Multi-Camera based Intelligent Traffic Surveillance using Deep Learning

Chris Antoinette Sahaaya Seelan, Hamsini Sankaran, Lavanya Bale, Suriya Shankar
Computer Engineering Department
San Jose State University
San Jose, USA
Email: {chrisantoinette.sahaayaseelan, hamsini.sankaran, lavanya.bale, suriya.shankar}@sjsu.edu

***Abstract*—Recently there has been an increased attention in traffic control and monitoring system due to the advancement in computer vision and IoT fields. Dynamic real time vehicle detection and tracking can provide essential data for efficient traffic planning and traffic management. The objective of this project is to develop a sophisticated and an efficient traffic monitoring system that improves traffic management, by detecting, tracking and classifying vehicles under different situations using deep learning algorithms. Even with the most advanced image and video processing, tracking of moving objects still proves to be a challenging task for traffic surveillance system. This is mainly due to the limitations of features extracted from a single camera. One of the main problems faced in outdoor video surveillance systems is visual tracking of objects. Some of the common issues faced in dynamic video tracking are illumination, background changes and occlusions. In this project, we propose to use data from multi-camera traffic surveillance system to carry out functionalities such as vehicle detection, tracking, re-identification and classification of vehicles. We also aim to solve traffic anomalies caused due to stalled vehicles, car crashes, traffic rule violation etc. Our approach uses deep learning algorithms to build an automated traffic surveillance system which will resolve scaling, illumination and occlusion issues which otherwise, are impossible to solve with single camera. Different deep learning algorithms will be used to process and analyze the data captured by the cameras. The information derived from the algorithm will help to provide real-time outputs such as warning signals, traffic signals, etc. The system will provide traffic information (traffic density, average speed of the vehicles) and send warnings in case of any accidents through user interface to the remote traffic monitoring center.**

***Keywords*—Approximated Nearest Neighbor, AI, Computer Vision, deep learning, GPU, SIFT, Vehicle re-identification.**

I. INTRODUCTION

There has been increased attention in traffic control system due to the advancement in the field of computer vision and deep learning. Traffic intersections are critical area in traffic system. It is reported by The Federal Highway Administration(FHWA), that every year reports approximately 2.5 Million accidents at traffic intersection [25]. In the United States over the last several years an average of one-quarter of traffic fatalities and roughly half of all traffic injuries are attributed to intersections[26]. Accidents caused at traffic intersection are dangerous and are caused mostly due to drivers errors. Many proposals have been made to address this issue and increase safety at intersections. The solution mainly focus on vehicle detection and vehicle tracking. Traffic intersections require more intelligent monitoring system to make them safer. Some of the solutions proposed involve LiDAR. LiDAR data is used for the generation of region proposals. It processes the three dimensional point cloud that it offers. Further, these candidate regions are processed by a CNN classifier that was fine-tuned for performing pedestrian detection. It is observed that the LiDAR space clustering approach offers a very efficient way of generating the region proposals leading to higher recall rates and fewer misses for pedestrian detection. Hence, it indicates that LiDAR data can provide auxiliary information for CNN-based approaches [23]. An approach to detect the traffic states of the roads with only the data from a single vehicle was performed. The biggest advantage of such an approach was the system can function properly even if there is only a smaller number of vehicles which are equipped with the system. This is used at the early deployment stage of a vehicle-to-vehicle (V2V) network or it is used for a large scale Intelligent Transportation System [24]. The machine learning mechanisms are utilized to perform the classification of the traffic state by extracting the behavior of the motion of a vehicle. This development and performance evaluation use very accurate vehicle traces that are collected at several real-world intersections with Lidar. It is observed that the this approach can achieve a detection accuracy of 88.94 percent [24].An efficient and increased performance can be got by implementing deep learning and computer vision techniques to draw more information on surveillance systems [1]. Though there is a great advancement in Computer vision applications for detection and tracking of vehicles, there is still some set-back with the data collected from a single camera especially in the case of outdoor surveillance systems. This is because of the factors like illumination, occlusions background changes etc. Multi-camera based video surveillance can be used for vehicle detection, tracking, re-identification and classification within supervised area and are addressed by Koutsia et al in [6]. In this project, we use the data from multiple cameras to develop an intelligent and a reliable Traffic surveillance system at traffic intersection. The system supports multiple functionalities such as vehicle detection and tracking, vehicle re-identification, classification and detection of traffic anomalies to alert the public. The project focuses on the traffic anomalies caused due to speed violation and car crashes.Our approach uses

various deep learning algorithms to build an automated traffic surveillance system which will resolve scaling and occlusion issues that arise due to the presence of a single camera. The data from the multiple cameras are used by deep learning to generate valuable insights which in turn can be used in the traffic monitoring system. The vehicles which violate the traffic rules at the intersection are reidentified. The information derived from the algorithm will help to provide real-time traffic warning to the public.

II. RELATED WORK

Various methods and models in which the traffic surveillance can be set up and executed was studied before proposing our current methods. The current method mainly uses a single camera to monitor the behavior of the objects and people. From the following paper we understood that Tracking with a single camera requires correspondence of frames over time so multiple-camera tracking is a correspondence problem between tracks of objects seen from different viewpoints at the same time instant is delivered. Several different techniques like 3D reconstruction methods (Shape- from-Silhouette) [11], alignment approaches and feature matching techniques are used for solving multi- camera calibration and correspondence issues. We also realized that using multiple cameras, Experiments on indoor and outdoor sequences with different cameras proved that the approach is simple, fast and accurate [12].The most important aspect of any deep learning model is the data set. The quality of the output depends hugely on the quality of the corresponding data set for which the model is being developed for. The KITTI dataset which has been used recently is less complicated than the data set from the VSLAB for the following reasons like Complicated road scenes, Diverse accidents, Crowded streets like the number of moving cars, motorbikes, and pedestrians per frame are typically larger than other datasets[1].The method to capture the live data is by CCD cameras or highly mounted video cameras. The methods have been discussed in the following [8],[14] and there are some limitations to the quality and clarity of the video captured. To improve the quality we used a super HD, high PoE camera Reolink RLC-410 for more clarity and increased details to the objects being captured. Rather than the conventional methods of storing data and retrieving them manually to the server for testing, we used the nanobeam bridge that acts as an interface between two networks to retrieve the data continuously. A full deep learning multi-camera people detector with a new three-view data-set with more accurate calibration in terms of consistency of a projection across all the views is presented in [7]. A Boosting algorithm is proposed, which merges the proposal generation and the detection steps. The cascade learning is formulated as Lagrangian optimization of a risk accounting for both accuracy and complexity and integrates both hand-crafted and convolutional features. The downstream classifier of the extension of R-CNN known as Faster R- CNN [2] was recently shown to degrade the performance, and it was proposed to be replaced by boosting forests. The resulting method does not use hand-crafted features and reaches state of the art accuracy. To improve the robustness to occlusion, a novel technique dubbed input dropout, consists of augmenting the input data by artificially masking part of the signal. While the masking is adapted to the morphology of the positive class, it is done on both positive and negative samples during training, so that it does not provide erroneous discriminating cues to the network. Based on the following researches we proposed to use fast RCNN and LSTM for the tracking.

III. PROBLEM STATEMENT/PROJECT ARCHITECTURE

The continuous change in the traffic and road infrastructure creates the necessity in improvements of traffic surveillance and monitoring. The proposed solution solves the problem of this maintenance and it is expected to be future proof. As every method has its own disadvantages such as occlusion, lighting condition and similarities of the vehicles. Moreover, lot of criminal activities involves vehicles, so it become an important problem to solve in the present era. We are proposing a generalized model that will be able to monitor the traffic irrespective of the lighting conditions. In this project we develop a more reliable vehicle detection and tracking system to perform real time multi-camera, vehicle detection, tracking and re-identification. We choose deep learning algorithms that are fast enough to support real time analysis of vehicles within each camera and provide a model which will support combining partial tracks within and between multiple-cameras.

A. Overview of the project setup:

Our set up to capture the live traffic video comprises of the following main components, IP cameras, network and nanobeam to transfer the data to the processor. Cameras are placed on roof tops of two consecutive signals. The data processing servers may not be necessarily close to the place of video capture, the solution for that has been discussed in[18]. To transmit the data from the roof top to the servers in our lab we have made use of Nanobeam bridge devices. The camera and the bridge is connected to a router that communicates with the Universitys network. Once the data is transferred to the processing centers the video is processed. We have made use of the open sourced tensorflow framework. For detecting the vehicles in the video we have made use of Single shot Detector Mobilenet, the output from it is then used by the Kalman filter which keeps track of the vehicle continuously by prediction and updation as long as the vehicle is in the field of view of the camera. The number of vehicles in the intersection is calculated using Tensorflow count API. Another algorithm that runs in parallel is the anomaly detection that is based on DSA RNN and LSTM. This continuously looks for accident cues and distributes the cues to the candidate objects. If the accident is predicted or any vehicle violates the traffic rules is found then the vehicle is tracked and the details sent to the traffic authorities. The re identification of these vehicle is done by nearPy ANN a search optimizer.

Abbreviations and Acronyms

The Acronyms and Abbreviations referred in the report are compiled and mentioned below. (ANN)Approximate

Nearest Neighbour, (SIFT) Scale Invariant Feature Transform(ALPR) Automated License Plate Recognition, (Re-id) Re-identification, (ITS) Intelligent Transportation System, (SSD)Single Shot Detector , (LSH) Locality Sensitive Hashes, (CNN)Convolution Neural Network, (KNN)k-Nearest Neighbour, (SVM)Support Vector Matrix,(CCTV)Closed Circuit Television.

IV. SYSTEM DESIGN

The video data is captured using cameras at various locations in Silicon Valley. The videos are recorded at 25 frames per seconds (fps), with resolution of 960540 pixels. There are more than 140000 frames in the training dataset and 8250 vehicles are manually annotated using LabelImg image annotation tool. In this project we use NVIDIAs Tesla P100 GPU accelerator. NVIDIAs CUDA toolkit supports majority of Deep learning libraries like TensorFlow, Pytorch, Caffe, etc. The annotated dataset are trained using TensorFlow deep learning framework using the Single Shot Multibox model on a Tesla P100 GPU.

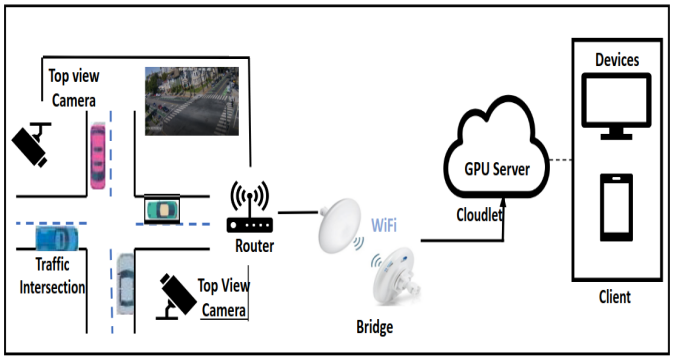

Fig. 1
System Architecture

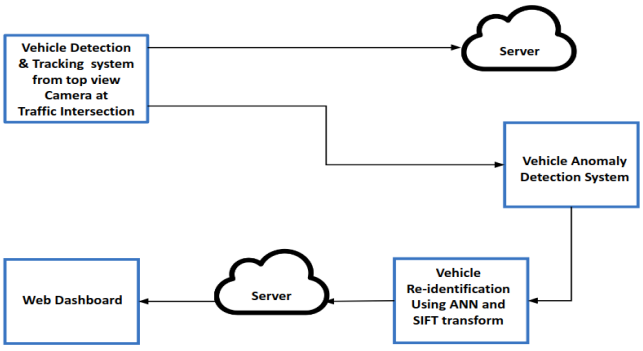

Fig. 2
Flow Diagram

A. Dataset

The initial step towards AI solution is getting the right dataset. Right data needs to be gathered which can correlate the outcomes to be predicted [3]. The data should be aligned with problem to be solved. In our proposed system we have used NVIDIA dataset for vehicle detection, tracking and re-identification. For anomaly detection we use ACCV dataset (VSLab Research) which consists of negative and positive examples of driving through diverse road scenes involving any accidents. After training the model we have used our dataset collected at San Jose State Universitys intersection at 7th street and San Fernando Street.

a. NVIDIA Dataset for vehicle detection, tracking and Re-identification: The dataset contains videos captured for several hours at multiple intersections and along highways in Silicon Valley and in Iowa. The dataset consists of 15 videos, with each video around 0.5-1.5 hours long. The videos are recorded at 30 fps and 1080p resolution (19201080)[4]. Videos are named in particular convention ad LocXY.mp4, where X is a location ID and Y is the video ID. Additional file associated with each video, LocXYmeta.txt, will contain metadata for the video, including location, category (highway or intersection), GPS coordinates, orientation, file size in bytes, video length, resolution, frames per second, and total number of frames[4].

Fig. 3
NVIDIA Dataset

B. Dataset for Anomaly Prediction:

The data set that we used to train the model for anomaly prediction was received from VSLab. It contained a dataset with many accident scenario in a high resolution. These datas were captured from dash cameras attached in the cars. It included diverse accident scenes pertaining to motor cycles and car hits. The dataset was complex compared to the KITTI data set which is collection of traffic data from Europe. In the current data set the number of moving cars, motorbikes, and pedestrians per frame are typically larger than other datasets. Accidents involving cars, motorbikes, etc. are all included in the data set. The street signs, billboards and road scenes are congested leading to a complex dataset[5]. The video clips contain positive and negative cases of accidents in the ratio

of 1:2 and there are around 100 frames per video clippings. The dataset is divided into training and testing datasets in the ration of 3:1. For testing the live video feed we captured the traffic videos using the camera set up at the roof top. The video received will be a non-annotated format which is then sent to our models for the analyses.

Fig. 4
VSLab Research Accident Dataset

C. Overall Neural Network Design for Intelligent Traffic Surveillance system

Recently a lot of research and models have been developed for the purpose of traffic surveillance using deep learning and machine learning. These use extensive computational resources and are data centric. In one of the existing state of art solutions, the vehicles are recognized using any of the deep learning algorithms like YOLO or SSD mobile net and are meant to perform dynamic video content analysis of the traffic video data and scale to large scale traffic camera video data [7]. The video frames are trained using SSD MobileNet model to achieve object detection and tracking. The SSD requires input images and true boxes for every vehicle during the process of training. The small sets of various aspect ratios are evaluated at every location of many feature graphs with a variety of scales. SSD which is a feed forward convolution neural network outputs a fixed size collection of bounding box and scores for the presence of object class. SSD addresses the low resolution issue in YOLO by making predictions based on feature maps taken at different stages of the convolutional network. It predicts the objects future location and reduces noise introduced by inaccurate detections [9]. Once the detection is made Kalman filter uses the result to keep track of the vehicle, it assigns binding box once it enters the view of the camera and removes the box when it leaves the field of vision. The filtering is based on the prediction and updating. The prediction step uses previous states to predict the current state and the update state uses the current measurement, such as detection bounding box location, to correct the state.

D. Vehicle Detection and Tracking

SSD MobileNet model is used for detecting and tracking of multiple vehicles using network of cameras placed on top of building to capture the view of traffic intersection. Mobile Nets combined with Single Shot Detectors (SSDs) provide a more efficient and accurate model to detect vehicles using deep learning. For vehicle detection, the captured image is taken as input to produce bounding box as output. TensorFlow Object Detection API, which is an open source framework built on top of TensorFlow to deploy object detection model[9]. The API also consists of pretrained collection of detection models using COCO dataset. COCO dataset consists of 90 different classes, with 14 classes related to vehicles, including bicycle, car, and bus, etc[9]. SSD MobileNet model meets our goal of obtaining balance between bounding box accuracy and good run time in detecting vehicles. Faster RCNN has better accuracy but lacks in speed. Whereas, YOLO model is fast but lacks in accuracy. SSD model balances the speed and accuracy and works best for vehicle detection in real time. SSD model addresses issues faced in YOLO model such as low resolution and low accuracy by predicting based on feature maps collected at different stages of convolution network. Predictions are made at each feature map for the given class labels and bounding box. The output of this model gives bounding boxes coordinates for detected vehicles in the format of ”[y-up, x-left, y-down, x-right]”.

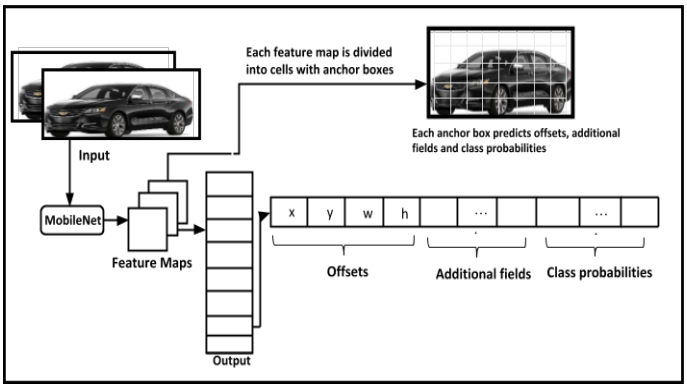

Fig. 5
TensorFlow SSD MobileNet architecture for vehicle detection

Tracking is achieved using Kalman filter algorithm which predicts vehicles future location. Some of the features of this algorithm are reduction of noise produced due to inaccurate detections, correction of errors in prediction based new findings. It also facilitates association process of multiple vehicles on the tracks[9]. Kalman filter has two main steps: i) Prediction ii) Update. The algorithm predicts the current state based on previous steps. It corrects the state using current measurement such as location of the bounding box . Following formulas are for the calculation[9]:
Prediction phase: notations
X: state mean
P: state covariance
F: state transition matrix
Q: process covariance
B: control function(matrix)
U: control input

Prediction phase: equations

$$\overline{X} = Fx + Bu \tag{1}$$

$$\overline{P} = FP^T + Q \tag{2}$$

Update phase: notations H: measure function (matrix) Z: measurement R: measurement noise covariance Y: residual K: Kalman gain
Update phase: equations

$$Y = z - H\overline{X} \tag{3}$$

$$K = \overline{P}H^T(H\overline{P}H^T + R)^-1 \tag{4}$$

$$X = \overline{X} + Ky \tag{5}$$

$$P = (1 - KH)\overline{P} \tag{6}$$

E. Speed Anomaly

Speeding is the most common anomaly and its hard to targeting the person who commits it. The pixel scale from the image recorded is calculated precisely to predict the speed of the vehicle on the intersection. For our scenario the camera is mounted to point the intersection near the school, So the speed of the vehicle should be less than 30 mph. We are concentrating on two directions 1. From west San Fernando to east San Fernando and 2. From 7th street to East San Fernando. The vehicles are detected in each and every frame. The vehicle detected in the first frame will be in the same order of detection in the second frame and on the further frame it appears. So it is easier to keep track of the same vehicle. The video is recorded in 30 fps, therefore applying the scale constant we can able to predict the speed of the vehicle using simple calculation considering how much pixels the vehicle travelled in this frame compared to the previous frame. Once we identify the speed of the particular vehicle we just need to check whether the vehicle speed exceeds the threshold which is in our case 30 mph. If it doesnt then we can continue checking for the speed violation. In case any vehicle exceeds the speed limit then the vehicle is been captured and sent to the the application that keeps track of the incoming vehicles beside this intersection.

F. Traffic Anomalies Anticipation of Accidents

Anticipation of accidents is very important in an AI based traffic system. It was found that Googles autonomous cars in 2009 lead to few minor accidents. Hence, it is necessary that vehicles should be in a position to anticipate the accidents. The traffic accidents are anticipated using Dynamic Spatial Attention (DSA) Recurrent Neural Network (RNN). The DSA RNN models can perform the following functions The distribution of soft-attention to candidate objects to bring the cues together Perform the model temporal dependencies of all the cues to predict the road accidents.

The anticipation of accidents is concentrated less when compared to the detection of traffic road lane detection. In this feature, we are implementing the state of art methodologies object detection to detect the candidate objects and also bring in a complete-frame and appearances of object-based detection. It is expected to arrive at the highest mean precision of above 75 percent, thereby bringing in the possibilities of RNN.

Here are few functionalities of the neural network model that we are using to anticipate the accidents DSA RNN It anticipates the accidents in every frame and learns to do the distribution of the attention to the candidate objects. RNN This neural network is used for performing the sequence modelling. It is used along with LSTM (Long Short-term Memory) which will help in the modelling of long term dependencies. Exponential Loss: We are adopting the exponential loss functions to estimate the loss functions for positive examples.

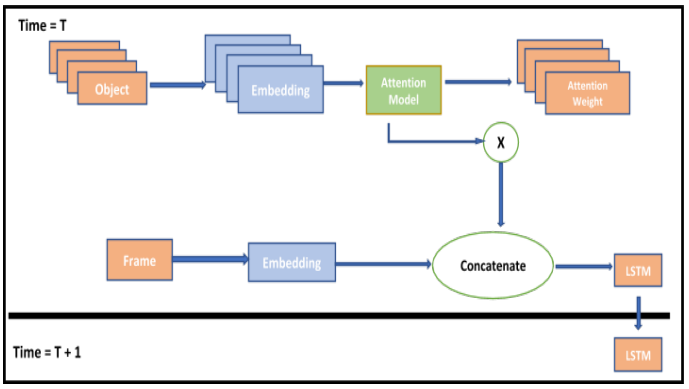

Fig. 6
DSA RNN and LSTM for anticipation of the accidents.

Features. The appearance and the motion cues are crucially important for the anticipation of the accidents. The appearance is captures using VGG network. The PCA is used to diminish the trajectory features to dimension 100. RD motion features are used in designing a Relatively- motion (RM features). Candidate objects. The proposed model with be distributing its soft-attention to the regions. This fine tuned detector is achieving the mean average precision of 52.3 percent. Model-Learning Our experiments make use of a learning rate of 0.0001 and bring an epoch of 40 steps. The model is implemented using TensorFlow.

G. Re-Identification of vehicles

The vehicle that violated speed is cropped out from the whole frame and sent to the function to identify it in the traffic. The same object detection model using SSD is incorporated here which detects every vehicle on the frame and we extract the CNN features from those detection. These detections are matched with the interested vehicle feature that has been cropped from the speed violation. We are using weighted cosine distance to identify the anomaly as it is quicker and that helps in tracking the wanted vehicle in real time. Once the features are matched with the cosine the objects are ranked by ordering from minimum distance scores. We are retrieving 10 vehicles from the similarity matching. As it is compared immediately 10 vehicles will be sufficient to get the desired

and it also makes it easier by reducing the further computation. Those 10 vehicles are matched again with the wanted one using SIFT transform which gives us the vehicle that violated the speed limit. By this way we can continuously apply the same algorithm on consecutive camera intersection to keep track of the vehicle.

H. Color Classification and Speed Monitoring

Color recognition and speed monitoring of vehicles is important part of intelligent traffic system. It plays important role in vehicle recognition and contributes significantly in re-identification of vehicles. This section explains in detail about how effective features are extracted and suitable classifiers are designed for the recognition of vehicle color and also how vehicle speed is monitored. Color classification and speed monitoring are implemented using TensorFlows SSD MobileNet model. Following is the block diagram indicating technologies used for performing vehicle speed monitoring system and color classification system. Vehicle speed prediction model is designed by implementing OpenCV through image pixel manipulation and calculation. For vehicle color detection, K-Nearest Neighbors Machine Learning Classification Algorithm and trained with color histogram.

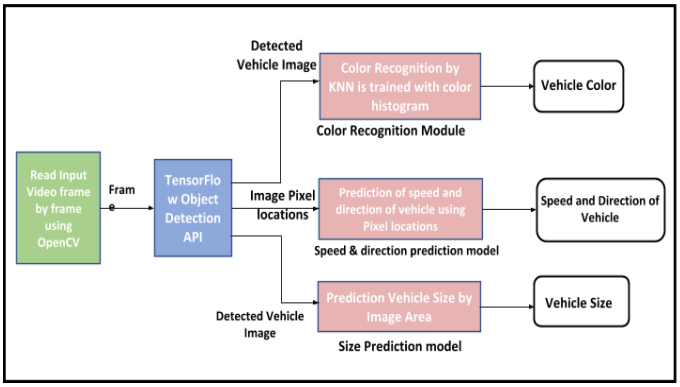

Fig. 7
Vehicle Color recognition, speed monitoring, and prediction of the type of the vehicle.

V. RESULTS

A. Results of our model for vehicle detection

The models that were trained using the SSD mobilent model are evaluated on the dataset gathered by fixing multiple reolink cameras at various angles facing the traffic intersection for capturing the live stream data. Two cameras are used to get the stream from different views. There is no proper answer to determine as to which model is the best for the accuracy. Our model SSD MobileNet was chosen and evaluated based on the following parameters namey the feature extractors, the output strides, the resolution of the input images, the strategy of matching and IoU threshold, the non-max suppression in the threshold, the positive vs the negative anchor ratio, the

SSD Mobilenet	No of Images	Detection Accuracy
Training	1284	83%
Testing	466	83%

TABLE I
Comparing accuracy for vehicle detection and tracking.

proposal number or the predictions, the encoding of boundary box, augmentation of the data and the training dataset. It is observed that SSD MobileNet is the fastest with the best accuracy and within the fastest detectors. It performed very well for large objects and even outperforms the faster RCNN in terms of accuracy. The accuracy of a model is defined in terms of the confusion matrix which is as follows.

$$Accuracy = \frac{t_p + t_n}{t_p + tn + f_p + f_n} \tag{7}$$

$$t_p - truepositives, f_p - falsepositives$$

$$t_n - truenegatives, f_n - falsenegatives$$

The few models that were used for vehicle detection namely SSD MobileNet, YOLO, CNN with adopting an architecture from RCNN have different accuracy and inference time. Around 750 traffic video clips consisting of 100 frames (5 seconds) were used for training. The training was performed for 3 hours. The initial vehicle detection at the traffic intersection using SSD MobileNet are depicted in figure 7 and figure 8. The total number of vehicles that are used for training are shown in the table below. The following screenshot shows the training process using TensorFlow SSD MobileNet model.

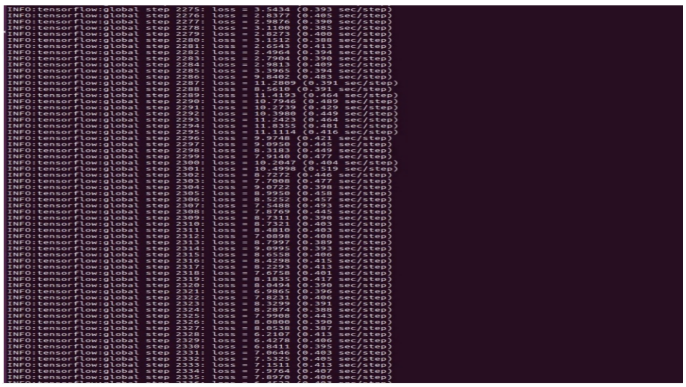

Fig. 8
Vehicle Training using Tensorflow SSD MobileNet

In order to compare the performance between the various detection algorithms, the CNN architecture was also employed to perform the vehicle detection. The weights of YOLOv1 are used to train the dataset. It appeared to be 20x faster than the popular SVM + HOG pipeline. The whole image was run through the convolution neural network(CNN) to generate one or more bounding boxes directly. The regression on the whole image are used to generate the bounding box. The

Fig. 9
Vehicle Detection using SSD MobileNet at the traffic intersection

Fig. 11
Vehicle detection using YOLO

Fig. 10
TensorBoard depicting the losses encountered after training using SSD MobileNet

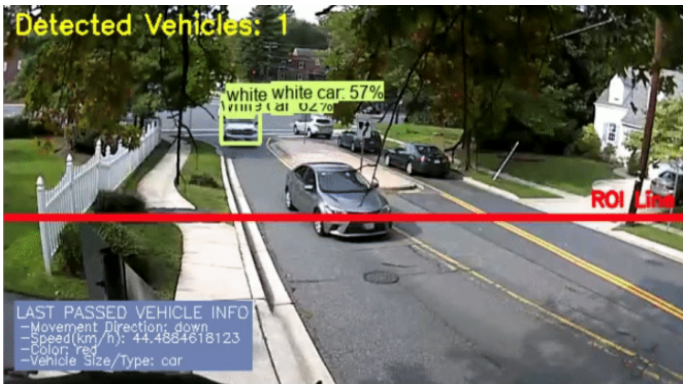

Fig. 12
Vehicle detection, speed monitoring and type of the vehicle

generated bounding boxes are directly from the CNN. The following screenshot indicate the vehicle detection at the traffic intersection using YOLO weights.

The second feature of the intelligent traffic surveillance system is the vehicle tracking which is running on Kalman filter tracking algorithm on top of the SSD MobileNet model. The prediction phase of the Kalman tracking algorithm was used to evaluate the tracking.

B. Traffic Color detection and Speed Monitoring System

The vehicle speed monitoring, direction and the count of the vehicles are performed using TensorFlow sSSD MobileNet model. The TensorFlow object counting API is used to make the count of the number of vehicles in the datasets. This open source framework is employed in the project to track the number of vehicles crossing the traffic intersection. The below screenshot shows the number of vehicles getting detected.

The other most important feature that was implemented in the project were the traffic anomalies using SSD MobileNet. The vehicles violating the speed threshold are detected are detected, helping the traffic surveillance system to alert the public. In addition to this, we also implemented the detection of traffic congestion using TensorFlow inception model which classifies the vehicle into low, medium and high traffic congestion.The system identifies the highly congested traffic image and using inception model classifies the traffic congestion. A very high traffic congestion of 0.925 is observed.

The SSD MobileNet model had a mean accuracy of 84.1 percent and gave better results in comparison with YOLO model and faster RCNN .

Model	mAP(%)
SSD MobileNet	84.1
YOLO	75
Faster RCNN	70.4

TABLE II
Comparing accuracy for vehicle detection and tracking.

Network Model	Frames per second(FPS)
SSD MobileNet	58
YOLO	45
Faster RCNN	7

Fig. 13
Traffic congestion detection using TensorFlow Inception model.

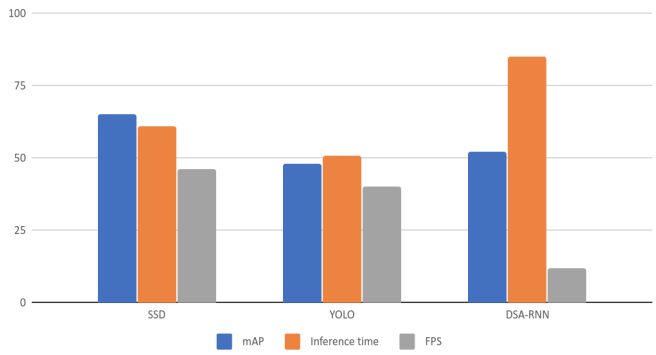

Fig. 15
SSD vs Other Models Performance

It is also observed that SSD MobileNet has the highest frame per second (FPS) on our vehicle dataset.

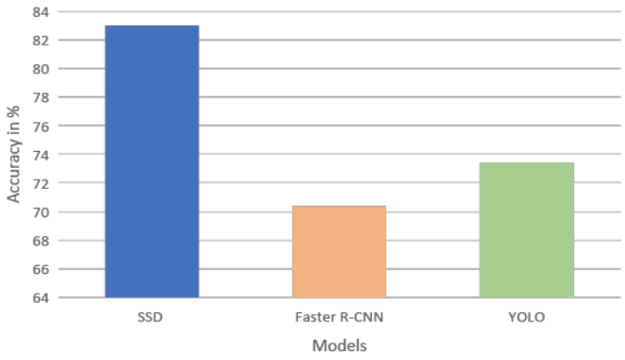

Fig. 14
Accuracy comparison between SSD, Faster RCNN and YOLO

C. Traffic Anomaly detection

i) Accident detection using DSA RNN The DSA RNN model was employed on a dashcam dataset to predict the various traffic anomalies. It anticipates the accidents in every frame and learns to do the distribution of the attention to the candidate objects.This neural network is used for performing the sequence modelling. It is used along with LSTM (Long Short-term Memory) which will help in the modelling of long term dependencies. We adopted the exponential loss functions to estimate the loss functions for positive examples.. The appearance and the motion cues are crucially important for the anticipation of the accidents.The PCA is used to diminish the trajectory features to dimension 100. The proposed model distributes its soft-attention to the regions. This fine tuned detector is achieving the mean average precision of 52.3 percent. Our experiments make use of a learning rate of 0.0001 and arrive at an epoch of 30 steps. The model is implemented using TensorFlow.

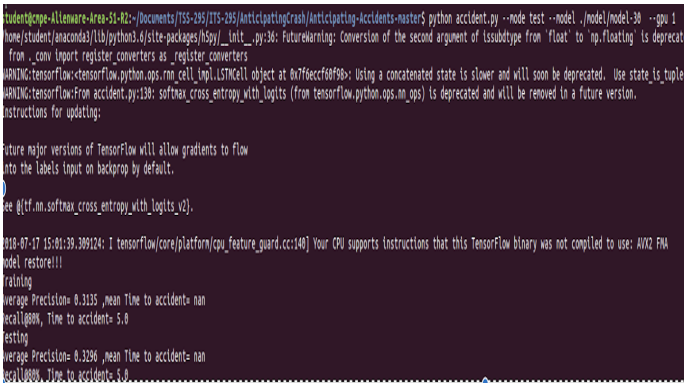

Fig. 16
Accident precision detection in the terminal

In order to avoid miscellaneous errors we increased the threshold from 30 to 35 mph. As far we witnessed the speed shown by the system is comparable to what we can see in the video. As it can be easily identified in the video by just comparing the number of frames the vehicle present in the video. If the vehicle present in the video for longer time that its obvious that the vehicle moving slower and similarly if the vehicle is in the video just for two or three frames it is most likely violated the speed. And one more problem is that when the vehicle is in halt that wont be consider or the speed of the vehicle at that state is zero. While visualizing it we should keep that in mind. Our system matches the visualization when we tried to show whenever the vehicle violated the speed and the vehicle doesnt displayed in the system more than four times as it went past the field of view in just four frames. The result below shows that in the figure 1 as the vehicle speed is under the limit it keeps on processing further without any interruption. And similarly in the figure 2 as soon as the vehicle went past the threshold it stops the processing and shows us the vehicle picture and send the cropped image for processing.

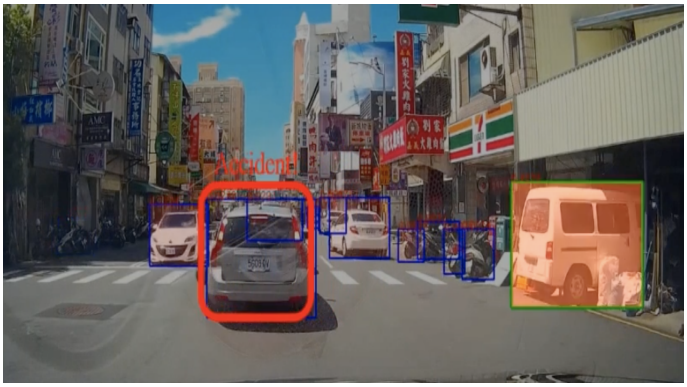

Fig. 17
Accident detection after running DSA RNN LSTM model

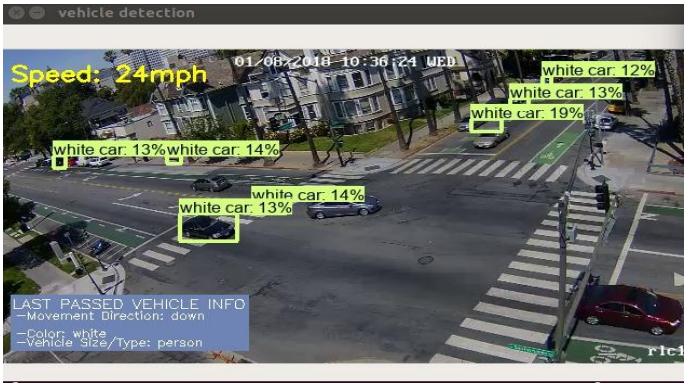

Fig. 18
Vehicle Speed, color and direction detection using SSD MobileNet

ii) Vehicle Reidentification We tried the vehicle re identification part in the highways where we didnt get the results as expected. It was mainly due to the fact that the number of vehicles that exists in the free in a given amount of time. There is also more exits and connections at regular distances which makes it hard to re identify the vehicles in the freeway. However in the intersection the re identification is relatively simpler as there wont be as many vehicles as in a freeway and also the vehicle speed is lot lesser than on the freeway which helps in avoiding image distortion when detecting vehicles. The same reflects in the result as well. Almost all the times we were able to identify the vehicle and it is relatively faster than what we got in the freeway due to the same reason. As we initially use the approximate nearest neighbor to reduce the options in re identification, it reduces the time exponentially and also the tried and tested out SIFT algorithms didnt fail for our scenario as well. It performed exceptionally and also a little quicker due to the smaller image of detected cropped vehicle.

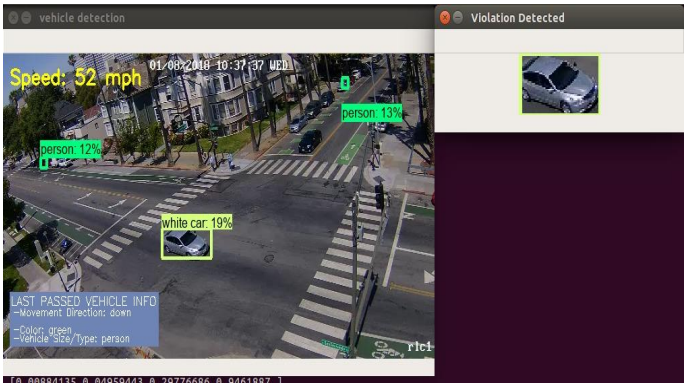

Fig. 19
Vehicle Violating the speed is reidentified

VI. Conclusion and Future Work

A well rounded solution for developing an intelligent traffic surveillance system at the traffic intersection has been proposed in our paper and it overcomes the drawbacks of currently existing traffic system provided by Computer vision techniques and single camera. Our solution also adds on many useful features such as vehicle detection, vehicle tracking, traffic anomaly detection, and vehicle reidentification, thereby making it as a sophisticated deep learning system. The SSD MobileNet model is used as the foundation for custom network model has been developed for identifying vehicles at the traffic intersection. Further, it is also used to perform other traffic monitoring features such as speed count using TensorFlow API and vehicle direction. From the results obtained we can observe that there is a reduction in the inference time for the with SSD MobileNet (19.98ms) when compared with YOLO model. The SSD MobileNet has a very good mAP value of 83.1 percent when compared with the other models.In addition, we also implemented the traffic anomaly detection using DSA RNN and vehicle reidentification using approximate nearest neighbor, a deep learning paradigm. We performed a complete traffic violation system at the SJSU traffic intersection by placing two reolink cameras on top of the engineering building. Using the powerful combination of deep learning and computer vision, we reidentified the vehicles violating the speed at the junction. In addition, the SSD model also identified the color and the direction of the vehicle. Our model has more accuracy and is with lots of sophisticated features which are required for traffic surveillance system when compared to a single camera based traffic monitoring system using computer vision techniques. .We have achieved vehicle detection, tracking, count estimation, direction and color classification using a powerful combination of deep learning and computer vision. For the future work, we plan to improvise the vehicle re-identification using the Siamese Neural Network which is a combination of identical neural networks to identify the similarity between two images. We also plan to improve the accuracy and the precision of the DSA RNN model to predict the road accidents. These optimized methods will yield a more

sophisticated and an efficient traffic surveillance system.

REFERENCES

[1] M. O. Mehmood and A. Khawaja, Multi-camera Based Human Tracking with Non-overlapping Fields of View, Multi-camera Based Human Tracking with Non-overlapping Fields of View - IEEE Conference Publication.

[2] Ullah and H. J. Lee, Moving Vehicle Detection and Information Extraction Based on Deep Neural Network Int'l Conf. IP, Comp. Vision, and Pattern Recognition, 2017.

[3] Datasets and Machine Learning, Skymind. [Online]. Available: https://skymind.ai/wiki/datasets-ml.

[4] Data and Evaluation, AI CITY CHALLENGE. [Online]. Available: https://www.aicitychallenge.org/?page-id=9.

[5] F.-H. Chang , Y.-T. Chen , Y. Xiang, and M. Sun, Anticipating Accidents in Dashcam Videos, aliensunmin.github.io, 20-Mar-2017. [Online]. Available: https://aliensunmin.github.io/project/dashcam/

[6] S. Ren, K. He, R. Girshick, and J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 11371149, Jun. 2016.doi:10.1109/TPAMI.2016.2577031

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. E. Reed. SSD: single shot multibox detector. CoRR, abs/1512.02325,2015.

[8] F.-H. Chan, Y.-T. Chen, Y. Xiang, and M. Sun, Anticipating Accidents in Dashcam Videos, Computer Vision ACCV 2016 Lecture Notes in Computer Science, pp. 136153, 2017.

[9] kcg2015/Vehicle-Detection-and-Tracking, GitHub, 25-Jul-2018. [Online]. Available: https://github.com/kcg2015/Vehicle-Detection-and-Tracking.

[10] S.-J. Kim , H.-M. Seo, and W. C. Park, Network Bridge System for Interoperation of ZigBee-UPnP Network - IEEE Conference Publication, Design and implementation of autonomous vehicle valet parking system - IEEE Conference Publication, 15-Dec-2011. [Online]. Available: https://ieeexplore.ieee.org/document/6104709/.

[11] T. Chavdarova and F. O. Fleuret, Deep Multi-camera People Detection, 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Jan. 2018.doi:10.1109/ICMLA.2017.00-50

[12] Koutsia, T. Semertzidis, K. Dimitropoulos, N. Grammalidis, and K. Georgouleas, Intelligent traffic monitoring and surveillance with multiple cameras, 2008 International Workshop on Content-Based Multimedia Indexing, 2008.

[13] S. Kuciemba and K. Swindler, Transportation Management Center Video Recording and archiving best general practices Technical report 2016.

[14] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR. (2015)

[15] NVIDIA Tesla P100: The Most Advanced Data Center GPU Ever Built, Tesla P100 Most Advanced Data Center Accelerator—NVIDIA.

[16] V. Sze, Y.-H. Chen , and T.-J. Yang , Efficient Processing of Deep Neural Networks: A Tutorial and Survey - IEEE Journals Magazine, 20-Nov-2017.

[17] H. S. Parekh, D. G. Thakore, and U. K. Jaliya, A Survey on Object Detection and Tracking Methods, Int. J. Innov. Res. Comput. Commun. Eng. IJIRCCE, vol. 2, no. 2, 2014.

[18] X. Liu, W. Liu, T. Mei, and H. Ma, PROVID: Progressive and Multimodal Vehicle Reidentification for Large-Scale Urban Surveillance, IEEE Transactions on Multimedia, vol. 20, no. 3, pp. 645658, Sep. 2017.doi:10.1109/TMM.2017.2751966

[19] C. Li, L. Guo, and Y. Hu, A New Method Combining HOG and Kalman Filter for Video-based Human Detection and Tracking, Int. Congr. Image Signal Process. CISP2010, 2010.

[20] Ihsan Ullah and Hyo Jong Lee, Moving Vehicle Detection and Information Extraction Based on Deep Neural Network International Conference IP, Computer Vision and Pattern Recognition, IPCV'17

[21] S. Han, and N. Vasconcelos, Object-Based Regions of Interest for Image Compression, University of California, San Diego.

[22] W. L. Khong, W. Y. Kow, H. T. Tan, H. P. Yoong, and K. T. K. Teo, Kalman Filtering Based Object Tracking in Surveillance Video System, CUTSE Int. Conf., 2011.

[23] D. Matti, H. K. Ekenel, and J.-P. Thiran, Combining LiDAR space clustering and convolutional neural networks for pedestrian detection, 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2017.

[24] C. Wang and H.-M. Tsai, Detecting urban traffic congestion with single vehicle, 2013 International Conference on Connected Vehicles and Expo (ICCVE), 2013.

[25] Home. (n.d.). Retrieved from https://www.autoaccident.com/statistics-on-intersection-accidents.html

[26] Intersection Safety. (n.d.). Retrieved from https://safety.fhwa.dot.gov/intersection/