MULTI SENSOR MULTI OBJECT TRACKING IN AUTONOMOUS VEHICLES

A Thesis

Submitted to the Faculty

of

Purdue University

by

Surya Kollazhi Manghat

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

December 2019

Purdue University

Indianapolis, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Mohamed El-Sharkawy, Chair

Department of Engineering and Technology

Dr. Maher Rizkalla

Department of Engineering and Technology

Dr. Brian King

Department of Engineering and Technology

Approved by:

Dr. Brian King

Head of the Graduate Program

This thesis is dedicated to my parents and ever supportive husband

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Mohamed El-Sharkawy for his motivation, expertise guidance, constant support and patience, without which the successful completion of my thesis would have been impossible. I would like to thank entire department of Electrical and Computer Engineering for the help and support through out my course work in IUPUI. A special mention goes to Sherrie Tucker who made my life so easy throughout my course work.

I would also like to extend my sincere gratitude towards my fellow lab mates Dewant Katare, Akash Gaikwad, Durvesh Pathak and Sreeram Venkitachalam, who made my time in the university more enjoyable. I would also thank my family and friends who gave support throughout the tough times.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

ABBREVIATIONS

MOT	Multi Object Tracking
NN	Neural Network
GPS	Global Positioning System
FCW	Forward Collision Warning
LiDAR	Light Detection And Rangig
IMU	Inertial Measurement Unit
GNSS	Global Navigation Satellite System
DA	Data Association
IOU	Intersection Over Union
CPU	Central Processing Unit
2D	Two Dimension
3D	Three Dimension
FOV	Field Of View
KF	Kalman Filter
EKF	Extended Kalman Filter
CV	Computer Vision
RNN	Recurrent Neural Network
ROI	Region Of Interest
MOTA	Multi Object Tracking Accuracy
MOTP	Multi Object Tracking Precision
MT	Mostly Tracked
ML	Mostly Lost
PF	Particle Filter
ADAS	Advanced Driver Assistance System

FP	False Positive
FN	False Negative
GT	Ground Truth

ABSTRACT

Kollazhi Manghat, Surya. M.S.E.C.E., Purdue University, December 2019. Multi Sensor Multi Object Tracking in Autonomous Vehicles. Major Professor: Mohamed El-Sharkawy.

Self driving cars becoming more popular nowadays, which transport with it's own intelligence and take appropriate actions at adequate time. Safety is the key factor in driving environment. A simple fail of action can cause many fatalities. Computer Vision has major part in achieving this, it help the autonomous vehicle to perceive the surroundings. Detection is a very popular technique in helping to capture the surrounding for an autonomous car. At the same time tracking also has important role in this by providing dynamic of detected objects. Autonomous cars combine a variety of sensors such as RADAR, LiDAR, sonar, GPS, odometry and inertial measurement units to perceive their surroundings. Driver-assistive technologies like Adaptive Cruise Control, Forward Collision Warning system (FCW) and Collision Mitigation by Breaking (CMbB) ensure safety while driving.

Perceiving the information from environment include setting up sensors on the car. These sensors will collect the data it sees and this will be further processed for taking actions. The sensor system can be a single sensor or multiple sensor. Different sensors have different strengths and weaknesses which makes the combination of them important for technologies like Autonomous Driving. Each sensor will have a limit of accuracy on it's readings, so multi sensor system can help to overcome this defects. This thesis is an attempt to develop a multi sensor multi object tracking method to perceive the surrounding of the ego vehicle. When the Object detection gives information about the presence of objects in a frame, Object Tracking goes beyond simple observation to more useful action of monitoring objects. The experimental results conducted on KITTI dataset indicate that our proposed state estimation system for Multi Object Tracking works well in various challenging environments.

1. INTRODUCTION

1.1 Introduction to Autonomous Vehicles

Autonomous cars have started as distant dream of researchers and automotive industry. An autonomous car can be seen as a huge system with various sensors attached to it and various algorithms running to take an adequate at precise time. This is also known as driver-less car or self driving car where all the actions of a driver is done by the autonomy implemented in the car. Computer vision has major role in making this possible. Autonomous vehicles sense their surroundings with various sensors attached and the algorithms interpret the raw data from sensors to identify presence and dynamic nature of obstacles in the navigation paths. There are mainly 3 primary sensors such as LiDAR, RADAR and camera for autonomous vehicles, These sensors together update the surrounding visual information for the vehicle in real-time, allowing the vehicles to be aware of their positions in unknown environments. There are additional sensors which gives the ego vehicle information on yaw, pitch, roll. Others which gives the map of its surroundings like travel path, and the vehicles relative position in space. In order to come up with this, an autonomous car uses sensor equipment such as Light Detection And Ranging Sensors (LiDAR), Radar sensors, Cameras, GPS Units, Ultrasound Sensor and Inertial Measurement Unit (IMU). LiDAR is continuously rotating LiDAR kept on top of the car to get the 3D view of the surrounding. The measurements from LiDAR combined with other sensors are used to get the complete view of the surrounding. GPS units in the car is used to determine the car's position in Global Positioning System units. This is perceived by using 3 satellites. Cameras are commonly used technology for lane detection and obtaining other information. The advancement in the image processing gives the details of the sign boards, lights and vehicles in ego vehicle's surrounding.

There are short range and long range radars available in the market. It uses time of flight for operation and it is an active sensor which can derive the actual location of objects in the surrounding in world coordinates.

The perception task of vehicle is collecting information from surrounding using multiple sensors which can be a visual data or dynamic information and interpreting the surrounding using the collected data. The collected data will be in reference with the vehicle position or sensor position. Further processing of this information using mathematical model will give the actual information of ego vehicle's surrounding. At the same time human driver can perceive this by looking in to the scene. Recent works in computer vision uses data from multi sensor technologies and advanced deep learning model can fuse this to interpret the environment.

1.2 Motivation

Self driving vehicles are the future of automotive domain. Computer Vision plays major role in achieving this. Recently, a lot progress has been made in Autonomous Driving field [10] [26], but complete autonomy is still challenging. A complete autonomous system should have the ability to perceive its surrounding environments, then make decisions based on what it perceives and finally manipulate the environment or actuate a movement. The ADAS system in Autonomous Vehicle's such as Forward Collision Warning (FCW), Auto Cruise Control (ACC) etcetera not only detect the but also keep track of the surrounding vehicles.

There are several algorithms present for object detection, with YOLO and SSD among the most popular. Detecting objects in every frame is like throwing all the information that you have from previous frames and starting it all over again. In object tracking we use image measurements to estimate position of object, but also incorporate position predicted by dynamics. This can reduce the noise in the measurements added due to the sensor degree of error. Moreover, tracking reduces the work to look for the object as it restrict the search area with the prediction. But, object detection in every frame then you have to search over whole scale space. An object tracking method will be able track even when viewpoint changes, an object detection method can not do it if not trained for. Using series of measurements in each video frame made over time, motion tracking can estimate, predict present and future locations [1] [39] [40]. This help the tracking to predict the position of object which is failed to detect in the sensors.

The advancement in the detection methods would help to implement a more accurate and less complex MOT system. There are a lot of research are going on to implement a real time tracker with good accuracy. This thesis is a humble trial to implement a lean MOT system using multi sensor data with better accuracy.

1.3 Contribution

This thesis implements a Multi Object Tracking methodology using multi sensor data. The method uses Camera and LiDAR data. The tracking methodology goes beyond simple observation by detection, it allows to know the position and dynamic information of all the moving objects present in the environment. This thesis implements a multi target tracking method by following a lean way of multi-target tracking implementation. The thesis has done in two steps. As the first step MOT has implemented on the camera data from KITTI [4] dataset. The 2D tracker results are converted to 3D coordinates. The second step is implementing 3D Multi Object Tracking on LiDAR data and fusing the outputs of both sensors. This method improved the MOT accuracy and resulted in an high grade MOT system. Object Tracking is an integral part of environment sensing, which enables the vehicle to estimate the surrounding objects trajectories to accomplish motion planning. The advancement in the object detection methods [7] [9] greatly benefits when following the tracking by detection approach. This thesis implements an online tracking method by following tracking by detection approach. In contrast to the other works which tend to focus more on accuracy of the system than system complexity, this thesis focuses on to develop an accurate, simple and real-time MOT system. We show that our work which implemented in lean way works good compared to the state-of-the-art performance available on the KITTI [4] dataset. The lean way of implementing our system results in running the system at a rate 200 frames per second.

2. OVERVIEW OF AUTOMOTIVE SENSOR TECHNOLOGIES

This chapter gives a brief overview of the various sensor technologies applied in autonomous vehicles and the concept of sensor fusion. The first part presents the most common sensors that build up the perception systems: Ultrasonic, RADAR, LiDAR, cameras, IMU and GNSS.

2.1 Various Sensor Technologies

Various sensor technologies introduced here are approached in a different way by exploring the physical foundations behind each sensor's operation. Electromagnetic spectrum that are used by various sensors are different. So studying these by considering these spectra of operation will give deep understanding about the technologies.

2.1.1 LiDAR - Light Detection and Ranging

One of the mostly used technology for 3D mapping of environment is LiDAR sensor. It uses time of flight criteria for the measurement operation. Light from the laser diode is emitted and received back by receiver. The time taken by this is counted towards measurement. The light emitted are in infrared range (905 nm or 1550 nm). The energy level is different for these rays. The first one requires less energy than the second. LiDARs are classified according to the information they provide. It can be 1D, 2D or 3D LiDAR. The 1D LiDAR measures the distance to the object using one coordinate system. 3D is a rotating LiDAR which can give 3D map of the environment.

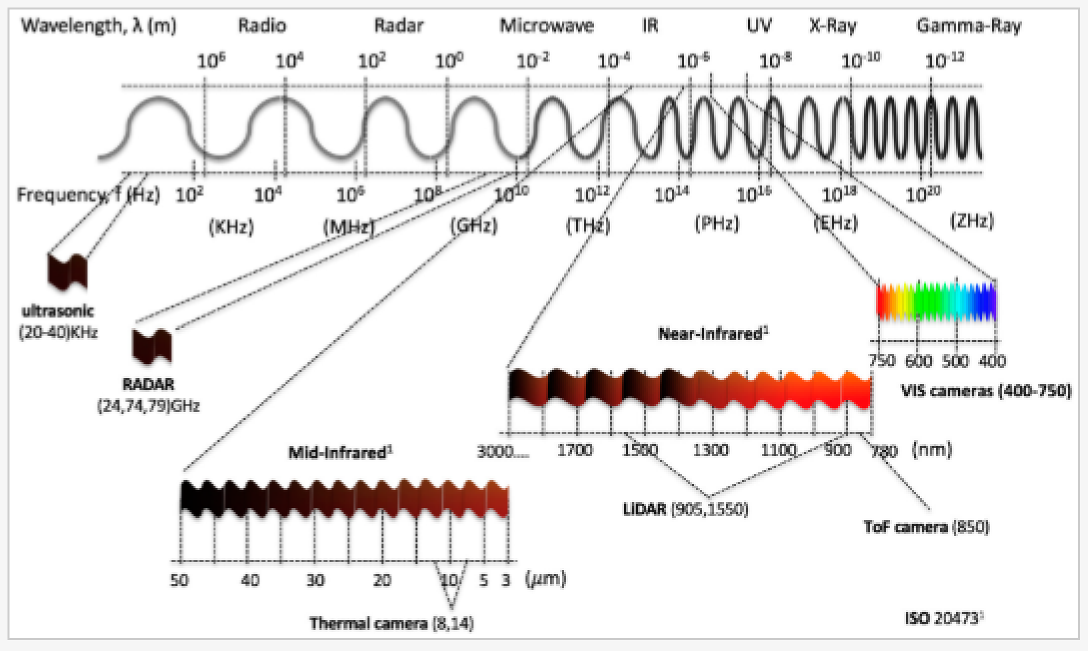


Fig. 2.1. Different Spectra used in Perception Systems.

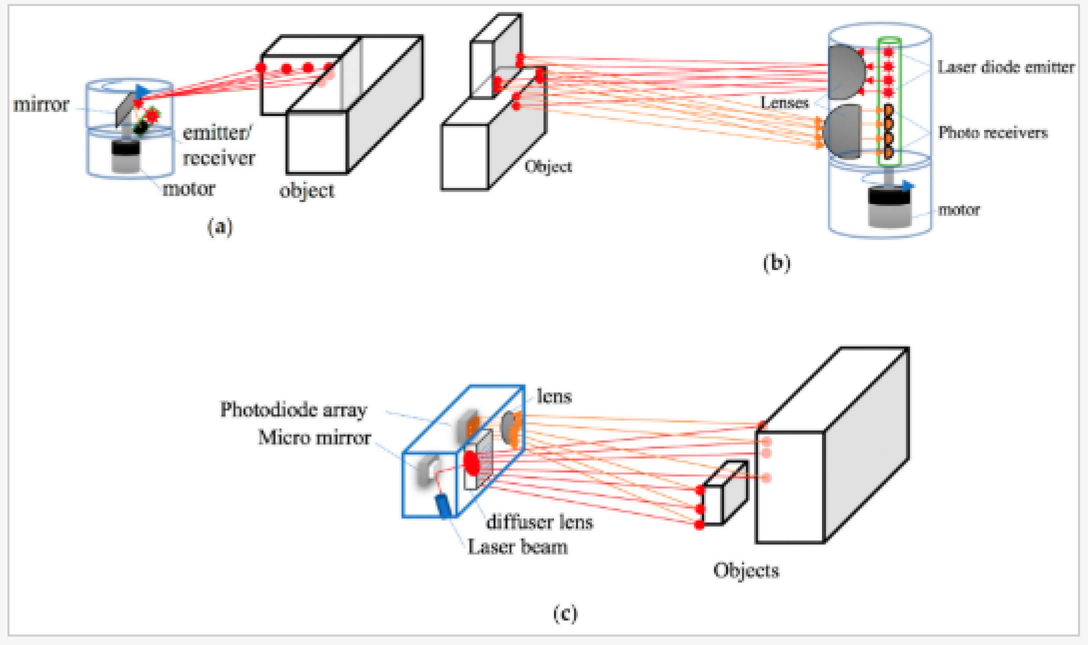


Fig. 2.2. Various LiDARs (a) Spinning 2D LiDAR, (b) Spinning 3D LiDAR, (c) SSD 3D LiDAR

2.1.2 Camera

Camera can be classified to 2 types according to the wavelength it is operating on. It can be a visible light based camera or Infrared based camera. Cameras can give colored images or grey scale. The colored images will have less resolution. The usual cameras will be 2D representation of the capturing 3D environment. It can not give actual distance measurement of the object. Camera can be a 2D camera or stereoscopic camera. The second one uses 2 cameras to capture the same scene and from the disparity information it gives the depth to the object in world coordinates.

2.1.3 RADAR (Radio Detection and Ranging)

Radar which is Radio Detection and Ranging uses radio waves or microwave to measure the surrounding object's location. Radar has started it's presence in military applications first. This can be used to detect any objects with long range. The absorption by the environment will be comparatively low in this. Radar measure the distance to the object by emitting the radio waves and calculating the time of flight of these signals and the received signal in the receiver. Radar is an active sensor, it not only give the distance to the targets but also provide the speed and angle of direction of the targets precisely. Radars are widely used in autonomous car for various ADAS systems in combination with other sensors. In self driving cars radar are kept as an array of antennas which will generate lobes that allow the car to increase the coverage and see targets.

2.1.4 Ultrasonic Sensor

Ultrasonic senors are another category of sensor found in self driving cars. As the name indicates it uses sonic waves for it's operation. The wavelength used by the emitted rays are in Kilo Hertz. This sensor also work on the time of flight principle. Sonic waves are emitted from the emitter and receive back by receiver.

$$d = c^2 \times ToF \tag{2.1}$$

The c velocity of the wave is in meters per second and ToF is time of flight in seconds. This calculates the distance to the object.

2.1.5 IMU- Inertial Measurement Unit

Inertial measurement unit is a proprioceptive sensor which gives the information of it's own. It measures force, angular measurements and magnetic field of the car.It is composed of a three axis gyroscope and accelerometer, together the IMU will act as 6-axis IMU. 3 axis magnetometer can be added to this to make the IMU 9 axis.

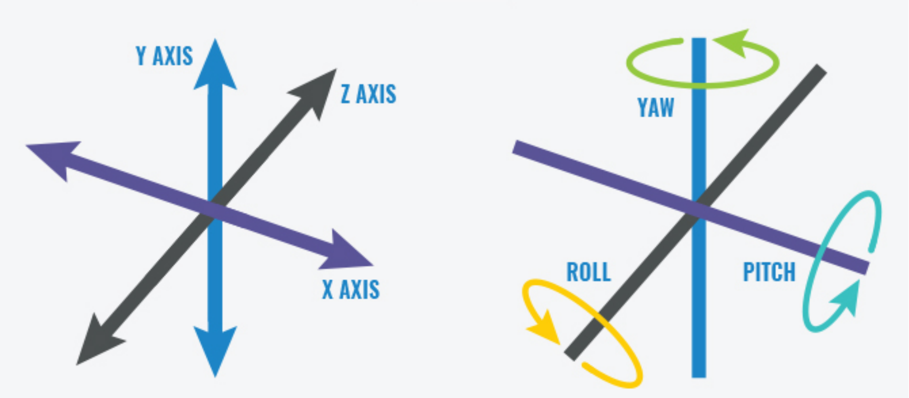


Fig. 2.3. Six Degrees of Freedom

2.1.6 Global Positioning Systems

Global Navigation Satellite System (GNSS) is the most widely used technology for vehicle positioning on land, sea and air. The operation of GPS is on time of flight principle. It provides an absolute position of a antenna using 3 fixed satellites orbiting approximately 20,000 km from the earth surface. The satellites emit signals containing information on satellite, its position, orbital parameters, etc. The receptor receives this and process to come up with the location information.

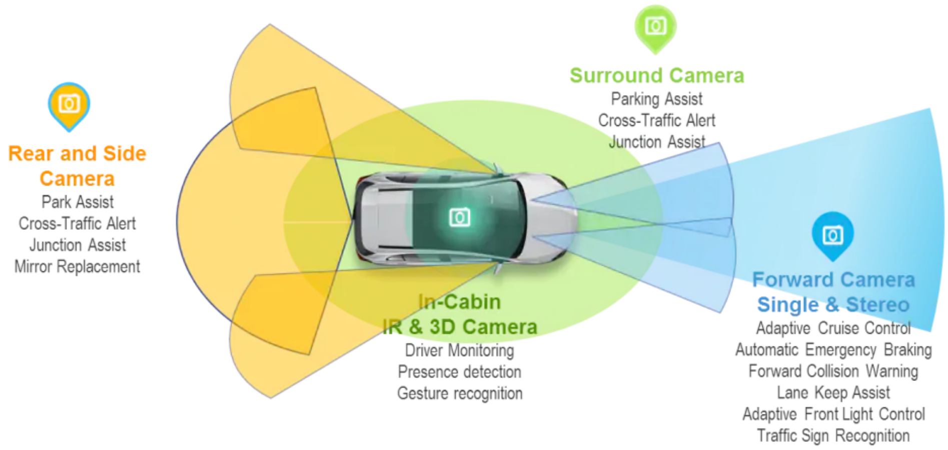


Fig. 2.4. General Architecture of the Perception Task [21]

2.2 Sensor Fusion

Perception of environment gives the data to the autonomous car algorithm to take action according to presence and absence of obstacles in it's path. So the perception should be accurate to make sure that the car is taking appropriate action in expected and unexpected hazardous situations. Using a single sensor will not be adequate as each sensor has it's own accuracy and degree of error. So this includes a variety of sensors placed in different arrangements to get accurate knowledge of the surroundings. Incomplete and inaccurate information can result in fatal scenarios.

Sensor fusion can be explained as the process of combining incomplete and inaccurate information together and coming up with a more detailed and accurate information; which gives better understanding of the surrounding of ego vehicle. It is unreasonable to assume there is a single sensor with accurate and detailed information.

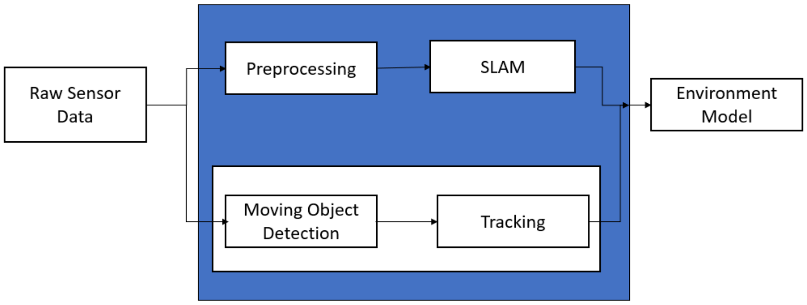
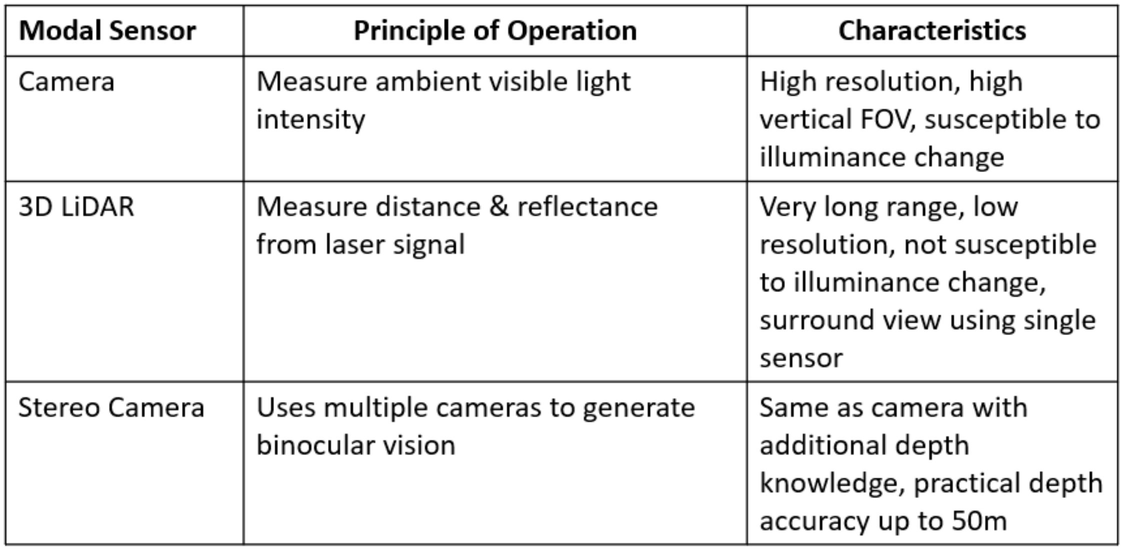


Fig. 2.5. General Blocks of the Perception Task



Modal Sensor	Principle of Operation	Characteristics
Camera	Measure ambient visible light intensity	High resolution, high vertical FOV, susceptible to illuminance change
3D LiDAR	Measure distance & reflectance from laser signal	Very long range, low resolution, not susceptible to illuminance change, surround view using single sensor
Stereo Camera	Uses multiple cameras to generate binocular vision	Same as camera with additional depth knowledge, practical depth accuracy up to 50m

Fig. 2.6. Comparison of Sensor Technologies [15]

There are many sources of information such as characteristic of a particular state, prior beliefs about possible states, or knowledge about certain constraints and relations, which can be used to obtain the required knowledge. In order to use the information from sensors effectively we should be able to describe the information accurately and how it can be used to present the actual information. Three elements - the state, the observation model and the decision rule, are the essential components of the data fusion problem. In data fusion problems the crucial step is to adopt a model with desired uncertainty associated with it. These models will have a state and observation variable to represent each targets. Probability method is the mostly used method for describing and manipulating the uncertainties. Estimation is an integral part of sensor fusion process. It gives an optimum estimate of the quantity. This is based on a decision principle by taking the measurements as input and coming up with an estimate. We receive a number of observations from multiple sensors and using this information we find estimate of the true state of the environment.

In automotive sensor fusion applications, there are different kind of classification available for sensor fusion. Depending on the level of abstraction there are two types of fusion methods. The first is low level fusion. Here the fusion is done on the raw data collected form the sensor before processing it. The second method is the high level fusion. This method fuse the data after prepossessing sensor data, after collecting features and information from raw data. The hybrid method make use of advantages of the above two methods. Other category of classification is based on how sensors are organised to perceive the surrounding. It can be complementary arrangement, competitive arrangement or collaborative arrangement. The inaccurate information by single sensor can be the result of sensor related issues it can be a hardware fault, calibration problem or accuracy of sensor. Other issues are occlusions, climate issues.

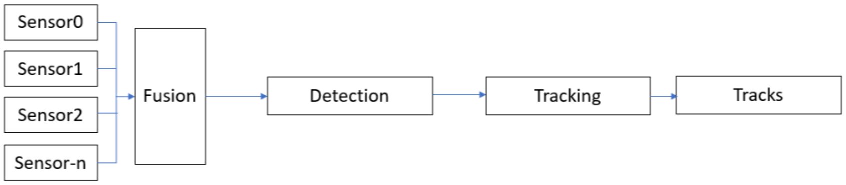


Fig. 2.7. Low Level Fusion

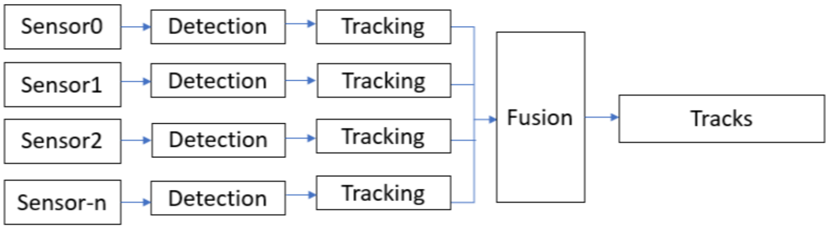


Fig. 2.8. High Level Fusion

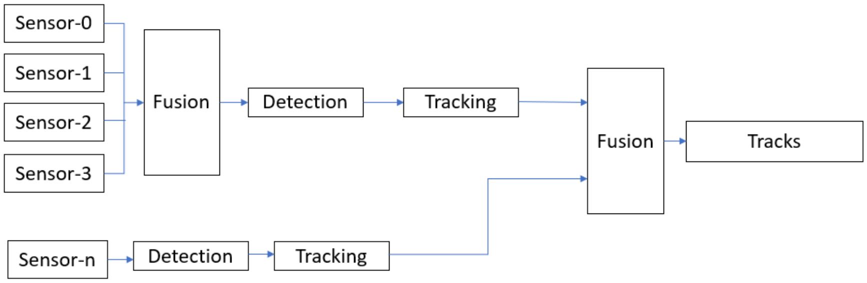


Fig. 2.9. Hybrid Level Fusion

3. CONCEPTS OF MULTI TARGET TRACKING

Object Detection and Tracking are the main methods of computer vision to infer information from sensors. When the Object Detection gives information about the presence of objects in a frame, Object Tracking goes beyond simple observation to more useful action of monitoring objects. An autonomous vehicle should have knowledge of position and other information about all the objects in it's surrounding to take appropriate action. By utilizing the information from a sequence of frames which measured with timestamp, autonomous vehicle can come up with an estimate of the positions of object in future timestamp. In forward collision avoidance, camera based object tracking helps to distinguish the potential collision threats in terms of their relevance to the decided path of the vehicle. The knowledge of moving objects around the vehicle enables a driver assistant system to alert a driver of potential collisions and dangers.

Target tracking has been studied for decades with numerous applications [1] [16]. Many methods has been introduced to solve the real time efficiency [1] and the occlusion problem [14]. There are offline tracking algorithms [22] which evaluate on past and future frames to generate efficient tracklets. But when comes to real time applications tracking should be online tracking methods [1, 2, 5]. With recent approaches in the detection domains including CNN based [24] and traditional approaches with feature vectors, the missed detections can be decreased, the precise bounding box can be reported. The advancement in the detection and higher frame rates simplifies the tracking method. The simple IOU tracker [11] introduced a method of data association without using visual information, thus reported a decrease in the computational complexity and processing time. The method proposed by Bochinski, Erik et al. integrated visual information too to handle longer occlusion with the increase of complexity in computation. Many ADAS techniques are using object detection

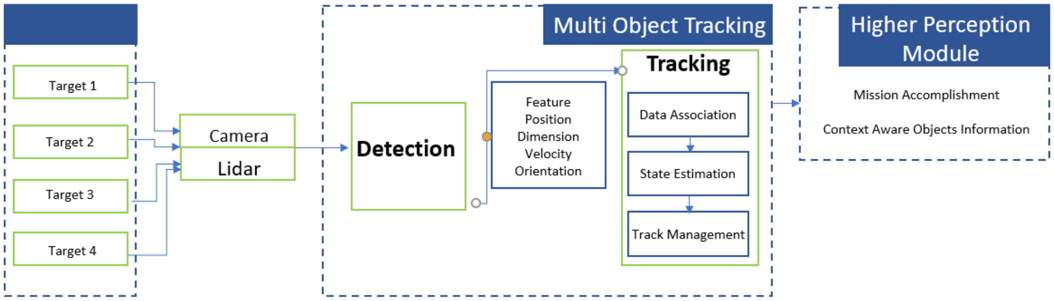


Fig. 3.1. Object Tracking Summary

and tracking as major part of the perceiving system. The knowledge of presence of obstacles and their position are important for an autonomous vehicle. The vehicle take control actions according to the presence, location and state of the object. The tracking has major role in finding the dynamic motion of these objects. The ability of tracker to track pedestrian motion will help the vehicle in appropriate breaking action.

The object tracking method widely uses tracking by detection method [2] [3]. This is the less complex and real-time method of tracking. Online tracking methodology help the tracker to update it's tracker accordingly, this makes the method less complex and popular. The first stage of tracking by detection is detection from the sensor data. On detection stage, the raw measurement is translated in to meaningful features, in other words objects are located through detection. After this the located objects are fed to the filter.

The state of the object in the surrounding are represented using random variable concept. As this variable possess different states with probability assigned to it. According to the probability value, the state of the system is given importance. These variables will have the data which gives the state of the system. Statistical theories are mostly used to design these kind of system, in that Bayes filtering [39] is highly

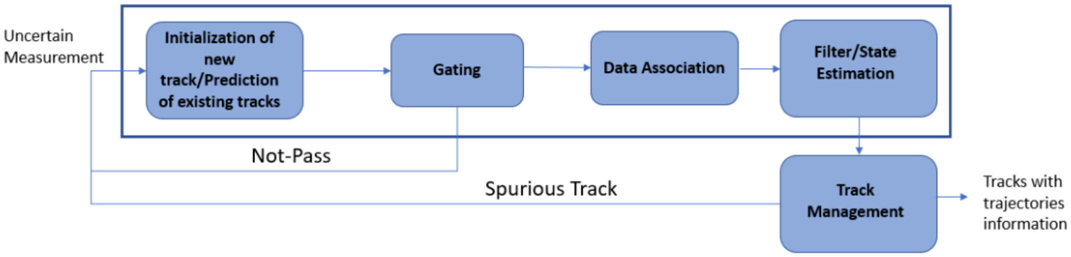


Fig. 3.2. Object Tracking Flow

used to find the trajectories of object's in motion. Here the state of the target is a random variable with adequate uncertainties added, each possible state is represented with a probability assigned with it in this scenario.

The feature based object tracking can be in to the steps as depicted in Figure 4.3. Raw data is collected by the sensors in the first step. The next step is filtering the data from measurements of the environment. The data association stage finds correspondence between measurement and tracks. Many methods exist for association of tracked objects with measurements. The update stage of the object's state estimate is the backbone of the tracking procedure. The object management module decides whether objects should be tracked When the measurements are missing. When new detections are triggered the management module decides whether it should be added as new tracker.

3.1 Bayes Theorem in Object Tracking

Measurement state of each objects are modelled as random variable in object tracking problems along with the uncertainties from environment. Bayes filtering approach is fundamental of complex methods used to solve object tracking problems. Multi object tracking calculates the number of object present in the frame and it also

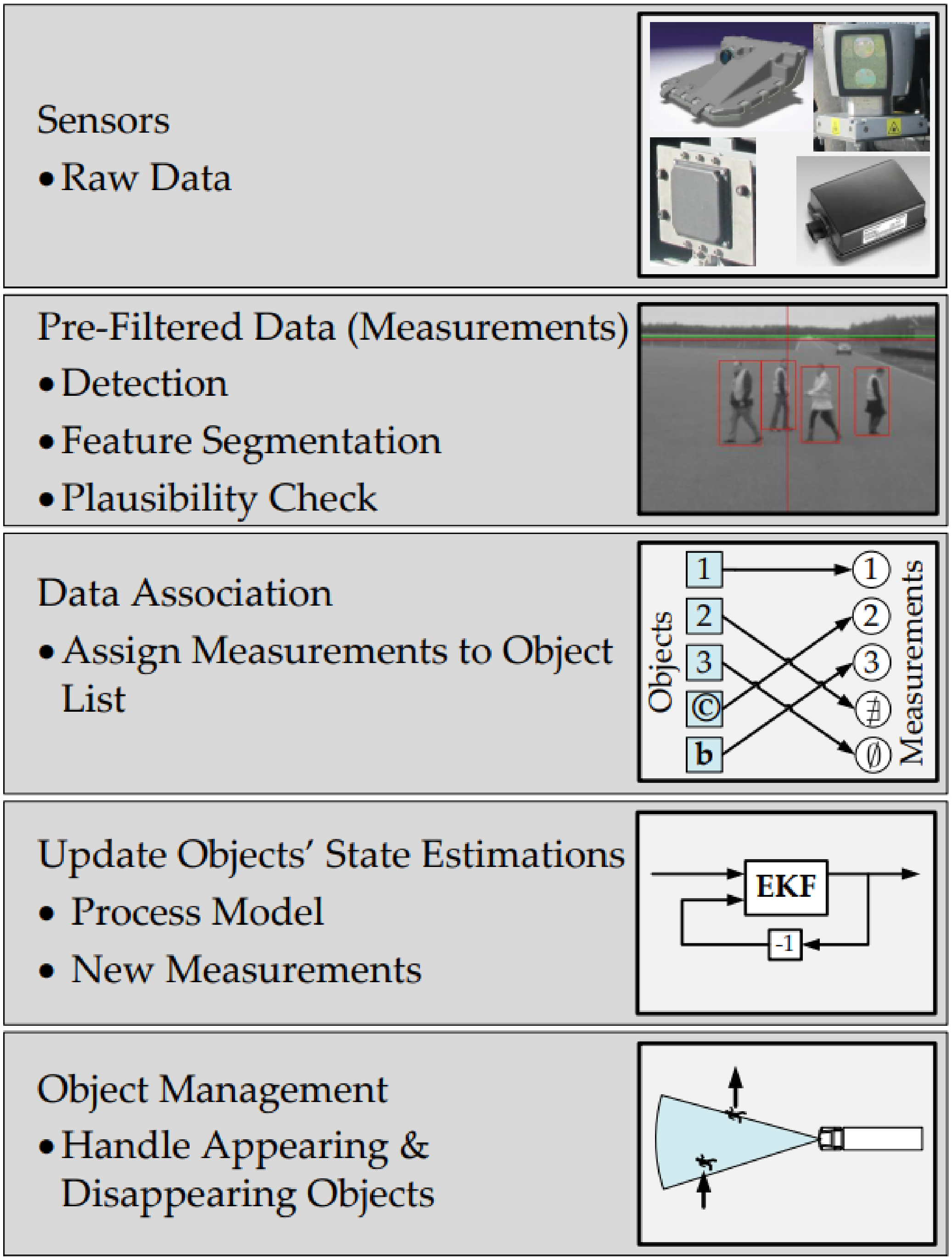


Fig. 3.3. Basic Steps of the Object Tracking Procedure [15]

calculate the objects states over time making use of measurement data z(k) (which is inaccurate, noisy or occluded) from various sensors and the estimate using process model allotted for dynamics of the object (Z|X), where X are all object states [18]. Applying the Bayesian rule with knowledge of all measurements Z(1:k) = (z(1), . . . , z(k))^{T} until time k, knowledge base at time k can be computed using:

$$p(x(k)|Z(1:k)) = \frac{p(Z(1:k)|x(k))p(x(k))}{p(Z(1:k))} \tag{3.1}$$

$$p(x(k)|Z(1:k)) = \frac{p(z(k)|Z(1:k),x(k))p(x(k)|Z(1:k-1))}{p(z(k)|Z(1:k-1))} \tag{3.2}$$

The probability distribution function in Equation 3.2 depends on all preceding measurements until time point k. Here the assumption is that current It is assumed that the current measurement depends only on the current state and not on the state history or measurement history. So the equation can be simplified as follows:

$$p(x(k)|Z(1:k)) = \frac{p(z(k)|x(k))p(x(k)|Z(1:k-1))}{\int p(z(k)|x(k))p(x(k)|Z(1:k-1))dx(k)} \tag{3.3}$$

3.1.1 Kalman Filter

Kalman filter algorithm works using 2 procedures. First is prediction and the second is update. Kalman filter will have a process model allotted to the targets in motion. In the first step which is prediction, using process model the filter calculates the current estimate of the state along with uncertainties. In the second step filter uses the measurement data from the sensors available and the estimate will be updated using this data in weighted average. Kalman filter is a recursive algorithm. It runs using the estimate from previous frame and the current measurement. Data

Association has a major role in the update step, as it finds the association between estimate and measurement. The below equation explain the algorithm:

$$X_{\text{n}} \ = FX_{\text{n-1}} + Bu_{\text{n}} + w_{\text{n}} \tag{3.4}$$

$$Z_{\text{n}} = HX_{\text{n}} + v_{\text{n}} \tag{3.5}$$

in this X $_{\text{n}}$ ∈ Rn, z_{n}∈ R_{m} and u_{n}∈R^{p} are the track state, observation, and control input vectors, respectively, at the time t_{n} ; $v_{\text{n}} \in R^{\text{q}}$ and $w_{\text{k}} \in R^{\text{m}}$ are the process and measurement noise sequences with co-variance matrices Q(t) ∈ $R^{\text{q} \times \text{q}}$ and R(t) ∈ $R^{\text{m}\times\text{n}}$ respectively; and F ∈$R^{\text{n}\times \text{n}}$, B ∈$R^{\text{n}\times\text{p}}$ and H_{n} ∈$R^{\text{m}\times\text{n}}$ are the discrete-time transition matrix, input gain, noise gain and measurement matrix, respectively.

The kalman filter solution can be found by following the below methods:
Predicted state:

$$\hat{X}(n|n-1) = FX(n-1|n-1) + Bu_{\text{n}} \tag{3.6}$$

where $\hat{\text{X}}$(n |n-1) predicts the state X(n) based on the previous estimate of X(n-1). This is the first of kalman filter with a process model assigned according to the motion or trajectory of the target. At the time of missing detection one can come up with the estimate of object location from this step.
Predicted measurement:

$$\hat{Z}(n|n-1) = H\hat{X}(n|n-1) \tag{3.7}$$

where $\hat{\text{Z}}$(n|n-1) represents the predicted measurement at n $^{\text{th}}$ iteration based on the sensor model and predicted state information.
Kalman gain:

$$K(n) = P(n+1|n)H^{\text{T}}S(n+1)^{\text{-1}} \tag{3.8}$$

where K(n) represents the optimal Kalman gain that decides the importance of the sensor data over the predicted state estimate using process model.
Updated state estimate:

$$\hat{X}(n|n) = \hat{X}(n|n) + K(n)\vartheta \text{ (n)} \tag{3.9}$$

Updated co-variance:

$$P(n|n) = P(n|n-1) - K(n)S(n)K^{\mathrm{T}}(n) \tag{3.10}$$

After update with the measurement data the state will be the optimum possible result with minimum error. This will be again a possible state with co-variance associated with it.

3.2 Data Association Methods

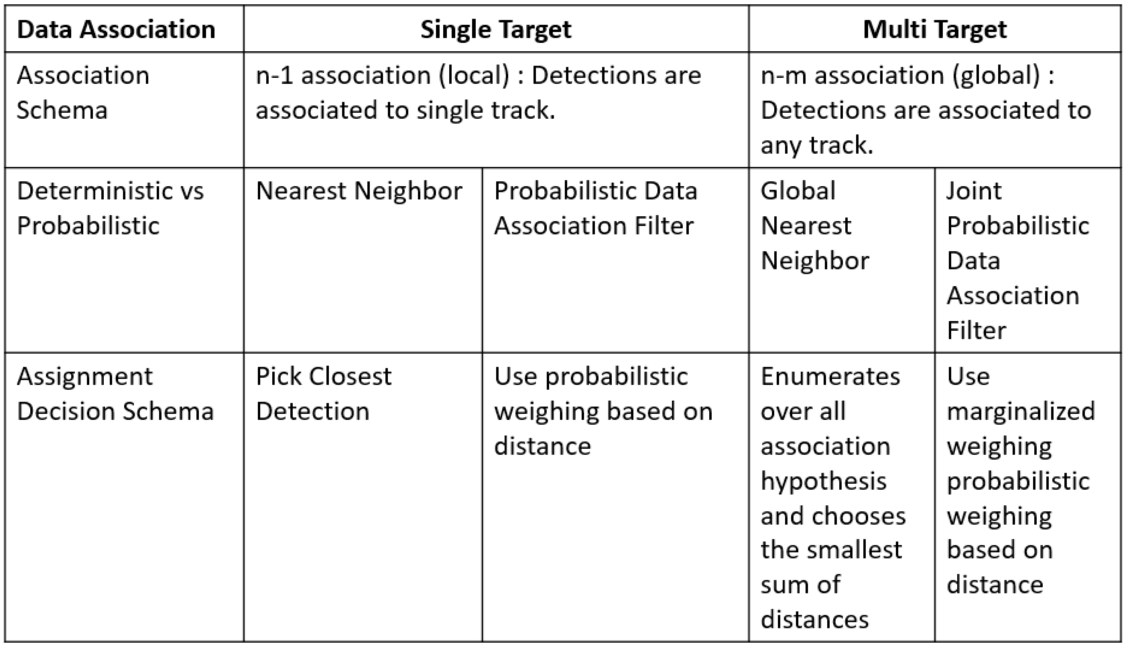
One of the most significant step of Multi Object Tracking is Data Association. This is mandatory when the vehicles are continuously going in and out of the sensor coverage. Data association can be seen as a problem which find the correspondence between measurement and existing tracks. A perfect data-association method is required to keep track of the continuously changing objects.

The Data Association should satisfy the following to be a good algorithm:

1. It should be able to find the appropriate track-measurement pairs in a clutter environment and other unexpected conditions.

2. It should be able maintain a set of tracked trajectories (vehicles going in and out of the sensor coverage).

The scenarios which need data association in multi object tracking are Clutter Scenario and Dynamic Dataset Scenario. Great care has to be taken when dealing with objects with overlapping measurement data. The bad association will result in loss of efficiency, this can result in ID switches.



Data Association	Single Target		Multi Target	
Association Schema	n-1 association (local) : Detections are associated to single track.		n-m association (global) : Detections are associated to any track.	
Deterministic vs Probabilistic	Nearest Neighbor	Probabilistic Data Association Filter	Global Nearest Neighbor	Joint Probabilistic Data Association Filter
Assignment Decision Schema	Pick Closest Detection	Use probabilistic weighing based on distance	Enumerates over all association hypothesis and chooses the smallest sum of distances	Use marginalized weighing probabilistic weighing based on distance

Fig. 3.4. Summary of Different Data Association Methods

4. OBJECT DETECTION

The process of finding the location of an object and classifying it in to different categories, is called object detection. Human eye can very easily locate and classify the objects present in the images. The visual system is very complex and it can perform various complex task fast and accurate. The pipeline of object detection model has different stages: region selection, feature extraction and classification.

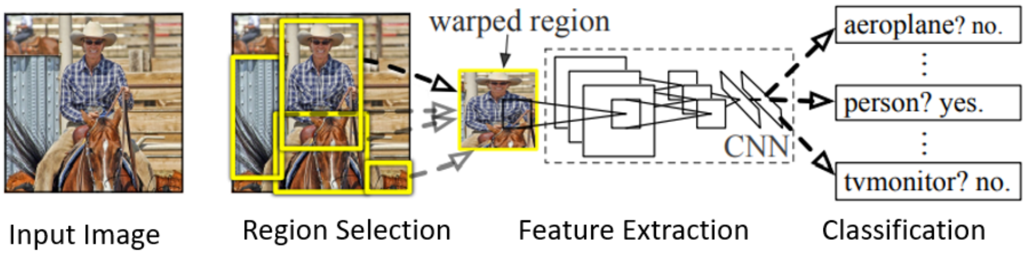


Fig. 4.1. Object Detection Pipeline [12]

The first and foremost step of Tracking by Detection methods are detecting the objects in each frame. The quality of detected object region is a necessary component of MOT system. The below sections briefly explains the Object Detection adopted by for the data from camera and LiDAR. For the better accuracy and fastness of the MOT system detection is done using state-of-the-art detectors from official KITTI [4] leader-board.

4.1 3D Detection on LiDAR Point Cloud

Our Multi Object tracking system uses the data from the LiDAR data. The data needs to go through the detection system before tracking the targets. This helps the MOT to detect and delete unnecessary targets from tracking. Deep Learning has achieved high progress in recent years for the detection of objects in 2D and other segmentation applications. 3D object detection is crucial in many real world applications. Recently proposed 2D object detections has the capability to understand the varying pose and background clutters. 3D detection still faces many challenges. One of the main sensors in autonomous vehicles for 3D data is LiDAR. This gives 3D point clouds to capture the 3D structures of the scenes. PointRCNN [24] is used for 3D object detection from point cloud data. PointRCNN [24] is a precise and powerful method for 3D detection. This method work on 3D Point clouds for detecting the objects. The structure of this method contains two steps. The initial step generates 3 Dimensional box in a bottom-up manner. Using this 3 Dimensional bounding boxes the initial step segments the foreground points. Then it produce a number of bounding box proposals from the segmented points. The next step of this method does canonical 3D box refinement.

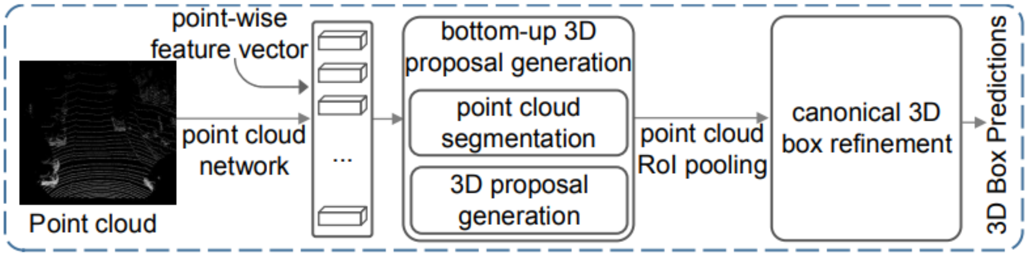


Fig. 4.2. 3D Proposals from LiDAR Raw Data [24]

PointRCNN [24] outperforms other methods with remarkable performance and it has

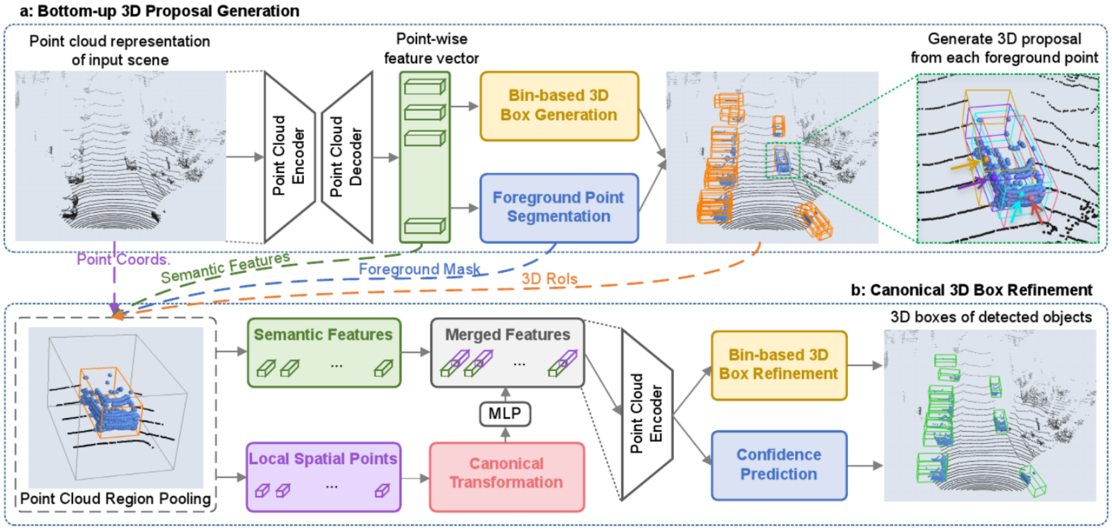

a: Bottom-up 3D Proposal Generation
Point cloud representation of input scene
Point Cloud Encoder
Point Cloud Decoder
Point-wise feature vector
Bin-based 3D Box Generation
Foreground Point Segmentation
Generate 3D proposal from each foreground point
Point Coords.
Semantic Features
Foreground Mask
3D RoIs
b: Canonical 3D Box Refinement
Point Cloud Region Pooling
Semantic Features
Merged Features
MLP
Local Spatial Points
Canonical Transformation
Point Cloud Encoder
Bin-based 3D Box Refinement
Confidence Prediction
3D boxes of detected objects

Fig. 4.3. The method of Object Detection from LiDAR 3D Data [24]

high rank on the KITTI [4] dataset. Choosing a high accuracy detection will result in higher accuracy of tracker while following the Tracking by Detection method. This method achieves better performance by using only LiDAR point cloud data compared to the other methods which used LiDAR and RGB image. This method showed poor performance compared to the methods which used multiple sensors on pedestrian data. This due to the fact that dimension of the pedestrians is small. An image can capture high characteristics than an image also be a reason for poor performance on pedestrian.

4.2 2D Detection on RGB Images

Faster R-CNN consists of two networks, region proposal network and a convolution neural network. Region proposal network is responsible for proposing the regions or the bounding box whereas the CNN is responsible for the classification. The faster R-CNN architecture makes use of feature maps to generate region proposals. As this

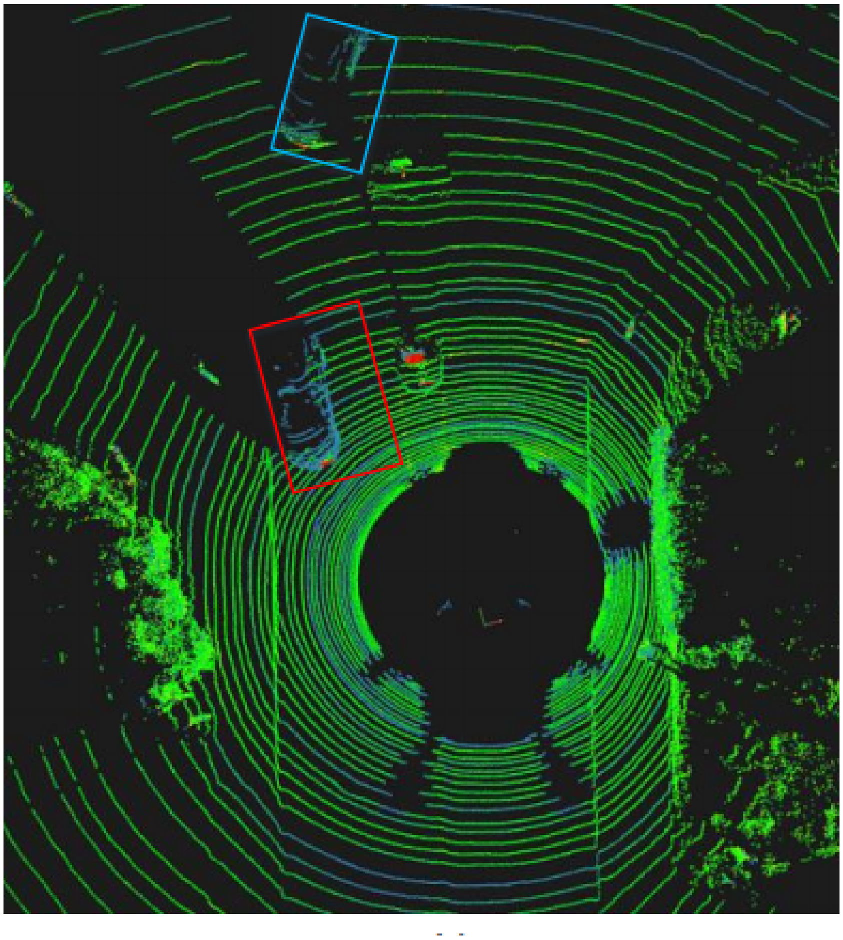


Fig. 4.4. Sample Detections on Point Cloud

approach reuses the feature map generated through CNN and does not use a brute force technique such as the sliding window the computation cost is low The further section discusses the components of faster R-CNN.

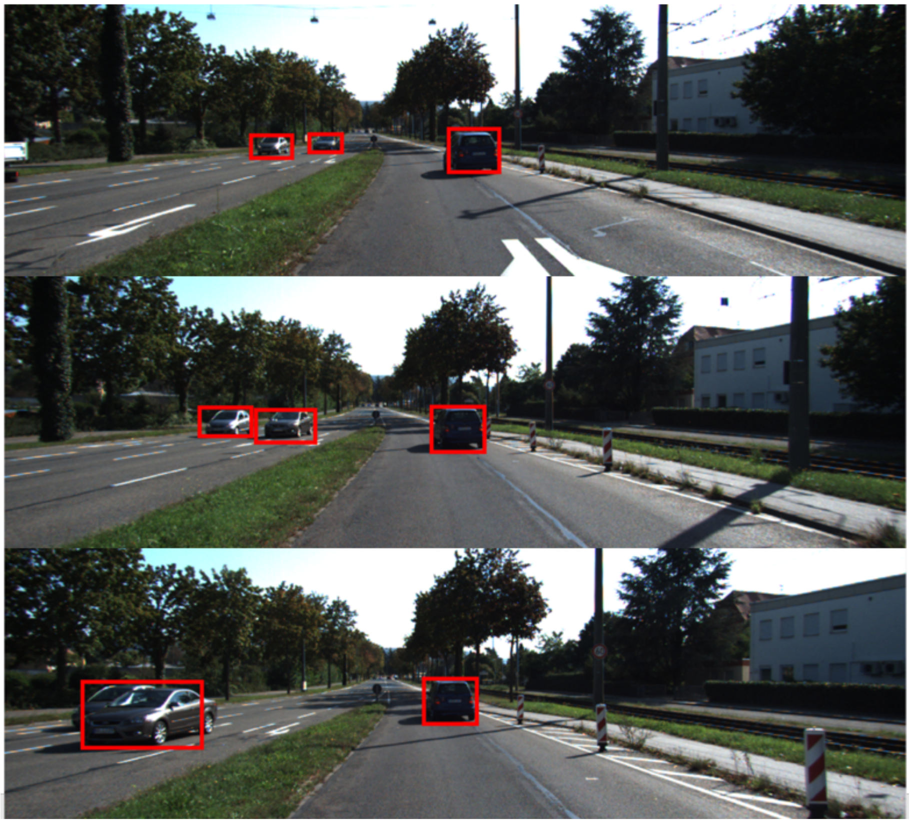


Fig. 4.5. Sample Detections on RGB Images

Anchors:

The anchor is the most crucial components in faster R-CNN [17]. Anchors are boxes in faster R-CNN configuration with different aspect ratios of 1:1, 1:2, 2:1. It is a very direct approach to train a neural network with four output X_i, Y_i, H_i, W_i to detect the bounding box of the object but this approach fails when there are multiple objects in the image. This issue can be resolved by running the anchors at each spot on feature maps generated by CNN [17]. The object detection accuracy is measured using Intersection Over Union (IOU), so at the time IOU of a specific anchor and ground truth label have a large intersection, the regression associated with that anchor provides fine-tuned bounding box.

Region Proposal network:

The features learned from the convolution neural network will be passed through the Region Proposal Network. The RPN has 2 objectives; it is responsible for providing class scores, which is whether the object present or not present for each anchor. Secondly, it is responsible for predicting the bounding box coordinates for the object present in the image for each anchor, e.g., if the input feature map is 100×100. RPN generates $2 \times 9 \times 100 \times 100$ class scores for whether the object is present or not and also generate $4 \times 9 \times 100 \times 100$ coordinates for bounding boxes X_i, Y_i, H_i, W_i.

$$L_{loc}(t^u, v) = \sum Smooth_{L_i}(t^u - v_i) \tag{4.1}$$

where,

$$Smooth_{L_i}(x) = \{ \;0.5x^2 \quad if \mid x \mid < 1 \mid x \mid -0.5 \;\; otherwise, \tag{4.2}$$

The above formula is the loss function for regressors [17].

The Classifier:

The classifier is an important part of the Faster R-CNN network,and the CNN is responsible for extracting features from the image and classifying the image. This part of faster R-CNN can be modified, and various other CNN architectures can be used to extract features for RPN. This network of CNN and RPN can be trained jointly to classify images and predict bounding box. Object detection pipeline uses region of interest pooling layer. It is used to find the region of interest. It scales the image to a defied size.

5. MOT SYSTEM DESIGN

This section gives the explanation of Multi Object Tracking implementation on a publicly available camera and LiDAR data. The camera based MOT system uses 2D detection results from faster R-CNN and LiDAR MOT system uses 3D bounding box detection results from the PointRCNN [24]. Generally, there are two types of MOT systems. The first is offline tracking, which uses tracker results from current and future frames. Tracklets are generated by linking the detections from the frames and associating iteratively to construct the trajectory of objects in the entire sequence. But Online MOT algorithms [1] [2] estimate the trajectory using the detections from past and current frames, are more applicable to real time applications such as ADAS, FCW and Navigation.

5.1 2D Tracking on Camera Data

The tracking can be viewed as combination of as combination of Object Detection, Propagating the detection using Motion Model, Data Association and Managing the Tracklets. The 2D object detector uses KITTI right camera images as the input and output the best t bounding box of the detected objects. The major online tracking methodology is tracking by detection. The major input of this tracking methodology is detection results from the sensor data. The efficiency of this tracker highly depend on the efficiency of detection method used. As discussed in the Section 5.2 the faster R-CNN [17] method gives the detections on the image input. The input from the detector is 2D bounding box data in KITTI tracking data format which is extreme left, top, right and bottom coordinates of the detected objects. This coordinates are used to find the center pixel, width and height of the bounding box to feed to the state estimation algorithm.

5.1.1 State Estimation

The Estimation Model represents the target motion from frame to frame. Knowing the likely position of target in the future frame reduces the search area, hence increases the accuracy of association. The popular motion models are categorized in to linear and non-linear motion models. The linear motion model follows a linear movement with constant velocity or constant turn rate. The non-linear model can represent a non linear model accurately than linear one. A standard Kalman filter is an optimum estimator when the state transitions are linear. This works under the assumptions that the noise is Gaussian. If the model is not linear and the noise is Gaussian, Extended Kalman Filter yields good results [40].

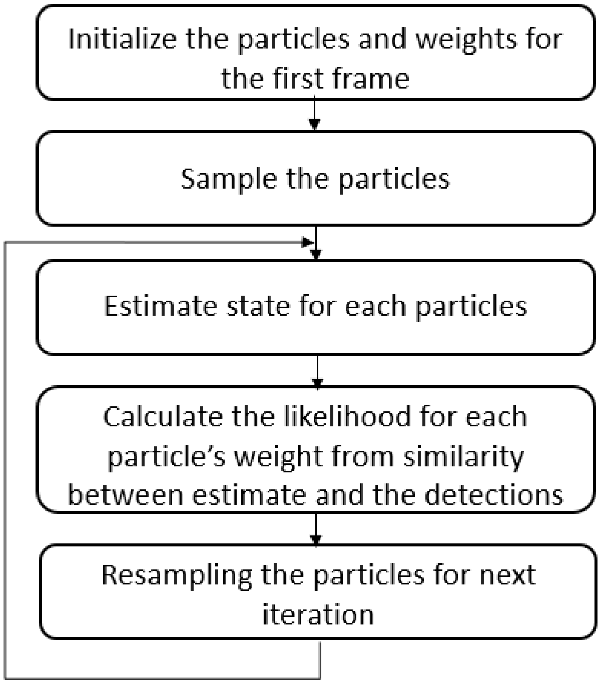

Initialize the particles and weights for the first frame
Sample the particles
Estimate state for each particles
Calculate the likelihood for each particle's weight from similarity between estimate and the detections
Resampling the particles for next iteration

Fig. 5.1. Estimation Model [1]

The proposed system uses Particle Filter, which is a sampling based recursive Bayesian algorithm with each of the particle is selected to represent a possible state. The filter starts with an assumption of a uniform distribution of the particles. The particle Filter which is also known as bootstrap lter or survival of the ttest, represent the posterior density function using a set of random samples with associated weights and compute the estimate with these samples and the weights. It predicts and corrects the future states and optimal possible state is found with smallest possible variance error. As the number of samples increase, this becomes an equivalent representation of posterior probability function and the estimate approaches the optimum value. Prediction of lter model provides a reliable region which helps decreasing the missed rate and reduced uncertainty of measured noise. The state of each particle is modelled as [x, y, w, h, $\mathrm{x_{vel}}$, $\mathrm{y_{vel}}$, $\mathrm{w_{vel}}$, $\mathrm{h_{vel}}$], where x and y represent the horizontal and vertical pixel location of the center of the target, while the scale w and h represent the width and height of the targets bounding box respectively.

5.1.2 Data Association

The core of Multi Target Tracking is data association and track to track association methods. The goal is to identify a correspondence between the sensor measurements/detections and the pre-existing tracks. Erroneous assignment of newly detected objects to the existing tracks will result in significant drop of accuracy. Generally, multi-scan methods are preferable in situations where there are a lot of false alarms and missed detections. However delaying the association to include future information will negatively affect the real-time capabilities.

$$IOU(a,b) = \frac{Area(a) \cap Area(b)}{Area(a) \cup Area(b)}$$

When using sufficiently high frame rates, detections of an object in consecutive frames have high overlap IOU (intersection-over-union) [1]. If the above requirements are met, tracking becomes simpler and can be implemented even without using image

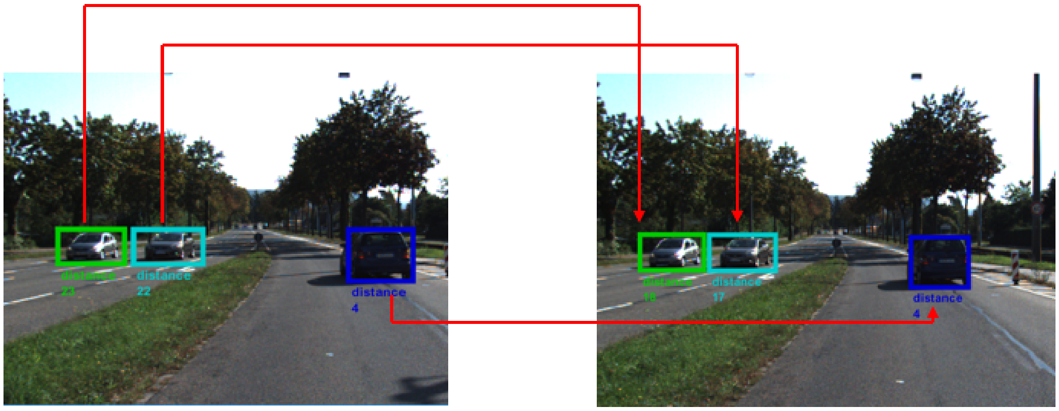


Fig. 5.2. Data Association Between Multiple Frames. Unique ID has given for Unique Tracker.

information. We use a simple IOU tracker which essentially continues a track by associating the detection with the highest IOU to the tracklets predicted from the previous frame, if a certain threshold IOU is met. The overall complexity of this method is very low compared to other trackers. As no visual information used it will result in fast ltering procedure. As shown in the Figure 6.2 unique tracks has given unique ID which is mapped to a unique color for the bounding box. This way the ego vehicle can have the dynamic nature of the moving object.

5.1.3 Tracklet Management

When new objects enter the frame and leave, unique identities need to be created and destroyed accordingly. Any detection with overlap less than σ_{IOU} (The minimum criteria for the association) is considered as an not tracked object and created a new identity for it. The tracker is initialized with the bounding box parameters such as [x, y, w, h] with velocity set to zero. Tracks which are not detected for N_{lost} frames are terminated. This will prevent the growth in number of tracks and accumulating error from the prediction with out having any detection to correct. The N_{lost} is set to 3 frames for object re-identication if available and did not make it a high value as this will increase the total tracks, thus computation of the tracks which might have left the frame already.

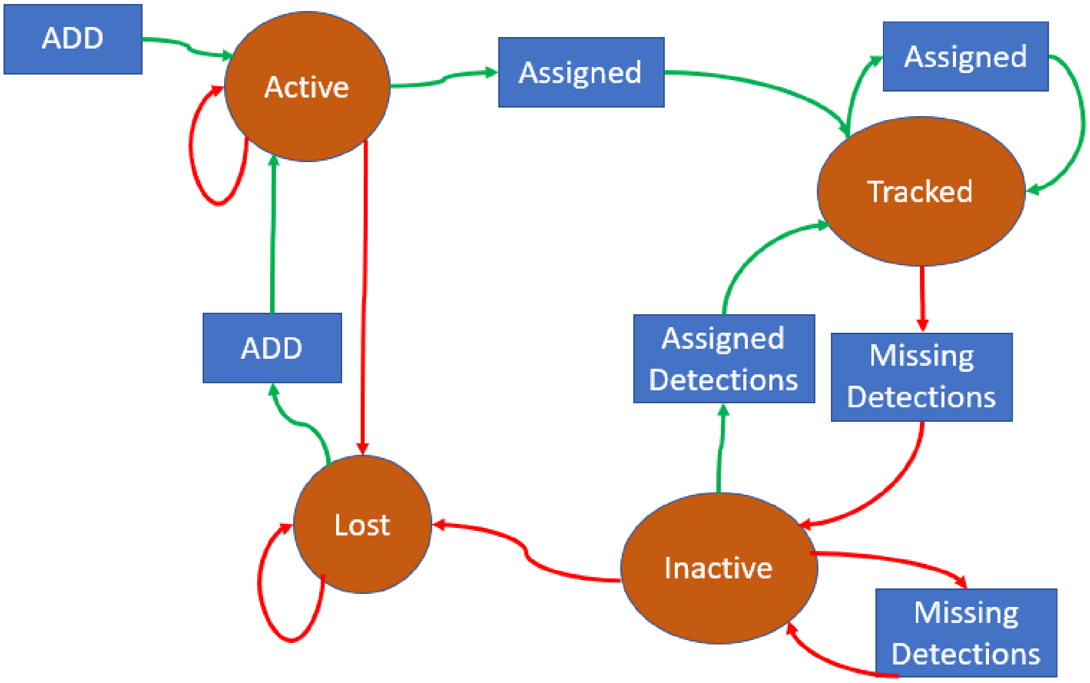


Fig. 5.3. Track Management for Online Tracking

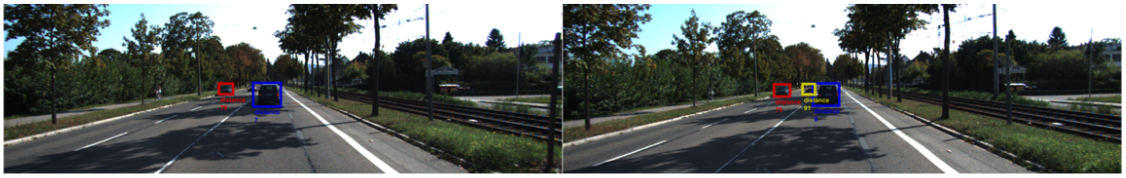


Fig. 5.4. Sample Frames for New Track Creation

5.2 3D Tracking with Point Cloud Data and Image Data

We use 3D detector described in Section 4.1 to obtain bounding boxes of the objects present in LiDAR point cloud data. Then 3D Kalman filter has used for the state estimation of each tracks for the next time frame. Later Hungarian algorithm [11] has adopted for data association. All the modules in the process are standard and we made sure to use this way to make the model to run as real time. As in the previous camera based MOT method, the 3D tracking explained here is not in 2D image space. This method does the tracking in complete 3D domain. The standard MOT benchmark from KITTI supports only 2D MOT evaluation. The convention to evaluate the 3D MOT is to project the 3D MOT tracker outputs to image plane. The 3D MOT system implemented has the same steps as in previous 2D MOT. The difference is in the domain tracking is implemented. As the method tracks in 3D space, the design includes 3D appearance and motion in 3D plane. As our system is an online method at each stage of estimation and association it uses the state parameters from the current time and the tracker information form the previous frame. The 3D MOT is then associated with the previous 2D MOT after converting the 2D MOT to 3D in point cloud plane. The architecture utilizes both a deep network and geometric optimization method. ResNet-34 CNN extracts a feature vector from the image. These are then fed to a fully-connected network with three layers to output the 3D box As discussed in Section 4, the advancement in the detection methods contribute highly to Tracking by Detection method. The object detection used here gives a set of 3D bounding box at each time frame. Each detection can be represented as (x, y, z, l, w, h, , s).

5.2.1 State Estimation

The 3D MOT uses standard Kalman FIlter for the state estimation. The system has modelled using 9 parameters T = (x, y, z, l, w, h, v_x, v_y, v_z). A complete mathematical explanation of the Kalman Filter is given Section 3.1.1. The state of

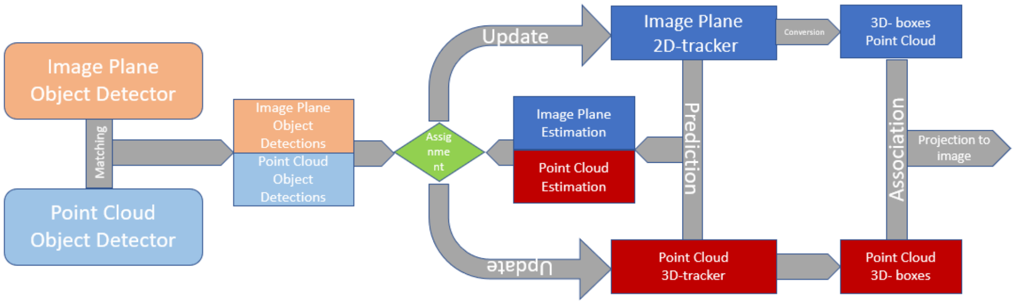
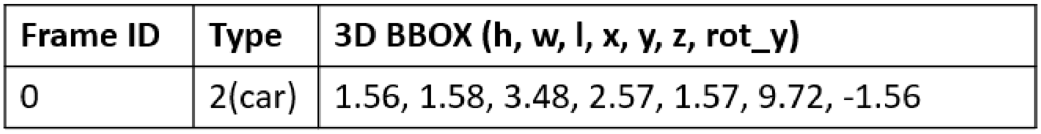


Fig. 5.5. LiDAR and Camera Based 3D MOT System Overview



Frame ID	Type	3D BBOX (h, w, l, x, y, z, rot_y)
0	2(car)	1.56, 1.58, 3.48, 2.57, 1.57, 9.72, -1.56

Fig. 5.6. Format of 3D Detection Result from LiDAR Point Cloud

the system is propagated through time using constant velocity model. The parameter in state representation are as follows:

x - X coordinate of the center 3D of bounding box

y - Y coordinate of the center 3D of bounding box

z - Z coordinate of the center 3D of bounding box

l - Length of 3D box

w - Width of 3D box

h - Height of 3D box

v_x - Velocity along X coordinate

v_y - Velocity along Y coordinate

v_z- Velocity along Z coordinate

5.2.2 Data Association

As in the case 2D Multi Object Tracking, Data Association plays a crucial role in 3D Multi Object Tracking methods also. The data from the detections has to be matched with the predicted trackers for the updation of prediction with sensor measurements. We use Hungarian method [11] for data association in our 3D MOT on LiDAR data. The affinity matrix is calculated for an $n \times m$dimension, where n and m are the count of detections and predicted trackers respectively. This computation is done by finding out the 3D IOU between each pair of detection and tracker. Then, the bipartite graph matching problem can be solved using hungarian algorithm in polynomial time. We have set an IOU $_{\text{min}}$ to reject the matching when the IOU is less than this value.

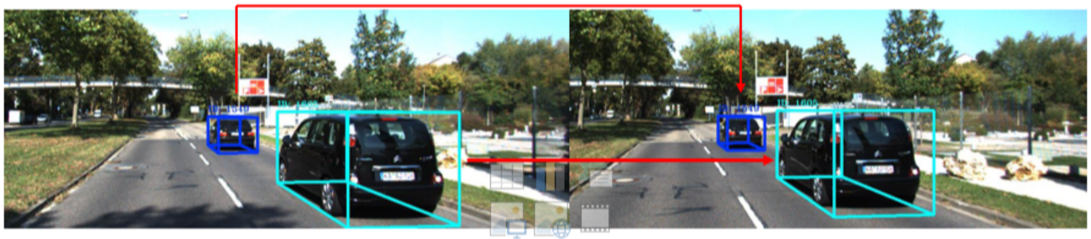


Fig. 5.7. Sample Results of Data Association Between Frames

5.2.3 Tracklet Management

This stage take care of the memory used while implementing an online Tracking methodology. When an existing object leaves the frame and another one enters, the tracklet management should be able to remove the first and add the second to the memory. We consider all the unmatched detections are as new object coming in to the sensor field of view. But to reduce the tracking of false positives, new trajectories will not be created until it has been continually detected in the next F_{min} frames. When the new object has detected more than F_{min}, it would be added to the trackers with a new unique ID and the state of the tracker is initialized with the position coordinates from the detection and velocities are set to zero. We also consider unmatched trajectories as objects leaving the sensor FOV. To avoid deleting true positives which is result of missing detection in certain frames, we keep tracking these object for a Age $_{\text{max}}$ before deleting it from the memory. This can reduce the multi object tracking accuracy.

6. MOT METRICS AND DATASET

6.1 MOT Performance Metrics

The evaluation of performance of multi object is done using available evaluated MOT benchmark metrics proposed by Milan et. al [5]. The evaluation metrics defined and utilised are described below.

6.1.1 Multi Object Tracking Accuracy (MOTA)

Multi Object Tracking Accuracy accounts for the FP (False Positve), FN (False Negative) and Identity Switches (IDSW) to measure the total performance of the implemented tracking methodology.

$$MOTA = 1 - \frac{\Sigma_{\mathrm{t}}(FN_{\mathrm{t}} + FP_{\mathrm{t}} + IDSW_{\mathrm{t}})}{\Sigma_{\mathrm{t}} GT_{\mathrm{t}}} \tag{6.1}$$

where t is time stamp and Ground Truth is represented using GT.

6.1.2 Multi Object Tracking Precision (MOTP)

Multi Object Tracking Precision calculates the preciseness of the location estimated by the tracked and the ground truth. This calculates the overlap region between the estimation and Ground Truth.

$$MOTP = \frac{\Sigma_{\mathrm{t,g}} d_{\mathrm{t,g}}}{\Sigma_{\mathrm{g}} d} \tag{6.2}$$

$\mathrm{d}_{\mathrm{t,g}}$ is the overlap quantity between tracked object with it's Ground Truth. The overlap is calculated using intersection over union.

6.1.3 Track Quality Issue

This calculates in total how many trajectories are precisely tracked and how many are lost from the tracking. A method is adopted by diving the track comparing with ground truth trajectories. Mostly Tracked (MT) correspond to at least 80% coverage, and Mostly Lost (ML) are the track which is tracked for less than 20%.

6.2 KITTI Dataset

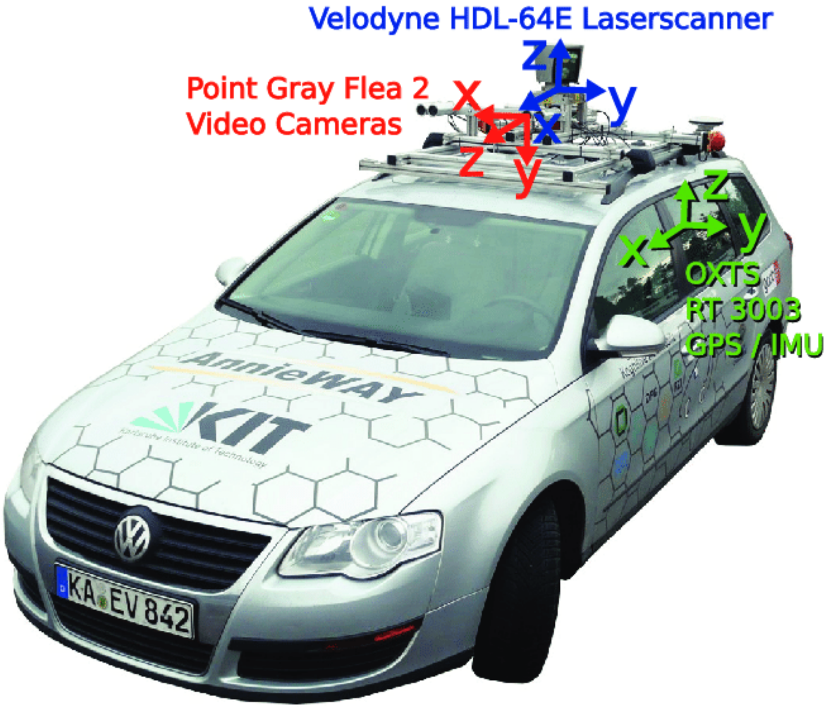


Fig. 6.1. KITTI Data Recording Platform Setup [4]

After implementing an algorithm it is crucial to test the algorithm in real time data. For an autonomous vehicle environment testing in synthetic data will not give an accurate results. The verification in this case is done using on-synthetic data. The dataset - KITTI [4] which is a publicly available dataset, provides real world recording using various sensors on the car drove in Karlsruhe, Germany. The KITTI dataset contains 21 training and 29 testing video sequences. For each sequence it has LiDAR point cloud, RGB images and calibration matrices of each sensor. The total number of frames available is 8008 and 11095 for training and testing. KITTI gives a 2D Multi Object Tracking Evaluation, in this the evaluation is done using 2D boxes projected in the image plane. If the tracker is a 3D tracker the evaluation is done by projecting the points on the image plane.

7. RESULTS AND DISCUSSIONS

7.1 Camera 2D MOT

This method tracks the detected 'Car' in the surroundings of ego vehicle in image plane. The results are in 2D coordinates in the image plane. This method uses constant velocity model to propagate the tracker. Most of the complex Multi Object Tracking methods achieve high efficiency at the cost of run time performance. But for an autonomous vehicle the real time processing is critical. The method of implementation here is taken care this. The system proposed is considered this in every stage of its implementation and reduced the processing complexity by removing appearance features from tracker algorithm. The proposed model used one camera sensor for tracking. This method put forward a low cost method for tracking by using a single sensor data. The proposed tracker method can run at 250 Hz (frames per second) on Intel i7 2.5GHz machine.

7.1.1 Results

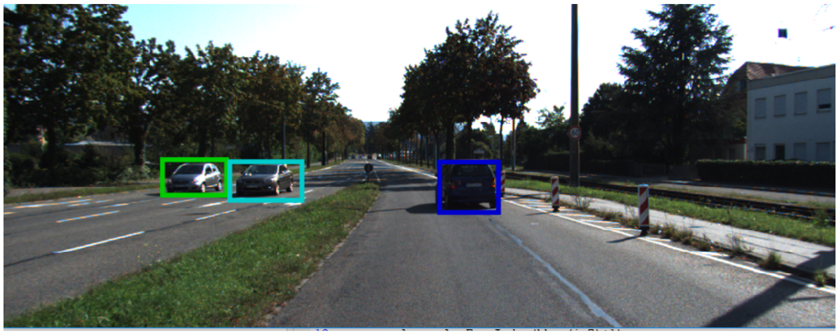


Fig. 7.1. The Tracklets with Unique Identity has given Unique Color.

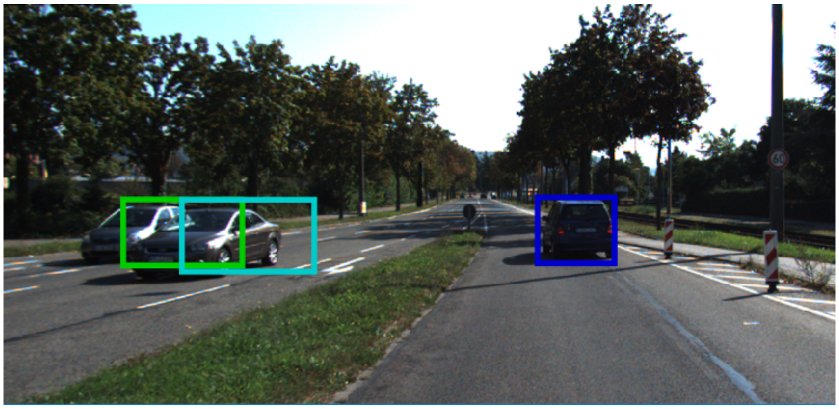


Fig. 7.2. The Tracklet with Occlusion in Camera due its Orientation Against Ego Vehicle has been Tracked

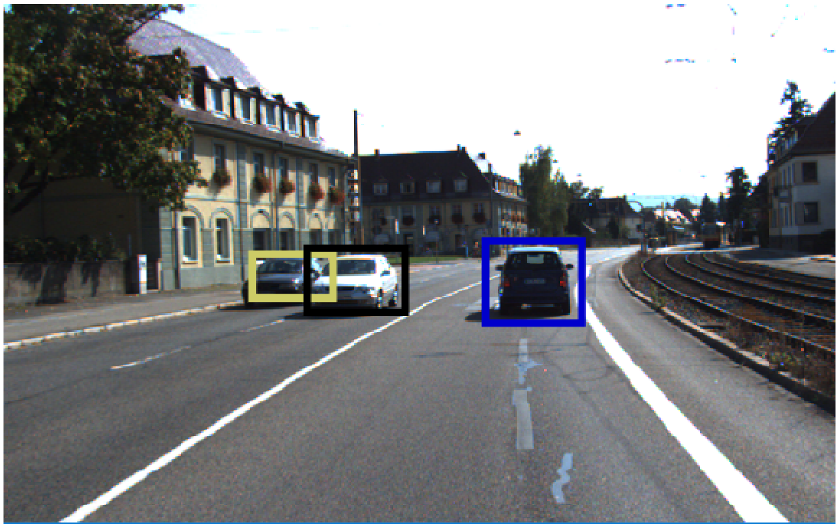


Fig. 7.3. The Tracklet with Missing Detection has been Tracked

Figures 7.2 and 7.3 gives sample scenarios where tracker could go beyond detection for detecting the presence of objects in the frames. The below table provides quantitative results of the MOT system. The evaluation is based on CLEARMOT metrics. The definition of CLEARMOT metrics are given in Section 6.1. The comparatively low

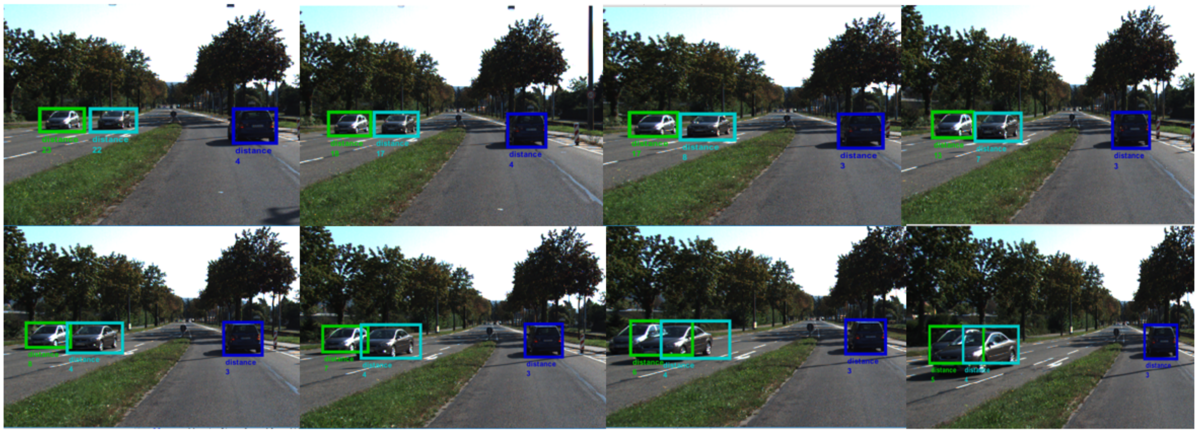


Fig. 7.4. Results of 2D MOT on KITTI Video Sequence-0056 of City Category

Table 7.1.
Results for 2D MOT on Camera Data

Tracker Name	MOTA (%)	MOTP(%)	MT(%)	ML(%)
Proposed 2D Tracker	32.2	71.2	10.8	30.8

accuracy and precision accounts for the false positive tracks, false negative tracks and the association issues. The model has implemented using camera data, which will have a degree of error. The result can be modified using multi sensor technology. The multi sensor technique gives the system capability to compensate the error of one sensor with the other. As a result high accuracy and precision can be achieved.

7.2 3D MOT with Multi Sensor Data

This method uses LiDAR data detections and the tracked trajectories from camera data projected to 3D coordinates. The fusion of two sensor help to reduce the measurement uncertainties of each sensor and gives an accurate detections than when use a single sensor. In addition to this fusion of multiple trackers can produce a better output in accuracy and dependability. The 3D MOT system implemented is tested on data available KITTI dataset and evaluated on CLEARMOT metrics. As the tracking system does not need any training, we used training sequences for evaluation.

7.2.1 Results

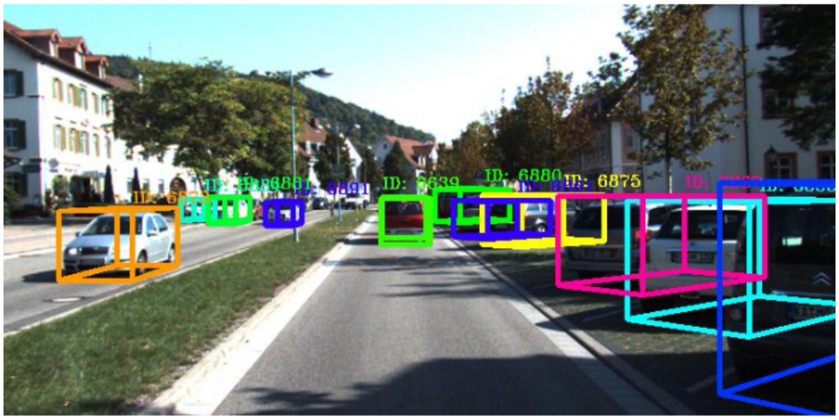


Fig. 7.5. 3D MOT in Cluttered City Traffic

7.2.2 Comparison with the Benchmarks

The quantitative comparison of our system and previous state-of-the-art shows that our MOT system with LiDAR and Camera perform good in all MOT evaluation metrics. Moreover our proposed system runs at a rate of 180 frames per second on Intel i7 2.7GHz machine. The MOTA scores reflect that the tracker has high degree of accuracy with 84.62%. The MOTA is lowered chiefly by the false positives which are tracked even the object leaves sensor FOV. The number of FP and IDSW are comparatively low. The MOTP score is also shows improvement compared to the state of art system on KITTI benchmark board. Despite perfect tracking, this

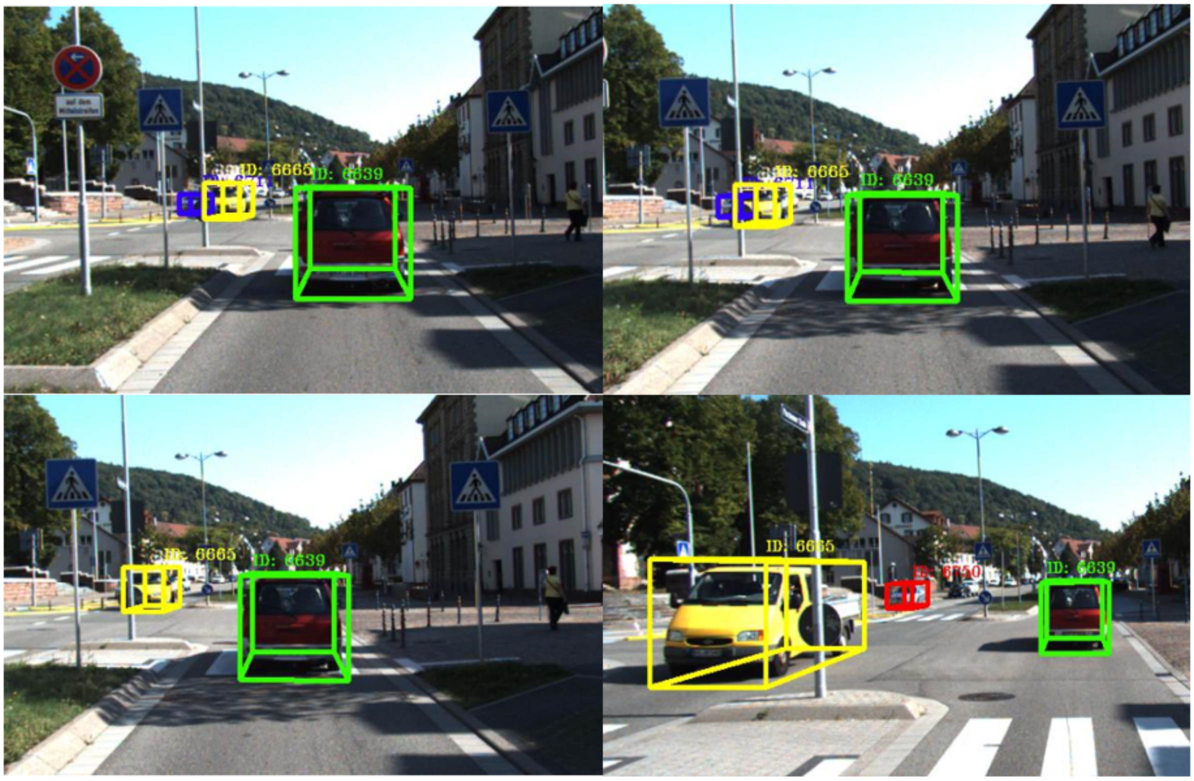


Fig. 7.6. Scenario of ID Switching Illustrated

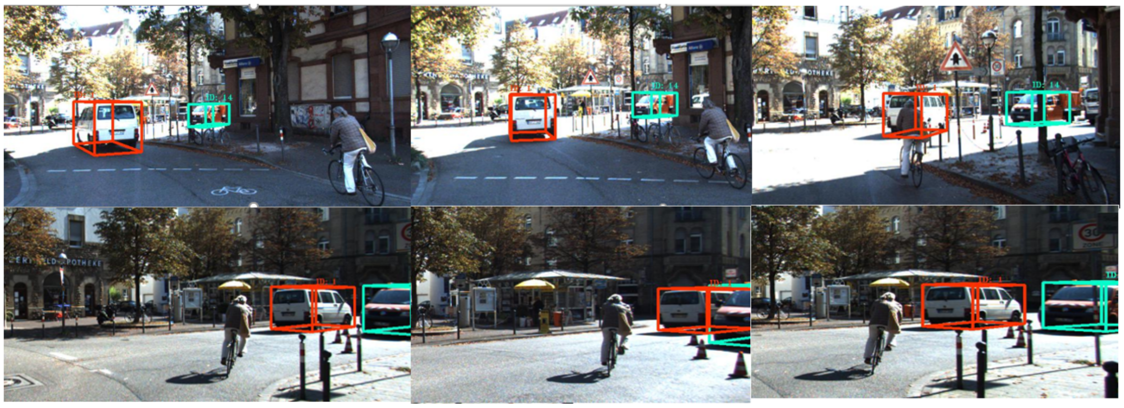


Fig. 7.7. Results of 3D MOT on KITTI City Category

Table 7.2.
Results for 3D MOT on LiDAR + Camera Data

Method	Type	MOTA(%)	MOTP(%)	MT(%)	ML(%)
Proposed 3D MOT	3D	84.62	85.81	74.1	2.48
BeyondPixels	2D	84.24	85.73	73.23	2.77
3D-CNN/PMBM	2.5D	80.39	81.26	62.77	6.15
extraCK	2D	79.99	82.46	62.15	5.54
DSM	3D	76.15	83.42	60.00	8.31

account for the partial overlap of tracker with the GT. Partial information is available when object enters frame from a long distance. This causes the MOTP to 85.81%.

8. SUMMARY

The motivation behind this thesis has been the development 3D Multi Object Tracking using the sensor data from camera and LiDAR. During this thesis the tracking has done in camera images first with 2D bounding box detection and then assigning a dynamic model to it for the state estimation. The predicted object locations are updated with the measurement at each time frame. The estimation model used is particle filter which is based on Bayesian theorem. The data association has done by finding affinity score from the intersection over union. The object tracklets is then converted to 3D box having 8 co-ordinates. This method implemented a lean way of tracking with a single sensor technology, which is a single camera sensor. Later the same procedure has implemented on LiDAR point cloud from the same data set. This tracker is an 3D tracker in 8 co-ordinate system with constant velocity dynamic modelling. The state estimation has done using Kalman filter algorithm. The predicted states of objects are associated to the measurements using Hungarian algorithm. Both trackers associated with simple fusion technique for better accuracy. The system has evaluated on CLEARMOT [5] metrics. Chapter 7 in the thesis highlights the results for both 2D tracking and 3D tracking.

REFERENCES

REFERENCES

[1] Manghat, Surya Kollazhi, and Mohamed El-Sharkawy. ”Forward Collision Prediction with Online Visual Tracking.” 2019 IEEE International Conference of Vehicular Electronics and Safety (ICVES). IEEE, 2019.

[2] Bochinski, Erik. ”High-Speed Tracking-by-Detection Without Using Image Information” Proceedings of the 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2017 [online] [http://elvera.nue.tu-berlin.de/files/1517Bochinski2017.pdf] [Accessed on 6th October 2018].

[3] Seung-Hwan, Bae ”Abstract Robust Online Multi-Object Tracking based on Tracklet Confidence and Online Discriminative Appearance Learning ” Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[4] Geiger, Andreas, Philip Lenz, and Raquel Urtasun. ”Are we ready for autonomous driving? the kitti vision benchmark suite.” Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2012.

[5] Bochinski, Erik. ”Extending IOU Based Multi-Object Tracking by Visual Information” Proceedings of the IEEE International Conference on Advanced Video and Signals-based Surveillance, Auckland, New Zealand 2018.

[6] Bernardin, Keni. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics, Hindawi Publishing Corporation EURASIP Journal on Image and Video Processing Volume 2008, Article ID 246309, 10 pages.

[7] Mousavian, Arsalan. ”3d bounding box estimation using deep learning and geometry.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[8] Qi, Charles R. ”Frustum pointnets for 3d object detection from rgb-d data.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[9] Wang, Li. ”Evolving Boxes for Fast V ehicle Detection” Proceedings of IEEE International Conference on Multimedia and Expo (ICME), 2017.

[10] Zhang, Xinyu. ”Real-time vehicle detection and tracking using improved histogram of gradient features and Kalman filters.” Proceedings of International Journal of Advanced Robotic Systems, 2018.

[11] Bewly, Alex. ”Simple Online and Realtime Tarcking.” Proceedings of the IEEE International Conference on Image Processing (ICIP), 2017.

[12] Arsalan Mousavian, Dragomir Anguelov, John Flynn, Jana Kosecka. ”3D Bounding Box Estimation Using Deep Learning and Geometry.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[13] Katare, Dewant, and Mohamed El-Sharkawy. ”Embedded System Enabled Vehicle Collision Detection: An ANN Classifier.” Proceedings of 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, 2019.

[14] Yoon, Ju Hong. ”Structural Constraint Data Association for Online Multi-object Tracking.” Proceedings of International Journal of Computer Vision. 2018.

[15] Mousavian, Arsalan. ”3d bounding box estimation using deep learning and geometry.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition , 2017.

[16] Dequaire, Julie. ”Deep tracking in the wild: End-to-end tracking using recurrent neural networks.” Proceedings of International Journal of Robotic Research, 2017.

[17] Wang, Li et al. ”Evolving Boxes for Fast Vehicle Detection” IEEE International Conference on Multimedia and Expo (ICME), 2017, pp. 1135-1140.

[18] Girshik, Ross ”Fast R-CNN”2015 IEEE International Conference on Computer Vision (ICCV) , 2015.

[19] Dongliang, Zheng,”Planning and Tracking in Image Space for Image-Based Visual Servoing of a Quadrotor” IEEE Transactions on Industrial Electronics, 2018.

[20] Zhang, Zhang ”Toward Occlusion Handling in Visual Tracking via Probabilistic Finite State Machines ” Published in: IEEE Transactions on Cybernetics Page(s): 1 - 13, 2018.

[21] Boksuk, Shin. ”Vision-based navigation of an unmanned surface vehicle with object detection and tracking abilities” Published in Journal Machine Vision and Applications archive Volume 29 Issue 1, January 2018 Pages 95-112

[22] Zhang, Li. ”Global Data Association for Multi-Object Tracking Using Network Flows” Published in IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[23] Baser E, Balasubramanian V, Bhattacharyya P and Czarnecki K, 2019. FANTrack: 3D Multi-Object Tracking with Feature Association Network. In IEEE Intelligent Vehicles Symposium.arXiv preprint arXiv:1905.02843 [online] [https://arxiv.org/pdf/1905.02843v1.pdf] [Accessed on 6th November 2019].

[24] Shi S, Wang X, and Li H, 2018. PointRCNN: 3D Object Proposal Generation and Detection from Point Cloud. In Computer Vision and Pattern Recognition. arXiv preprint arXiv:1812.04244 [online] [https://arxiv.org/pdf/1812.04244.pdf] [Accessed on 10th November 2019].

[25] Voigtlaender P, Krause M, Osep A, 2019. MOTS: Multi-Object Tracking and Segmentation. arXiv preprint arXiv:1902.03604v2. [online] [https://arxiv.org/pdf/1409.1556.pdf] [Accessed on 10th November 2019].

[26] Girdhar, Rohit. ”Detect-and-track: Efficient pose estimation in videos.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[27] Wang, Qiang. ”Fast online object tracking and segmentation: A unifying approach.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[28] Milan, Anton., 2016. MOT16: A benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 [online] [https://arxiv.org/pdf/1409.1556.pdf] [Accessed on 10th November 2019].

[29] Xu, Danfei, Dragomir Anguelov, and Ashesh Jain. ”Pointfusion: Deep sensor fusion for 3d bounding box estimation.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[30] Milan, Anton. ”Online multi-target tracking using recurrent neural networks.” Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[31] Garcia, Fernando. ”Sensor fusion methodology for vehicle detection.” IEEE Intelligent Transportation Systems Magazine 9.1 (2017): 123-133.

[32] Girshick, Ross. ”Fast r-cnn.” Proceedings of the IEEE international conference on computer vision. 2015.

[33] Girshick, Ross. ”Rich feature hierarchies for accurate object detection and semantic segmentation.” arXiv preprint arXiv:1311.2524. 2013 .

[34] Venkitachalam, Sreeram. ”Realtime Applications with RTMaps and Bluebox 2.0.” Proceedings on the International Conference on Artificial Intelligence (ICAI). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2018.

[35] Qi, Charles R. ”Frustum pointnets for 3d object detection from rgb-d data.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[36] Wen, Longyin. ”UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking.” arXiv preprint arXiv:1511.04136 (2015).

[37] CS231n: Convolutional Neural Networks for Visual Recognition. [online] [http://cs231n.github.io/convolutional-networks/] [Accessed on 6th October 2018]

[38] Chen S. Y. ”Kalman filter for robot vision: a survey.” IEEE Transactions on Industrial Electronics 59.11 (2011): 4409-4420.

[39] Catlin, Donald E. Estimation, control, and the discrete Kalman filter. Vol.71. Springer Science and Business Media, 2012.

[40] Faragher, Ramsey. ”Understanding the basis of the Kalman filter via a simple and intuitive derivation.” IEEE Signal processing magazine 29.5 (2012): 128-132.