Obstacle Detection and Tracking Based on Multi-sensor Fusion

Shuyao Cui, Dianxi Shi, Chi Chen, Yaru Kang

HAL Id: hal-02197780

https://inria.hal.science/hal-02197780

Submitted on 30 Jul 2019

Obstacle Detection and Tracking Based on Multi-sensor Fusion

Shuyao Cui*, Dianxi Shi, Chi Chen, Yaru Kang

National University of Defense Technology, Changsha, Hunan Province, China
Shuyao8023you@163.com, dxshi@nudt.edu.cn,
darrenchan1992@163.com, 13272233899@163.com

Abstract. In the obstacle detection system，a great challenge is the perception of the surrounding environment due to the inherent limitation of the sensor. In this paper, a novel fusion methodology is proposed, which can effectively improve the accuracy of obstacle detection compared with the vision-based system and laser sensor system. This fusion methodology builds a sport model based on the type of obstacle and adopts a decentralized Kalman filter with a two-layer structure to fuse the information of LiDAR and vision sensor. We also put forward a new obstacles-tracking strategy to match the new detection with the previous one. We conducted a series of simulation experiments to calculate the performance of our algorithm and compared it with other algorithms. The results show that our algorithm has no obvious advantage when all the sensors are faultless. However, if some sensors fail, our algorithm can evidently outperform others, which can prove the effectiveness of our algorithm with higher accuracy and robustness.

Keywords: Multisensor, Data fusion, Kalman filer, Obstacle detection, Tracking

1 Introduction

The sensor is a component or device that can sense surrounding environment information and can convert these messages into corresponding useful output signals according to certain rules [1]. The development of sensor technology could drive the development of unmanned system technology to some extent, because the unmanned system's perception of the surrounding environment is based on sensor information [2]. However, because of the inherent limitations of sensors (such as the low precision of the ultrasonic sensor [3], the insensitivity of the laser sensor to the transparent object and the failure of the visual sensor in the dark environment, etc.), single sensor system cannot achieve high accuracy in obstacle detection. Although improving the performance of various sensors may help solve this problem [4], a new alternative

has aroused increasing attention: multi-dimensional heterogeneous data fusion technology. Compared with single sensor information, multi-sensor data fusion appears more competitive in the following aspects: fault tolerance, complementarity, real-time and economy [5], so it has been gradually applied.

In recent years, more and more people have used this technology for obstacle detection in robotics and automatic driving. Among them, Michael S. Darms et al. [6] used and fused LIDAR and RADAR sensor information from vehicles to track objects around the vehicle. Fernando García et al. [7] proposed a high-level fusion scheme that makes it possible to improve the classical ADAS (advanced driving assistance) system by integrating data provided by laser sensors and computer vision.

Our fusion system consists of LiDAR and computer vision sensors can be used to detect objects that may pose safety risks to drivers and pedestrians and this fused information provides a full understanding of the environment.

The rest of the article is organized as follows: Section II shows state of the art. Obstacle classification is introduced in Section III. In section IV, the process of tracking and data association fusion procedures based on decentralized Kalman filter with a two-layer fusion structure are described. Section V shows the obtained results. Finally, conclusions and future works are presented in Section VI.

2 Tracking procedures based on a decentralized Kalman filter

There are two main steps are related to tracking: estimation methods and data association techniques. The former is the estimation of the motion position at the next moment based on the previous motion state of the obstacle, and the latter deal with the matching of the new detection with the previous detection.

2.1 Estimation Method

In order to predict the position of the obstacle at the next moment more accurately, we use a decentralized Kalman filter with a two-layer fusion structure (DKF) to fuse the data of the vision sensor and LiDAR to fight the inherent limitation of sensors. We used the sequential sensor approach [8] that independently treats observations from various sensors and feeds them sequentially to the DKF estimation process.

The moving objects we are interested in (vehicles and pedestrians) have their own movement rules and constraints. For example, a pedestrian can move in any direction while the movement of the vehicle is limited by non-integral constraints [9]. Therefore, we establishes different motion models for different types of obstacles. For the vehicle, a sudden lateral movement should not be considered because of the road restrictions. We approximate it has a constant velocity motion in the two detection intervals and establish a constant velocity model for it.

The vehicles are difficult move at a constant speed, so our model should contain a certain amount of processing noise in the process of vehicle state prediction. The processing noise Q matrix is represented by the equations (1):

$$Q = \begin{bmatrix} \frac{a_x^2 t^3}{3} & \frac{a_x^2 t^2}{2} & 0 & 0 \\ \frac{a_x^2 t^2}{2} & a_x^2 & 0 & 0 \\ 0 & 0 & \frac{a_y^2 t^2}{3} & \frac{a_y^2 t^2}{2} \\ 0 & 0 & \frac{a_y^2 t^2}{2} & a_y^3 \end{bmatrix} \tag{1}$$

For pedestrians, we use the Constant Turn Rate and Velocity (CTRV) model because of its unlimited behavior. In this model, the state transfer function of CTRV is:

$$x(t + \Delta t) = \begin{pmatrix} \frac{v}{\omega}\sin(\omega\Delta t + \theta) - \frac{v}{\omega}\sin\theta + x(t) \\ -\frac{v}{\omega}\cos(\omega\Delta t + \theta) + \frac{v}{\omega}\cos\theta + y(t) \\ v \\ \omega\Delta t + \theta \\ \omega \end{pmatrix} \tag{2}$$

So, the formula for calculating the noise covariance matrix Q in the CTRV model is:

$$Q = \begin{bmatrix} \left(\frac{1}{2}\Delta t^2\sigma_a\cos\theta\right)^2 & \frac{1}{4}\Delta t^4\sigma_a^2\sin\theta\cos\theta & \frac{1}{2}\Delta t^3\sigma_a^2\cos\theta & 0 & 0 \\ \frac{1}{4}\Delta t^4\sigma_a^2\sin\theta\cos\theta & \left(\frac{1}{2}\Delta t^2\sigma_a\sin\theta\right)^2 & \frac{1}{2}\Delta t^3\sigma_a^2\sin\theta & 0 & 0 \\ \frac{1}{2}\Delta t^3\sigma_a^2\cos\theta & \frac{1}{2}\Delta t^3\sigma_a^2\sin\theta & \Delta t^2\sigma_a^2 & 0 & 0 \\ 0 & 0 & 0 & \left(\frac{1}{2}\Delta t^2\sigma_\omega\right)^2 & \frac{1}{2}\Delta t^3\sigma_\omega^2 \\ 0 & 0 & 0 & \frac{1}{2}\Delta t^2\sigma_\omega & \Delta t^3\sigma_\omega^2 \end{bmatrix} \tag{3}$$

After establishing motion model for obstacles, we used a distributed Kalman filter with a two-layer structure to fuse the data of multiple sensors. In Fig.1, each sensor subsystem estimates the states and makes fault detection independently. If any sensor subsystem is faulty by detection, it is isolated and restored. Instead, it is sent to the first fusion layer with a netted parallel structure in which the estimation errors of every pair of sensors are fused to determine the cross-covariance between them, while the estimations and covariances are sent to the second fusion layer.
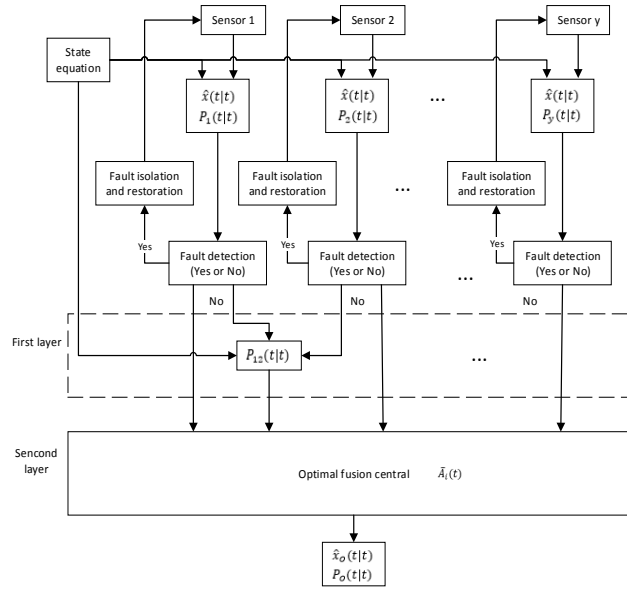
Fig. 1. The decentralized Kalman filter with a two-layer fusion structure.

The second fusion layer is the final fusion center in which the estimations and the cross-covariance matrices among the faultless local subsystems are fused to determine the optimal matrix weights and obtain the optimal fusion filter. On the other hand, after the faulty sensors are recovered, they can rejoin the parallel fusion structure. Due to the decentralized structure, the computational burden of fusion center is reduced, and the fault tolerance and reliability is assured.

2.2 Data Association Technology

The first step in the association process is to give up the impossible associations, so we have to create a threshold procedure for all trajectories. We use

this threshold procedure to create a region containing all the most likely combinations. Associations that do not belong to the threshold area will be discarded. In this step, the probability of all possible assumptions and the connection probability is calculated. The joint incidence probability represents the probability of the associated event m in time k tracking the joint associated event of J.

$$P(\theta|Z_k) = P_D^{M-n}(1-P_D)^n P_{FA}^{m_k-(1-m)} \prod_{j=1}^{m_k} g_{i,j} \tag{4}$$

Where P_D is the detection probability, and P_{FA} is the false alarm probability, M is the number of tracking targets, and n is the track assigned to clutter (no correlation).

Next, an assignment matrix should be created, in which each row denotes an observation value and each column indicates a trajectory. As a result, the probability of all combinations is calculated, and the allocation is carried out according to this matrix. This means that only one track is assigned to a given observation. All the tracks that are not assigned will increase their no-detection counters, and they will be eliminated if the counter reaches a given value.

2.3 Track Creation and Deletion Logic

Track creation and deletion policy is based on the definition of two different kind of tracks: united and non-united. Former are those tracks that were confirmed by both sensors at the same time or in subsequent scans. Later refers to tracks detected by a single sensor, it is not reliable enough since the other sensor have not confirmed it. Algorithm 1 defines the policy for track creation as well as when the track is updated.

```
Algorithm 1 Track Creation and Updated Logic
Case1: current state = non-united
  If the other sensor = detected, then track united.
  If the same sensor = detected, then track non-
united & track updated.
  If both sensor = detected, then track united.
  If consecutive_no_detection > 4, then track elimi-
nated.
Case 2: current state = united
  If single sensor = detected, then track updated.
  If both sensor = detected, then track updated.
```

```
  If consecutive_no_detection > 5, then track elimi-
nated.
Case3: current state = no match
  If single sensor = detected, then new non-united
track.
  If both sensor = detected, then new united track.
end
```

3   Results

We tested in a simulated environment to examine the feasibility of single-sen-
sor system and the fusion system independently.

We first tested the accuracy of the vision sensor and the LiDAR sensor, and
analyzed the accuracy of the sensor by using the missed rate and the false
detection rate as a measure. Table 1 shows that the performance of vision and
LiDAR scanners system when tested.

Table 1. Accuracy of vision and LiDAR

| Sensor Type	Fault Detection（%）	Miss Detection（%）
Vision	1.14	23.4
LiDAR	10.03	0.24

For LiDAR, the probability of missed obstacles is very low, which means
that most obstacles can be detected. However, one problem with laser sen-
sors is that they have higher false positives. The result of visual sensor is the
opposite. Visual algorithms can be used to increase the confidence of these
laser scanners because they with low false alarm rates. However, a low posi-
tive rate means that this algorithm is not robust enough, and fusion is a nec-
essary condition for increasing robustness.

3.1   Tracking Algorithm Performance

We use three sensors in the system, in the simulation, setting T (sampling pe-
riod) = 0.01. We calculated the position and velocity errors of local and decen-
tralized filters in 0-100 seconds and compared them with a centralized filter.
It can be seen from Fig. 2 that when all the sensors are not faulty, the accuracy

of the optimal fusion dispersion filter is higher than that of each local filter and close the accuracy of the concentrated filter.

To test the fusion filter fault-tolerant and robust, we assume that the first sensor is faulty. The fusion filter is indicated by red solid lines, the centralized filter is indicated by blue lines, and the real value is indicated by black line. From the simulation graph Fig.3, it can be seen that when the first sensor fails at 50 < t < 100, the centralized filter diverges, but the decentralized filter can still track the target.

(a) (b)
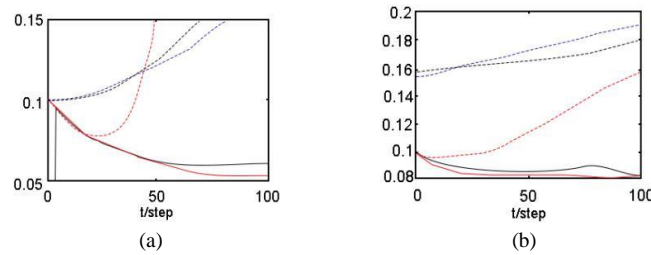
Fig. 2. Comparison of the precision of local filters, decentralized filter and centralized filter when all sensors are faultless: (a) filtering error of position; (b) filtering error of velocity. Local filter 1, red dotted line; local filter 2, black dotted line; local filter 3, dark blue line; centralized filter, red solid line; decentralized filer, black solid line.

The results show that when any sensor fails, the information fusion dispersion filter with double-layer structure has better fault tolerance and robustness characteristics.

(a) (b)
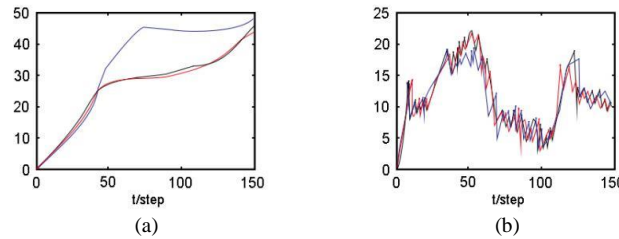
Fig. 3. The tracking performance comparison of the decentralized filter and centralized filter when the first sensor is faulty: (a) position; (b) velocity.

3.2 Fusion System Performance

After the fusion of vision and LiDAR sensor information, the accuracy of detection has been greatly improved. The fault detection rate and the miss detection reached 2.56 percent and 0.77 percent, respectively. Base on the data

mentioned above, we can say the requirement for improving the detection system performance can be achieved due to the fusion procedure.

4   Conclusion

The main goal of the presented work was to provide sensor fusion methodology for intelligent vehicles, able to overcome the limitations of each sensor, providing a robust and reliable safety application for road environment. The results provided show that by fusing the information of the camera and LiDAR, it was possible to accomplish the complex task of safe vehicle detection in inter-urban scenarios. Furthermore, the presented systems give the possibility to increase the set of sensors thanks to its extensibility.

5   Acknowledgment

6   Reference

1. Waltz E L. Multisensor Data Fusion[M]// Multisensor data fusion /. Artech House, 1990:25-42.
2. Nagla K S, Uddin M, Singh D. Multisensor Data Fusion and Integration for Mobile Robots: A Review[J]. Iaes International Journal of Robotics & Automation, 2014, 3(2).
3. Hall D L, Llinas J. An introduction to multisensor data fusion[J]. Proceedings of the IEEE, 2002, 85(1):6-23.
4. Michael S. Darms, Paul E. Rybski, Christopher Baker, et al. Obstacle detection and tracking for the urban challenge[J]. IEEE Transactions on Intelligent Transportation Systems, 2009, 10(3):475-485.
5. Luo R C, Kay M G. Multisensor Integration and Fusion in Intelligent System[J]. IEEE Transactions on Systems Man & Cybernetics, 1989, 19(5):901-931.
6. Wang G. Hall David L, McMullen Sonya AH: Mathematical Techniques in Multisensor Data Fusion[J]. Biomedical Engineering Online, 2005, 4(1):23.
7. Michael S. Darms, Paul E. Rybski, Christopher Baker, et al. Obstacle detection and tracking for the urban challenge[J]. IEEE Transactions on Intelligent Transportation Systems, 2009, 10(3):475-485.
8. Milanés V, Llorca D F, Villagrá J, et al. Intelligent automatic overtaking system using vision for vehicle detection.[J]. Expert Systems with Applications, 2012, 39(3):3362-3373.
9. Mitchell H B. Multi-Sensor Data Fusion: An Introduction[M]. Springer Publishing Company, Incorporated, 2007.