Mitigating autonomous vehicle accidents through sensor fusion

Aarush Sharma
West Windsor-Plainsboro High School North

Abstract

One of the most rapidly developing sectors of transportation is autonomous vehicle technology known as Advanced driver-assistance systems (ADAS). In some cars, this emerges as a simple feature, such as pedestrian detection for quick braking, while in others, as a full autonomous algorithm, such as driving a vehicle autonomously using cruise control, various sensor inputs, and machine learning algorithms to better acclimate it to different situations. Most techniques today rely on relative positioning and sensor inputs for controlling different parts of the vehicle itself. Take a car as an example: a car could have a camera on the front to detect objects ahead of the car's path, or sensors on the side to detect if it is too close to a curb. Inputs to the car's system such as these are optimal for driving on main roads and highways, and can aid the driver's performance on the road. However, fusing relative inputs with absolute localization can allow for smoother movements, accurate real-time positioning, and can provide accident prevention through GPS absolute localization. Through this paper, methods of Kalman-filtering absolute and relative forms of input are explored to improve the overall precision of localization, and the results of increased accuracy and less noise in the data after the Kalman filter shows the applications of this algorithm in autonomous vehicles.

1. Introduction

Autonomous vehicles are a large step forward in automation and the reliance of data through sensors. While it has various benefits from an environmental and comfort standpoint, the precision from sensors and automated tracking could help avoid a potential safety hazard for passengers. In fact, according to the United States Department of Transportation, the average increase of the number of fatalities from car accidents has increased by 7.2% despite people driving less during the pandemic. The summary furthers the argument about the necessity of better road safety by illustrating the $19 billion in funding to improve the general road safety, much of which is being invested in autonomous vehicles. From general technologies such as software tracking to the actual hardware behind creating better cameras, the investment in autonomous vehicles is one that is vital and would help in better road safety as a whole.

There are two ways of localizing a vehicle: one using absolute coordinates relative to a fixed reference frame (like the case of GPS), and the other being relative to itself (such as with onboard sensing through camera, lidar, or radar inputs). Relative input mapping and localization is one of the most widely used forms of controlling an autonomous vehicle in practice. It provides data on how far a vehicle is from other objects in close proximity and allows it to control inputs such as heading, speed, and position on a road. However, occasionally having slower update times and worse overall precision can lead to inaccuracies on the road. This is where absolute localization is useful and can be done through an array of techniques. Instead of relying on just sensor input, absolute localization works off of displacement from a known location in 3D space. An example of this would be moving a toy car and tracking the number of rotations of the wheels to determine how far forward or back the car is from its starting position. Given that this form of localization is completely independent of surroundings and doesn't need more than displacement to calculate the overall position of a vehicle in 3D space, it can lead to greater accuracy and faster speeds when reading position, which can be better for accident prevention and can increase the safety for drivers.

Some common techniques of this form of localization include but are not limited to wheel based odometry, visual SLAM, and for vehicles, GPS tracking devices that are already built into most mainstream cars today. Raw wheel odometry calculates a displacement of position relative to a fixed starting point based on the number of rotations of a car wheel. This system tends to drift over time because of how it moves non-linearly due to the slippage of the wheels in parallel and perpendicular directions making it hard to estimate a fully accurate position. Visual SLAM relies on a similar technology, but instead of mapping position based on wheel rotations, it uses information from cameras and inertial measurement units to find the accurate displacement from a relative start position. VSLAM tends to work great for short distances, but has drift in displacement over time, as the environment and overall readings vary as a vehicle gets further from known locations. That leaves the GPS system which already is used in most cars. A global positioning system is notorious for having minimal drift over time or travel, but often lacks precision under 30-200cm. This furthers the argument that local sensing is a recommended tool for precise positioning (parking, obstacle avoidance, lane keeping, etc.), but a GPS would help in correcting for drift in a system over time and be used for higher-level decision making, such as knowing which street an accident is on. If given the location of the car on a map, the exact location can be used to create a safer driving environment as a whole.

For example, with accident prevention, a GPS system could automatically slow down your vehicle's speed based on knowing an accident at position X_f and current position and X_i. This would create more safety, especially at high speed roads in harsh weather conditions, where driver's could benefit from an automated system such as this one. Another benefit of GPS location is the advantages it has when used in conjunction with relative feedback from cameras and external sensors. Using the global xy-coordinates of the vehicle on the road, a fully automated system can be used to control the car with better precision and accuracy, while also simultaneously allowing for more overall safety. The issue with relative feedback is that it doesn't work great in roads without defined key points for sensors such as cameras to identify and track in real time. These key points would include road signs, traffic lights, lane markings, and other identifying information for any intelligent car system.

2. Defining the system

The aim of this paper is to compare the benefits of relative, absolute, or combined localization in an effort to create an optimal solution to automated driving systems. Relative localization, especially with computer vision or sensor feedback, returns a signal with the result of it being actuated. For example, a touch sensor would return TRUE if being pressed, or FALSE otherwise. Similarly, a distance sensor would return the distance between the sensor and an object in real time. Having multiple sensor inputs provide the same data necessary to make calculations for following a path can increase overall accuracy and provide redundancy in data, creating a fallback in case one sensor fails.

The main problem is combining the sensor inputs in an effective manner. There are various ways to combine inputs such as using a Bayesian network, Particle Filter, Gauss-Newton filter, or other sensor fusion algorithms. The algorithm chosen by this paper is a Kalman filter, which comprises different variations such as plain Kalman filter (KF), Extended Kalman Filter (EKF), or a Unscented Kalman Filter (UKF). A Kalman filter uses various measurements at different intervals of time from multiple inputs and fuses them, often yielding results that are closer to the actual value needed. This process of fusion is normally used when there are doubts about the actual quality of the raw, unfiltered results of the two different sensor inputs, and if implemented correctly, the Kalman filter can provide more reliable data that can be used to increase performance. Furthermore, Kalman Filters that use multiple sensors will never make a given estimate worse than with a single sensor, a theoretical property that is advantageous in the field of autonomous vehicles. While this paper uses a plain Kalman Filter to fuse feedback from the given sensors, an Extended Kalman Filter can be better for nonlinear systems because a plain KF relies on a linear, time-invariant model, and that a (potentially different) linear model is approximated depending on the state of the system through EKFs. Although it requires more computation, a higher level of accuracy can be gained through this method of sensor fusion.

The way the Kalman Filter itself works is by using the feedback from sensor measurements. The equations below model how a basic system would use a Kalman Filter:

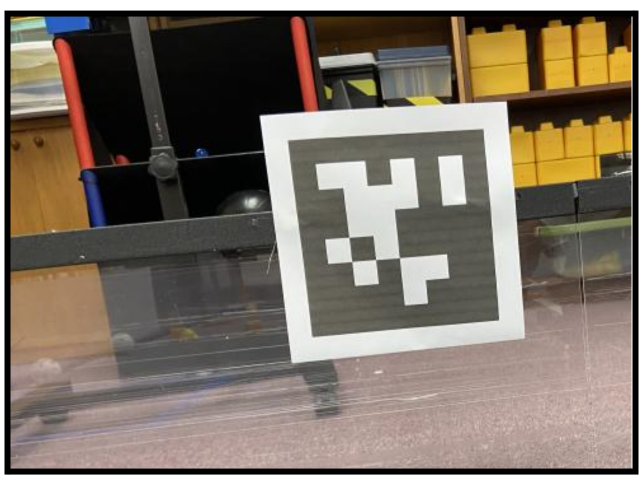
$$x_t = x_{t-1} + u_t$$
$$P_t = P_{t-1} + Q$$
$$K_t = P_t \frac{1}{P_t + R}$$
$$x_t = x_t + K_t(z_t - x_t)$$
$$P_t = (1 - K_t)P_t$$

x_t is defined as the current state estimate
x_{t-1} is defined as the previous state estimate
u_t is defined as the input at a certain time
Q is the model covariance
R is the sensor covariance
P is the initial guess
K is the Kalman Gain

P consists of the two key sensor inputs that are important when dealing with this type of a filtering system. The first sensor input, in the case of this study, is the raw data from the T265 motion tracking camera. This camera uses VSLAM (Visual-Inertial Simultaneous Localization And Mapping) technology combining absolute data from fisheye lens cameras and an embedded inertial measurement unit to give an accurate location from a starting position based on the total displacement of the camera itself. This camera was mounted directly to the robot, and was synced up with the data from the second sensor, which was the regular webcam tracking the marker in real-time.

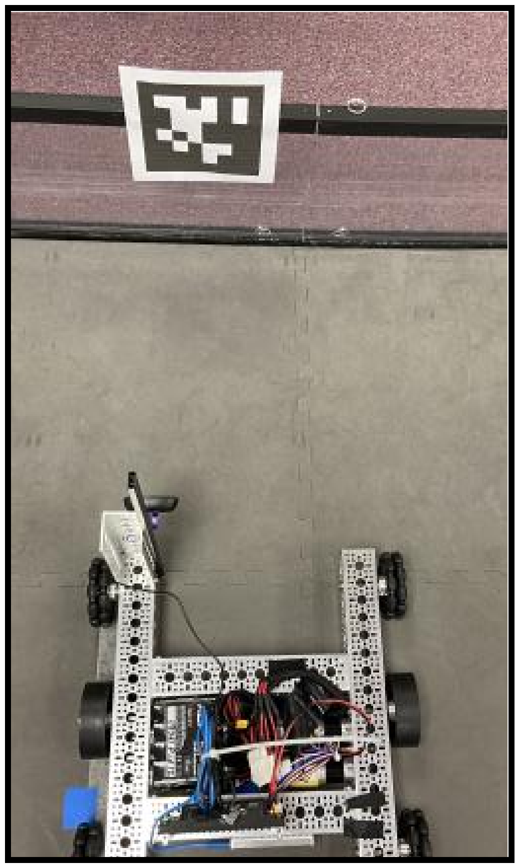
3. Collecting and testing data

To Kalman filter two sensor inputs of data, this paper chose to model the prediction algorithm from a situation in smaller scaled robots. These robots were readily available and are typically found in the FIRST (For Inspiration and Recognition of Science and Technology) Tech Challenge competition and were modified to accompany sensors to record data. The robot in use was a simple drivetrain that could move around the floor, and had two main aspects to aid this project. One was a basic webcam, specifically a Logitech C270 webcam, that allowed for the collection of relative data input (how far the robot was away from an object). The object in play here was a fiducial marker from the 36h11 group, and AprilTag, a computer vision detection algorithm, was used to detect the marker shown in picture 1:



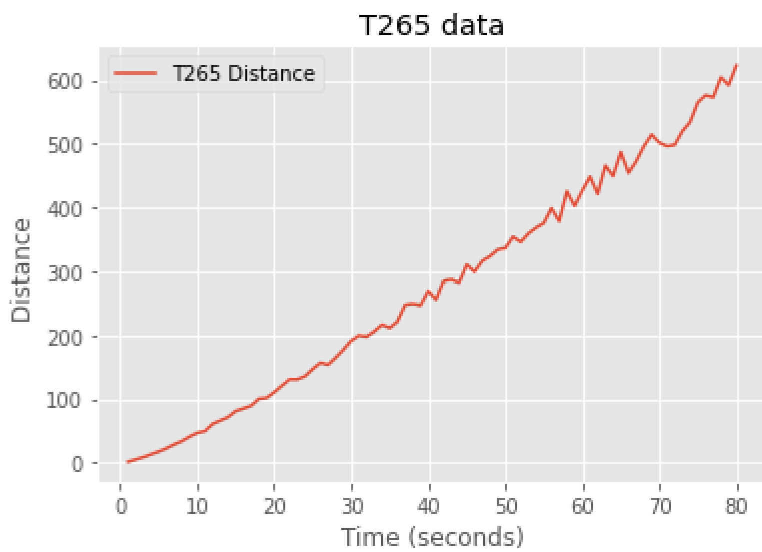
Picture 1 shows a 36h11 group marker stuck onto a perimeter

The fiducial marker serves as a base point to capture data from a basic webcam, as it yields the most precision from a simple computer vision algorithm. The design of the marker allows a computer vision algorithm to tag onto corners, and sense the image pattern in the frame of view, which can also yield a 3d position of an x-coordinate, y-coordinate, and robot rotation at a certain point of time. The robot camera was pointed at the fiducial marker as see in picture 2:

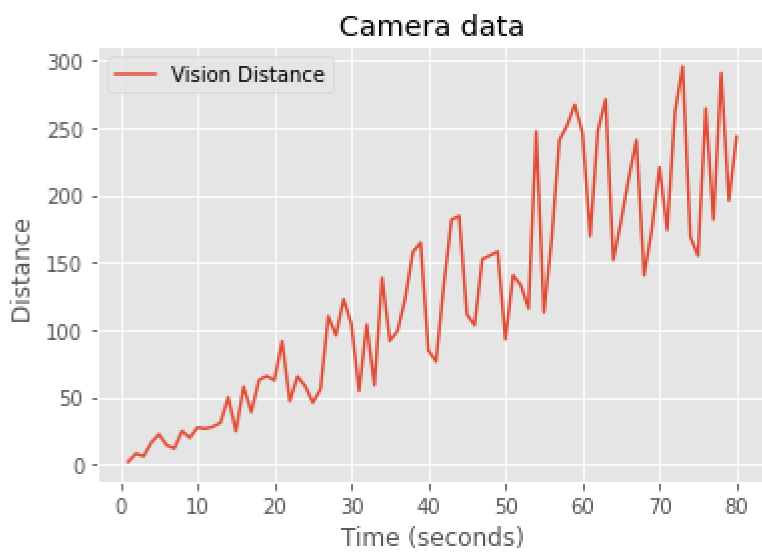


Picture 2 shows camera pointed at fiducial marker

To collect data from a second input, a T265 camera is being used to calculate displacement from a relative start position. This camera, while now discontinued, provides accurate measurements on real-time displacement of the robot through a 3D vector using X-direction, Y-direction, and theta heading. After sampling data 2 times a second for 40 seconds providing a total of 80 real-time measurements of position, it was clear that fusing the inputs and using a Kalman filter created better data outputs to be used.

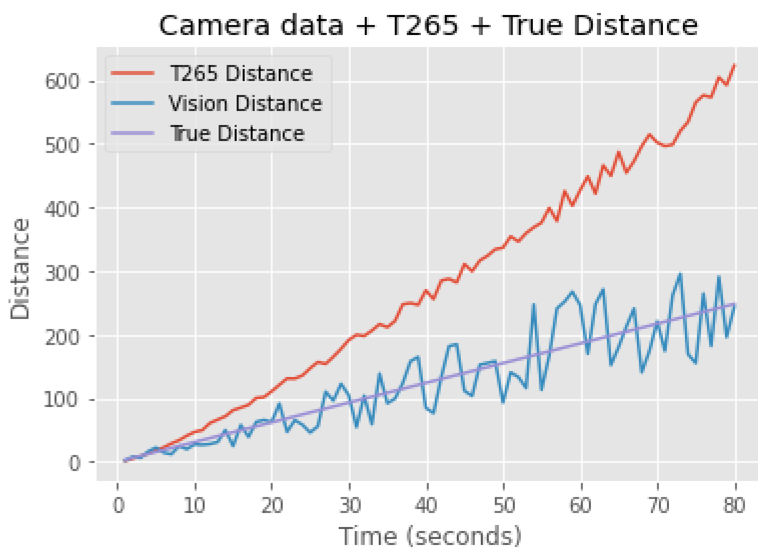


The raw data from the T265 camera shows that it had decent accuracy as distance was at



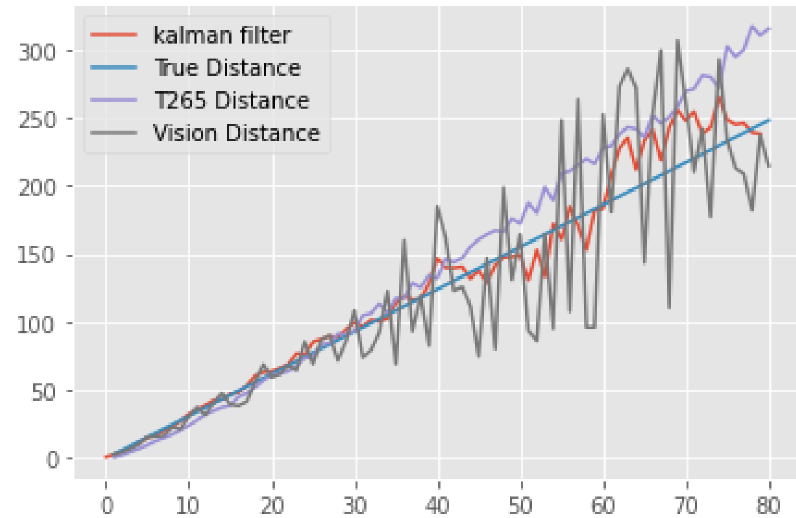
a lower number, and slowly drifted with higher displacement values. While this model does not directly apply to a GPS in a vehicle, it still shows how absolute forms of measurement can drift due to external factors.

The raw data from the camera on the other hand doesn't drift over time or displacement, but rather the noise and the distance has a linear relationship. The noise tends to increase at higher distances because it becomes harder for a computer vision algorithm to detect objects at large distances. This can be seen in the real-world applications where most automated vehicles have a problem in detecting road signs, cars, and other objects at certain distances away from the sensor, but become more clear as they come close.



The graph above is a direct comparison between both inputs to the real data

The Kalman filter takes advantage of the properties of the two different inputs: the low drift provided by the vision input, as well as the low noise values from the T265 input. By applying feedback between the T265 and the vision system, it pushes the estimate closer to the values of the second input (the vision system), and by increasing or decreasing the Kalman gain, the rate of convergence to the second input is changed accordingly. The tuning for the Q and R values in an effort to optimize P was done through an iterative solution finding the best combinations that provided the lowest root mean squared error relative to the true data



Graph with Kalman Filter values shown

4. Conclusion and future insight

By combining two independent absolute and relative control inputs through a Kalman Filter, vehicles can benefit from the increased precision to avoid accidents. This model can be applied to real-world situations with autonomous vehicles, where the accuracy of the sensor is going to increase due to higher industry standards and better control systems. The changes would mainly be in the sensors, where the absolute input would come from data from a GPS, and the relative input would come from sensors included with a vehicle, as seen already in the majority of cars today.

Another way to better this system as a whole is to use an Extended Kalman Filter instead of a regular one. Essentially, this sensor fusing algorithm is a nonlinear version of a regular Kalman Filter, and is commonly used with systems in vehicles. Applying the same mathematical model to fuse these sensor inputs could potentially yield better results because of estimating and calculating inputs non-linearly. Every update, a vehicle doesn't necessarily have to move linearly in space, so estimating the data with a nonlinear system theoretically would create more accurate data points to use in a path-following algorithm.

The application of this data itself doesn't simply end in the realm of localization. Other major components to prevent accidents include robust path-following for autonomous vehicles, reacting to sudden changes in input, filtering out unnecessary data for even more optimal solutions, and merging data with readily available sources. For example, a source talking about a thunderstorm a mile south from where a vehicle is, the system would automatically know to slow down in worse conditions based on the location to the storm calculated by the GPS and it's sensors.

On the other hand, autonomous vehicles provide certain dangers to routine passengers if they fail to work as planned. Ensuring that a system has well-thought out safety measures, and that the inputs for position, velocity, 3d-location through the GPS, as well as the general software controlling the vehicle are all heavily tested in different conditions is crucial to the success of these types of vehicles. For example, while this testing worked in a controlled environment with ample lighting, less external factors than on road, and accurate fiducial markers, the real-world scenarios presented to autonomous vehicle models could cause general issues with the main algorithm. However, with empirical testing and simulating this environment, this issue can be resolved, proving the viability of such vehicles.

5. References

- Media, NHTSA. "2020 Fatality Data Show Increased Traffic Fatalities during Pandemic." NHTSA, NHTSA, 3 June 2021, www.nhtsa.gov/press-releases/2020-fatality-data-show-increased-traffic-fatalities-during-pandemic#:~:text=While%20Americans%20drove%20less%20in,36%2C096%20fatalities%20reported%20in%202019.

- Gutta, Surya. “Benefits of Autonomous Vehicles.” Medium, Geek Culture, 4 Aug. 2021, medium.com/geekculture/benefits-of-autonomous-vehicles-e90ebfd324e.
- (www.kalmanfilter.net), Alex Becker. “Online Kalman Filter Tutorial.” Kalman Filter Tutorial, www.kalmanfilter.net/default.aspx.
- “Tracking Camera T265.” Intel® RealSense™ Depth and Tracking Cameras, 12 May 2021, www.intelrealsense.com/tracking-camera-t265/.
- “AprilTag.” Apriltag, april.eecs.umich.edu/software/apriltag.
- Caunt, Benjamin. CTRL ALT FTC, www.ctrlaltftc.com/.
- Hancock, P. A., et al. “On the Future of Transportation in an ERA of Automated and Autonomous Vehicles.” PNAS, National Academy of Sciences, 16 Apr. 2019, www.pnas.org/content/116/16/7684.
- Kaur, Kanwaldeep, and Giselle Rampersad. “Trust in Driverless Cars: Investigating Key Factors Influencing the Adoption of Driverless Cars.” Journal of Engineering and Technology Management, Elsevier, 3 May 2018, www.sciencedirect.com/science/article/pii/S0923474817304253.