Article

Fluid-based field representation for collision risk assessment on the road scenes

Xuanpeng Li[1], **Lifeng Zhu**[1,*]

[1] School of Instrument Science and Engineering, Southeast University, Nanjing 210096, Jiangsu, China; li_xuanpeng@seu.edu.cn (X.L.), lfzhulf@gmail.com (L.Z.)

* Correspondence: lfzhulf@gmail.com (L.Z.)

Abstract: Prediction of the likely evolution in the traffic scenes is a challenging task because of high uncertainty of sensing technology and dynamic environment. It leads to failure of planning for intelligent agents like autonomous vehicles. In this paper, we propose a fluid-based physical model to present the influence of surrounding object's motion on driving safety. In our pipeline, the input sensor could be LiDAR, camera, or multi-modal data. We use a Kalman filter to estimate the state space of each detected object, and adopt the properties of stable fluid to build a riskmap based on the density field. The noisy state space are then modeled as the boundary conditions in the simulation of advection and diffusion process. We test our approach on the public KITTI dataset and find this model could handle the short-term prediction in case of misdetection and tracking failure caused by object occlusion, which shows promising in collision risk assessment on road scenes.

Keywords: Navier-Stokes solver; multi-object tracking; collision risk assessment; road scenes

1. Introduction

Assessing risks of collision is an essential task for autonomous vehicles driving on the road. It requires predicting the likely evolution of the current traffic situation, and assessing how dangerous that future situation might be [1]. A common measure of collision risk is time-to-collision (TTC) which is insufficient for handling complex situations, e.g., on a curved road. Moreover, once the object failed to be tracked due to hard occlusion, it would lead to less effective assessment of collision risks.

Collision risk assessment employs information from sensors and uses grid-based representation to estimate potential threats [4]. An occupancy grid of the scene is a typical model by updating with the Bayesian occupancy filter to continuously present the surrounding environments [2,3]. The spatial information of vehicles, cyclists, pedestrians and other obstacles is quantified into a distribution of occupancy by using probability or belief mass. The occupancy grid can be constructed by using multiple sensors like LiDAR, radar, cameras and fusion together.

However, only the occupancy in the visible region is presented, except for the occluded objects, which would be really potential threats. If an object is failed to be tracked due to occlusion, it has high probability to reappear. Therefore, in order to improve driving safety, we aim to develop a better representation of dynamic objects around ego-vehicle. It could reveal spatio-temporal information even if tracking is lost. Moreover, we expect to improve the efficiency of the prediction with physical model in the traffic scenes.

Physical model is employed to encode the spatial-temporal information for the interaction between objects. In 2011, Hamrick *et al.* [5] performed experiments to suggest that humans use internally physical models to predict how dynamical systems evolve. A recent line of work on long-term prediction based on physical simulators achieves some progress by fusion of data-driven learning method. For example, robots can quickly learn manipulation skills when predicting the consequences of physical interactions [6]. Pixel-level video prediction is another topic achieving great progress recently. It focuses on prediction in future by observing the generated frames in the past [7,8].

Different from a data-driven method, we explore a fluid-model-driven approach to simulate the evolving distribution of substance around surrounding objects and predict their behaviors. The advection and diffusion process in the fluid model help reveal hidden spatial information in presence of missing data. In order to analyze objects' interaction and estimate collision risk, we modify the standard diffusion process and put emphasis on the diffusion based on the object's state space.

In the following sections, we firstly make a short survey about related work in Sec. 2. Then, the formulation of fluid model and construction of riskmap is presented in Sec. 3. We evaluate our method with real-world public data from KITTI dataset in Sec. 4 and finally make a discussion and conclusion in Sec. 5.

2. Related Work

In this section, we review recent studies that have contributed to 3D object detection and tracking and risk representation on road scenes. In recent years, with the development of deep neural networks, research on the tasks of computer vision, such as image classification, object detection and semantic segmentation, achieve significant progress. Especially for the object detection, there are two categories of network frameworks: (1) two-stage network, e.g., R-CNN, Faster R-CNN and Mask RCNN; (2) one-stage network, e.g., YOLO and SSD. These methods show great improvement over traditional methods like SVM and AdaBoost. Some ideas are employed in 3D object detection towards point cloud from LiDAR, RGB-D and stereo cameras.

The 3D object detection based on the point cloud mainly has three categories of approaches: (1) projection from 3D point cloud to 2D image, such as Complex-YOLO [10] and BirdNet [11]; (2) voxel feature based detection, like 3DFCN [12], Vote3Deep [13], and VoxelNet [13]; (3) multi-modal fusion approaches, such as MV3D [15] and AVOD [16]. Recent research in multi-object tracking (MOT) has focused on the tracking-by-detection and learning-to-track principals. Both batch methods [17,18] and online methods [19,20] have explored how to learn a similarity function for data association. More recent studies on MOT have integrated the hierarchical features from deep convolution networks [21, 22] and correlation filter [9]. In addition, reinforcement learning algorithm has been proposed to link data in online MOT, and Markov decision processes (MDP) have been proved to be suitable for dynamic environment [23]. Multi-object can be modeled as multi-agent which has its own lifetime to perform certain tasks and maintain certain states.

Prediction of objects' motions and risk assessment in traffic scenes is an important issue. Trajectory analysis [24,25] and occupancy grid mapping [2,26] are two major approaches to compactly represent other objects' motions. Trajectory analysis mainly focuses on the temporal characteristic of vehicle's motion. Sivaraman *et al.* [24] attempt to obtain vehicles' trajectories in front of the ego-vehicle from a monocular camera and a stereo camera. The behaviors of the observed trajectories are then learned by unsupervised learning method. In [25], panoramic camera arrays are used to capture a full surround view, and trajectories of surrounding vehicles are then extracted and classified by the hidden Markov model (HMM) into certain state like overtaking and lane change. On the other side, occupancy grid mapping essentially relies on probabilistic spatial information, e.g., using recursive Bayesian filtering [2]. A probabilistic grid-based approach is used to exploit surrounding vehicles' movements and infer the presence and location of occluded road surface as a complementary method to already existing road detection systems [26].

In addition, the field-based approaches have been proposed for automatic vehicle guidance for a long history. Electric field model is used to interpret the vehicle's motion as an electron within an electric field, and the system is transformed into a riskmap reflecting the risk at a certain position in the dynamic environment [27]. Wolf and Burdick [28] introduce a vehicle collision avoidance system in a full two-dimensional field with lane, road, car, and velocity potential function components. Recent work [29] assesses collision risks by risk potential modeling of predicted motion of the surrounding vehicles in various driving conditions. Wang *et al.* [30] propose a hybrid field model to assess the driving safety and use it in pre-collision warning system. We have proposed a field-based representation of surrounding vehicles from the forward-moving monocular camera [31]. But it is just a first attempt, which do not take the scale, orientation and velocity of object into account. In this work, we make improvement on building the fluid-based field with the properties of objects, which gives an impressive representation of collision risk.

3. Approach

In this paper, the key point is using the fluid model to demonstrate the interaction of moving objects in the traffic scene, no matter which modal of the input data. Our framework would fit multi-modal sensor data such as LiDAR, radar and camera. The 3D LiDAR point cloud data from the public KITTI dataset is employed, which could provide accurate objects' location and classification via the state-of-the-art 3D object detector [32]. Our system pipeline consists of two components: (a) 3D object detection and tracking; (b) fluid-based field computation and representation. In the follows, we will present the details of each step.

3.1. 3D Object Detection and Tracking

3D object detection module provides n 3D bounding boxes $\mathbf{D}_t = \{\mathbf{D}_t^1, \mathbf{D}_t^2, \dots, \mathbf{D}_t^n\}$ at the frame t, which contains the 3D coordinate of the object center x^i, y^i, z^i , object's size l^i, w^i, h^i, and heading angle θ^i for each detected object $\mathbf{D}_t^i = (x^i, y^i, z^i, \theta^i, l^i, w^i, h^i)$. We directly adopt the state-of-the-art 3D detector *PointRCNN* with their models pre-trained on the KITTI training set to get the detection results.

Inspired by the work [33], we use the Kalman filter to predict the entire state spaces of n detected object trajectories $\mathbf{T}_{t+1} = \{\mathbf{T}_{t+1}^1, \mathbf{T}_{t+1}^2, \dots, \mathbf{T}_{t+1}^n\}$ with constant velocity model. The state space of each object trajectory is formulated as a 10-dimensional vector $\mathbf{T}^i = (x^i, y^i, z^i, \theta^i, l^i, w^i, h^i, v_x^i, v_y^i, v_z^i)$. The motion model of the Kalman filter is formulated as

$$\mathbf{T}_{t+1} = \mathbf{F} * \mathbf{T}_t + \mathbf{W}_t \tag{1}$$

with

$$\mathbf{F} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{W}_t \sim \mathcal{N}(0, \mathbf{R}_t).$$

Since we don't observe the initial velocities of objects, the covariance matrix $\mathbf{P}$ of high uncertainty should be given at the inital step.

Then, a data association module is employed to match the next detection $\mathbf{D}_{t+1}$ with predicted trajectories $\mathbf{T}_{t+1}$ by using the metrics of 3D IoU and Hungarian algorithm [34]. The outputs of the data association module are the matched pairs of detection and predicted trajectories $(\mathbf{D}_{t+1}, \mathbf{T}_{t+1})_{\text{match}}$. Using the matched pairs, the measurement model of the Kalman filter is set as

$$\mathbf{D}_{t+1} = \mathbf{H} * \mathbf{T}_{t+1} + \mathbf{V}_{t+1} \tag{2}$$

with

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \text{ and } \mathbf{V}_{t+1} \sim \mathcal{N}(0, \mathbf{Q}_{t+1})$$

In order to manage the birth and death of the trajectories, unmatched detection is considered as potential objects entering the perspective field and new trajectory would be created. In addition, unmatched trajectories are regarded as potential objects leaving the perspective field, and once each unmatched trajectory exceeds several frames, it will be deleted which means the death of the trajectory.

The output of the Kalman filter is the corrected trajectories of objects based on the matched detection. However, it is still hard to handle the occlusion issue due to missing detection. We employ the fluid-based representation module to manage the short-term occlusion problem thanks to the physical modeling of the fluid.

3.2. Fluid-based Field Representation

With the trajectory of tracked objects, we are able to compute the 2D riskmap in the bird-eye view based on a fluid-based field representation. In the field of physics, a fluid whose density and temperature are nearly constant is described by a velocity field $\mathbf{u}$ and a pressure field p [35]. Given that the velocity and the pressure are known for the initial time $t = 0$, the evolution of these quantities over time is given by the Navier-Stokes equations [36]:

$$\begin{aligned} \nabla \cdot \mathbf{u} &= 0 \\ \frac{\partial \mathbf{u}}{\partial t} &= -(\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{1}{\rho}\nabla p + \mathrm{v}\nabla^2 \mathbf{u} + \mathbf{f} \end{aligned} \tag{3}$$

where v is the kinematic viscosity of the fluid, ρ is its density and $\mathbf{f}$ is an external force. The "$\cdot$" denotes a dot product between vectors, while the vector spatial partial derivatives is denoted as $\nabla = (\partial/\partial x, \partial/\partial y)$ in 2D and $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)$ in 3D. And the shorthand notation $\nabla^2 = \nabla \cdot \nabla$. In our pipeline, let $\mathbf{x}_t^i = (x^i, y^i)_t$ denote the spatial coordinate of the ith object at the time t, because we only consider the 2D fluids. The relative velocity of the ith object (to ego-car) at the time t is denoted as $\mathbf{u}_t^i = (v_x^i, v_y^i)_t$. Note that if the object at the last frame is failed to be tracked, we assume its velocity unchanged.

As stated by the Helmholtz-Hodge Decomposition [37], any vector field $\mathbf{w}$ can be written uniquely as $\mathbf{w} = \mathbf{u} + \nabla q$, where $\nabla \cdot \mathbf{u} = 0$ and q is a scalar field. The velocity field is a vector field, and thus it can be projected into a "divergence free" part regarded as $\mathbf{u} = \mathbf{P}\mathbf{w} = \mathbf{w} - \nabla q$ by an operator $\mathbf{P}$. The Eq. 3 can be transferred to

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{P}(-(\mathbf{u} \cdot \nabla)\mathbf{u} + \mathrm{v}\nabla^2 \mathbf{u} + \mathbf{f}) \tag{4}$$

and it removes the pressure term of the scalar field by using the fact that $\mathbf{P}\mathbf{u} = \mathbf{u}$ and $\mathbf{P}\nabla p = 0$.

In this module, we compute the distribution of the substance around each object as a density field and adopt incompressible flow to model the behavior of substance. Suppose at the time t, we take the trajectories $\mathbf{T}_t$ to describe the involved computation of the corresponding density field as a riskmap $\mathcal{D}(\mathbf{x}_t)$. After $\mathcal{D}(\mathbf{x}_t)$ is obtained, we advance to the computation at the next time step. The input and output are illustrated in Fig. 1, where we also show the ego-car in magenta and other detected vehicles in black.

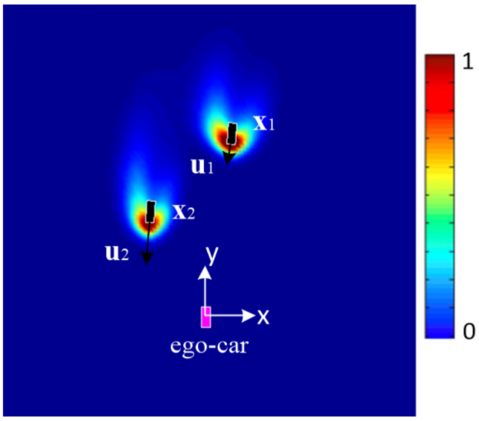

Figure 1. Riskmap: the ego-car in magenta and other detected vehicles in black. The red region means high collision risk, while the blue means no collision risk.

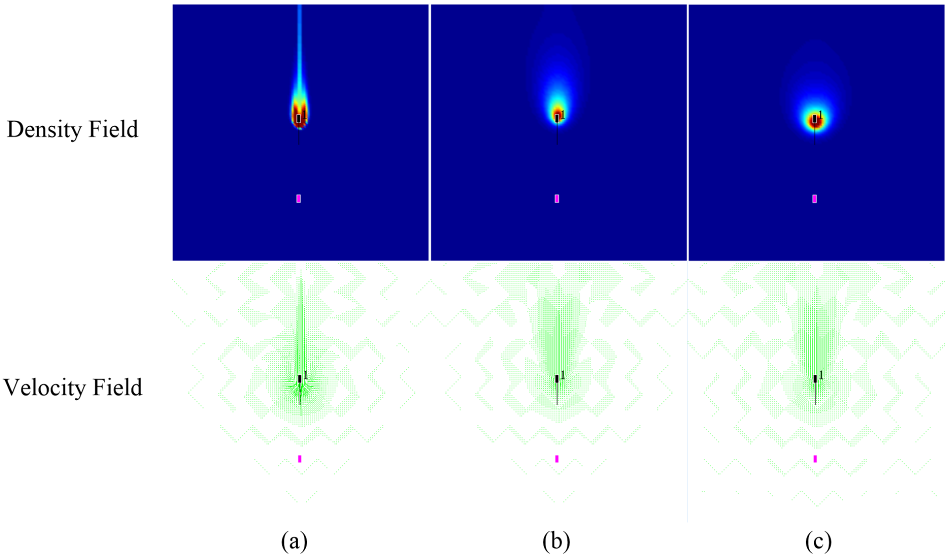

Figure 2. Advection and diffusion: (a) only advection; (b) only diffusion; (c) advection and diffusion.

This process starts from the solution $\mathbf{w}_0(\mathbf{x}) = \mathbf{u}(\mathbf{x}, t)$ of the previous step and then sequentially resolve each term of Eq. 4. Similar to the work [35], we decompose the computation of the field into an advection stage, a projection stage and a diffusion stage of a periodic process. The steps are:

$$\mathbf{w}_0(\mathbf{x}) \overbrace{\longrightarrow}^{\text{advect}} \mathbf{w}_1(\mathbf{x}) \overbrace{\longrightarrow}^{\text{diffuse}} \mathbf{w}_2(\mathbf{x}) \overbrace{\longrightarrow}^{\text{project}} \mathbf{w}_3(\mathbf{x}).$$

This process involves the advection and diffusion without adding the force term $\mathbf{f}$. The density field periodically evolves from the advection to the diffusion. The effects of the advection and diffusion are illustrated in Fig. 2.

As objects move, the velocity can be modeled as the boundary condition in the advection process. Specifically, we first solve the differential equation as

$$\frac{\partial \mathbf{u}}{\partial t} = -\mathbf{u} \cdot \nabla \mathbf{u} \tag{5}$$

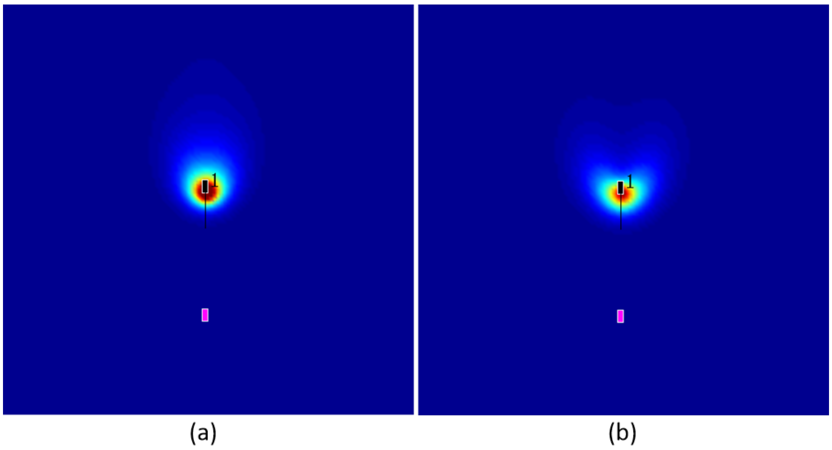

Figure 3. Different diffusion: (a) isotropic diffusion; (b) anisotropic diffusion.

with $\mathbf{u}(0) = \mathbf{w}_0$ and $\mathbf{w}_1 = \mathbf{u}(\Delta t)$ for the velocity field $\mathbf{u}$ with the boundary condition. Since the substance is incompressible under the advection guided by $\mathbf{w}_1$, the advection of the substance in the density field $\mathcal{D}$ is computed by solving

$$\frac{\partial \mathcal{D}}{\partial t} = -\mathbf{w}_1 \cdot \nabla \mathcal{D}. \tag{6}$$

Then, diffusion is the other effect of the moving vehicles on substances. Each vehicle is supposed to continually play a role on the entire field. In this case, the existence of surrounding vehicles can be persistently observed even with a periodic advection of the density field. Formally, we solve the diffusion equation:

$$\frac{\partial \mathcal{D}}{\partial t} = -\lambda \Delta \mathcal{D} + s, \tag{7}$$

where λ controls the rate of diffusion and s measures the strength of the source of diffusion at the position of each object. For preventing the space being fully filled with substances, a damping step is proposed at the end of each time step. The density field is damped as $\mathcal{D} = \omega \mathcal{D}$ after the diffusion, where the damping coefficient ω is set to 0.95 in our implementation.

The advection and diffusion equations as well as the projection for incompressibility are solved by the solver proposed in [35]. The diffusion equation is solved by iteratively applying isotropic Gaussian filter as shown in Fig. 3(a); thus, the distribution of the density field around each object is isotropic. For better presenting the possible risk, we introduce anisotropy along the velocity of the tracked objects in the diffusion step as illustrated in Fig. 3(b). The diffusion stencil in isotropic diffusion is uniform around each grid, i.e., $w(i \pm 1, j \pm 1) = \lambda$. We take the velocity field $\mathbf{u}$ into account to modify the stencil. The stencil at neighboring grids is updated as $w(i + sgn(v_x), j) = \lambda + a \cdot v_x$ and $w(i, j + sgn(v_y)) = \lambda + a \cdot v_y$, if the velocity at the grid (i, j), a is the parameter controlling the anisotropy and the sign function is

$$sgn(x) = \begin{cases} \frac{d}{dx}|x| & \text{if } x \neq 0 \\ 0 & \text{otherwise} \end{cases} . \tag{8}$$

Note that more complex anisotropic such as Laplace operator could be used here, but we only adopt such updating scheme because it is a simple and efficient encoding method for the short-term prediction of surrounding objects.

4. Experiments

In this section, we evaluate the proposed method on the KITTI tracking dataset, which contains 21 LiDAR data clips and corresponding image sequences. Currently, we take the vehicles as example for validation and ignore other objects like pedestrians and cyclists. We directly employed the pre-trained PointRCNN model to get the results of 3D object detection from the LiDAR point cloud. Then, the 3D Kalman filter is adopted to track multiple objects in each sequence. All tracked data are illustrated in a bird-eye view with a fluid model in order to assess collision risk. The fluid-based riskmap is demonstrated in Fig. 4, where we project the 3D multi-object tracking results on the synchronized image sequence. The process of fluid modeling is computed on a single CPU thread with 2.5GHz about 25 milliseconds. The field is updated frame by frame with a resolution 512 × 512, corresponding to an area of $80m \times 80m$. Fig. 4(a) shows the 3D tracking results projected on the continuous image frames. The advection and diffusion process makes the density field of corresponding objects varying frame by frame until they disappear from the view as shown in Fig. 4(b). The region with the red color represents high collision risk area, and the blue color means no collision risk. When objects leave from ego-car's perception field, the substance of the density field would dissipate immediately; it means there is no more risk from these objects.

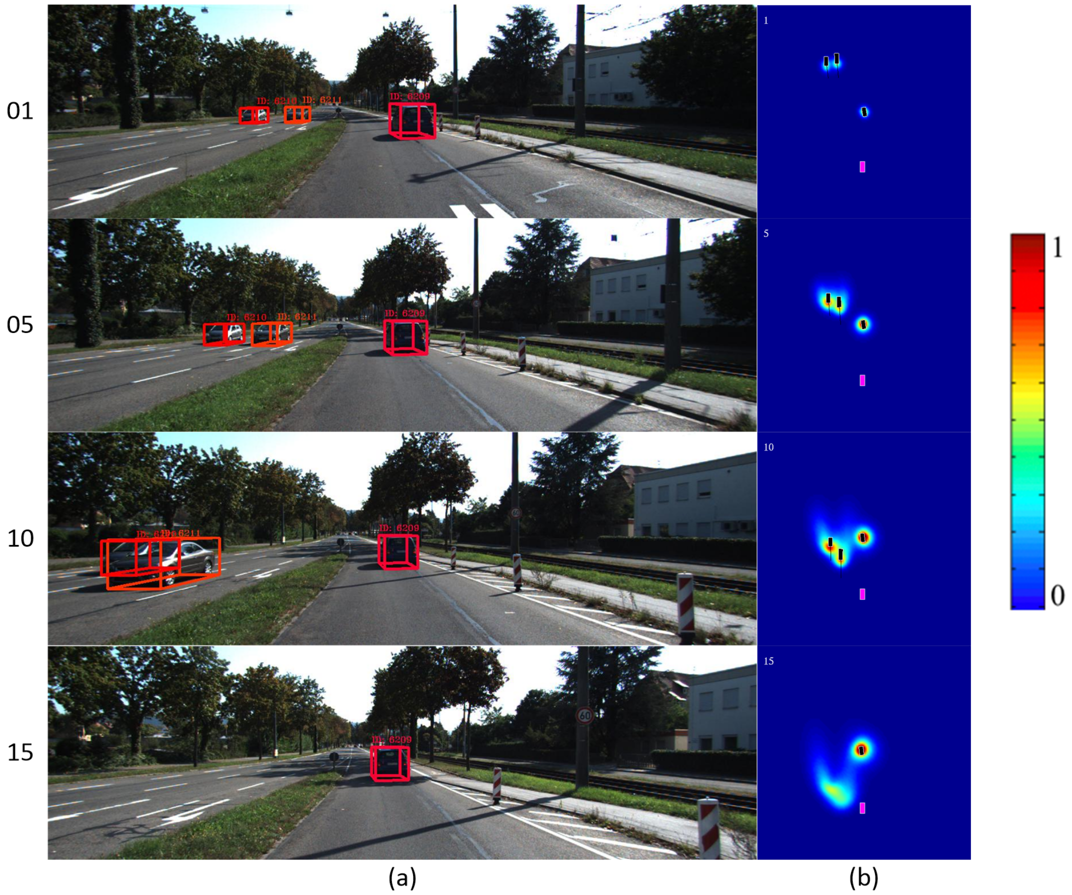

Figure 4. Fluid-based field representation on the *KITTI tracking training set 0010*: (a) 3D multi-object tracking results projected on the image sequence; (b) density field representation.

We find that it is relatively stable with respect to noise in the multi-object tracking, especially when there is tracking failure as shown in Fig. 5. Let's look at the area marked by red circles in the riskmap. At the first frames 122, 123, 124, it is successfully detected and tracked (3D bounding box in blue); then, it is fully occluded by another vehicle and no more being tracked in the following frames from 125 to 127. At the last two frames 128, 129, we can see that this vehicle is detected and tracked as a new object. In this sequence, the riskmap could still show the influence field of this object by advection and diffusion, even though it is not detected and tracked.

In cases with no tracking failure, our method presents a riskmap with the evolving spatial and temporal information, while conventional risk assessment methods produce a scalar signal to measure

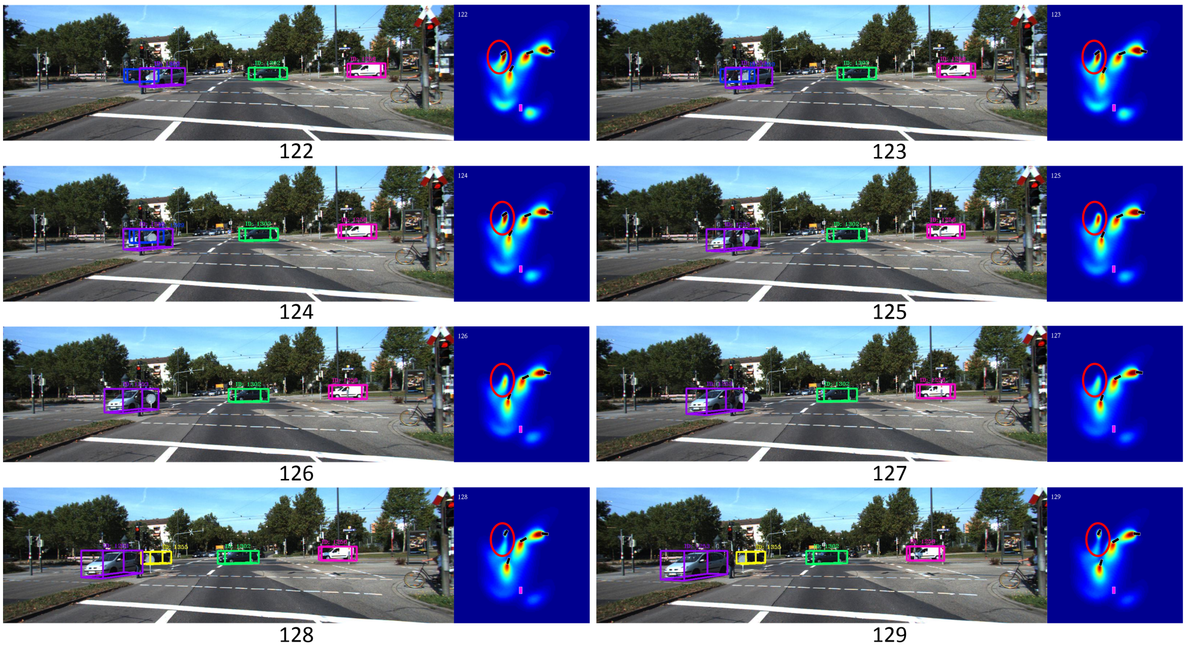
122
123
124
125
126
127
128
129

Figure 5. The fluid model provides a field-based representation of failed-tracking object. The red circles indicate the riskmap changing when the vehicle becomes lost due to occlusion.

the risk level. The density field presents potential risks as the ego-vehicle moves. Comparing with occupancy grid which only shows the static occupancy information, our method could assess collision risk by dynamic prediction. We test our method with more complex scenarios, and the results are shown in Fig. 6. The regions with potential risk are filled with warmer color when the ego-vehicle is waiting at a crossroad or making a turn. The passing vehicles around ego-car show their possible collision region on the orientation as shown in Fig. 6(a). Fig. 6(b) illustrates that when the ego-car turns, the surrounding objects make a reasonable density-field representation of possible collision risk.

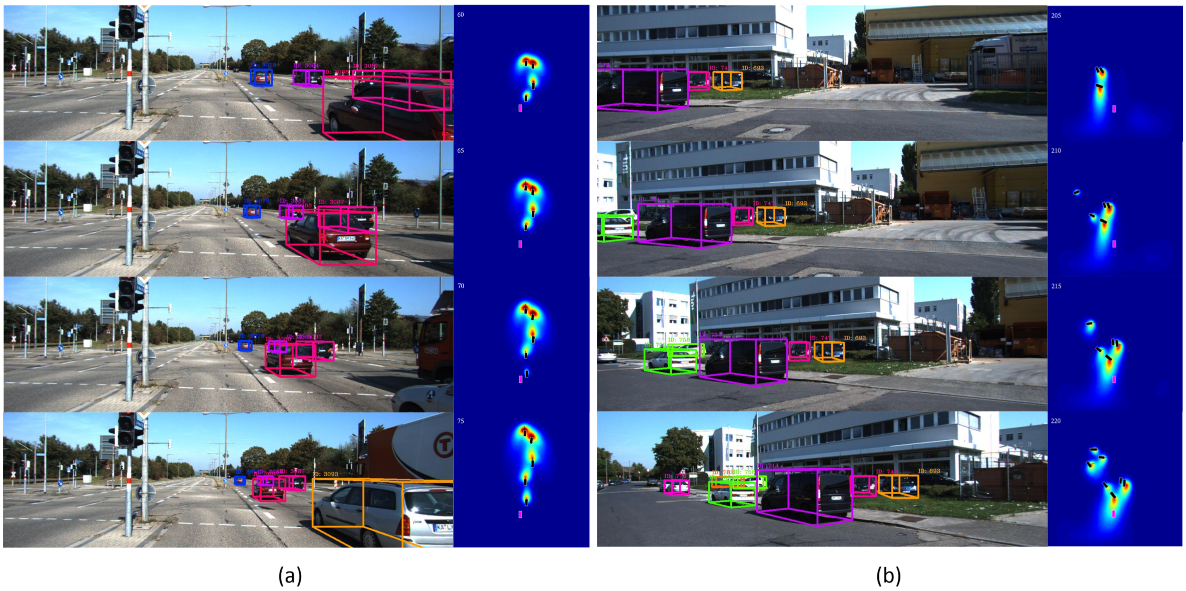
(a)
(b)

Figure 6. The fluid-based representation works in various scenarios: (a) the ego-car is waiting at a crossroad, and the density map shows the surrounding cars pass; (b) the riskmap illustrates the possible collision risk when the ego-car makes a turn.

5. Conclusion

In this paper, we propose an approach of the fluid-based risk assessment and representation, which can be easily integrated with different modal data, like LiDAR point cloud and camera image sequence. We use a state-of-the-art 3D detector and the traditional Kalman filter for multi-object tracking to estimate the state space of surrounding objects (using vehicle as example). Then, the advection and diffusion process handles the physically evolving density field in order to build the dynamic riskmap for intelligent agent like autonomous vehicle. This method enables a short-term prediction of the distribution along the trajectory of each object, even though object detection and tracking fails and sensor data is imperfect. Besides, our method can also be used in the inference of object's visibility, e.g., certain object becomes occluded. The efficiency of the proposed approach is evaluated on the public KITTI dataset.

Driving assistance and automated driving are high-level tasks which require the understanding of the driving semantics or intention, while our method only takes the estimated trajectories and creates information at the conceptual level. In the future we will explore how to combine more road elements like lane marking, geometric information, static obstacles for planning and decision-making further. Although we do not focus on tracking occluded objects in this work, we do believe a stable tracking is beneficial for risk assessment. Therefore, a field-based systematic method to deal with the challenges of tracking surrounding vehicles is also part of our future work.

Author Contributions: conceptualization, X.L. and L.Z.; methodology, X.L.; software, X.L. and L.Z.;validation, X.L.; investigation, X.L.;writing—original draft preparation, X.L.; writing—review and editing, X.L. and Z.L.; visualization, X.L.; funding acquisition, X.L. and L.Z.

Funding: This research is funded by the Natural Science Foundation of Jiangsu Province grant number No. BK20160700, and the Fundamental Research Funds for the Central Universities No. 2242018K40067 and 2242019K40039.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Lefèvre, S.; Dizan V.; Christian L. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH journal* **2014**, *1(1)*, 1, doi:10.1186/s40648-014-0001-z.
2. Coué, C.; Pradalier, C.; Laugier, C.; Fraichard, T.; Bessière, P. Bayesian occupancy filtering for multitarget tracking: an automotive application. *Int J. Rob. Res.* **2006**, *25(1)*, 19–30, doi:10.1177/0278364906061158.
3. Nguyen, T. N.; Michaelis B.; Al-Hamadi; A.; Tornow, M.; Meinecke, M. M. Stereo-camera-based urban environment perception using occupancy grid and object tracking. *IEEE Trans. Intell. Transp. Syst.* **2011**, *13(1)*, 154–165, doi:10.1109/TITS.2011.2165705.
4. Laugier, C.; Paromtchik, I. E.; Perrollaz, M.; Yong, M.; Yoder, J. D.; Tay, C.; Négre, A. Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety. *IEEE Intell. Transp. Syst. Mag.* **2011**, *3(4)*, 4–19, doi:10.1109/MITS.2011.942779.
5. Hamrick, J.; Battaglia, P.; Tenenbaum, J. B. Internal physics models guide probabilistic judgments about object dynamics. In Proceedings of the 2011 Annual Conference of the Cognitive Science Society, Boston, MA, USA, 20-23 Jul 2011; vol. 2.
6. Villegas, R.; Yang, J.; Zou, Y.; Sohn, S.; Lin, X.; Lee, H. Learning to generate long-term future via hierarchical prediction. In Proceedings of the 2017 International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 3560–3569.
7. Villegas, R.; Yang, J.; Hong, S.; Lin, X.; Lee, H. Decomposing motion and content for natural video sequence prediction. In Proceedings of the 2017 International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
8. Hsieh, J. T.; Liu, B.; Huang, D. A.; Li, F. F.; Niebles, J. C. Learning to decompose and disentangle representations for video prediction. In Proceedings of the 2018 Conference on Neural Information Processing Systems, Montréal, Canada, 2–8 December 2018; pp. 517–526.

9. Park, S. H.; Lee, K.; Yoon, K. J. Robust online multiple object tracking based on the confidence-based relative motion network and correlation filter. In Proceedings of the 2016 IEEE International Conference on Image Processing, Phoenix, AZ, USA, 25–28 September 2016; pp. 3484–3488.
10. Simon M.; Milz S.; Amende K.; Gross, H. M. Complex-YOLO: An Euler-Region-Proposal for Real-Time 3D Object Detection on Point Clouds. In Proceedings of the 2018 European Conference on Computer Vision, Munich, Germany, 8–14 Septermber 2018; pp. 197–209.
11. Beltrán, J.; Guindel, C.; Moreno, F. M.; Cruzado, D.; Garcia, F.; De La Escalera, A. Birdnet: a 3d object detection framework from lidar information. In Proceedings of the 2018 International Conference on Intelligent Transportation Systems, Maui, HI, USA, 4–7 November 2018; pp. 3517–3523.
12. Li, B. 3d fully convolutional network for vehicle detection in point cloud. In Procceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vancouver, Canada, 24–28 September 2017; pp. 1513–1518.
13. Engelcke, M.; Rao, D.; Wang, D. Z.; Tong, C. H.; Posner, I. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation, Singapore, 29 May–3 June 2017; pp. 1355–1361.
14. Zhou, Y.; Tuzel, O. Voxelnet: End-to-end learning for point cloud based 3d object detection. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4490–4499.
15. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3d object detection network for autonomous driving. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Singapore, 29 May –3 June 2017; pp. 1907–1915.
16. Ku, J.; Mozifian, M.; Lee, J.; Harakeh, A.; Waslander, S. L. Joint 3d proposal generation and object detection from view aggregation. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, 1–5 October 2018; pp. 1–8.
17. Butt, A. A.; Collins, R. T. Multi-target tracking by Lagrangian relaxation to min-cost network flow. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1846–1853.
18. Kuo, C. H.; Huang, C.; Nevatia, R. Multi-target tracking by online learned discriminative appearance models. In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 685–692.
19. Bae, S. H.; Yoon, K. J. Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June 2014; pp. 1218–1225.
20. Hong Yoon, J.; Lee, C. R.; Yang, M. H.; Yoon, K. J. Online multi-object tracking via structural constraint event aggregation. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 1392–1400.
21. Leal-Taixé, L.; Canton-Ferrer, C.; Schindler, K. Learning by tracking: siamese cnn for robust target association. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 33–40.
22. Tang, S.; Andres, B.; Andriluka, M.; Schiele, B. Multi-person tracking by multicut and deep matching. In Proceedings of the 2016 European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 100–111.
23. Xiang, Y.; Alahi, A.; Savarese, S. Learning to track: online multiobject tracking by decision making. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 4705–4713.
24. Sivaraman, S.; Morris, B.; Trivedi, M. M. Learning multi-lane trajectories using vehicle-based vision. In Proceedings of the 2011 IEEE International Conference on Computer Vision Workshops, Barcelona, Spain, 6–13 November 2011; pp. 2070–2076.
25. Dueholm, J. V.; Kristoffersen, M. S.; Satzoda, R. K.; Moeslund, T. B.;Trivedi, M. M. Trajectories and maneuvers of surrounding vehicles with panoramic camera arrays. *IEEE Trans. Intell. Veh.* **2016**, *1(2)*, 203–214, doi:10.1109/TIV.2016.2622921 .

doi:10.20944/preprints201908.0161.v1

26. Casapietra, E.; Weisswange, T. H.; Goerick, C.; Kummert, F.; Fritsch, J. Building a probabilistic grid-based road representation from direct and indirect visual cues. In Proceedings of the 2015 IEEE Intelligent Vehicles Symposium, Seoul, Korea, 28 June–1 July 2015; pp. 273–279.
27. Reichardt, D.; Shick, J. Collision avoidance in dynamic environments applied to autonomous vehicle guidance on the motorway. In Proceedings of the 1994 Intelligent Vehicles Symposium, Paris, France, 24–26 October 1994; pp. 74–78.
28. Wolf, M. T.; Burdick, J. W. Artificial potential functions for highway driving with collision avoidance. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 3731–3736.
29. Kim, K.; Kim, B.; Lee, K.; Ko, B.; Yi, K. Design of integrated risk management-based dynamic driving control of automated vehicles. *IEEE Intell. Transp. Syst. Mag.* **2017**, *9(1)*, 57–73, doi:10.1109/MITS.2016.2580714.
30. Wang, J.; Wu, J.; Zheng, X.; D. Ni; Li, K. Driving safety field theory modeling and its application in pre-collision warning system. *Transp. Res. Part C Emerg. Technol.* **2016**, *72*, 306–324, doi:10.1016/j.trc.2016.10.003.
31. Zhu, L.; Li, X.; Lu, W.; Zhang,Y. J. A field-based representation of surrounding vehicle motion from a monocular camera. InProceedings of the 2018 IEEE Intelligent Vehicles Symposium, Changshu, Suzhou, China, 26 June–1 July 2018; pp. 1761–1766.
32. Shi, S.; Wang, X.; Li, H. Pointrcnn: 3d object proposal generation and detection from point cloud. In Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16 June–20 June 2019; pp. 770–779.
33. Weng, X.; Kitani, K. A baseline for 3D multi-object tracking. *arXiv* **2019**, arXiv:1907.03961.
34. Kuhn, H. W. The Hungarian method for the assignment problem. *Nav. Res. Logist. Q.* **1955**, *2(1-2)*, 83–97, doi:10.1002/nav.3800020109.
35. Stam, J. Stable Fluids. In Proceedings of the 1999 ACM Siggraph Computer Graphics, New York, NY, USA, July 1999; vol. 99, 121–128.
36. Chorin, A. J.; Marsden, J. E. A mathematical introduction to fluid mechanics. In *Texts in Applied Mathematics 4. Sencond Edition*; Publishing Hourse: Springer, New York, USA, 1990.
37. Bhatia, H.; Norgard, G.; Pascucci, V.; Bremer, P. T. The Helmholtz-Hodge decomposition–a survey. *IEEE Trans. Vis. Comput. Graphics* **2012**, *19(8)*, 1386–1404, doi:10.1109/TVCG.2012.316.