Preprints, 10th IFAC International Symposium on Advances in Automotive Control
Columbus, Ohio, USA, August 28-30, 2022

Entropy Based Metric to Assess the Accuracy of PNT Information ⋆

M. Saim * U. Ozguner *

* *Ohio State University, Center for Automotive Research, Columbus,OH 43212 USA (e-mail: saim.1@osu.edu, ozguner.1@osu.edu).*

Abstract: Entropy measures uncertainty present within the data. Highly Automated Vehicles (HAVs) can navigate safely and efficiently if location information of occluded dynamic objects is available. It is assumed that dynamic objects have GPS receivers, and location information can be acquired through a fast communication link. However, GPS info can be easily modified or suffers from high error because the transmission link is not secure or due to Non Line of Sight (NLOS) between transmitter and receiver. To solve this problem, an entropy metric is introduced to ascertain the value of the supplied information and reject information with a high amount of error present within the data. This work focuses on pedestrians as dynamic objects and uses Finite State Machine (FSM) based hierarchical control to navigate HAVs. It is shown that the entropy metric can improve the efficiency of the control of HAVs.

Keywords: Control of automotive systems, Intelligent driver aids, Kalman Filtering techniques in automotive control, In-vehicle communication networks, Entropy, Finite State Machine, Scan Matching.

1. INTRODUCTION

Entropy is a tool frequently used in statistics to assess the average level of uncertainty present in the samples of a random variable. Highly Automated Vehicles (HAVs) have a level of autonomy 5 as per SAE standard, in an urban environment, HAVs are highly dependent on position information of occluded dynamic objects. Position information of occluded dynamic objects is critical for safe and efficient navigation for all the stakeholders. When collected through sensors, this information is vulnerable to hacking and high errors due to urban canyons. These errors can cause accidents and unnecessary delays in the navigation of HAVs. An entropy-based metric can be used to assess the value and reliability of data received. This work uses a simulation setup with occluded pedestrians, and control of HAVs to assess whether using such a metric would benefit the operation of HAVs. For explanation, three such scenarios are presented in section 1.2 and illustrated in Fig. 1.

The entropy of a random variable gives the value present within the information. In the case of highly likely events, information is less valuable and vice versa. Consider a discrete random variable $X = \{x_1, x_2, ..., x_n\}$ and Probability Mass Function (PMF) as P(X). Then entropy can be explicitly written as shown in (1)

$$H(X) = -\sum_{i=1}^{n} P(x_i) log_b P(x_i). \tag{1}$$

Where b is the base of the logarithm, we have used $b = 10$, which is used when the probability of the event occurring is 1/10. If $H(x)$ has a high value, we can classify it as poor quality, meaning information with high error and data can be rejected.

1.1 Literature Review

Entropy has been used to control automated vehicles to assess the reliability of data received on several occasions. In Jwa et al. (2008), a method is presented for data fusion to track, recognize, and monitor Intelligent Transportation Systems (ITS). In this problem, Robust data alignment is done for successful data fusion. A cost criterion based on entropy is proposed for outlier rejection. In Adamey and Ozguner (2011) multiple targets must be tracked with multiple dynamic sensing agents. Mobile sensing agents plan their motion so that tracking can be efficient and accurate. An entropy-based cost function is utilized to reject information with an unacceptable error range.

In Adamey et al. (2015) a scenario is established where three types of vehicles exist on a highway. Namely, fully equipped, partially equipped, and not-equipped. A fully-equipped vehicle has local sensors and communication capability. In contrast, a partially-equipped vehicle can only communicate, and a not-equipped vehicle cannot communicate and does not have local sensors. The objective is to maintain a tracking list of all the vehicles present in the scenario with a tracking list in the partially and fully equipped vehicles. A Kalman Filter (KF) is used for data fusion and correction, and a covariance matrix

⋆ This work was partially supported by the U.S. Department of Transportation (USDOT) under Grant 69A3552047138 for the CARMEN University Transportation Center (UTC). The first author also acknowledges the Fulbright Master's and PhD Program Funded by the United States Department of State. The first author also acknowledges study leave granted by Bahria University, Islamabad, Pakistan for study in US for postgraduate studies

generated through KF is used to measure the system's entropy. Data is rejected and considered unreliable if the entropy is beyond a threshold. This is used to determine location of occluded pedestrians in "occupancy grid"
In the present work, the pedestrian case is chosen because the safety of all the stakeholders is critical. As local sensors of ego vehicles cannot detect occluded pedestrians, communication setups have to be established. In our simulation setup, we use the information of occluded pedestrians through LTE/4G.
In case pedestrians are occluded, several methods are utilized to detect a pedestrian. In one method, using off-camera on the streets in an industrial area is used to detect the pedestrians through WiFi Borges et al. (2012). However, this solution is costly and not scalable. Another method by Gelbal et al. (2017) utilizes software-defined radio to transmit the position of pedestrians that, in turn, is also very expensive as dedicated DSRC modules are too expensive. In Flores et al. (2018) 802.11 b/g/n communication method is used so that HAVs can receive position information of occluded pedestrians, and the data is fused with LIDAR to get accurate results. However, this solution is also too expensive and not scalable. Currently, we do not have a vast network of WiFi routers in the road infrastructure. Sugimoto et al. (2008) used 3G/WLAN to communicate pedestrian information to the vehicle. However, it was not fast enough that the problem could become scalable and choke the network if the number of vehicles or pedestrians increased. With 4G/LTE and 5G, pedestrians can send their position information, and the communication modules are affordable. We suggest this communication protocol for sharing pedestrian location information with the vehicle. We have assumed all pedestrians have cell phones with GPS sensors available for pedestrian localization. The problem with GPS sensors is that it suffers from Non-Line of Sight (NLOS) in urban areas due to high-rise buildings and can have a high amount of position, navigation, and timing errors, which can have errors up to 50 meters. Also, GPS transmitting frequency can be easily generated, and fake information can be generated to misguide the sensor, resulting in accidents or unnecessary collision avoidance measures from the ego vehicle.
We have developed a process to detect occluded dynamic objects in our proposed method. We assume that the typical profile of errors in GPS position information is known, and we will use baseline controllers to compare with our design controller of ego vehicle to prove our system is more efficient and safe using performance metrics.

1.2 Simulation Case Study

To explain Vulnerable Road Users (VRU). We are presenting three cases in Fig. 1 taken from ISO standard 22737 section 3.1 ISO (2022). Fig. 1(a) is inspired from section 3.1.1 and Fig. 1(b) and Fig. 1(c) is inspired from section 3.1.2. in (b), a bicyclist comes from an alleyway and appears suddenly in front of the ego vehicle. Similarly, in (c), a bicyclist appears from an uncontrolled intersection. Collectively pedestrians and bicyclists are classified as VRUs. The implemented simulation setup is shown in Fig. 1(a). A static pedestrian who is falsely reporting its position with high errors. An occluded dynamic pedestrian jaywalking and a parked vehicle on one road lane. Such a setup is created to show that using entropy and extra information

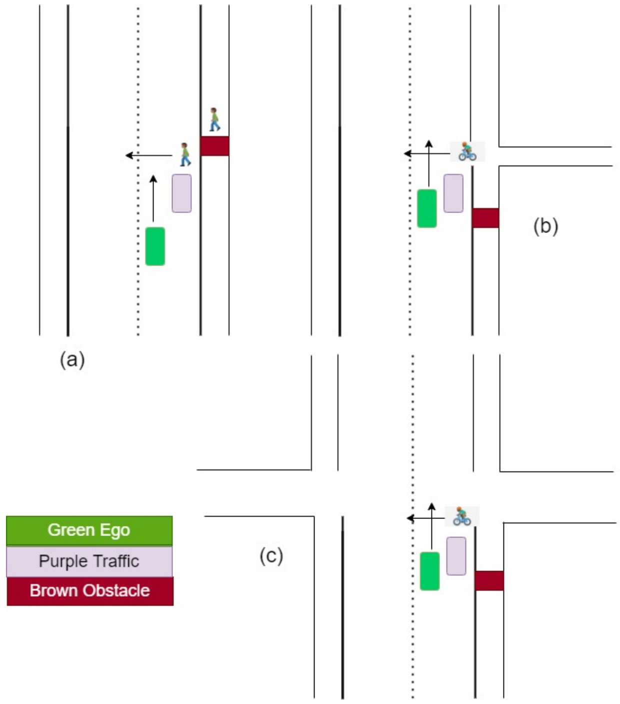

Fig. 1. Scenarios: (a) pedestrian on street, (b) Bike from alleyway, (c) Bike in uncontrolled intersection

from the environment can result in the safe and efficient control of HAVs. The goal is to test our proposed method with multiple baselines to show our method's effectiveness using some performance metrics. Goals achieved in this case study are listed below

- Introduced an entropy-based metric that will assess the reliability of position information received
- Designed some metrics to measure the performance of HAVs in terms of energy consumed and minimum distance to a dynamic object maintained by HAVs
- Designed Finite State Machine based hierarchical control of ego vehicle

In the next section entire architecture of how communication setup is created and how entropy is performed to assess the position accuracy of the data and the metrics utilized to measure the performance of the process is defined, and results are provided in the next section with a concluding remarks in the last section.

2. METHODOLOGY

In this section, the whole process of how occluded pedestrian is detected and avoided successfully is explained. The complete process is shown in Fig. 2. Each block of the process will be explained in subsequent sections, and finally, brief information about the baseline controllers used for comparison is presented.

2.1 Environment

Ego vehicle is modeled with NVIDIA PhysX model which is similar to dynamic bicycle model. Streets and intersections are built in the environment. Position, velocity, and acceleration information can be extracted from the environment, and steering, throttle, and brake to control

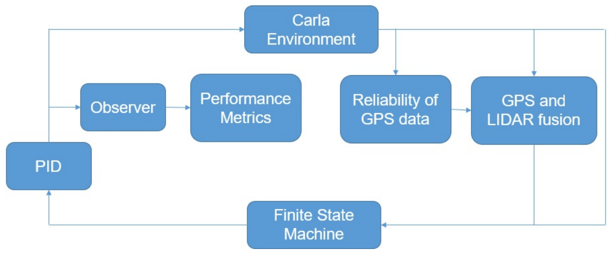

Fig. 2. Block Diagram of the Process

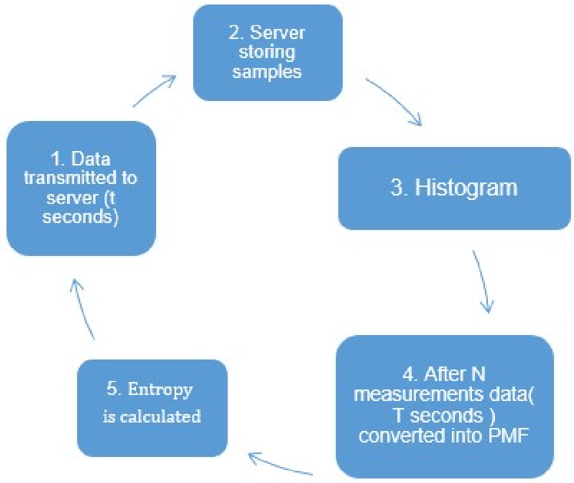

Fig. 3. Flow diagram for entropy calculation

the vehicle. Pedestrians with a point mass model are also added within the environment, with speed and heading control available for the pedestrian. Their position can be extracted from the environment. The environment can be seen in Fig. 1(a).

2.2 Reliability of GPS data

It is assumed that all the pedestrians and vehicles have GPS receivers and 4G/5G transceivers present. Pedestrians are transmitting their position information to a cloud server. The cloud server will then send the data to vehicles in the vicinity of pedestrians to modify their trajectories to avoid collisions. In Urban scenarios GPS can have high error Miura et al. (2015). The primary reason for this error is high-rise buildings that block the transmitter and receiver's Line of Sight (LOS). A process is designed to assess the reliability of the amount of error present in the data. The process is explained in Fig. 3. In the first step, N samples are collected. N has to be small so that process can be executed in real-time. After that, L_2 norm of each two-dimensional sample is calculated. Afterward, a histogram is calculated as shown in (2). where s_j is each GPS sample collected from a pedestrian. In Fig. 3, it is to be noted that t seconds is different from T seconds as to evaluate histogram, we need some samples; therefore, in our work, T=5t.

$$P(s_j) = \frac{Histogram(s_j)}{\sum_{i=1}^{n} Histogram(s_i)}. \tag{2}$$

Then in the next step entropy is calculated as shown in (3) .

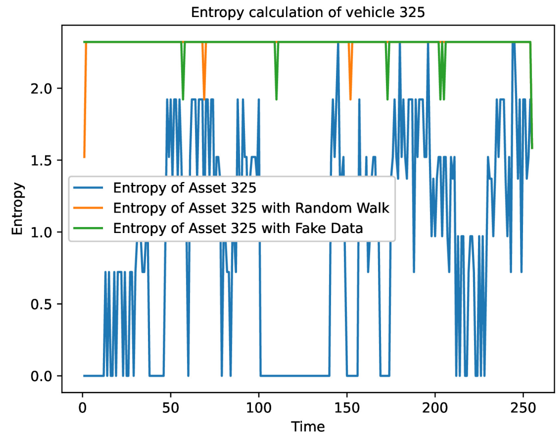

Fig. 4. Entropy of GPS data from Dept of Cincinnati

$$H(S) = -\sum_{j=1}^{n} P(s_j) log_b P(s_j). \tag{3}$$

The process was applied on real-time data taken from the Dept of public health, city of Cincinnati Cincinnati (2021) with two types of error added random walk and Gaussian Noise with shifted mean and high variance. Results in Fig. 4 show that a threshold (approximately 1.89) can be established to ascertain the variance of the data. Moreover, data can be rejected if the data has a high variance. The blue curve is the original data collected, while the green curve is when Gaussian error $X \sim \mathcal{N}(\mu = 0, \sigma^2 = 10)$ is added into the data to show the effectiveness of the method. Similarly, in orange curve random walk error $X \sim \mathcal{N}(\mu = 0, \sigma^2 = 3N)$ is added into the original data. where $N = \{1, 2, 3, ...\}$. In Fig. 4 it can be seen that a threshold of approximately 1.8 can be used, and if the entropy value is below this threshold, data will be rejected.

2.3 GPS and LIDAR Fusion

To fuse data two step process is followed taken from Adamey et al. (2013). Scan Matching is performed as shown in (4)

$$\min_{x_i^{sm}(t)} \left(\frac{1}{2} e_{ij}^T(t) P_{rel} e_{ij}(t) + \frac{1}{2} e_i^T(t) P_{gps} e_i(t)\right). \tag{4}$$

$$e_{ij}(t) = z_{ij}(t) - \max(\|x_i^{sm}, x_j^{gps}\|_2). \tag{5}$$

$$e_i(t) = x_i^{sm} - x_j^{gps}. \tag{6}$$

where x_i^{sm} is realigned position of ego vehicle and x_j^{gps} is the GPS measurement from pedestrian j. While $z_{ij}(t)$ is the relative position measurement from LIDAR where i is the ego vehicle and j is the jth pedestrian and P_{rel}, P_{gps} are covariance matrices.

The benefit of using scan matching is that not only LIDAR and GPS data is fused, but the error is also corrected. The next step in the process is Kalman Filtering (KF). The benefit of using KF is twofold we get the next predicted state of the pedestrian while the amount of error in the position is corrected. The model used to predict the next state of a pedestrian is based on the point mass model.

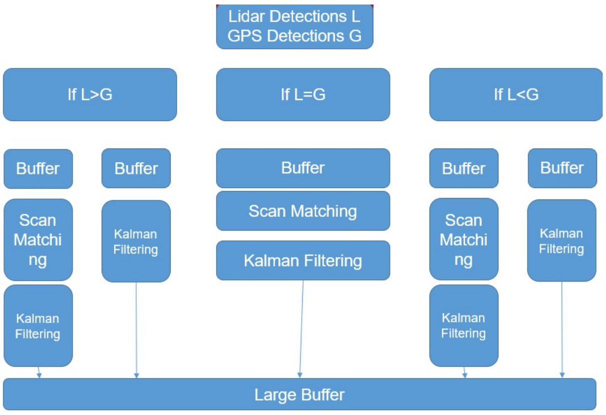

Fig. 5. GPS and LIDAR data Fusion

The next challenge in the process is that there are three possible detections because of communication errors and local sensor (LIDAR) limitations. The three cases and how they are handled to create a final list of position information of all dynamic objects presented are shown in Fig. 5. A similar approach is followed in all three cases. First, LIDAR detections are matched with GPS data using the vicinity rule. The vicinity rule means that in the Horizontal Alert Level (HAL) of LIDAR detection, it is assumed that GPS detection will be available and then processed through KF to get the final predicted positions. The rest of the data is processed through KF to generate a final list of detections.

2.4 Finite State Machine based Hierarchical Control

Finite State Machines (FSM) is a mathematical model through which a finite number of mathematical computations can be performed through a finite number of interconnected states. FSM is widely used to control machines requiring a sequence of operations. A machine can be in one of a finite number of states at any given time. State transitions happen based on indicators from the environment indicating the machine's state. Hierarchical FSM control is utilized to control the vehicle, and at a low-level PID control is implemented. The FSM controller is shown in Fig. 6. It has two higher-level states straight and turn. With a straight node, we have further three states: follow, brake, emergency brake, and similar three states in higher node turn but with different indicators to switch between states. The detail and function of each indicator are shown in table 1.

Table 1. Flags and their Interpretation

Symbol	Definition
F_I	Intersection Flag
α	Distance to start decelerating
D_{d2p}	Distance to closest pedestrian
β	Distance to start hard decelerating
θ	Distance to start decelerating while turning
ξ	Distance to start hard decelerating while turning

2.5 Performance Metrics

To measure the performance of the designed process, three metrics are used. The first metric consists of using the vehicle's lateral position and angle, implying the collision avoidance measure used to avoid the pedestrian. The metric is shown in (7)

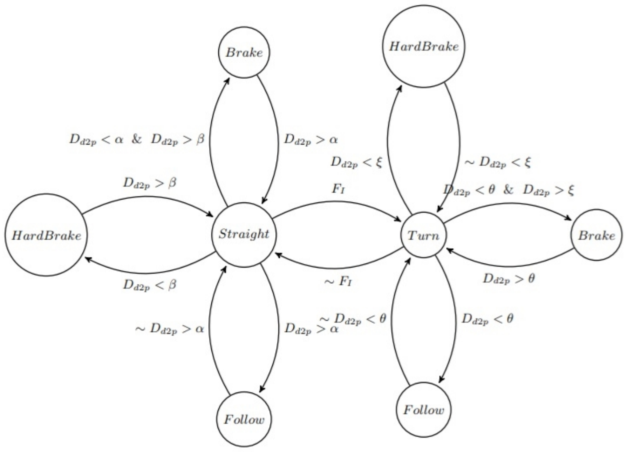

Fig. 6. Hierarchical Finite State Machine

$$J1 = \sum_x x^T Q x + u_1^T R u_1. \tag{7}$$

Q and R are positive definite and positive semi-definite matrices, respectively. Where x is the state vector with states, $[y\ \theta]$ where y is the vehicle's lateral position with respect to the origin and θ is the angle with respect to the ego vehicle. Furthermore, u_1 is the input vector with steering angle γ given to the vehicle.

This metric $J1$ is derived by implementing the observer model using the lateral position and angle of the dynamic model as shown in Acarman et al. (2001). $J1$ is a convex function and can measure the energy consumed by the vehicle while performing evasive action to avoid the vehicle.

The second metric is based on the distance to the closest pedestrian maintained by the vehicle at all times and is shown in (8)

$$J2 = \frac{1}{D_{d2p}} + k\xi. \tag{8}$$

Where D_{d2p} is the distance of the vehicle with the closest pedestrian and k is a very high gain, and ξ is a binary value which is one when the vehicle collides with the pedestrian.

This metric is derived such that when the vehicle gets closer to the pedestrian, it penalizes the metric heavily as it is inversely proportional to the distance to the vehicle and gives a very high penalty if the vehicle collides with the pedestrian. A binary term is added with a high gain to indicate a collision. The additional binary term is added because the distance is measured from the vehicle's center to the center of the pedestrian. It is to be noted that the binary term is piece-wise defined, and when a pedestrian is in close vicinity of the vehicle, it is considered a collision.

Finally, the last metric measures deceleration within the vehicle when it is performing evasive action to avoid the vehicle. The metric is shown in (9)

$$J3 = \sum h(u_2)^T Q h(u_2). \tag{9}$$

where $h(u_2)$ is shown in (10) and u_2 is the acceleration of the vehicle when it is performing an evasive action to avoid the pedestrian

$$h(u_2) = (-1 + sign(u_2))u_2. \tag{10}$$

2.6 Baseline Controllers for Comparison

To show a better performance. Three baseline controllers were used, the first one is vanilla PID control with longitudinal control to avoid dynamic objects, and the other two have similar architecture as in Fig. 2. Nevertheless, some blocks were removed. In the second baseline, the block labeled reliability of GPS data is removed. In the third baseline controller, both the blocks' reliability of GPS data and fusion of LIDAR and GPS functionalities were removed, and the performances were compared. Vanilla PID is utilized only in metric J2 because the FSM-based controller implemented has PID at a low-level hierarchical control. Consequently, if the proposed method performs better after removing the blocks, it will perform better if only vanilla PID is compared. Vanilla PID is used in J2 to demonstrate that vehicle collides with a pedestrian in the absence of the proposed controller.

3. RESULTS AND DISCUSSION

The simulation environment used to test the designed process is CARLA CARLA (2022) open-source environment. The scenario is described in Fig. 1 part (a). The pedestrian moving has a GPS error of $X \sim \mathcal{N}(\mu = 0, \sigma^2 = 3)$ and the pedestrian which is stopped on the sidewalk has an error of $X \sim \mathcal{N}(\mu = 0, \sigma^2 = 10)$. These errors are chosen because GPS errors in the literature are modeled using the Gaussian error model Cui and Ge (2003), and variance is chosen such that it remains under the acceptable range. This reference Maier and Kleiner (2010) shows that the error variance is less than 10 meters. The required speed is 20 m/s because the maximum speed limit in US streets is 45mph, approximately equal to 20 m/s. The vehicle has to start decelerating if the distance to the pedestrian is less than seven meters and has to go into hard deceleration mode if the distance to the closest pedestrian is less than four meters. The results with each metric are shown in Fig. 7, Fig. 8 and Fig. 9.

The metrics are measured only when the vehicle takes action to avoid a collision with the pedestrian. In Metric $J1$ that is shown in section 2.5, we can see the results in Fig. 7. Until 15 seconds, the pedestrian is not detected. No energy is consumed to avoid the pedestrian. However, after that, it can be seen that until 40 seconds, baseline two and baseline three algorithms mentioned in section 2.6 are still consuming energy because the pedestrian on the sidewalk is reporting its position in the collision range of the vehicle and our process can reject this data using entropy metric as described earlier. In the second figure, Fig. 8 the baseline controller one collides with the pedestrian while other baselines and our complete process can avoid the pedestrian. The reason for this is that the setup provides occluded pedestrian's position beforehand, and it has knowledge well before there is a possibility of a collision, and the vehicle performs preemptive action. Moreover, the last metric $J3$ also shows similar results

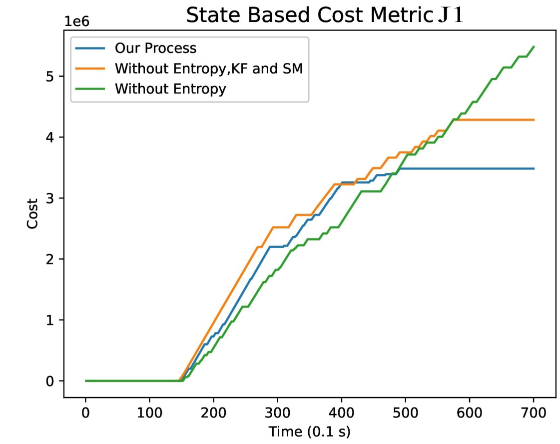

Fig. 7. State based Cost Metric

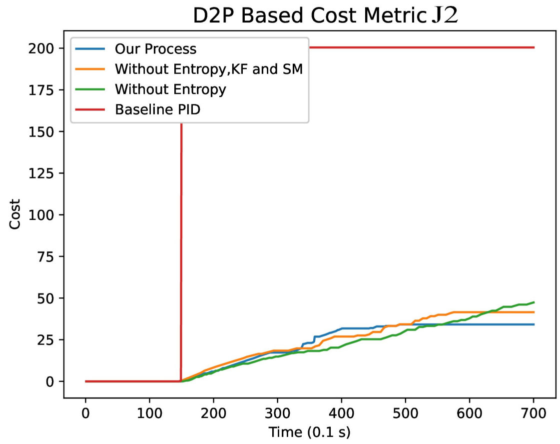

Fig. 8. Distance to Pedestrian based Cost Metric

in Fig. 9 because of the same reasons described above. Vanilla PID is not used in Fig. 9 and Fig. 7 because low-level control of FSM includes PID, and the proposed method performs better when scan matching and KF are implemented within the system. Consequently, if the system outperforms when only specific features are turned off, it will certainly perform better than vanilla PID. In Fig. 8 it is explicitly used to show that the ego vehicle collides with the pedestrian and which is a safety hazard. Our process outperforms baseline controllers and can avoid pedestrians based on the metrics. However, it can be observed that baseline methods outperform the proposed method in some instances. This is because the entropy metric is not always perfect, and sometimes erroneous information results in extra energy consumption of energy, but the overall proposed method outperforms baseline methods. Another noteworthy thing is that having extra information prevents collision. Suppose such a system, as described above, had been developed and deployed. Arizona's fatal crash of a semi-automated vehicle with a pedestrian could have been avoided.

4. CONCLUSION

In this paper, a communication setup was provided to receive position information of the occluded so that col-

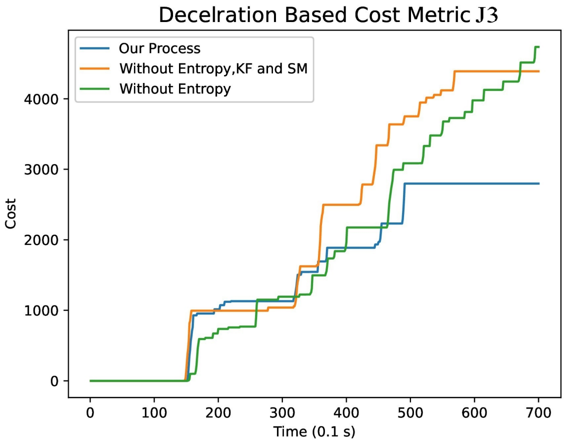

Fig. 9. Deceleration based Cost Metric

lisions can be avoided. This solution is scalable because the communication protocol and access can be easily obtained as almost every pedestrian carries a GPS sensor and LTE/4G module in their cell phones. However, creating a solution in such a way creates a problem if the information has a significant amount of errors present within the information, which will reduce vehicle efficiency significantly. So to solve this problem, an entropy-based threshold is introduced to solve this problem. Then, the fusion process of LIDAR data with GPS data to remove other inaccuracies was introduced. Metrics were presented to show the performance of the proposed method. Results show that the proposed method can make HAVs safer for all the stakeholders present within the environment and make HAVs more efficient in terms of energy consumption to evade pedestrians. Since most pedestrians have LTE/4G modules and GPS devices within their cellular phones, Hardware Infrastructure is already present. Nothing new needs to be established. The method proposed is practical in the real world.

REFERENCES

Acarman, T., Pan, Y., and Ozguner, U. (2001). A control authority transition system for collision avoidance. In *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 01TH8585)*, 466–471. IEEE.

Adamey, E., Kurt, A., and Özgüner, U. (2013). Cooperative traffic mapping using onboard sensing and v2v communication in mixed-traffic environments. In *Second International Symposium on Future Active Safety Technology, FAST-zero*, 1–6.

Adamey, E., Ozbilgin, G., and Ozguner, U. (2015). Collaborative vehicle tracking in mixed-traffic environments: Scaled-down tests using simville. Technical report, SAE Technical Paper.

Adamey, E. and Ozguner, U. (2011). Cooperative multitarget tracking and surveillance with mobile sensing agents: A decentralized approach. In *2011 14th International IEEE conference on intelligent transportation systems (ITSC)*, 1916–1922. IEEE.

Borges, P.V., Tews, A., and Haddon, D. (2012). Pedestrian detection in industrial environments: Seeing around corners. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4231–4232. IEEE.

CARLA (2022). Carla. URL `https://carla.org/`.

Cincinnati, C.o. (2021). Vehicle gps data: Department of public services: Open data: Socrata. URL `https://tinyurl.com/GPSCincinnati`.

Cui, Y. and Ge, S.S. (2003). Autonomous vehicle positioning with gps in urban canyon environments. *IEEE transactions on robotics and automation*, 19(1), 15–25.

Flores, C., Merdrignac, P., de Charette, R., Navas, F., Milanés, V., and Nashashibi, F. (2018). A cooperative car-following/emergency braking system with prediction-based pedestrian avoidance capabilities. *IEEE Transactions on Intelligent Transportation Systems*, 20(5), 1837–1846.

Gelbal, S.Y., Arslan, S., Wang, H., Aksun-Guvenc, B., and Guvenc, L. (2017). Elastic band based pedestrian collision avoidance using v2x communication. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, 270–276. IEEE.

ISO (2022). Iso 22737:2021. URL `https://www.iso.org/standard/73767.html`.

Jwa, S., Ozguner, Ü., and Tang, Z. (2008). Information-theoretic data registration for uav-based sensing. *IEEE Transactions on intelligent transportation systems*, 9(1), 5–15.

Maier, D. and Kleiner, A. (2010). Improved gps sensor model for mobile robots in urban terrain. In *2010 IEEE International Conference on Robotics and Automation*, 4385–4390. IEEE.

Miura, S., Hsu, L.T., Chen, F., and Kamijo, S. (2015). Gps error correction with pseudorange evaluation using three-dimensional maps. *IEEE Transactions on Intelligent Transportation Systems*, 16(6), 3104–3115.

Sugimoto, C., Nakamura, Y., and Hashimoto, T. (2008). Prototype of pedestrian-to-vehicle communication system for the prevention of pedestrian accidents using both 3g wireless and wlan communication. In *2008 3rd International Symposium on Wireless Pervasive Computing*, 764–767. IEEE.