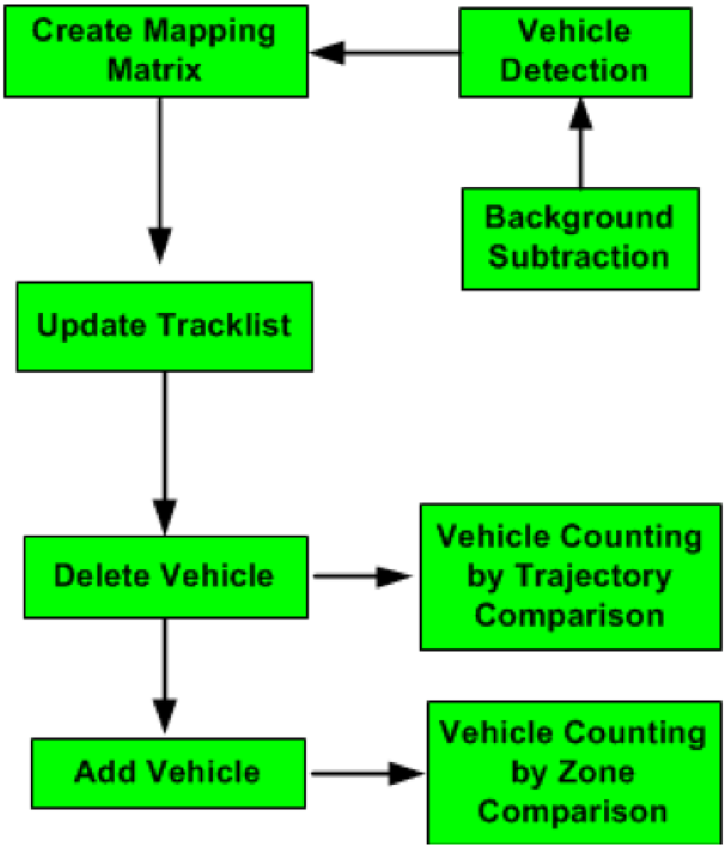

Figure 11: Vehicle Tracking System

Figure 11 shows block diagram of their tracking system. Gaussian mixtures is used to model background and moving objects are detected as pixels that do not fit any of the K background Gaussian models. The detected vehicles is then tracked and the trajectory is obtained. The obtained trajectory could be robust/complete or it could be occluded. So, their proposed solution switch between a zone comparison module and a trajectory comparison module. Their trajectory comparison module uses least common sub-sequence distance to take care of broken trajectories using typical scene paths.

[9] uses video footage from a properly placed camera as input data. Vehicle detection using background subtraction becomes challenging sometimes. Different lighting, weather conditions can effect the background. Also, vehicles cant be considered always moving and slow moving vehicles can effect background subtraction algorithms. So, they vehicle movement using optical flow algorithm and produce turning movement counts at a intersection. They detect all the vehicles and assign them unique IDs. They track these vehicles to produce turning movement counts. Figure 12 shows their proposed architecture.

In [10] they developed a system for counting vehicles by tracking in a video. They track vehicles and seperate them into a set of trajectories. They use open source Traffic Intelligence tracker to track vehicles and derive counts. They also claim that their architecture rapidly analyzes videos and yield detailed turning movement counts.

In [11] they developed a system for counting, behaviour analysis of pedestrians and vehicles at signalized intersections.

Figure 13 shows their system overview. This semi-automated framework takes manually labelled positions into video frames and compute paths from it. This system provides trajectories by tracking and using LCSS for path recognition. From this trajectories corresponding turning

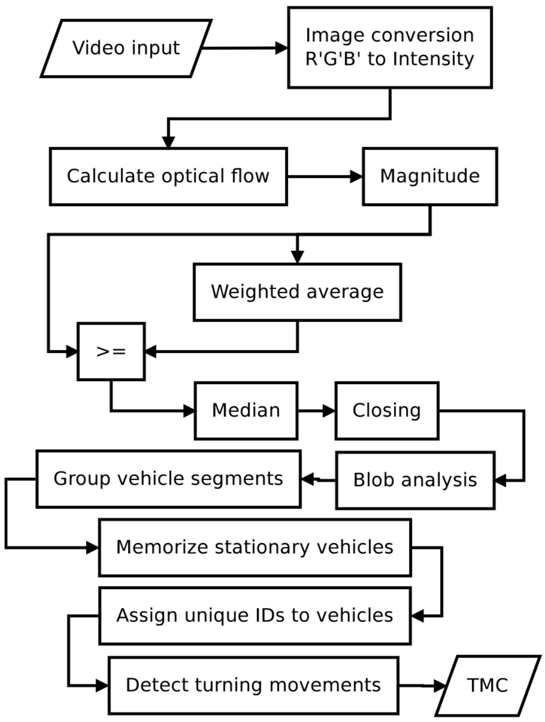

Figure 12: Proposed Solution

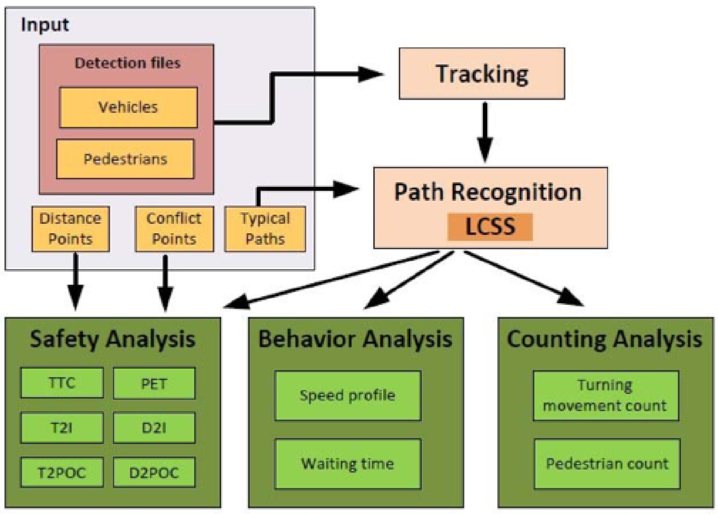

Figure 13: Proposed System Overview

movements are calculated. Their system also provides behaviour analysis at intersections by estimating speed profile of each Turning Movement and accurate waiting times.

References

[1] Anthony Chen, Piya Chootinan, Seungkyu Ryu, Ming Lee, and Will Recker. An intersection turning movement estimation procedure based on path flow estimator. *Journal of Advanced Transportation*, 46:161 – 176, 04 2012.

[2] M. C. Bell P. T. Martin. Network programming to derive turning movements from link flows. In *Transp. Res. Rec., no. 1365*, pages 147–154. Transportation Research Board, 1992.

[3] J. Wu and C. Thnay. An o-d based method for estimating link and turning volume based on counts. In *Transp. Res. Rec., no. 1365*, pages 865–873. ITE Dist 6, 2001.

[4] Kun Xu, Ping Yi, Chun Shao, and Jialei Mao. Development and testing of an automatic turning movement identification system at signalized intersections. *Journal of Transportation Technologies*, 03:241–246, 01 2013.

[5] and and. Real-time estimation of turning movement proportions based on genetic algorithm. In *Proceedings. 2005 IEEE Intelligent Transportation Systems, 2005.*, pages 96–101, Sep. 2005.

[6] M. S. Ghanim and K. Shaaban. Estimating turning movements at signalized intersections using artificial neural networks. *IEEE Transactions on Intelligent Transportation Systems*, 20(5):1828–1836, May 2019.

[7] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Wang. Traffic flow prediction with big data: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):865–873, April 2015.

[8] M. S. Shirazi and B. Morris. Vision-based turning movement counting at intersections by cooperating zone and trajectory comparison modules. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 3100–3105, Oct 2014.

[9] A. Abdagic, O. Tanovic, A. Aksamovic, and S. Huseinbegovic. Counting traffic using optical flow algorithm on video footage of a complex crossroad. In *Proceedings ELMAR-2010*, pages 41–45, Sep. 2010.

[10] A. Lessard, F. Belisle, G. Bilodeau, and N. Saunier. The counting app, or how to count vehicles in 500 hours of video. In *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1592–1600, June 2016.

[11] M. S. Shirazi and B. Morris. A typical video-based framework for counting, behavior and safety analysis at intersections. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 1264–1269, June 2015.

[12] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Wang. Traffic flow prediction with big data: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):865–873, April 2015.

[13] Bidisha Ghosh, Biswajit Basu, and Margaret O'Mahony. Multivariate short-term traffic flow forecasting using time-series analysis. *Trans. Intell. Transport. Sys.*, 10(2):246–254, June 2009.

[14] S. D. Schrock. Estimating turning movements at roundabouts using bluetooth technology.

[15] P. Lin and J. Rasp. A short-count method for the estimation of traffic turning movements, dept. statist., florida state univ. 673.

[16] M. C. Schaefer. Estimation of intersection turning movements from approach counts. 58:41–46.

[17] Hillel Bar-Gera, Pitu Mirchandani, and Fan Wu. Evaluating the assumption of independent turning probabilities. *Transportation Research Part B: Methodological*, 40:903–916, 12 2006.

[18] Hai Yang, Yasunori Iida, and Tsuna Sasaki. The equilibrium-based origin-destination matrix estimation problem. *Transportation Research Part B: Methodological*, 28:23–33, 02 1994.

[19] B. T. Morris and M. M. Trivedi. Learning, modeling, and classification of vehicle track patterns from live video. *IEEE Transactions on Intelligent Transportation Systems*, 9(3):425–437, Sep. 2008.

[20] B. T. Morris and M. M. Trivedi. A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Trans. Cir. and Sys. for Video Technol.*, 18(8):1114–1127, August 2008.

[21] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik. A real-time computer vision system for measuring traffic parameters. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 495–501, June 1997.

[22] E. D. Arnold S. L. Jones and M.Zanin. Estimating intersection turning movement. VTRC 99-TAR8.

[23] Mohammad Shokrolah Shirazi and Brendan Morris. Vision-based turning movement monitoring: Count, speed waiting time estimation. *IEEE Intelligent Transportation Systems Magazine*, 8:23–34, 04 2016.

[24] R. Scheneider. Comparison of turning movement count data collection methods for a signal optimization study.

[25] R. S. Lyles A. P. Tarko. ”development of a portable video detection system for counting turning vehicles at intersections.

[26] Roxana Velazquez-Pupo, Alberto Sierra-Romero, Deni Torres-Roman, Yuriy V. Shkvarko, Jayro Santiago-Paz, David Gmez-Gutirrez, Daniel Robles-Valdez, Fernando Hermosillo-Reynoso, and Misael Romero-Delgado. Vehicle detection with occlusion handling, tracking, and oc-svm classification: A high performance vision-based system. *Sensors*, 18(2), 2018.

[27] Adaptive turning flow estimation based on incomplete detector information for advanced traffic management. In *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, pages 830–835, Aug 2001.

[28] Hediye Tuydes-Yaman, Oruc Altintasi, and Nuri Sendil. Better estimation of origindestination matrix using automated intersection movement count data. *Canadian Journal of Civil Engineering*, 42(7):490–502, 2015.

[29] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik. A real-time computer vision system for measuring traffic parameters. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 495–501, June 1997.

[30] S. Lee, H. Baik, and J. H. Park. Visual traffic movement counts at intersection and origin-destination (o-d) trip table estimation. In *2007 IEEE Intelligent Transportation Systems Conference*, pages 1108–1113, Sep. 2007.

[31] Sokemi Rene Emmanuel Datondji, Yohan Dupuis, Peggy Subirats, and Pascal Vasseur. A survey of vision-based traffic monitoring of road intersections. *Trans. Intell. Transport. Sys.*, 17(10):2681–2698, October 2016.

[32] H. Veeraraghavan, O. Masoud, and N. P. Papanikolopoulos. Computer vision algorithms for intersection monitoring. *IEEE Transactions on Intelligent Transportation Systems*, 4(2):78–89, June 2003.