Forward Collision Prediction with Online Visual Tracking

Surya Kollazhi Manghat and Mohamed El-Sharkawy
IoT Collaboratory IUPUI, Department of Electrical and Computer Engineering
Purdue School of Engineering and Technology Indianapolis
sukoll@iu.edu melshark@iupui.edu

***Abstract*—Safety is the key aspect when comes to driving. Self-driving vehicles are equipped with driver-assistive technologies like Adaptive Cruise Control, Forward Collision Warning system (FCW) and Collsion Mitigation by Breaking (CMbB) to ensure safety while driving. This paper proposes a method by following a lean way of multi-target tracking implementation and 3D bounding box detection without processing much visual information. Object Tracking is an integral part of environment sensing, which enables the vehicle to estimate the surrounding object's trajectories to accomplish motion planning. The advancement in the object detection methods greatly benefits when following the tracking by detection approach. This will lead to less complex tracking methodology and thus decreasing the computational cost. Estimation based on particle filter is added to precisely associate the tracklets with detections. The model estimates and plots bounding box for the objects in its camera range and predict the 3D positions in camera coordinates from monocular camera data using a deep learning combined with geometric constraints using 2D bounding box, then the actual distance from the vehicle camera is calculated. The model is evaluated on the KITTI car dataset.**

***Index Terms*—Autonomous vehicles, camera, tracking, KITTI, object detection, FCW, ADAS.**

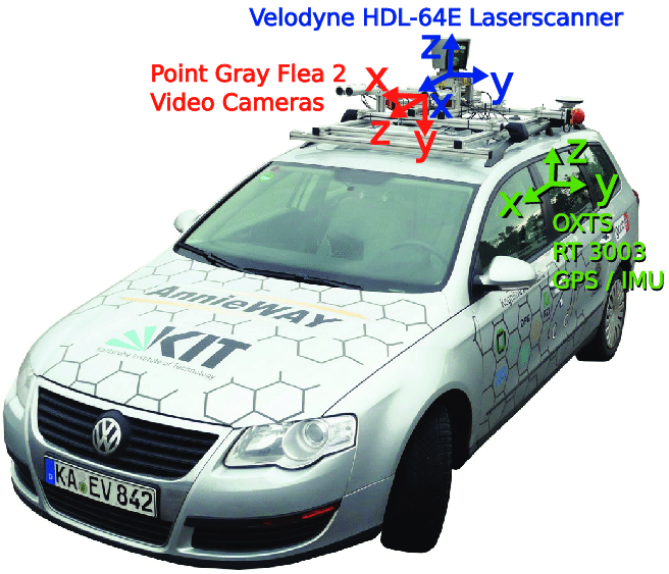

Fig. 1. KITTI Data Recording Platform Setup [3]

I. Introduction

The Autonomous cars combine a variety of sensors such as radar, lidar, sonar, GPS, odometry and inertial measurement units to perceive their surroundings. Interpreting the sensory information aids in perception of the environment, which can be used for Navigation and Control of a vehicle. Object Detection and Tracking are the main methods to perceive 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 must be aware of the position and dynamic information of certain moving objects encountered in the environment. By using a series of measurements in each video frame made over time, motion tracking can estimate, predict present and future locations.

The detection and tracking of objects of interest assist the intelligent autonomous vehicle for forward collision warning and path panning. In the case of forward collision avoidance, visual object tracking helps to distinguish the potential collision threats in terms of their relevance to the planned path of the vehicle [18]. The knowledge of moving objects around the vehicle enables a driver assistant system to alert a driver of potential collisions and dangers [12]. Other applications in the intelligent transportation systems is the design of an optimal trajectory for one vehicle under normal conditions or when overtaking a single moving vehicle on a predetermined road. Path planning for autonomous vehicles is to plan in real time a collision-free path in the presence of dynamically moving objects and with a limited sensing range. Here visual target tracking can help the autonomous vehicle pass through every area in the camera range and avoid obstacles.

Visual Multi Object Tracking estimate the object trajectories according to image sequence identities. Among numerous types of proposed MOT methods, Tracking by detection is the popular one. Tracking by detection methods can be broadly categorized in to online and offline tracking (batch or semi batch) methods. The offline methods use current and future frame for the detection results. 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 estimate the trajectory using the detections from past and current frames, are more applicable to real time applications such as ADAS, FCW and Navigation.

This is the author's manuscript of the article published in final edited form as:

Manghat, S. K., & El-Sharkawy, M. (2019). Forward Collision Prediction with Online Visual Tracking. 2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES), 1–5. https://doi.org/10.1109/ICVES.2019.8906291

Target tracking has been studied for decades with numerous applications [20]. Many methods has been introduced to solve the real time efficiency [1, 4] and the occlusion problem [19]. There are offline tracking algorithms [21] 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, 9, 10]. With recent approaches in the detection domains including CNN based [17] 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 [10] introduced a method of data association with out 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. Numerous MOT methods directly utilize the first- order or the second-order independent motion models to locate objects (Bae and Yoon 2014) and associate accurately. Here we present a Forwarding Collision Warning system which uses the detected object and incase of occlusion it utilizes the predicted track of objects to warn the Ego Vehicle about the surroundings. This real-time application uses online MOT with real-time as the motivation and a deep learning model to predict the objects location in camera coordinates from the 2D bounding box estimated. A strong emphasis is kept on efficiency to facilitate real time tracking.

II. Methodology:

Multi Object Tracking can be viewed as combination of Object Detection, Propagating the detection using Motion Model, Data Association and Managing the Tracklets.

Object Detection: The proposed 2D object detection uses visual based detection algorithm. Generally simple appearance models are used like raw pixel template representation,while color histogram is the most popular method of appearance modelling. Other approaches use covariance matrix representation, pixel comparison representation, SIFT-like features, or pose features. Recently, deep neural network architectures have been introduced for appearance modelling.

The 2D Object Detector in the proposed method uses raw images as input and output the best fit bounding box of the detected objects. The appearance features are ignored other than the detected bounding box for tracking. The position [x, y, w, h] and size of the 2D bounding box are used for further processing. Object re-identification in track will incorporate complexity and adds significant overhead into the tracking framework, which potentially limiting its use in real-time applications. This paper exploit recent advances in visual object tracking to solve the problems of online tracking by detection method, rather than aiming to be robust to detection errors. Object detection algorithms have been improved significantly. This results in improved object detectors, which allows implementation of much simpler tracking by detection methods. The position [x, y, w, h] and size of the 2D bounding box are used for the further processes. The appearance features are ignored other than the detected bounding box for tracking. The position [x1, y1, x2, y2] and size of the 2D bounding box are used to calculate the center. The tracking is done using center coordinates, width and height. The detections are available for 3 categories of KITTI dataset - car, pedestrian and cyclist. We aim to concentrate on tracking scenario of car category.

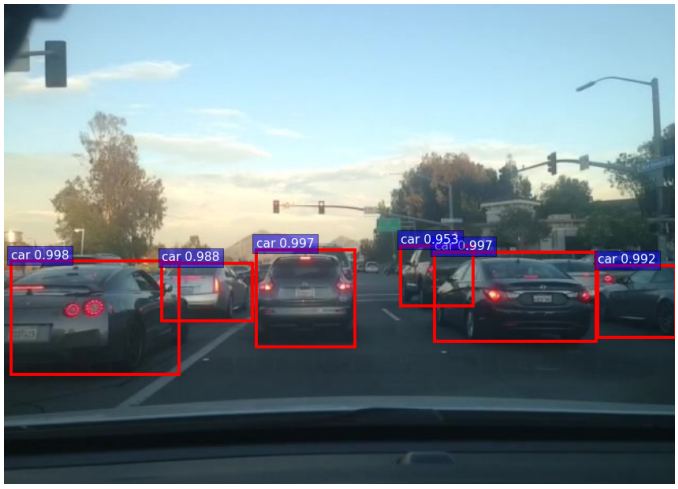

Fig. 2. 2D Object Detection [3]

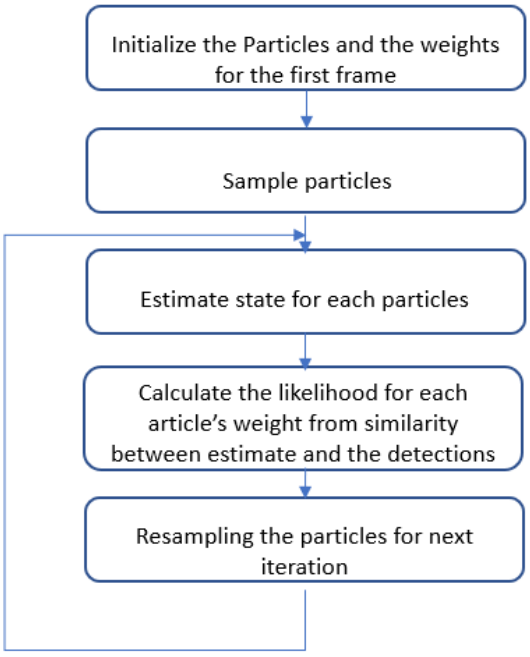

Fig. 3. Estimation Model

Estimation Model: 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, EKF yields good results.

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 filter or survival of the fittest, 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 filter 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, x_{vel}, y_{vel}, w_{vel}, h_{vel}], where x and y represent the horizontal and vertical pixel location of the centre of the target, while the scale w and h represent the width and height of the targets bounding box respectively.

Data Association: The most important stage of Multi-Target Tracking is data association and track to track association methods. The aim of this method is to identify the resemblence between the sensor measurements/detections and the pre-existing tracks. Incorrect assignment of newly detected objects to the existing tracks will result in remarkable decrease of accuracy. Generally multi-scan methods are recommended in situations where there are a lot of false alarms and missed detection. But delaying the association to include future information will negatively affect the use in real-time applications. So for associating tracklets Online trackers uses past and current frames and done in faster way possible.

when using sufficiently high frame rates, detections of an object in consecutive frames have high overlap IOU (intersection-over-union) [1].

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

If the above requirements are met, tracking becomes simpler and can be implemented even without using image 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 σ_{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 filtering procedure.

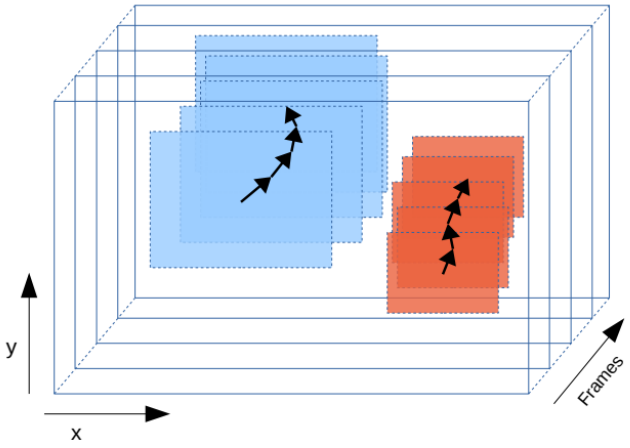

Fig. 4. IOU Data Association [1]

Managing Track Identities: SORT algorithm [10] with modifications is applied for a lean implementation of Multi Object Tracking. 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 untracked object and created a new identity for it. The tracker is initialized with the bounding box parameters [x, y, w, h] and velocity set to zero. Tracks which are not detected for T_{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 T_{lost} is set to 3 frames for object re-identification 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.

III. Forward Collision Prediction

The proposed Forward Collision Prediction uses a 3D Object detection algorithm, which relies on simple image. The input of the model is mono camera image and a 2D bounding box containing the target objects, outputs the 3D bounding box estimated for the targets. The architecture is inspired from the work of Mousavian et al. in [6] which utilizes a deep network and geometry for the calculation of 3D bounding box from a single image. The input image is cropped to 2D bounding box, re-sized and fed to the ResNet-34 CNN that extract feature vector. The feature vector is then input to the fully connected network of 3 layers to output (2*8 + 3+ 1) values. Algorithm is conceptually simple, but this method out performs complex and computationally expensive algorithms. The first 16 values in the output are the regressed residual for the pixel values of the projected eight corners of 3D bounding box on image plane. The next three values are regressed residuals for the size dimension of the 3D box. The last value is the regressed residual for the the distance of 3D box center from the camera center. The regressed residuals are used to find the 3D box parameters [x, y, z, h, w, l, θ]. This is estimated using geometry, by minimizing the difference

between the regressed pixel coordinates and the ones obtained by projecting the estimated 3D box onto the image plane. The architecture is implemented in python using PyTorch. The proposed method used the ResNet-34 model as in the extended FrustumPointNet [7] architecture and the network was trained using the Adam optimizer.

The estimate of 3D bounding box is obtained by minimizing the objective function

$$f(x,y,z,h,w,l,\theta) = f_{\mathrm{p}}(x,y,z,h,w,l,\theta) + \alpha f_{\mathrm{d}}(x,y,z) + \beta f_{\mathrm{s}}(h,w,l) \quad (1)$$

In equation-1 f_p computes the pixel coordinates for the 3D bounding box corners, f_d calculates 3D bounding box distance and penalize the difference with the regressed values. f_s penalize for the difference in (h, w, l) with regressed one. The regressed size estimate is believed to be more accurate than the distance estimate, so assigning $\beta > \alpha$.

The bounding box from the tracking is given as the input to the Distance Predictor using the 3D box estimation. In each time frame the actual coordinates from the vehicle camera is predicted for each tracks and distance is used to warn the vehicle about the collision chances.

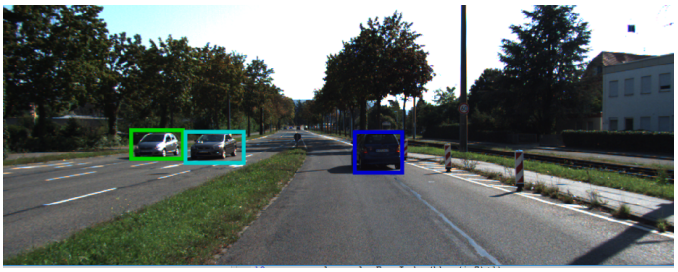

Fig. 5. The tracklets with unique identity has given unique color of bounding box. Distance (m) is predicted for each target in camera coordinates

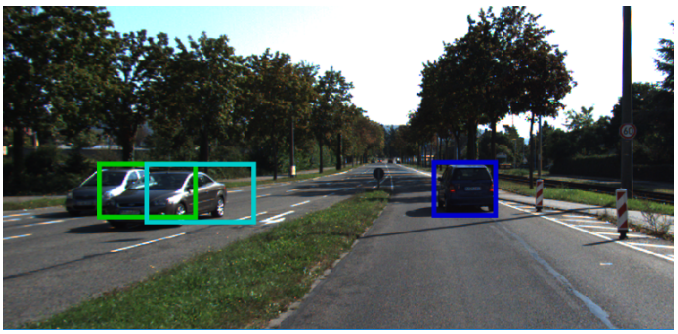

Fig. 6. The tracklet with occlusion in camera due its orientation against ego vehicle has been tracked

IV. Results

The Multi Object tracking methodology used by the model should be evaluated on the following criteria to talk about its efficiency. It should detect all the objects and estimate the location of the objects in the all the frames as precisely as possible. It should also keep track of these objects over time. Each detected object is given unique ID which stays constant throughout the sequence. The proposed model is tested on CLERAMOT [5] metrics. The proposed tracker method can run at 250 Hz (frames per second) on Intel i7 2.5GHz machine. Figure 5, 6, 7 shows the visual results of tracking. Table 1 shows the quantitative results. Figure 8 is the visual output of distance calculated from 3D coordinates and Table 2 shows the quantitative results.The 3D estimation gives low accuracy compared to RGB-D, Lidar models as the proposed system uses only camera information.

The accuracy of the tracker is discussed on:
MOTA(): Multi-object tracking accuracy
MOTP(): Multi-object tracking precision
MT(): number of mostly tracked trajectories. i.e. target has the same label for at least 80% of its life span.
ML(): number of mostly lost trajectories. i.e. target is not tracked for at least 20%of its life span.

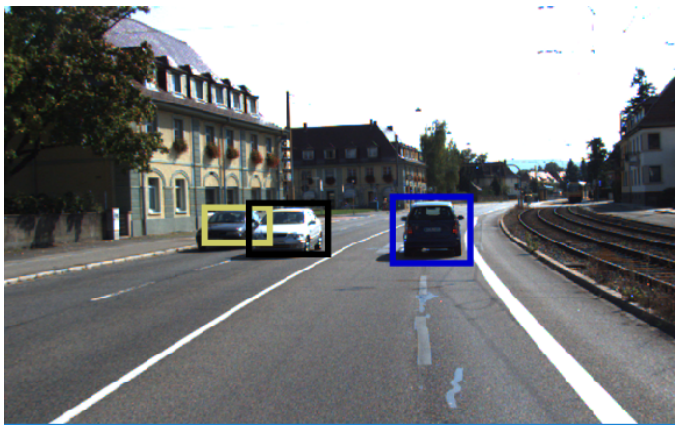

Fig. 7. The tracklet with missing detection has been tracked

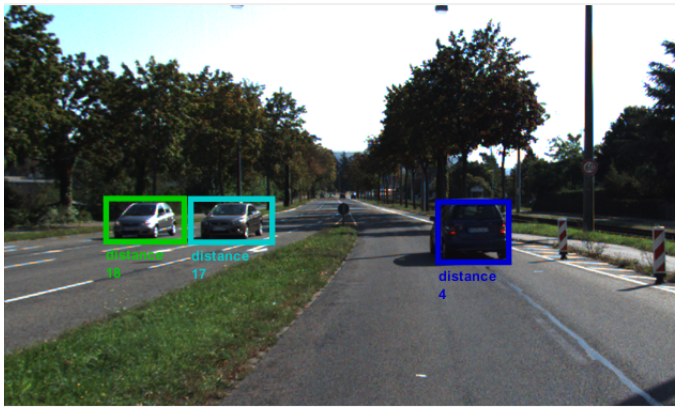

Fig. 8. The tracklet with distance (m) calculated in camera coordinates

V. Conclusion

This paper proposes an approach to implement FCW with online Multi-Target Tracking method by following efficient method of real time implementation. Most of the complex Multi Object Tracking methods achieve high efficiency at the

TABLE I
OUTPUT OF KITTI DATASET ON MONOCULAR CAMERA 2D TRACKER ARCHITECTURE

Method	MOTA	MOTP	MT	ML
Proposed Tracker- KITTI	32.2	71.2	10.8%	30.8%

TABLE II
OUTPUT OF KITTI DATASET ON MONOCULAR CAMERA 3D ESTIMATION ARCHITECTURE

Method	Easy	Moderate	hard
3D box predictor(Image)-KITTI	15.68	12.91	12.42
3D box predictor(Image)-KITTI (50%)	46.35	33.81	30.34

cost of run time performance. But for an autonomous vehicle the real time processing is critical. The FCW 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 and FCW. This method can be extended by integrating lidar and radar data for better accuracy of the tracker.

REFERENCES

[1] Bochinski, Erik, et al. "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.

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

[3] 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.

[4] Bochinski, Erik, et al. "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.

[5] Bernardin, Keni, et al. 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.

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

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

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

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

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

[11] 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.

[12] 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.

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

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

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

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

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

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

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

[20] Boksuk, Shin et al. "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

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