TRAJECTORY EXTRACTION FOR ANALYSIS OF UNSAFE DRIVING BEHAVIOUR

C. Koetsier[1], S. Busch[1], M. Sester[1]

[1] Institute of Cartography and Geoinformatics, Leibniz University Hannover, Germany
(koetsier, busch, sester)@ikg.uni-hannover.de

KEY WORDS: Trajectory Extraction, Deep Learning, Surveillance Video Analysis, Trajectory Analysis

ABSTRACT:

The environment of the vehicle can significantly influence the driving situation. Which conditions lead to unsafe driving behaviour is not always clear, also not to a human driver, as the causes might be unconscious, and thus cannot be revealed by expert interviews. Therefore, it is important to investigate how such situations can be reliably detected, and then search for their triggers. It is conceivable that such insecure situations (e.g. near-accidents, U-turns, avoiding obstacles) are reflected, for example, as anomalies in the movement trajectories of road users.
Collecting real world traffic data in driving studies is very time consuming and expensive. However, a lot of roads or public areas are already monitored with video cameras. In addition, nowadays more and more of such video data is made publicly available over the internet so that the amount of free video data is increasing. This research will exploit the use of such kind of opportunistic VGI. In the paper the first step of an automatic analysis are presented, namely: to introduce a real time processing pipeline to extract road user trajectories from surveillance video data.

1. INTRODUCTION

Getting insight into critical driving maneuvers is an important prerequisite for improving the safety in traffic. As a matter of fact, the environment of a driver has a big influence on the driving behaviour. Such information ranges from the static environment in terms of lanes, the infrastructure, but also the dynamic environment in terms of other traffic participants, changing conditions (such as weather). The knowledge about this kind of information is important for human drivers, but even more so for autonomous cars. For autonomous vehicles this information about the surrounding has to be highly accurate and current to directly interpret and evaluate the surrounding, measured by sensors. The richer the information is, the better a vehicle can judge the situation, predict next steps and react. The surrounding of the vehicle can significantly influence the driving situation: e.g. obstacles, limited visibility due to vegetation, mix of many traffic participants (cyclists, pedestrians, car drivers), busy situation on street due to event (e.g. in front of cinema). Which conditions lead to unsafe driving behaviour is not always clear - even more so, as they might be unconscious, and thus cannot be revealed by expert interviews. Thus the idea of so-called Naturalistic Driving Studies (NDS) is to capture normal traffic situations with a set of different sensors, e.g. cameras inside and outside the car (Campbell, 2012). Such sensor data have the potential to detect critical situation - and once they are detected, it is possible to infer their triggers in the data as well. It is conceivable that such unsafe situations (e.g. near-accidents, U-turns, avoiding obstacles) are reflected, for example, as anomalies in the movement trajectories of road users. Thus, the idea in this paper is to observe the behaviour of traffic participants in terms of movement trajectories and analyze it.

Collecting real world traffic data in driving studies (e.g. (Barnard et al., 2016)) is very time consuming and expensive. Also, NDS can suffer from the problem that the user behaviour may be influenced by the knowledge of being observed. On the other hand, a lot of roads or public areas are monitored with video cameras today. Also, more and more of such video data is made publicly available over the internet so that the amount of free video data is increasing. The disadvantage of this kind of data is that it typically was not acquired for this purpose, hence, the setup of the cameras is not optimized, nor is the quality of the images. Thus, this kind of data can be considered as so-called opportunistic Volunteered Geographic Information (VGI) (Goodchild, 2007). One challenge of this research is therefore to handle and exploit such data. The long term goal of this research is to provide a mechanism to extract potential critical situations (both spatio-temporally, but also abstract in terms of events and relationships) from observed traffic trajectories. Critical situations can on the one hand be spatio-temporal, e.g. a dangerous maneuver of a car triggered by a UPS-van parking in the street; on the other hand also a generic relationship, e.g. a situation, which occurs when many people wait at a traffic light (at any place and time). In this paper, the first steps of this research are presented, namely: (a) to introduce a real time processing pipeline to extract road user trajectories from surveillance video data and (b) to sketch possible ways to analyze the trajectories with respect to anomalies.

2. RELATED WORK

In order to capture road users trajectories, different measurement systems can be used, ranging from Floating Car Data, via sensors in road-side infrastructure towards sensors mounted on aerial vehicles (see e.g. (Krajewski et al., 2018). Whereas the advantage of aerial sensors is that they provide mostly occlusion free trajectories with low or no perspective distortion, such approaches cannot be permanently operated. Thus, the framework presented in this paper relies on object detection in (mono) images from surveillance cameras. Such cameras are widely available; due to their long-term operation, they are also able to track and observe seasonal changes in the environment. The use of mono-cameras, however, has to take distortions due to deviation of the objects from an assumed plane into account.

After video streams have been captured, the moving objects

(traffic participants) have to be interpreted in the images. With increasing computational power in the last years also the performance of object detection in terms of classification accuracy and run-time increased.

Deep Learning approaches for detecting object bounding boxes in images like SSD (Liu et al., 2016), Faster R-CNN (Ren et al., 2015) and YOLO (Redmon et al., 2016) outperform classic approaches like optical flow, background subtraction or a sliding window feature classifier. In this work we chose YOLO as object detector since our goal is to create a real time processing pipeline and YOLO is currently the fastest detector while still providing a high detection accuracy. YOLO handles the object detection as a regression problem to spatially separated bounding boxes with associated class probabilities. The prediction happens in a single evaluation for a given input image. To connect the detections of each video frame to a trajectory, filters like a Kalman filter (Kalman, 1960) or a particle filter (Smith, 2013) can be used.

There are also networks, like Mask R-CNN (He et al., 2017), for not only detecting bounding boxes of an object, but already segmenting the image. While increasing the object detection accuracy in comparison to bounding box detectors they have a higher run-time. Furthermore, beside the separation of detection and tracking also neural networks exists which directly tracking generic objects based on pretrained features like Re3 (Gordon et al., 2018).

There are many approaches which try to analyze behaviour from given trajectories. An example is the inference of intesion of traffic participants (Varytimidis et al., 2018); another is the prediction of future movements taking the other traffic participants into account - which is accomplished using an LSTM-neural network (Cheng & Sester, 2018). The underlying idea of the approach in this paper is to define an unsafe situation as one, which is different from the normal situation, i.e. can be considered as an anomaly. To this end, a Hidden Markov Approach to describe components of anomalous behaviour can be applied (Huang et al., 2014); this model is able to detect traffic situations such as route repetitions or U-turns. In the domain of trajectory analysis several methods have been proposed to group trajectories in order to find mean values (e.g. (Ester et al., 1996, Kuntzsch et al., 2016, Ahmed et al., 2015)), and then determine deviations thereof as anomalies. A survey on anomaly detection in trajectories is given by (Kumaran et al., 2019).

3. METHOD

For the surveillance camera pipeline we decided to use detection based tracking with a filter over direct tracking networks such as Re3. The reason is, we assume a higher trajectory precision as well as a faster run-time due to the fact, that we can add knowledge by modelling the general movement behaviour of the road users with a filter and do not need a high frame rate because of the filter prediction steps. In addition this setup is more flexible and different parts can be optimized later on.

In general the surveillance camera pipeline works as follows (see Figure 1): for each frame of the specified input video stream an object detection, using YOLO, will be performed, which returns a list of bounding boxes for all found objects (specified road user types) in the scene. For each found road user and their determined bounding boxes the center ground point will be estimated. Using a precomputed homography matrix for the

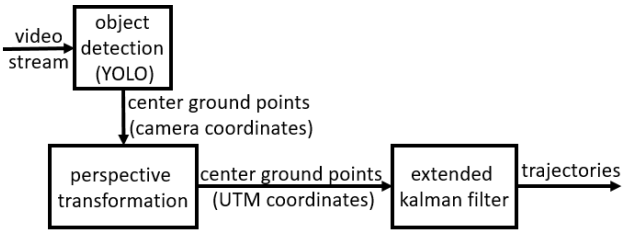

Figure 1. Surveillance camera pipeline structure

concrete scene this center ground point in the camera coordinates is projected to a global coordinate system. A extended Kalman filter is used to merge the detected road user positions. In this architecture, the detector and trajectory calculation are separated, while still running in real time. The detector reviews raw video frames and outputs bounding boxes with classifying annotations. While the detector only looks at each frame once, the trajectory calculation uses information over a specified amount of frames to smooth out variance in the detection and do a plausibility check to prevent false tracking and reject false detections. In the end the output trajectories can be further post processed and analyzed.

4. EXPERIMENT

The trajectory analysis was performed on trajectories gathered from the introduced surveillance camera pipeline, which was applied to a video sequence of a webcam live stream of the Main Street in Canmore, Alberta, Canada, hosted on YouTube (Alberta, 2019), for a timespan of five hours on the 26.01.2019 from 08:00 am to 01:00 pm. A sample image from the scene of the selected surveillance camera can be seen in Figure 2. The chosen video stream is available 24 hours a day, 7 days a week, which makes it interesting for long therm analyzes. It also allows to make investigations as to the quality of the trajectory tracking depending on the daytime. The video quality with a resolution of 1280 x 720 pixel and 30 frames per second is decent.

Since the surveillance camera pipeline should run with any input video stream, we assume that the video source can not be calibrated and thus the camera parameters are not known. Because of this the pipeline needs a precomputed homography matrix for each different scene. If the video stream is recorded in a controlled environment the camera could be properly calibrated and reference points in the camera image determined precisely.

The homography matrix can be computed by manually determining corresponding point pairs in the camera image and a georeferenced orthophoto. For the chosen webcam live stream of the Main Street in Canmore, Alberta, the city itself provides an free available orthophoto with a pixel resolution of 7 cm (Alberta, 2017). The scene contains good to map structures such as zebra crossing, other road markings, quadratic trash bins and trees. In this scene we selected 25 point pairs, which can be seen in Figures 2 and 3. The residuals in the ground control points have a mean value of 49,5 cm. Thus, this is the positional accuracy, which can be optimally achieved.

The road users are characterized by bounding boxes parallel to the image coordinate system. As only mono images are analyzed, no depth information is available. Therefore it is difficult to identify a suitable reference point of the objects. As a

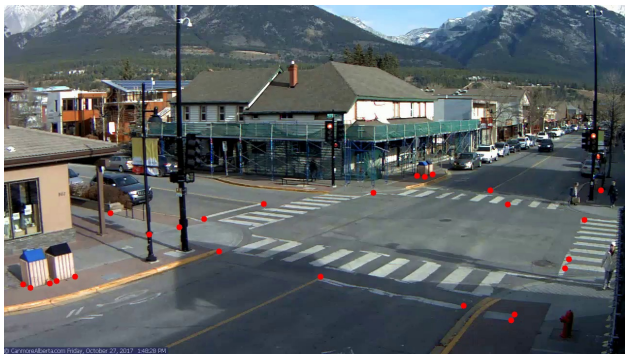

Figure 2. Homography points of the camera image

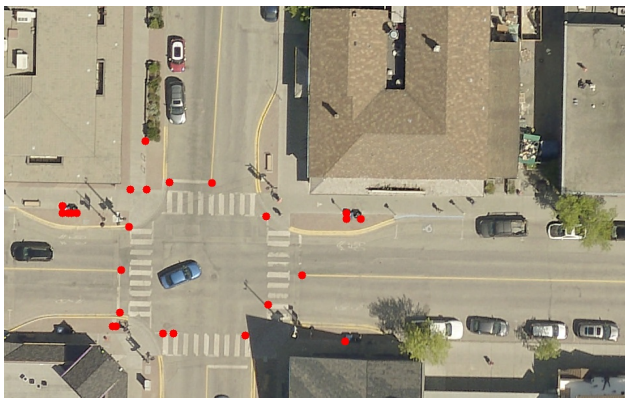

Figure 3. Homography points of the aerial image

simplification, the bounding boxes bottom center is used. This ensures that the point lies on the assumed road plane. Later, an analysis of the displacements induced by these assumptions is given. The homography is applied to each estimated center ground point of the detected road users for each frame.

The road users bounding boxes for each frame are detected by YOLO with the provided pretrained network weights and configurations by the YOLO authors (Redmon et al., 2018). The given network configuration of YOLO is able to detect 80 different classes. Since this work deals with the tracking of road users the detection results are filtered and only detections of the classes person, bicycle, motorbike, car, truck, bus are used.

The projected road user locations are aggregated to trajectories by use of an extended Kalman filter with the assumption of constant acceleration and yaw rate as well as the use of the bicycle model (see (Wang & Qi, 2001)). To eliminate situations where the detection failed for a longer period and the filter was not able to connect the detections properly all trajectories smaller than 15 meters were removed to have meaningful trajectories.

Subsequently, the trajectories were clustered with help of DBSCAN (Ester et al., 1996), in order to find suitable paths that road users usually take.

5. RESULTS & DISCUSSION

With the previously described method we were able to achieve the results, which can be seen in the following figures. First, some individual trajectories of vehicles and pedestrians are shown. Figure 4 visualized the tracks of cars turning left. It can be seen, that the assumption of the reference point being the bounding boxes lower center lead to systematic displacements of the trajectory towards the observing camera. These displacements are in the range of one to three meters, depending on the dimension and the orientation of the object.

Figure 5 shows that individual pedestrians can be tracked over large extents; it also shows the inaccuracies which are induced by assuming that the lowest point of the bounding box is always situated on the ground. This assumption seemingly violated, when people are walking and thus the lowest point is sometimes erroneously assigned to the lifted foot.

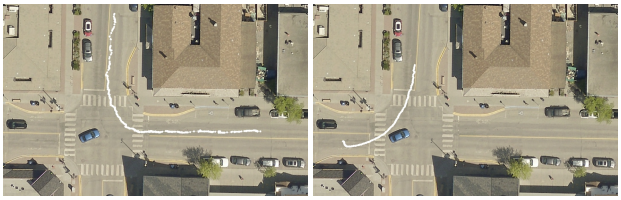

Figure 4. Individual vehicle trajectories

Figure 5. Individual pedestrian trajectories

Figure 6. Aggregated vehicle trajectories

By looking at the trajectory visualization of the vehicles (see Figure 6), it can be seen, that the crossing with its lanes as well as the turn maneuvers can be recognized. On the left hand side the trajectories suddenly stop and moving diagonally upwards. This pattern is caused by vehicles driving out of the camera view but are still detected while partially visible. On the right hand side trajectories leave the road to the bottom or join from the top. This vehicles join the lane form or leave the lane for a parking lot.

By looking at the trajectory visualization of the pedestrians (see Figure 7) it can be seen that the sidewalks and zebra crossing can be recognized. Furthermore some people do not follow the rules and illegally cross the road. This is the reason why some trajectories of pedestrians are on places of the road where they should not be.

Figure 7. Aggregated pedestrian trajectories

It can be observed, that the vehicle trajectories are displaced and are shifted depending on the location in respect to the camera, whereas the trajectories of the pedestrians are generally more accurate. This result can be explained by the choice of the center ground point. Persons in comparison to cars have a relatively small and quadratic stand space. That is why the center ground point for persons can be set as the bounding box bottoms center independently from the orientation of a person in respect to the camera (see Figure 8). This is not true for vehicles. Depending on the orientation of a vehicle in respect to the camera the bounding box bottoms center reflects different points of the vehicle (see Figure 9).

Figure 8. Pedestrians center ground points

Figure 9. Vehicles center ground points

The reason for the this is view angle of the camera in respect to the scenes ground plane. General speaking this angle can be any from 0°, parallel to the ground (side view), to 90°, orthogonal to the ground (top view). The smaller the angle the bigger the problem of determining the right center ground point.

To elaborate the approximate impact of the chosen center ground point for vehicles the following task was performed (see Figure 10): the blue car in the middle of the presented intersections aerial image was chosen as sample vehicle. For this car a bounding box with a width of 175 cm and length of 495 cm was manually determined and projected into the camera view. In the camera view the center ground point of the the projected car was detected as before as the bounding boxes center bottom. This selected detection center ground point in the camera image was than projected back into the georeferenced aerial image and compared with the original (ground truth) center point of the blue cars bounding box. This difference was calculated for each rotation of the blue car between 0°and 360°in 5°steps around the cars center on the three marked positions. For the left position the minimal error was found to be 85 cm at 55°, the maximum error 271 cm at 350°and the average error 214 cm. For the mid position the minimal error was found to be 88 cm, the maximum error 280 cm and the average error 220 cm. For the right position the minimal error was found to be 165 cm, the maximum error 340 cm and the average error 289 cm. That shows, that the inaccuracy of the vehicles center ground point increases the more far away a vehicle is from the camera.

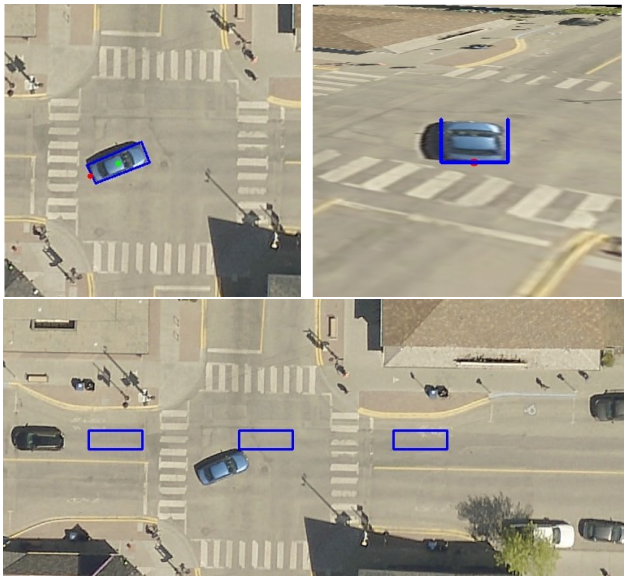

Figure 10. Error estimation

If the video stream would be recorded from out the air, for example by a drone, to a have a bird eye view like in Figure 3, determining the right center ground point would be much easier in comparison to a side view of a scene as in our image. In this case the center of a detected road users bounding box could be chosen as the center ground point.

As described above for pedestrians the center ground point can be chosen as the detected bounding boxes bottom center but their trajectories are affected by another problem. A known issue by YOLO is that small objects appearing in groups can not be detected accurately. In the chosen test video sequence this phenomenon was observed with pedestrians. It happened that groups of pedestrians were sometimes detected as a single pedestrian, sometimes each pedestrian was detected correctly separately. This behaviour changed for consecutive video frames (see Figure 11). For small groups of pedestrians up to three persons this is not a problem because due to a high frame rate there exist enough detections for each single pedestrian.

https://doi.org/10.5194/isprs-archives-XLII-2-W13-1573-2019

For larger groups the trajectory of a single pedestrian can be split or even not existing.

Figure 11. Pedestrian group detection

Another problem which decreases the trajectory accuracy is the changing size of the detection bounding boxes for an object, which results in jumping center ground points for an object. This effect is handled by the extended Kalman filter, which smooths the trajectories.

While the detection of road users using YOLO with the provided pretrained network weights by the YOLOs authors works most of the time reliably and accurately, sometimes the network fails in cases of occlusions. An example is shown in Figure 12, where vehicles are occluded by the house, a traffic light and a lantern.

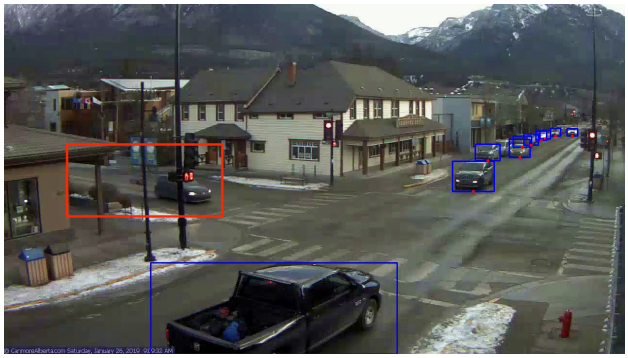

Figure 12. Problematic detection area (red)

Figure 13. Vehicle trajectory clusters

To analyze the trajectories, they were clustered with help of DBSCAN (epsilon=7, min_points=4, Frchet distance). In Figure 13 it can be seen, that each travel direction as well as the turn maneuvers are separated nicely into different clusters. Because of above discussed inaccuracies of the trajectories as well as trajectory separation resulting from failed detections e.g. due to occlusions, it happens, that one class is represented by multiple clusters and thus have to be merged in a subsequent step.

We also performed a small analysis on how the light conditions impact the detection and in consequence the trajectory accuracy. Therefor three different video sequences representing different light conditions were chosen. To represent the light condition as numeric value, the video frames of the chosen sequences were converted to gray scale images and the mean gray value was determined. As accuracy metric we chose the number of correct detections in respect to the number of total frames for a vehicle between the outer borders of the zebra crossing. Table 1 shows the average over ten cars per lane going straight. Since the scene has several light sources like shop windows, street lamps and traffic lights, which can be reflected by vehicles, the vehicles were divided into two groups. Light-colored vehicles including white and silver cars and dark-colored vehicles including black and dark blue cars. Beside the decreasing detection rate with darker light conditions, also the detected bounding box size varies more. It is interesting to note that even at dusk still a high detection rate can be achieved.

light conditions	vehicle color	
(mean gray scale)	light-colored	dark-colored
daylight (98.36)	100 %	100 %
dusk (48.00)	91.40%	88.33%
darkness (32.33)	43.12%	13.87%

Table 1. Vehicle detection accuracy under different light conditions

6. CONCLUSION & OUTLOOK

We showed that with the above described method it is possible to extract road user trajectories from a single surveillance video stream in real time with at least 30 frames per second. It was shown that the detection of road users was reliable - even in situations with poor light conditions. Due to discussed challenges the trajectory accuracy is in the range of meters.

For now the pipeline is suitable to extract meta information like from where to where how many cars are going, how the traffic flow changes over time and the detection of significant trajectory changes like u-turns and leaving the lane for a parking lot. For a more detailed trajectory analysis to investigate the driving behaviour the trajectory accuracy needs to be increased.

To increase the detection accuracy and also estimate the center ground point better, several improvements are planned: the next step is to try to segment the detected road users precisely, e.g. with the help of Mask R-CNN. In this way, the orientation of the vehicles can be determined and thus the center point can be better estimated. Another promising idea to achieve this is to try to learn the orientation or the displacement from the true centerpoint in a deep learning framework. To this end, a benchmark dataset with labelled ground truth trajectories is needed. This benchmark data will also allow to quantitatively evaluate the trajectory accuracy.

Once the surveillance camera pipeline is improved, it would be interesting to see how the pipeline performs against deep learning networks like Re3. Further investigations will focus on how the camera resolution as well as the recorded frames

per second affect the trajectory accuracy and how to find and trade-off between run-time and trajectory accuracy.

Up to now, the critical situations have been analyzed only by visual inspection. The next major step is to devise automatic analysis methods to identify those situations automatically. In a first step we will use clustering of the majority of the trajectories and identifying the outliers as critical situations. As in our approach we have the trajectories of all traffic participants (and even the video streams), we are then able to inspect these situations in the original data and try to infer the underlying reasons. This could, e.g. be another traffic participant, forcing a car to stop; or it could also be due to weather conditions, or the time of the day – all of this information is available in the original data and thus is accessible for automatic analysis.

REFERENCES

Ahmed, M., Karagiorgou, S., Pfoser, D., Wenk, C., 2015. A comparison and evaluation of map construction algorithms using vehicle tracking data. *GeoInformatica*, 19, 601–632.

Alberta, C., 2017. Canmore 2017 orthophoto - 7cm sid. `http://opendata-canmore.opendata.arcgis.com/datasets/143e49da51014228853bf09f9e760ae8`. Accessed: 2019-04-01.

Alberta, C., 2019. Canmore live webcam "main street". `https://canmorealberta.com/webcams/main-street`. Accessed: 2019-04-01.

Barnard, Y., Utesch, F., van Nes, N., Eenink, R., Baumann, M., 2016. The study design of UDRIVE: the naturalistic driving study across Europe for cars, trucks and scooters. *European Transport Research Review*, 8, 14.

Campbell, K. L., 2012. The SHRP 2 naturalistic driving study: Addressing driver performance and behavior in traffic safety. *Tr News*.

Cheng, H., Sester, M., 2018. Modeling mixed traffic in shared space using lstm with probability density mapping. *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 3898–3904.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X. et al., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd*, 96number 34, 226–231.

Goodchild, M. F., 2007. Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69, 211-221. http://www.springerlink.com/content/h013jk125081j628.

Gordon, D., Farhadi, A., Fox, D., 2018. Re3: Re al-Time Recurrent Regression Networks for Visual Tracking of Generic Objects. *IEEE Robotics and Automation Letters*, 3, 788–795.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn. *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

Huang, H., Zhang, L., Sester, M., 2014. A recursive bayesian filter for anomalous behavior detection in trajectory data. *Connecting a Digital Europe Through Location and Place*, Springer, 91–104.

Kalman, R. E., 1960. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82, 35–45.

Krajewski, R., Bock, J., Kloeker, L., Eckstein, L., 2018. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2118–2125.

Kumaran, S. K., Dogra, D. P., Roy, P. P., 2019. Anomaly Detection in Road Traffic Using Visual Surveillance: A Survey. *arXiv preprint arXiv:1901.08292*.

Kuntzsch, C., Sester, M., Brenner, C., 2016. Generative models for road network reconstruction. *International Journal of Geographical Information Science*, 30, 1012–1039.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A. C., 2016. Ssd: Single shot multibox detector. *European conference on computer vision*, Springer, 21–37.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2018. You only look once network weights v3. `https://pjreddie.com/media/files/yolov3.weights`. Accessed: 2019-04-01.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 91–99.

Smith, A., 2013. *Sequential Monte Carlo methods in practice*. Springer Science & Business Media.

Varytimidis, D., Alonso-Fernandez, F., Duran, B., Englund, C., 2018. Action and intention recognition of pedestrians in urban traffic. *arXiv preprint arXiv:1810.09805*.

Wang, D., Qi, F., 2001. Trajectory planning for a four-wheel-steering vehicle. *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, 4, IEEE, 3320–3325.